



SpatialAnalyzer

Users Manual



SpatialAnalyzer

Users Manual

2024.04.02

SPATIALANALYZER TRAINING MANUAL

Hexagon Metrology, inc.

04/02/2024. Hexagon Metrology, inc. All rights reserved. Copyright ©1994-2024.

Your rights to the software are governed by the accompanying software license agreement. The owner or authorized user of a valid copy of SpatialAnalyzer software may reproduce this publication for the purpose of learning to use such software. No part of this publication may be reproduced or transmitted for commercial purposes, such as selling copies of this publication or for providing paid-for support services.

Every effort has been made to ensure that the information in this manual is accurate. Hexagon Metrology is not responsible for printing or clerical errors.

Because Hexagon Metrology frequently releases new versions and updates to its applications and Internet sites, images shown in this book may be slightly different from what you see on your screen.

Hexagon Metrology, inc.
436 McLaws Circle
Williamsburg, VA 23185
+1 (757) 565-1500
www.kinematics.com

SpatialAnalyzer and New River Kinematics are trademarks of Hexagon Metrology, inc. , registered in the U.S.

Windows is a registered trademark of Microsoft Corporation.

iPhone and iPod are registered trademarks of Apple Inc.

OpenGL is a registered trademark of Silicon Graphics, Inc.

Other company and product names mentioned herein are trademarks of their respective companies. Mention of third-party products is for informational purposes only and constitutes neither an endorsement nor a recommendation. Hexagon Metrology assumes no responsibility with regard to the performance or use of these products.

TABLE OF CONTENTS

CHAPTER 1: WELCOME	1
About This Manual	2
Conventions	2
SA Packages	5
SA Professional	5
CHAPTER 2: GETTING STARTED	5
SA Ultimate	6
SA Analysis	7
SA Machine	7
Feature Summary Matrix and Details	8
System Requirements	8
Minimum Requirements	8
Recommendations for Optimal Performance	8
Check My Computer	8
Installation	9
Installing SA	9
Installer Command-Line Options	10
Instrument Drivers	11
Licensing	12
Traditional USB Dongle Licensing	12
Hexagon Universal License Library (HULL)	13
Nalpeiron Cloud Licenses	15
Certification	20
How to Get Help	20
SA Community Webpages	20
Documentation	20
Customer Support	21
Context-Sensitive Help	21
Tutorials & Sample Files	21
Navigating this Users Manual	22
An overview of fundamental concepts and guiding principles	22
Getting into the details of operation	22
The core of measurement - Points and feature measurement	23
Working with Instruments - SA Instruments Manual	23
How to effectively move and align measurements for analysis	24
Live view and build it functionality	24
Analysis, inspection refined alignment operations	24
Measurement Uncertainty and Unified Spatial Metrology Networks	25
Scripting and Automation	25
Specialized functionality	25

TABLE OF CONTENTS

References	26
CHAPTER 3: SA FUNDAMENTALS	27
Installation and Configuration	27
Installing Spatial Analyzer (SA).....	27
The Working Frame	29
The Default Collection.....	30
Measured Points vs. Cloud Points.....	30
Point Offset Compensation	31
Probe Offsets	32
Surface Normal Direction.....	33
Comparing Measurements to Points and Objects.....	34
Unbounded Geometry	35
Feature Measurement.....	36
Traceability	37
The Alignment Process.....	40
Selection Filtering.....	40
Architecture	41
Host SpatialAnalyzer Session	42
Instrument Interface	42
Instruments.....	43
CHAPTER 4: EXPLORING THE USER INTERFACE	47
Application Components.....	47
SA Ribbon Menus.....	49
SA TreeBar	54
Report Bar	57
Status Bar	57
MP Bar.....	58
Classic Toolbars	58
Main Toolbar.....	59
Graphical Control Toolbar.....	60
Color/WCF Toolbar	61
Instrument Docking Bar.....	62
SA Toolkit.....	62
UI Themes.....	63
Custom UI Profiles.....	63
Customizing Toolbars & Menus	65
Keyboard Shortcuts	67
Options.....	68
Language Translation.....	69
Character Sets.....	71

TABLE OF CONTENTS

Creating a Translation	71
Merging Language Files	73
Using .CSV Files For Translation	73
CHAPTER 5: THE GRAPHICAL VIEW.....	75
Navigating the Graphical View	75
Graphical View Controls	75
Zoom	75
Autoscale.....	76
Pan	76
Rotate.....	76
Center In View.....	77
Adjust View Components.....	77
Jumping to Previous/Next Views	78
Set Viewpoint From Frame.....	78
Set Viewpoint From Instrument Updates	78
Set Viewpoint From Object	81
Set Viewpoint From Surface Point.....	81
View Clipping	81
Multiple Graphical Views	83
Heads-Up Display	83
User Options Controls	84
Saved Viewpoints.....	86
Rendering Modes.....	87
Surface Backsides	89
Background/Highlight Color	90
Hardware Acceleration.....	91
Surface Rendering Options	92
Labels	94
The Hover Window.....	95
Object-Specific Rendering Modes	97
Visibility.....	98
Miscellaneous Settings & Commands	99
View Animation Steps.....	99
Arrowhead Sizes	99
Scale Indicator	100
Shadow Plane.....	100
Connection Indicator	101
Uncertainty Fields.....	102
Highlighting	103
View Refresh.....	104

TABLE OF CONTENTS

Copying the Graphical View to the Clipboard	104
Creating a Movie Capture of the Screen	104
Controlling Target Icon Size.....	105
CHAPTER 6: WORKING IN SA.....	107
Selection.....	107
Graphical Selection Process	108
Treebar Selection	111
F2 Selection	112
Selection By Color	115
Context Menus.....	116
Organizing Your Job	117
Folders.....	118
Collections.....	119
Relocating Items to Another Collection.....	120
Rearranging Points	121
Naming & Renaming Items	124
Deleting Items	128
Double Colon Format	131
Configuring SpatialAnalyzer	132
Job File Settings.....	132
Machine Settings.....	137
Security	139
System Diagnostics	141
Memory Monitor	141
Check Job Data Consistency	142
Regenerate References	142
Repeating Commands.....	142
Working With Items	143
Moving Objects.....	144
Mirroring Objects.....	149
Scaling Objects	150
Finding Items In The Tree	151
Network Browser	152
Remote Monitoring	153
Working With The Clipboard	154
Working Frame and Locked Objects	155
About the World Frame.....	155
Locked Items.....	156
Object Color	156
Accessing Item Properties.....	158

TABLE OF CONTENTS

The Calculator	159
User Notes	159
Adding Notes to Items	159
Item History	160
Print Setup	160
Command-Line Arguments	161
Launching SA with Command-Line Arguments	162
Example Command-Line Arguments	162
Multiple Instances and Exiting SA	163
CHAPTER 7: WORKING WITH FILES	165
SA Files	165
Creating New Files	165
Opening Files	166
Saving Files	166
Save As	167
Backing Up Files	167
Restore Points	169
Password-Protecting SA Files	169
File Import	170
Auto Import	170
Importing SpatialAnalyzer Job Files	171
Direct CAD Access	171
Importing Standard CAD Formats	177
Importing Predefined ASCII Formatted Files	178
Importing ASCII Formatted Points With Offsets	180
Importing XML Files	180
Cloud Files: Standard Formats	182
Custom Formats	183
Embedding Files	183
Importing Images	184
File Export	184
Exporting ASCII Formatted Points	184
Exporting Point Clouds	186
Exporting Vector Groups	187
Exporting Geometry	187
Exporting Object Frames	187
Exporting to CAD Formats	188
Q-DAS SPC Data	189
Exporting Window Placements	189
Templates	190
The Default Template	190

TABLE OF CONTENTS

Screen & Graphics Capture	191
CHAPTER 8: POINTS	193
About Points	193
Point Groups	193
Constructed Points	194
Measured Points	194
Point Characteristics	195
Point Tolerance	196
Notes	197
Targeting Offsets	197
Accessing A Point's Group Menu From the Graphical View	198
Measurements	199
How SA Computes a Measured Point's Location	199
About Target Observations	200
About Target Offsets	202
Characteristics Unique to Measured Points	203
Hidden Point Measurements	208
The Hidden-Point Bar Database	208
Gravity Frame	209
2 Point Hidden-Point Bar	210
N Point Bars or Fixtures	211
Setting Properties for Multiple Points	213
Working With Point Lists	214
Point Sets	216
Point Sets	216
Constructing Points	218
Entering Point Coordinates	218
Constructing a Point at the Center of a Line	218
Constructing Points at the Center of Spheres and Circles	219
Constructing Points at the Center of a Cylinder	219
Projecting to the Closest Point on Objects	220
Projecting Points Parallel to a Working Coordinate Frame Axis	221
Projecting Points Radially from a Working Coordinate Frame Axis	221
Projecting Points Spherically from a Working Coordinate Frame Origin	222
Projecting Points to a Curve	223
Creating Points at a 3-Plane Intersection	224
Creating Points at the Intersection of a Plane and Circle	224
Constructing Points at the Intersection of a Line and Plane	225
Constructing a Point at the Intersection of 2 Lines	226
Constructing Points at the Intersection of a Line and Circle	226
Constructing Points at the Intersection of a Line and Cylinder	227

TABLE OF CONTENTS

Constructing Points at the Intersection of a Line and Sphere	227
Constructing Points at the Intersection of 2 B-Splines	228
Constructing Points at the Intersection of a B-Spline and Surfaces	229
Constructing Points at the Intersection of Principle Object Axes and Surfaces	230
Shifting Points In a Direction	230
Shifting Points In Cylindric Coordinates.....	231
Laying Out Points on a Grid By Count	232
Laying Out Points on a Grid By Distance.....	233
Laying Out Points On a Circle	234
Generating Random Points.....	235
Laying Out Points On A Surface By Clicking.....	236
Laying Out Points On An Object Vertex By Clicking.....	236
Laying Out Points On An Object's Vertices	237
Laying Out Points Evenly On Curves	237
Laying Out Points On Curves With Prescribed Spacing	238
Laying Out Points On Curves by Clicking	238
Laying Out Points on Curves Max Chordal Deviation	239
Laying Out Points On Lines	239
Laying Out Points At The Center of Surface Holes.....	240
Laying Out Points In The View By Clicking	242
Copying Existing Points.....	242
Moving Existing Points Into A New Group.....	242
Creating a Subset of Points With Greatest Spacing.....	243
Auto-Corresponding Two Groups By Proximity.....	244
Auto-Corresponding 2 Groups by Inter-Point Distance.....	245
Averaging Points.....	245
Creating Points at the End of a Line	246
Creating Points at a Plane Cross Section:.....	246
Constructing Points Evenly Spaced on a Surface:	246
Constructing Points From Vectors	247
Constructing a Single Point through Patch Normal Shift.....	248
Scale Points	249
Copy Groups Excluding Obscured Points.....	250
Auto Filter to Faces	250
From Surface Faces	251
Closest points between 2 Surfaces	251
CHAPTER 9: FRAMES	253
6D Locations and Measurements	253
The Working Frame In SA.....	253
Frame Properties	256
Frame Transform Representation.....	258

TABLE OF CONTENTS

Frame Wizard	259
Additional Frame Construction	262
Frame Sets	266
CHAPTER 10: GEOMETRY	269
Locations and Measurement	269
Point Group	269
Objects vs. Geometry Relationships	272
Lines	272
Properties	272
Construction	273
Circles	275
Properties	275
Construction	276
Planes	277
Construction	277
Slots	279
Properties	279
Construction	279
Ellipses	280
Properties	280
Construction	280
Cylinders	281
Properties	281
Construction	282
Spheres	282
Properties	282
Construction	283
Cones	283
Properties	284
Construction	285
Paraboloids	285
Properties	285
Construction	285
Toruses	285
Properties	286
Construction	287
Perimeters	287
Properties	287
Construction	287
Graphical Entities	288

TABLE OF CONTENTS

Sphere Section.....	290
CHAPTER 11: INSTRUMENTS.....	293
Instrument Properties.....	294
Sensor Control	295
Scale Factor	296
Uncertainty Variables	298
Instrument Graphics (PCMM arms only)	298
Instrument Communication Controls.....	299
Instrument Shots.....	300
Instrument Shot Modes	301
Shot Colors	301
Angle Only Shots	302
Instruments and Measurements.....	302
Using Instruments	303
Instrument Options and Properties.....	304
The Instrument Interface	305
Starting an Instrument Interface	307
Running Multiple Instrument Interfaces.....	308
Connection Configurations.....	309
IP-Based Instruments	309
IP Address Basics	309
Computer and Instrument Setup	310
Changing a Static IP Address.....	311
Multiple IP Setup (Dual IP)	312
Measurement Simulation	312
Laser Tracker Simulation	312
Portable CMM Arm Simulation	315
Simulation with Other Instrument Types	316
Measurement Fabrication	316
Taking Measurements from SA	317
Auto Measure	317
Guided Points	324
Proximity Triggered Points.....	325
Synchronized Measurement	328
Synchronized Arm/Tracker Scans for Alignment	329
Crib Sheets and Measurement Templates	332
Monitoring.....	333
Watch Windows and Transformation Tracking.....	333
Tools	334
Scale Bars	334
Perimeter.....	334

TABLE OF CONTENTS

Target Detection	334
Gravity Measurements in SA	335
Measuring Level	335
Level Compensation Verses Measurement	337
CHAPTER 12: ALIGNMENT & TRANSFORMATIONS	345
Review of Available Alignment Options	346
Measured Point to Reference Alignments	346
Feature and Reference Point Alignments	347
Surface Alignments	347
Point Cloud/Mesh to Reference Alignments.....	347
Frame and Feature Based Alignments	347
Advanced Optimizations.....	348
Network Alignments	348
Saved Alignments.....	349
Repeat Last Alignment	349
Relationship Construction.....	349
Alignments Saved Separately	350
GD&T Datum Alignments.....	350
Measured Points to Reference Alignments.....	350
Best-Fit Transformations.....	350
Point to Point Best-Fit.....	351
Locate By Measuring Nominal Points.....	360
Drift Check.....	362
Feature and Reference Point Alignments	365
Nominal Geometry Relationships	365
Reference Point System (RPS) Alignment	366
Surface Alignments	369
Quick Align to CAD.....	369
Best-Fit Points to Surfaces.....	371
Network Alignments.....	375
Bundle Adjust.....	375
Unified Spatial Metrology Network(USMN)	378
Job Pre-Planning and Instrument Placement	378
How to Pre-Plan instrument placements.....	378
CHAPTER 13: WATCH WINDOWS & TRANSFORM TRACKING	381
How to Add a Watch Window.....	382
Watch Window Types	382
Watch Instrument's Point.....	383
Watch Point to Point	383
Watch Point to Point With View Zooming.....	384
Watch Closest Point	384

TABLE OF CONTENTS

Watch Point to Objects.....	384
Watch Projected Point to Objects.....	385
Watch Three Point Frame.....	386
Watch Two Point Axis Alignment.....	387
Watch Theodolite Observer Point to Point.....	388
Watch Theodolite Observer's Point.....	389
Watch Theodolite Observer's Data.....	389
Watch Window Options.....	390
Data Representation.....	390
Item Visibility.....	394
Background Window.....	395
Text Controls.....	395
Arrow Settings.....	396
Setting a Zero Reference.....	397
Graphical Guide.....	397
UDP Broadcast.....	399
Transformation Tracking.....	400
Relationships with Trans-Track.....	400
Transform in 6D using Instrument Updates.....	401
Big Data Window.....	408
UDP Broadcast.....	410
SA Remote and Metrology Remote.....	411
SA Remote.....	411
Initial Setup.....	412
Connecting the Phone, Computer and Tracker.....	412
Available Status/Commands.....	416
CHAPTER 14: RELATIONSHIPS.....	417
What is a Relationship.....	417
Definition of Relationship Types.....	418
Points Based Relationships.....	418
Object Orientation Relationships.....	418
Special Function Relationships.....	419
Geometry Relationships.....	420
General Relationship Properties & Controls.....	420
Visual Controls.....	420
File Management Tools.....	421
Projection and Fit Controls.....	422
Relationship Fit Settings.....	423
Reporting Options and Controls.....	426
Combined Relationship Reports.....	428
Recomputation.....	428

TABLE OF CONTENTS

Measurement Associations	429
Measurement Trapping and Trapping Controls.....	429
Properties & Controls - Special Function Relationships	430
Group to Nominal Group.....	430
Points/Groups to Surface Faces	431
Vector Group to Vector Group Relationships.....	431
Pipe Relationships	435
Using Pipe Relationships	437
Defining Pipe Relationships.....	437
Working With the Cut	441
Relationships with built in Alignments.....	443
Minimizing Relationships	444
Performing a Relationship Fit	445
Moving Objects by Minimizing Relationships	445
Sequential Alignment.....	453
Move Collections by Minimizing Relationships	454
CHAPTER 15: GEOMETRY FITTING & GR-FEATURES	457
Geometry Fitting	457
About Planar Offset Methods.....	458
About Measured Side Settings.....	461
Basic constructions and the Fit Interface.....	463
The Fit Interface.....	463
Point List	467
Definition of Geometry Relationship Types	472
Geometry Relationships	472
Properties & Controls in GR-Features	472
GR Properties and Report Options	472
Cardinal Points.....	476
Automatic Vector Groups.	477
Geometry Fit Settings:.....	478
Working with Cloud Data.....	479
Reporting Controls.....	481
Additional GR-Feature types	482
Single Point Geometry.....	484
Dynamic Geometry	484
Comparison Only Geometry Relationships.....	486
CHAPTER 16: FEATURE INSPECTION	489
Why Feature Based Inspection	489
Why GR-Features verses GD&T.....	490
Chapter Guide.....	490

TABLE OF CONTENTS

Building GR-Features	490
Nominals	491
New Feature Controls.....	492
Building New Features.....	494
Extract Geometry	498
Relationships.....	499
Feature Inspection	500
Instrument Alignment.....	501
Measurement Trapping	502
Helpful Tools for Inspection	505
Feature Trapping Controls and Automation.....	508
Workflow Reference Guide for Points based Inspection	511
How to Measure Features (Auto Detect) Using a Probing Device.....	511
Building Geometry from Existing Measurements.....	514
Building Center Points and Intersections from Existing Geometry.	516
How to build a coordinate system using measured features	521
How to measure a set of reference points.	524
Cloud Based Part Inspection with an Arm and CAD.....	528
SA Toolkit (Classic Menu Only)	535
Relationship tab	536
Sweep and Sweep & Move	538
Analysis Tab	539
Visualization	539
Construction	540
Fitting.....	541
Uncertainty	541
Scaling.....	542
Reporting Tab	542
Reports.....	542
Dimensions.....	543
Callouts.....	544
CHAPTER 17: POINT CLOUDS.....	545
Point Clouds.....	545
Point Cloud Formats	545
Point Cloud Visualization	546
Real-time Data Display	546
Working with Raw Point Cloud Data.....	547
Boundary Points	555
Real-Time Meshing.....	556
Voxel Cloud Display	558
Using Voxel Colors for Deviation and Analysis	562

TABLE OF CONTENTS

Clipping Planes	565
Cross Section Clouds	567
Cloud Filtering and Sub-Sampling	570
Auto Filter Point Status	572
Point Cloud Construction Options	572
Create Empty Scan Stripe Cloud	572
From Existing Clouds	572
From Existing Clouds Uniform Spacing	573
From Existing Cloud Points	573
Random Data.....	573
From Existing Point Group	573
Auto Filter to Faces	573
Auto Filter to Vectors	574
Auto Filter to Lines.....	574
Auto Filter to circular Arc B-Splines.....	574
On Object's Vertices.....	575
Auto Filter to Cross -Sections	575
From Frame Set	575
Building Points from Point Clouds	575
Converting Clouds to Points	576
Filtering Clouds to Points	576
Filtering Clouds to a Plane	577
Filtering Clouds to Vectors by Resolving Points	578
Extracting Center Points From Scanned Spheres.....	580
Cloud Based Alignment/Registration Tools	582
Aligning Clouds to CAD / Mesh / Clouds.....	582
Point Extraction and Best-Fit	585
Cloud to Object Relationship Optimization	585
Refit Scan Passes.....	585
Query Point Cloud to Point Cloud	586
Feature Extraction from Point Clouds	587
Auto Filter to Nominal Geometry >3D Extraction.....	587
Feature Specific Filtering	592
Auto Filter to Nominal Geometry >2D Geometry	597
Extract Geometry from Clouds.....	599
Meshing Point Clouds	601
Scan Stripe Mesh Operations	602
Extended Parameter Settings	604
Scan Stripe Mesh Editing	608
CHAPTER 18: GD&T	611
Importing CAD with GD&T Annotations	612

TABLE OF CONTENTS

Creating GD&T Annotations	612
Creation Options	613
GD&T Entities.....	621
GD&T Annotation Properties.....	622
GD&T Alignment.....	634
Pattern Datums	637
Surface Alignment.....	637
Disabling Datum Alignment	638
Simultaneous Evaluations	639
GD&T Evaluation	640
GD&T User Options	640
GD&T Evaluations versus RMS fits	643
Decimal Precision	643
Evaluation Zone/Length	644
Cylinder Evaluation & Coverage	644
Slots and Open Slots.....	645
Point Cloud Thinning Controls.....	648
Feature Check Specifics	650
Form Checks	650
Orientation Checks.....	654
Location Checks	655
GD&T Workflow	664
Guided Inspection.....	664
Inspection Bar Trapping and Direct Association	669
Show Only Active Feature Check or Datum.....	670
Remeasure Failed Checks Only	670
Feature Inspection Auto Filter.....	671
GD&T Reporting.....	673
Reporting Options	674
Adding Checks to a Report.....	679
CHAPTER 19: REPORTING	681
Building New Reports	681
Report Bar	682
Displaying the Report Bar.....	682
Building Reports Using the Report Bar	683
Generating Reports.....	684
Quick Reports	684
SA Report Builder Utility.....	685
Reporting User Settings	686
SA Reports	688
Creating an SA Report.....	688

TABLE OF CONTENTS

The Report Designer	689
Export Options	692
Adding Elements to the body of a Report	693
Report Fields	694
Tables	698
Modifying Items in the report.....	699
Report Templates.....	700
Object Report Options.....	702
Events.....	703
Queries	703
Query Basics	704
Cloud to Cloud Query	708
Vector Groups	710
Vector Group Properties.....	710
Vector Construction Options	722
Dimensions.....	725
Types of Dimensions	725
Working With Dimensions	726
Properties Common to all Dimensions	727
Type Specific Dimension Properties	730
Point to Point Dimensions.....	730
Point to Object Dimensions.....	732
Object to Object Dimensions.....	735
Diameter / Radius Dimensions.....	736
Callout Views	738
Creating a Callout View	738
Callout View Properties.....	739
The Layer Manager	742
Adding Callouts to a Callout View	744
Callout Styling	744
Callouts as Hyperlinks.....	745
Types of Callouts	746
Charting.....	750
Available Charts.....	751
Chart Plot Region Controls.....	752
Chart Statistical Results	754
Photos.....	756
Metrology Reporting	757
CHAPTER 20: MEASUREMENT UNCERTAINTY.....	761
What is Measurement Uncertainty?.....	761
Measurement Uncertainty and Visualization in SA	762

TABLE OF CONTENTS

Basic Uncertainty Computation Process for a Tracker/Total Station:.....	771
Basic Uncertainty Computation Process for a PCMM Arm.....	774
Uncertainty for Points with Multiple Observations	775
Measurement Analysis with Uncertainty Fields	776
Point to Point Distance Uncertainty	776
Geometry Fit Uncertainty	777
How to Perform a Geometry fit with Uncertainty:	780
CHAPTER 21: UNIFIED SPATIAL METROLOGY NETWORK	785
What is USMN	785
What is USMN?	785
Weighted Bundle.....	786
Using USMN	790
Observations and Point Group Requirements for USMN.....	790
USMN Dialog	793
Basic USMN Solution Process	807
Uncertainty Context Manager.....	808
Building a Network Solution.....	808
Accessing More Details	812
Working with Scale Bars	816
Incorporating Digital Levels	816
Expanding a Network Solution	817
CoVariance with SA.....	818
CHAPTER 22: WORKING WITH SURFACES	821
Surface Grouping.....	821
Constructing a Merged Surface.....	822
Dissecting Surfaces.....	822
Show Selected Faces Command	823
Duplicate Surface Faces	823
Surface Normal Conditioning.....	824
Indicating Point on Positive Side.....	824
Reversing Surface Normals	825
Auto-Reversing Surface Normals	825
Surface Construction Options.....	826
From a Collection of Surfaces.....	826
By Dissecting Surfaces	826
Select Faces by Projecting Points	826
From Objects	826
From Plane, Cylinder, Sphere, and Cone.....	826
Tubular Surface from a B-Spline	826
Advanced Surface Fitting.....	826
B-Splines	827

TABLE OF CONTENTS

Properties	827
B-Spline Construction	828
Advanced Surface Fitting	833
Polygonized Meshes	837
From Surfaces	837
From Cloud Points and from Selected Cloud Points	837
Colorized Mesh from Point Proximity	838
CHAPTER 23: MEASUREMENT PLANS	839
About Measurement Plans	839
Running a Measurement Plan	839
Quick MPs	841
Working With Measurement Plans	843
Protecting Scripts	844
Generating SDK Code	845
The MP Editor	846
Overview	847
About Commands	848
Entering Commands	852
Editing Commands	852
Resizing Columns	854
Hard Coding Argument Values	854
Referencing Arguments	855
Using Variables	856
Connections View	857
MP Find and Replace	858
Commenting Out Commands	860
Setting Breakpoints	861
Locking Commands	862
Entering Comments	863
Indenting Commands	863
Setting Command Colors	863
Grouping Commands	864
Working with Subroutines	865
Writing Your First Script	866
Debugging	869
Stepping Through a Script	869
Debug Mode	870
MP Watcher	870
CHAPTER 24: SA SDK	873
SA SDK	873
SDK Engine	874

TABLE OF CONTENTS

Automatic Code Generation	874
Setting Up an SDK Project.....	875
Sending Commands to SA.....	884
CHAPTER 25: THE OPC INTERFACE	889
What Is OPC?.....	889
SA & OPC.....	889
Prerequisites.....	890
Configuring the OPC Server.....	890
Step 1: Creating a DCOM User	890
Step 2: Disable Windows Firewall	891
Step 3: Adjust User Account Control Settings	891
Step 4: Install SA.....	892
Step 5: Install OPC Core Components	892
Step 6: Register the NRK OPC DA Server.....	892
Step 7: Enable the OPC Server in the SA Configuration	892
Step 8: Test the Server with a Local Client.....	892
Step 9: Configuring System-wide DCOM Settings	893
Step 10: Configure the OPC Server's Specific DCOM Settings	894
Configuring the OPC Client.....	895
Step 1: Create an OPC User Account & Install Components.....	895
Step 2: Configure System-Wide DCOM Settings	895
Step 3: Test the OPC Server Connection.....	896
Using SA's OPC Server.....	896
MP Support	897
CHAPTER 26: MACHINE CALIBRATION.....	899
SA Machine.....	899
Robots in SA	900
Controlling a Robot from SA	903
Collecting Calibration Measurements	904
Calibration Interface	905
Denavit-Hartenberg Notation.....	908
APPENDIX A: KEYBOARD SHORTCUTS.....	911
APPENDIX B: MATH REFERENCE	913
Linear Least Squares Adjustment.....	913
Basics of Spatial Transformations.....	914
The old viewpoint debate: Are we describing or transforming?	915
Unit Rotation Matrices.....	916
Compound Rotations.....	916
Notes on Decimal Precision and Rounding.....	918

TABLE OF CONTENTS

This Page Intentionally Left Blank.

Welcome

Congratulations on your purchase of SpatialAnalyzer, the world's leading portable metrology software platform. SpatialAnalyzer is a traceable, graphical metrology software package that can simultaneously communicate to virtually any number and type of dimensional measurement systems and perform complex analysis tasks with ease.

SA is used in many high-profile applications throughout a wide array of industries, from manufacturing to science and research. Some of these industries include:

- Aerospace (aircraft and spacecraft)
- Shipbuilding
- Particle Accelerators
- Nuclear Energy
- Automotive
- Heavy Equipment
- Machining

You are about to embark on learning how to adeptly use the most powerful portable metrology software ever created. We think this learning period will prove to be remarkably short. Most users are able to confidently acquire and analyze data after only one day. In fact, it's quite possible that you can start using SpatialAnalyzer simply by performing a few trial and error "test drives". The software is robust, and in general, it will politely inform you if you try something impossible.

We suggest that you may ultimately save time and grief by taking the time to read through this manual. Don't be too particular about memorizing anything—after all, this document is available from

within SpatialAnalyzer. We hope you will obtain an understanding of the philosophy behind SpatialAnalyzer by reading this manual. If you are unable to read the manual, it also serves as an excellent reference when you have questions.

Above all else, the goal of our work has been to develop a software package where users of all levels of expertise may actually enjoy taking and analyzing measurements and may have confidence in the numbers they report.


About This Manual

New River Kinematics (NRK) regularly updates this manual to incorporate the most recent updates and changes to SA. However, due to the fact that this manual is currently under development, and due to the rapid pace of NRK's software development, you may encounter out-of-date material or options that are not yet available in your version of SA. If you have any questions about the manual or its content feel free to reach out to us.

SpatialAnalyzer comes in several packages. In cases where it may be ambiguous, we try to make it clear if a capability is not available in a specific SA package.

If you have any comments or suggestions for this manual, or if you notice any errata, please report them to support@kinematics.com.

Conventions

 **Note:** An important note that accompanies the text appears in a box like this.

A standard set of conventions have been established to help ensure that this document is easy to read. The established standards are described here:

The titles of windows and dialogs will appear like this: *Best Fit Transformation*.

Menu choices appear like this: **Construct>Points>Enter**.

Checkboxes, dropdowns, and other control choices appear like this: *Show Scale Indicator in View*.


Button titles appear like this: **Apply**.


Keyboard combinations appear like this: **Ctrl+D**.

Named fields or boxes that subdivide areas of the user interface will appear like this: *Degrees of Freedom*.

Filenames appear like this: License10734.met.

Text that you should type in will appear like this: **Nominal**.

 **Warning:** A warning box, like this, will appear anytime there is information considered to be of a critical nature for either data integrity or measurement data accuracy. Failure to heed warnings could potentially lead to loss of data or incorrect results.

 **Tip:** Tips or recommendations that include best practices or time-saving techniques will appear in a box like this.

New terms will look like this: *Relationship Minimization*.

Hyperlinks or links to other locations in this document look like this:
support@kinematics.com.



A Closer Look

Additional information that is not critical, but may be of interest to curious readers, is printed in a box like this.

**WARNING: UNITED STATES COPYRIGHT LAW PROTECTS THIS MANUAL,
ASSOCIATED SOFTWARE, ALGORITHMS, AND VISUAL CONTENT.**

This Page Intentionally Left Blank.

Getting Started

CHAPTER

2

You are about to start your journey toward learning the world's most powerful portable metrology software package. But you can't take the first step until the software's installed!

SA Packages

Spatial Analyzer is only a single software package with a single installer. However, different license modules are available to enable access to different levels of content within it. There are currently four license levels or feature packages: SA Professional, SA Ultimate, SA Analysis and SA Machine. The details of the functionality provided in each of these packages is provided below.

SA Professional

SA Professional is the professional measurement, alignment, inspection, analysis, and reporting software for all portable metrology instruments. In addition to all essential measurement needs, it includes Native CAD import, several key inspection features, geometry inspection, and measurement automation.

MEASUREMENT

SA Professional offers 100% traceability from measurement to reporting. All measurement data is stored alongside time stamps, instrument information, weather data, and other measurement parameters. Log files track user actions, fit results, and more, allowing you to see the entire history of the job file.

ALIGNMENT

An essential part of the measurement process is the alignment of measurements to a known coordinate system. SA Professional includes a variety of alignment methods ranging from traditional 3-2-1 alignments to more advanced surface fits.

BUILD

The Relationship functionality offers an easy way to calculate and observe part deviation. Relationships are dynamic in nature and update

automatically if part alignment or data changes.

EVALUATION & ANALYSIS

A user-friendly interface permits both graphical and numerical depiction of measurement uncertainty, enhancing your perspective of measurement quality.

REPORTING

SA Professional offers quick, user-friendly reporting functionality. Quick Reports are ideal for on-location reporting, report templates are perfect for repetitive work, and the Report Designer allows you to drag and drop items to develop custom reports using tables, charts, and graphics.

GD&T INSPECTION

GD&T allows you to import CAD with GD&T annotations, create annotations manually, and inspect to GD&T standards with real-time reporting.

GEOMETRY INSPECTION

Geometry inspection enables you to define design-based inspection routines from a CAD model or primitive geometry.

NATIVE CAD

Native CAD provides import support for such major applications as CATIA V4, V5, Pro/Engineer, SolidWorks, Unigraphics, and other CAD formats.

SA Ultimate

SA Ultimate is the premier measurement, optimization, analysis, reporting, and automation software suite for all portable instruments. It includes everything contained in SA Professional, plus the following features:

REAL-TIME ALIGNMENT

Transformation Tracking allows you to track moving parts in real-time so that you can monitor a part's position as it is guided into place.

ADVANCED FIT OPTIMIZATION

Relationship minimization provides the power necessary for advanced alignments and can help bring an out-of-tolerance part back within tolerance. You can also define fit envelopes to satisfy multiple constraints/requirements.

COMPLEX INSTRUMENT NETWORKS

USMN is an extremely powerful feature that leverages the uncertainty characteristics of different instruments to provide a much more accu-

rate instrument network than that of traditional alignment methods. It is used by many metrology groups to solve large-scale networks such as accelerator rings, full submarine surveys, and large machinery analysis.

UNCERTAINTY ANALYSIS

The ability to include a level of confidence estimate with your measurements and your alignments is essential to accurate metrology. SA offers advanced uncertainty modeling capabilities for point computations, geometry fits and network alignment solutions.

AUTOMATION

The Measurement Plan and SA SDK functions can add a significant layer of automation to your processes.

Generating simple to complex scripts can greatly improve workflow and productivity, reducing analysis time from days to minutes while eliminating errors and saving significant resources.

PIPE FITTING

The Pipe Fitting function is used in large piping applications when precision cuts must be determined.

Precision measurements and optimization greatly improve accuracy, ultimately reducing the cost of rework.

DATABASE OUTPUT

Database output is an additional reporting feature that gives you the ability to transfer SA data to databases and track projects/data long-term.

SA Analysis

SA Analysis offers a reduced cost alternative to the SA Ultimate license. It includes all of the features of SA Ultimate but does not offer the ability to connect to instruments. This creates an ideal license for post processing and analysis applications.

SA Machine

SA Machine contains all features of SA Ultimate, and also allows users to interface with robots and CNC machines. This permits calibration and real-time compensation of robots and large-volume CNC machines to high degrees of accuracy. It also enables robotic scripting, teach pendant behavior, and linear or joint space robotic control.

Feature Summary Matrix and Details

A selection of Package Fact Sheets and a Feature Matrix comparison can be found on our website here:

<https://kinematics.com/about/brochures.php>

System Requirements

SpatialAnalyzer is capable of running on most any modern computing platform with a full windows operating system. In order to run on a Mac a Windows environment must be installed such as using Boot Camp or something similar.

Minimum Requirements

A basic computer configuration that supports SA would include:

- Windows 10™ (64-bit) or later
- 8 GB RAM
- 5 GB free disk space for installation (more for your data files)
- 1600 x 900 screen resolution
- Basic support for your instruments drivers.

Recommendations for Optimal Performance

The system outlined below will provide a good base platform for using the most advanced features of SA. With this equipment measurement data may be gathered with high data rates (1 kHz sampling rates) into complex models containing numerous CAD surfaces.

- 3.5 GHz or faster eight-Core or more 64-bit processor
- Windows 10™ (64-bit)
- 32 GB system RAM or more
- 1920x1080 or higher screen resolution
- Hardware-accelerated graphics card supporting OpenGL®, and ideally an NVIDIA card that offers CUDA processing.
- 500 GB of free hard disk space

For large volume scanning applications or very large CAD models more memory may be desirable.

Check My Computer


SA provides a utility named *Check My Computer* that detects certain system configuration issues that may result in errors or problems while running SA. While this tool does not detect all issues—and while

some checks may fail even though you never experience problems in SA—this is a useful diagnostic tool to use when you *do* encounter a problem. This tool does the following:

- Checks for certain problematic software known to interfere with certain aspects of SA operation.
- Verifies that a default printer is installed.
- Checks for TCP/IP availability (a network interface).
- Checks firewall settings.
- Ensures that your computer's network interface provides support for UDP broadcasting.
- Ensures that your computer's network interface provides support for socket communication and that it is not blocked.
- Checks your User Account Control settings (Windows 7 and later) to ensure they're not set too high.
- Provides convenient access to system information and network configuration settings.

If you encounter an issue running SA, it is recommended to run this utility to check for problems and to contact support if necessary to find a resolution.

To Run the Check My Computer Utility

- From SA, Select **Help>Check My Computer**, or
- From the Windows Start menu, in the SA install folder, select the **Check My Computer** entry.
- In the Main toolbar, click the SA Network  icon.

Installation

Installing SA

The SA Installer will guide you through installing the following:

- SpatialAnalyzer™ Application
- Documentation (this document)
- Hardware lock driver


The installer is included with your provided USB flash drive and is also available for download at www.kinematics.com/download.

To Install SpatialAnalyzer:

1. Double-click SpatialAnalyzer 2024.XX.XX-Installer.exe.
2. Choose a destination folder (the default folder is usually pref-

erable), then click **Next**.

3. Leave both components (hardware driver & SA application) checked unless you have a specific reason to deselect a component. (For instance, if you are an OEM running a software lock, you would not need to install the Sentinel Hardware Driver). Click **Next**.
4. Choose additional install options. The Check My Computer utility will check your system for some common configuration issues that often cause problems with instrument communication in SA. When done selecting options, click **Install**.

 **Note:** Updated SA versions may require a new version of the hardware lock driver, so it is best to leave that option checked. Older drivers will be updated, if necessary.

SA is continuously updated with improvements and new features, and new beta releases are posted frequently. If your maintenance supports upgrading, you are encouraged to download the latest version and explore the newest functionality.

Installer Command-Line Options

The SA Installer supports several command-line options. These are typically most useful for IT departments that wish to do a silent or automated install. The most command-line options can be combined and are as follows:

- **/?**. Lists the command line options.
- **/S**. Performs a silent install (this must be a CAPTIAL S). The installer will operate in the background with no user intervention.
- **/CREATESHORTCUT**. Creates a desktop shortcut when run in silent mode.
- **/TEMPUNITS=[Celsius, Fahrenheit]**. Sets the default temperature units for the install. If omitted, Fahrenheit will be used as a default, unless SA was installed previously, in which the temperature units will not change. Ex. usage: **/TEMPUNITS=Celsius**.
- **/LENGTHUNITS=[Inches, Feet, Millimeters, Centimeters, Meters]**. Sets the default length units for the install. If omitted, inches will be used as a default, unless SA was installed previously, in which the length units will not change. Ex. usage: **/LENGTHUNITS=Millimeters**.

To Install SA with Command-Line Options:

- At the command line, first use CD to set the working directory to where the SA installer resides, such as the downloads folder.
- Enter the executable name followed by the options desired. For instance, the following installs SpatialAnalyzer silently in the background while also creating a desktop shortcut and

setting the length units to millimeters: “SpatialAnalyzer 2021.2.71490.1-installer.exe” /S /CREATESHORTCUT /LENGTHUNITS=Millimeters

To Check for SA Updates:

- Select **Help>Check for SA Updates**,
- Choose **Help>SpatialAnalyzer License Management>View Maintenance Support and Subscription Status**, then click the **Check for Updates** button, or
- Navigate to <http://www.kinematics.com/products/spatialanalyzer/download-software.html>.

Instrument Drivers

Each measurement device (instrument) supported within SA has different driver and initial configuration requirements. Details on setting up an instrument can be found in the **Instruments Manual** built into SA and accessible under the Help menu in the ribbon. They can also be found by searching for a particular device by name in the SA community :

<https://kinematics.my.site.com/SA/s/>

Required drivers can be downloaded directly from our webpage here:

<https://kinematics.com/download/downloadindex.php>

Licensing

SpatialAnalyzer now offers both USB and Cloud Licensing. To determine which one is right for you please consult with your local sales representative.

Traditional USB Dongle Licensing

With this option SpatialAnalyzer is licensed via a USB hardware lock. A hardware lock driver (Sentinel), used to read the USB key, must be installed before using SpatialAnalyzer. This driver is provided as part of the software install.


Meta-key File Installation

USB Licensing uses Meta files (extension .met) to upgrade or extend a license Maintenance. A new USB License will have no need for a Meta file since the key contains the initial Meta file. When software licensing is extended, or renewed, a user will receive a Meta file. All the user must do is double click the Meta files(s) when received and they will auto extract into the correct directory. Should an issue occur or you need to retrieve the file, it can be manually placed / found at: C: Analyzer Data\License.

Meta-keys have filenames like "License12345.met", where the 5-digit number must match the serial number on the key.

License files can also be sent with the *.mez extension. This is a zip file that can contain more than 1 license. SA automatically recognizes both *.met and *.mez extensions so all you must do to install either format is double click on the file.

To install an SA Meta-key File:

 **Note:** If a hardware lock is moved to a different computer, the meta-key file must be transferred with it by copying the file into the new computer's Analyzer Data\License folder. Failure to do this will result in the hardware lock reverting to its original license settings.

1. Double-click the *.mez or *.met file you received from NRK and SA will automatically install it.
2. Ensure that the hardware lock with the matching serial number is installed in the USB port.
3. Run SpatialAnalyzer.
4. Choose the [Help>SpatialAnalyzer License Management>View Current License Information](#) menu option to verify your license.

Viewing Your License Information

Occasionally it may be necessary to examine the details of your SpatialAnalyzer license, such as the serial number, user and company, upgrade date, and available software components. You can view the details of the current license in use at any time.

To View the Current License Information

1. From the SA menu, select **Help>SpatialAnalyzer License Management>View Current License Information**.
2. A window will be displayed showing your license details, and this text will be automatically copied to the Windows clipboard.



A Closer Look

When SA starts up, it first looks for a hardware lock. It retrieves the license information and serial number from the hardware lock, then checks your `Analyzer Data\License` folder to see if there is a meta-key file matching the hardware key's serial number. If found, it uses the meta-key file to update the license information from the lock.

Hexagon Universal License Library (HULL)

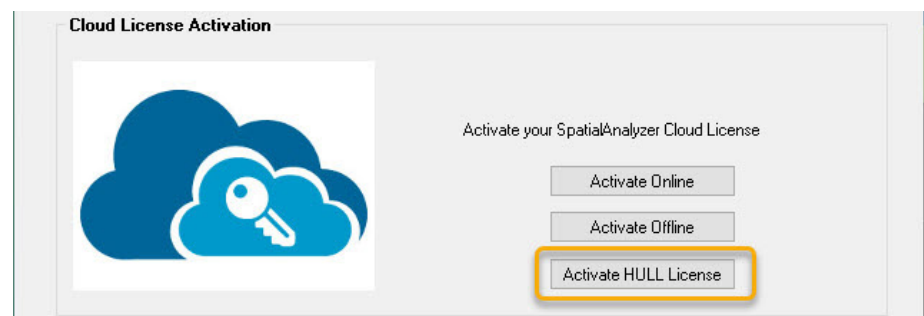
Starting in 2024.1 a new license method is now supported. This licensing system requires the installation of the HLM applications available for download from our ftp site here:

<https://www.kinematics.com/ftp/HLM/>

Once installed all licensing and configuration is taken care of through the HULL License Manager application. An “Entitlement” (cloud code) needs to be issued to open access to SA and its features.

When you start SA without a license an additional option will be presented which will directly open the HULL license Manager if installed where you can activate your license (Figure 2-1):

Figure 2-1. Accessing the HULL License Manager Application to register a license.



Select Activate Entitlement and enter the code provided by your salesman (Figure 2-2).

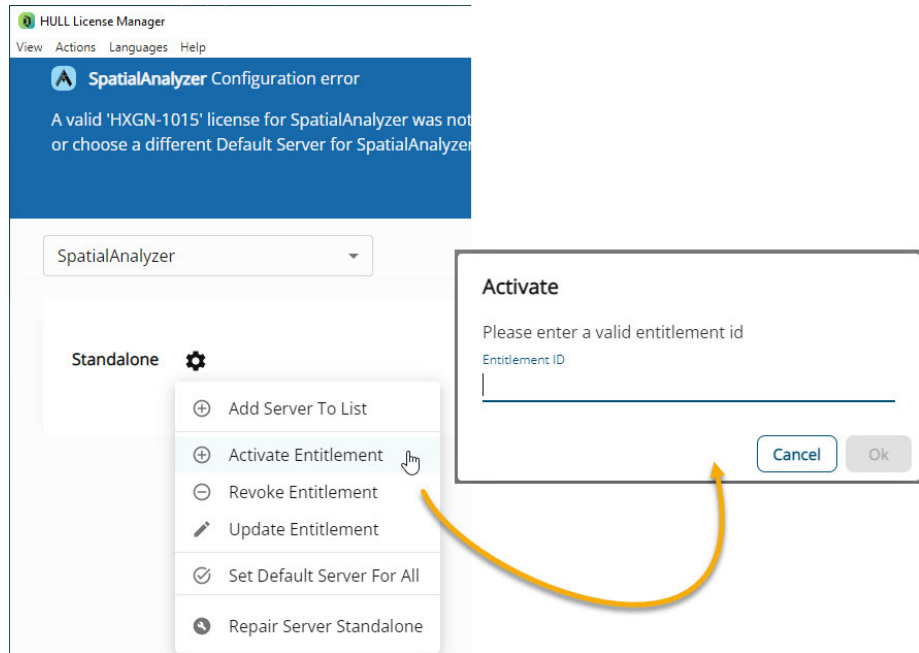



Figure 2-2. Activating an Entitlement within HULL License Manager.

The license manager will then access the cloud and register the license. A standalone license will then become available for use (Figure 2-3).

Standalone 

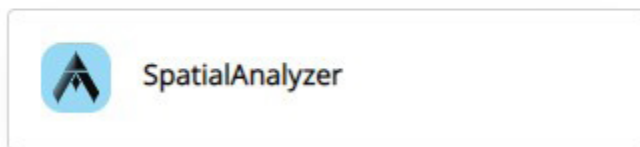
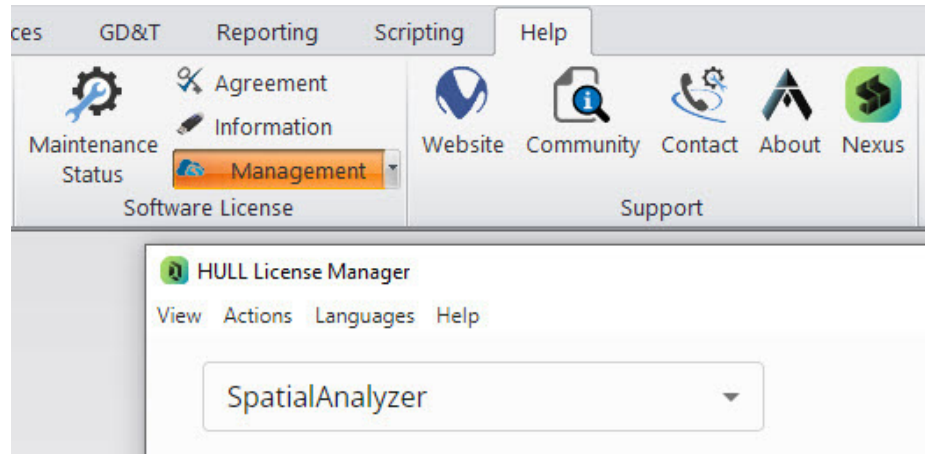


Figure 2-3. Available SA License.

Once a valid license exists it can be accessed from both the cloud Management button and the About buttons in the Help tab of the Ribbon Menu (Figure 2-4).

Figure 2-4. Accessing HULL license information from the Management button.



When you start SA it will automatically look to see if you have a valid HULL license and then check if additional licensing is available.

Contract Hexagon support for more information and current documentation regarding setting up and configuring licenses through the Hexagon License Manager application. Several configurations and network server options are available.

Nalpeiron Cloud Licenses

Cloud licenses provide a number of advantages over USB Dongles. They can easily be activated and deactivated on-line providing an easy way to share a single license with colleges in different facilities and they are not as prone to being lost or damaged. Cloud licenses use a license code consisting of a string of numbers and letters that can be used to activate SA on a particular machine.

On-line Cloud License Installation

To install a cloud license simply open SA. If no license is available you will see the following activation as an option on screen (Figure 2-5):

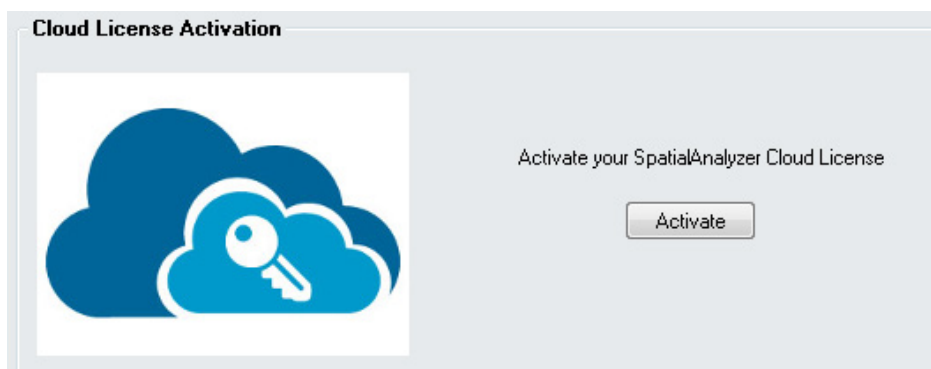
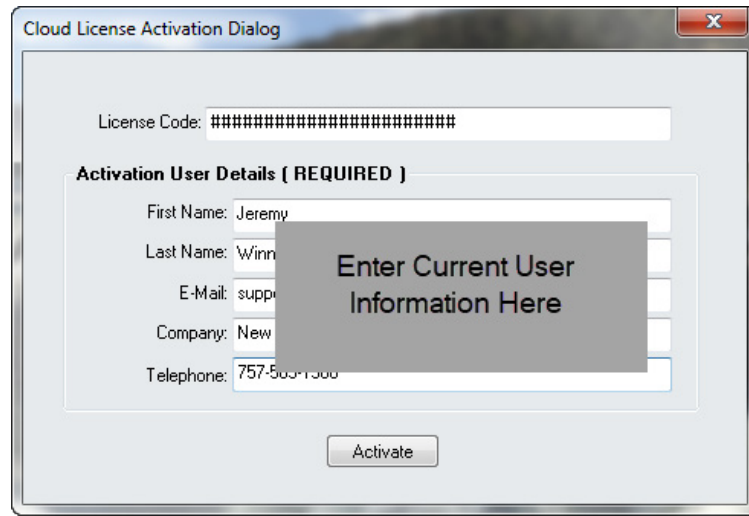


Figure 2-5. Cloud License Activation

1. Select **Activate** and enter the Cloud License number you were provided along with your User Details (Figure 2-6). Be sure to enter the code exactly (it is case sensitive). It can be helpful to copy and paste the string directly out of an email or text file.

The activation can only be performed with an active internet connection.

Figure 2-6. Entering Cloud License Information



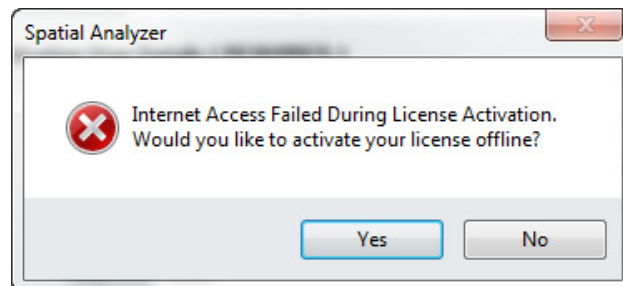
To check on the status of your license or to de-activate the license on a particular computer go to **Help > SpatialAnalyzer License Management > Cloud License Management**.

Offline Activation/Deactivation:

Many customer will not have direct internet access. License activation and de-activation can be still be accomplished by transferring a license certificate to a computer with access. To do so follow these steps:

1. Attempt activation (as described above) and you will see the following message(Figure 2-7):

Figure 2-7. No Internet Connection



2. Press **YES** to activate your license offline. This will bring up the following Cloud License Offline Activation Dialog with your current license information(Figure 2-8):

Cloud License Offline Activation Dialog

License Code: #####

Activation User Details (REQUIRED)

First Name: Jeremu
 Last Name: W
 E-Mail: jer
 Company: NF
 Telephone: 757-564-2519

1. Fill Out Above Fields and Generate Offline Certificate

Generate Activation Certificate

2. Save Certificate and Take to Computer with Internet Access

f5e4b8715754d47c18c31037703edf0c5a5a64f064b1be3db2400a04a9d62acf666e
 b809d05167b793b54366fd6fb4433f043295998b8d7ed5a8dc5a46e71f2c0f98703e
 d4bbb9d76e225e1a50ad7205ea477ddd72ae6a0c674ed399f0749c7e628087ae10
 74925c4020300cdc77be353388533944ab71afddb34608b2d45885a3596dfa8165f

<http://kinematics.com/spatialanalyzer/cloud.php>

3. Paste License Activation Response Certificate from Internet Computer Below

4. Press Activate for Certificate from Step 3

Activate Certificate

Figure 2-8. Offline Certificate Generation

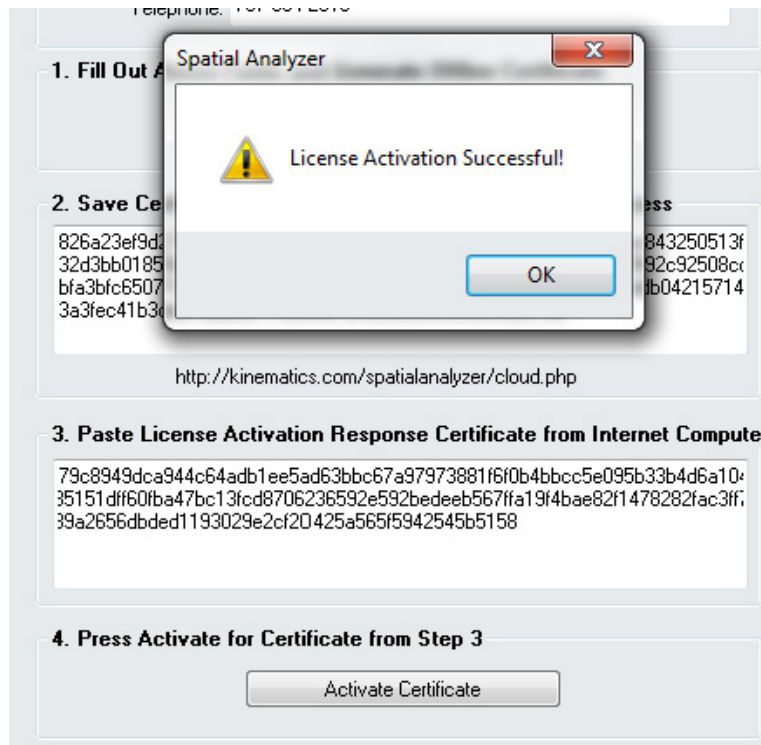
3. Select the *Entire* Certificate and copy it to a *.txt file or word document that can be saved and transferred to another computer. Then take this file to another computer with internet access.
4. Browse to our Cloud License Activation Site here: <http://kinematics.com/spatialanalyzer/cloud.php>

Figure 2-9. Cloud license offline registration site

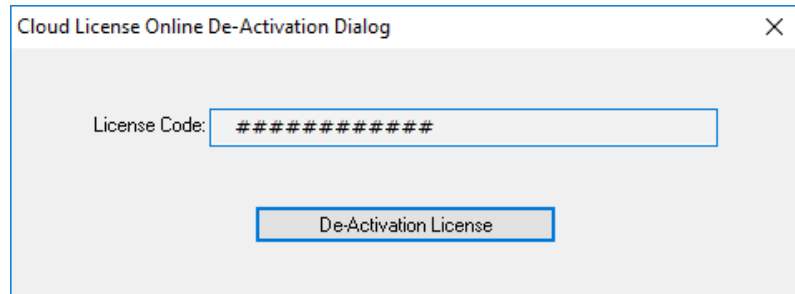


5. Paste the Activation Certificate into the site and press **Activate**. This will generate a license Certificate that can then be transferred back to the offline computer.
6. Once this License Certificate is entered into the Offline Cloud Activation Dialog the cloud license should be successfully activate for that machine (Figure 2-10):

Figure 2-10. License Activation Successful.



The deactivation process is the same. You will need to go to **Help>Spatial Analyzer License Management> Cloud License Management** and select **Deactivate** (Figure 2-11):

Figure 2-11. Deactivate Process

The same code generation will process will then follow and the de-activation can be done with the use of an activation certificate online as well.

Checking License Status

You may also wish to view your maintenance subscription information, which will indicate the date software maintenance will expire.

To View Your Maintenance Subscription Information:

- From the menu, select **Help > SpatialAnalyzer License Management > View Maintenance and Support Subscription Status**.

To Request Renewal for Your Maintenance and Support Subscription:

- Select **Help > SpatialAnalyzer License Management > Request Maintenance and Support Renewal**, or
- Choose **Help > SpatialAnalyzer License Management > View Maintenance and Support Subscription Status**, then click the **Request Upgrade/Renewal** button.

Certification

SpatialAnalyzer has been certified by the National Institute of Standards and Technology (NIST), the Physikalisch-Technische Bundesanstalt (PTB), and the National Physical Laboratory (NPL).

How to Get Help

SA Community Webpages

The SA Community webpage provides an index of key word searchable help articles, many of which are also linked to helpful YouTube videos. Its a great interactive place to start looking for basic information on SA and how to get started using it. It can be accessed using this URL:

<https://kinematics.force.com/SA/s/>

Documentation

If you need help, the definitive source of information on SA is the *SA User Manual* (this document). The *Instruments Manual* and *MP Command Reference* have been separated for clarity.

This document is in PDF format and is fully searchable.

To Access the SpatialAnalyzer User Manual do one of the following:

Click the User Manual icon  in the *Main Toolbar*,

Select **Help>User Manual**, or

In the Windows Start Menu, navigate to the SA install folder, then the *Documentation* subfolder, and select the SpatialAnalyzer User Manual link.


Every SA Release has a complete Readme file documenting every change to the software, organized by software version. This is often a useful source of information, particularly with determining instrument driver compatibility with a given SA version.

To View the SA Readme File:

- Select **Help>View the SA Read Me File**, or
- From the Windows Start Menu, navigate to the SA install folder, then the *Documentation* subfolder, and select the Release Notes link.

Customer Support

If you need help setting up or using SpatialAnalyzer, there are several ways that you can reach customer support:

 **Tip:** We recommend against contacting support personnel directly via their e-mail address or phone number. Doing so may result in a delayed support response, as the support individual may be out of the office or otherwise unreachable.


- **E-mail.** support@kinematics.com
- **Telephone.** Dial +1 (757) 565-1500 and choose the support option (option 1). Be aware that this will transfer you directly to a voice mailbox, but your call will be returned promptly.
- **Web.** Navigate to www.kinematics.com and choose the support link, or from within SA, select **Help>Visit NRK Website!**

Regardless of how you contact us, we try to respond to all support requests in less than 24 hours.

Context-Sensitive Help

SA has a context-sensitive help system wherein you can click on menu items, interface components, or select dialogs in order to be directed to that topic in this User Manual.

To Initiate Context-Sensitive Help:

1. In SA, press **Shift + F1**. The mouse cursor icon will change  to indicate that you are now in context-sensitive help mode.
2. Click on a menu item, toolbar item, or interface item to be directed to the topic in the user manual.

Some dialogs have a question mark in their upper-right corner. Click this question mark to be directed to the appropriate topic in the documentation.

Tutorials & Sample Files

SA comes with a variety of sample files to demonstrate some of the key features of SA. Most of these sample files are used in the Tutorials chapter toward the end of this manual.

To Open a Sample File:

From the menu, select **Help>Open Sample SA Files**.

Navigating this Users Manual

This users manual is quite large and detailed and it can be difficult to know where to start. The following is a grouping of chapter headings and a brief synopsis of what is contained within each group. The idea is to help you know where to go in order to find the information that you need.

An overview of fundamental concepts and guiding principles

SA Fundamentals

This chapter contains a select of fundamental principles that are considered central to the operation of the software. These include subjects such as:

- The architecture of how instruments are connected
- The basic definition of an object
- How measurements are linked to instruments
- How and when measurements are compensated
- Understanding the working frame

Getting into the details of operation

Exploring the User Interface

This chapter provides a review of the basic tool bars and how they are used to manage the data contained in a measurement job.

The Graphical View

As a graphical software, the ability to effectively manage and work with points, objects and surfaces in 3D space is essential. This chapter provides the tools to work effectively and report features contained in the graphical view.

Working in SA

The majority of basic operations can be found in this chapter. Its a catch all for general operation.

Working With Files

A big part of measurement is managing job files. This section includes important subjects such as:

- Creating new files
- Import and export operations
- Backups and restore points
- CAD formats

The core of measurement - Points and feature measurement

Points

Points are the basis from which virtually all analysis is performed in portable metrology. This chapter goes into depth on how points are architected from observations, how their properties are managed, and various ways to build points and use them for comparison.

Objects

While points are the key to measurement those measurements need to be used for analysis of features such as geometric objects. The basic geometry types, their inherent properties and construction is reviewed in this chapter (see [Analysis](#) for geometry fitting).

Working with Surfaces

Analysis is typically either a comparison of points to a regular geometric shape or a CAD model or surface. This chapter provides some fundamental concepts necessary for working with CAD models and surfaces.

Working with Instruments - SA Instruments Manual

Instruments

An instrument model and the measurements associated with it are saved within an SA Job. The functions essential to managing these instruments are included in this chapter and include:

- Instrument properties
- Scaling an instrument
- Instrument uncertainties

This chapter also provides an overview of how to connect to a live instrument

- Adding Instruments
- Overview of Instrument interfaces
- Connection methods
- Managing IP based instruments
- Measurement simulation

Measuring with Laser Trackers

Details on the laser tracker interface and its operation. Quick-Start information for specific trackers can be found at the end of this chapter.

Measuring with Arms

Details on the PCMM arm interface and its operation. Quick-Start in-

formation for specific arms can be found at the end of this chapter.

Measuring with Total Stations

Details on the Theodolite Manager and both theodolite and total station interfaces and their operation. Quick-Start information for specific theodolites and total stations can be found at the end of this chapter.

Measuring with Photogrammetry

Details on photogrammetry interfaces and their operation. Quick-Start information for specific photogrammetry devices can be found at the end of this chapter.

Measuring with the Laser Radar

Details on the Laser Radar interface and its operation.

Measuring with Room Scanners

Details on the room scanner interfaces and their operation. Quick-Start information for specific room scanners can be found at the end of this chapter.

Measuring Operating Laser Projectors

Details on the Laser Projector interfaces and their operation. Quick-Start information for specific photogrammetry devices can be found at the end of this chapter.

How to effectively move and align measurements for analysis

Alignment and transformations

This chapter provides a list of possible alignment options and the steps to conduct these alignments.

Live view and build it functionality

Watch Windows

Watch windows provide the ability to get a live feedback on a point or objects position relative to any other point or object in a job. This chapter discusses how.

Analysis, inspection refined alignment operations

Analysis

The fundamentals of geometry fitting, point offset compensation, and error visualization are presented here.

Relationships

Relationships are dynamic links between points and objects. They

can be used to build dynamic geometry, continually updating vector groups and for the most advanced of alignment operations.

GR-Feature Inspection

Geometry Relationships (GR-Features) are very powerful tools and have become the centerpiece for dynamic inspection. Feature Inspection provides both quick access to important menu features and a streamlined process flow.

GD&T

Standardized inspection is a separate analysis process intended to be true to the ASME or ISO standard of your choice.

Dimensions & Callouts

Dimensions and callouts provide the tools you will need to annotate the graphics both for operator feedback and reporting.

Reporting

This chapter is dedicated to report creation and editing.

Measurement Uncertainty and Unified Spatial Metrology Networks

Measurement Uncertainty

Review of uncertainty analysis

USMN

Advanced uncertainty based network alignments.

Scripting and Automation

Measurement Plan

SA provides a powerful scripting environment called Measurement Plan for easy automation of just about any task that can be performed.

Also refer to the [MP Command Reference](#) for details on each available MP command.

SA SDK

SDK provides a means by which to write custom applications that utilize Measurement Plan functionality within Visual C++, VB.NET, or other development environments that support ActiveX controls.

Specialized functionality

The OPC Interface

OPC is a communication standard that enables various devices in a manufacturing environment to communicate among themselves. SA Ultimate provides OPC data server functionality.

Machine Calibration

SpatialAnalyzer has the unique ability to integrate portable metrology measurements for the calculation of true metrology-compensated robot and CNC machine calibrations.

References

Tutorials

This chapter contains an assortment of tutorials covering SA's basic functionality.

Keyboard Shortcuts

A printable page of useful keyboard shortcuts.

Math References

A selection of useful references for basic calculation processes carried out within SA.

End Users License Agreement

The SA License Agreement is available in each SA version under the Help Menu.

It is important to have a good understanding of SA's "modus operandi". Having a good fundamental understanding of how SA operates will prove to be extremely useful and will make you a more effective user of the software. In this section, we'll describe some of the fundamental principles surrounding SA and its use.

Installation and Configuration

Installing Spatial Analyzer (SA)

SA is a Windows only application. To run SA on a mac a windows compatibility environment must be run through Boot Camp or a similar application.

The current version of SA, as well as a beta testing version, can be downloaded from the kinematics.com webpage here:

<https://kinematics.com/download/downloadindex.php>

To install the application double click on the SpatialAnalyzer####.exe file and follow the directions.

Displaying Ribbon Menus

Starting in 2019 a ribbon menu was added to SA. This ribbon menu was designed to incorporate the best of the classic menus and the Toolkit menu and streamline SA workflow. The display of the ribbon menu is optional and enabled with a check box option on the Display tab of the Users Options "**Show Ribbon Bar**" (Figure 3-21):

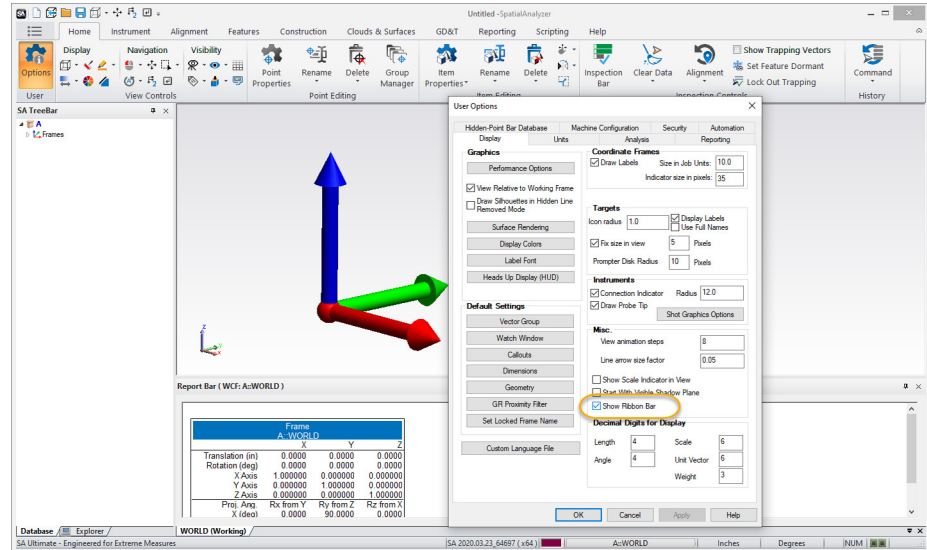


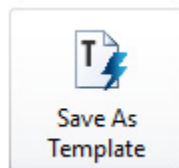
Figure 3-1. Default SA Window. The On/Off control for the Ribbon menu is found in the User Options on the Display Tab.

The following tutorials will use the Ribbon menu display. Buttons within the tabs are referenced as [Tab Name]>[Section Name]>[Button Name]>[Sub Button Name]. So as an example, you can rename a point by going to the **Home**(the tab to select)>**Point Editing** (ribbon section to look in)>**Rename** (the large icon). You can also rename many points using a naming patter by going one level deeper **Home>Point Editing>Rename>Points using a Name Pattern** , indicating to use the drop down button under Rename and select from within.

Customizing New SA Job Files

Much of SA can be customized. Some basic settings are saved in the registry and will persist when SA is closed and reopened. However, much of the settings are saved within a job file. To customize how a new job file looks you can save a template by doing the following:

1. Configure a job file such that it has the settings you would like to see when SA opens.
2. Select File>Save as Read-Only SA Template (Figure 3-2) and save the file within the C:\Analyser Data\Templates directory as "default.xit64".



Save as Read-Only SA Template

Used to define your current job as a read-only template. Templates provide a great way to setup a job file for a particular task or application customized for that purpose.

Figure 3-2. Saving a Job File as a Read Only Template

Additional template files can be saved and opened as needed but the default template is used for newly created job files automatically.

The Working Frame

Measurement is a process by which the location of something of interest is determined with respect to a reference of some kind. This can be to its self, for example when measuring the width of something (where you measure how far one side is from the other), but typically it is with respect to a base reference. In order to define both the position and orientation of an object you need both a base reference point and orientation. A coordinate system or coordinate frame is a common way to report this information and is a fundamental piece of any computer aided graphical design or inspection software.

When you start a new empty SA job file there will only be one item in the file (both the tree and the graphics), which is the “WORLD” frame. This is the base coordinate frame for the job file (Figure 3-3).

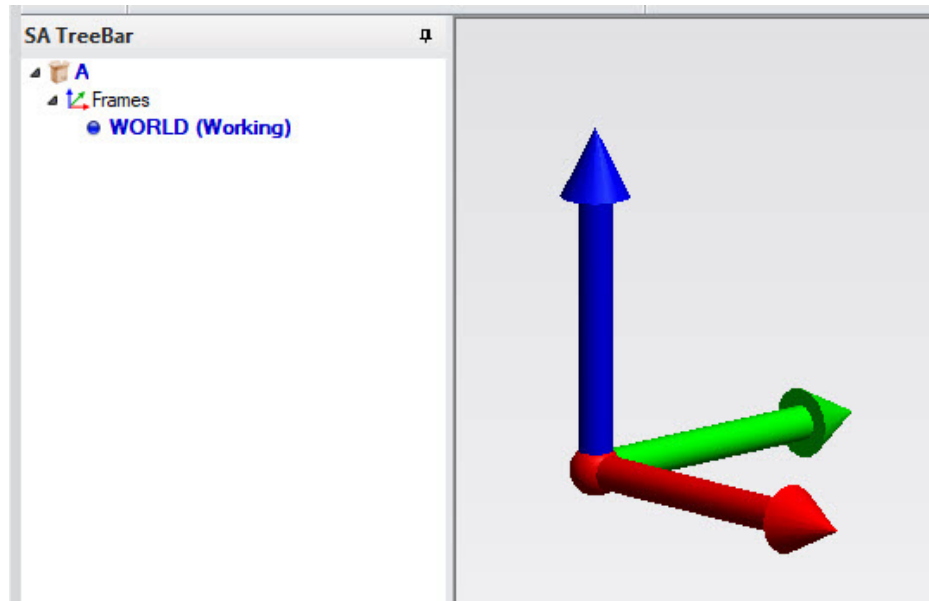


Figure 3-3. The base coordinate system for a new SA job file is the WORLD frame, which is also set to “Working” by default.

SA’s fundamental math comes from physics and engineering where a right-handed Cartesian coordinate system is a standard practice. This means that every point in a job file will have an X,Y, Z location with respect to the reporting frame. Meaning that its a given distance out along the X axis, the Y axis and Z axis from this common reference.

SA has a concept known as the *working frame*, which is also marked in the tree as (Working). The working frame is a coordinate frame through which all data is represented or mapped. Put another way, it is the active coordinate frame.

Many coordinate frames can be defined in a single job file and any coordinate frame can be made the working frame. When a frame is working, all data (point coordinates, delta values, rotations) are rep-

resented in SA with respect to that frame unless specified otherwise.

It is important to recognize that changing a coordinate frame does not physically move anything in space—it merely results in a change in the reference system used to represent that same position and orientation. This allows you to take measurements in any arbitrary default coordinate system and represent them in a system that better describes the data. For more information on Frames refer to “[Frames](#)” on page 253.



A Closer Look

You can assign a frame of reference to a watch window in order to see deviations represented in a given frame regardless of the working frame. You can also specify a reporting frame for an object in order to view that object’s properties on a report with respect to a specific coordinate frame. (Those values will still be represented outside of reports with respect to the working coordinate frame. That is, if you assign a reporting frame to a point group, the coordinates will be represented in the active coordinate frame unless viewing them in a report).

The Default Collection


Collections are organizational elements used to contain entities in a hierarchy in SA. Just as there is always exactly one working frame, there is always exactly one default collection. The default collection (colloquially referred to as the working collection) is the organizational container inside which all newly created entities are placed, unless specified otherwise. For example, if a circle is created as the result of a geometry fitting operation, that circle will be placed into the default collection.

Measured Points vs. Cloud Points


It is important to understand the distinction between measured points and cloud points, and to know when it is more appropriate to use one instead of the other when taking measurements. Both points and clouds are discussed in more detail in later chapters—however in this section we will discuss the broad differences between the two entities.

Measured points carry with them a relatively large amount of overhead. They are not simply an XYZ coordinate with a name, but actually store quite a bit of information. As described in “[Traceability](#)” on page 37, measured points store metadata in addition to the measurement itself and its offsets. Weather parameters (temperature, pressure, and humidity), RMS data, measurement mode information, timestamps, and other information are all stored with a point for traceability and documentation purposes. While this information is useful and great to have, it also means that a measured point puts

a relatively large strain on the data processing and computing capabilities of the machine. Storing and handling that wealth of data can become relatively stressful on a computer's resources if there is a very large number of points.

 **Note:** All points in a cloud should be measured with the same offset. If you change probes and the offset changes, put those points into a different cloud.

Cloud points, on the other hand, are designed for very large quantities of measurements and are optimized to be much more lightweight from a computer resources perspective. Cloud points only store an XYZ value. They have no metadata and no individual names or offsets, and therefore use the smallest possible computer resources. Instead, the point cloud itself (which is the container for the cloud points) stores the offsets for the measurements. Although cloud points (collectively known as a cloud) are typically created from scanning devices, they can be created by discrete measurements from instruments like laser trackers and portable arms as well.

 **Tip:** For best performance, measure into a cloud whenever recording a large number of points (>10,000).

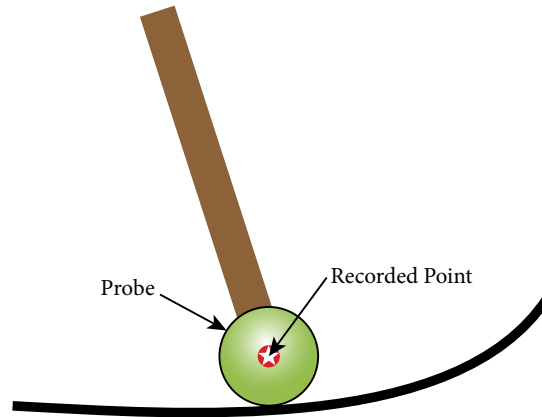
So the question to answer is: when should a measurement be taken as a regular measured point, and when should it be recorded as a cloud point? The general rule of thumb is that if you are performing any type of surface scanning consisting of a large (>10,000) number of points, you should use point clouds. Note that nothing in this rule of thumb refers to the type of instrument being used. That is—whether or not you are using an actual scanner is of no consequence. If you are “scrubbing” a surface with an SMR and scanning with a laser tracker, for instance, that is considered scanning and point clouds should be used.

Point Offset Compensation

When storing measurements, SA always records the point representing the center of the probe (SMR, ruby tip, etc). Unless non-contact or zero-offset measurement is being performed (such as measurement with a laser radar, scanner, point probe, or photogrammetry), the recorded point will not represent the point where the probe touched the measured surface, but instead will be at its center (see [Figure 3-4](#)). (For zero-offset measurement, the recorded point is at the point of contact). What this means, of course, is that the distance between the measured point and the measured surface is not zero, but is instead equal to the probe radius (also known as the planar offset).

In order to properly compensate for probe offsets, two values must be known: the probe offset, and the surface direction. These two properties are discussed in the following sections.

Figure 3-4. The recorded point for any measurement is always at the probe center.

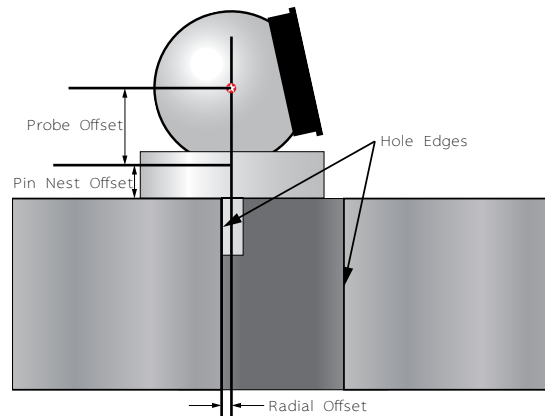


Probe Offsets

When a measurement is recorded, SA stores not only the probe center point, but also the corresponding offsets. There are actually two offsets used: a planar offset and a radial offset. The planar offset is an offset perpendicular (or normal) to the resulting geometry, whereas the radial offset is an offset in the plane of the geometry.

The planar offset is used to offset the measured point toward a surface, line, or curve. The radial offset is used to offset measurements inward or outward when measuring or fitting circles and ellipses using tooling that introduces a radial offset, such as a pin nest (see [Figure 3-5](#)).

Figure 3-5. The radial offset is used for shifting measurements inward or outward when using a pin nest.



If just a spherical probe is used (such as a ruby tip or an SMR), the radial offset is ignored (it is never used). If additional target tooling is used—such as the pin nest in [Figure 3-5](#), the offsets attributable to the additional tooling are included in the point's offset. For instance, the pin nest pictured introduces 0.25" of additional planar offset and 0.125" of radial offset on top of the SMR's 0.75" planar offset. This yields a total planar offset of 1.0" and a total radial offset of 0.125".

Whenever a measured point is compared to an object such as a CAD

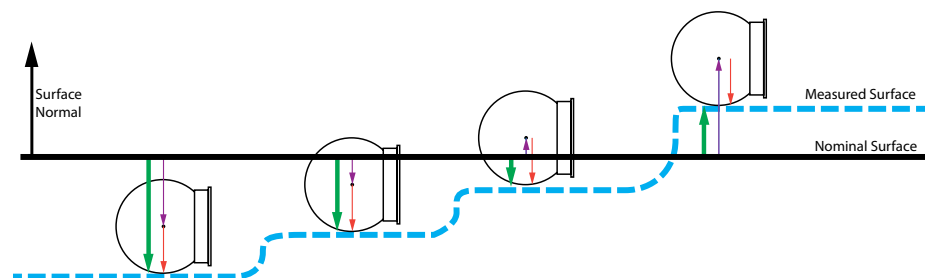
surface, line, or curve, probe offset compensation is automatically applied, and the reported deviation is actually the deviation between the point of contact and the corresponding closest point on the object of interest. Likewise, whenever geometry is created from measured points, the resulting geometry is automatically offset from the measured points, and the resulting geometry represents the surface actually measured.

Obviously, in order for SA to compensate for offsets correctly, the correct offsets must be stored with the measured points. Offsets should be set in the appropriate instrument interface whenever target tooling is changed, and as a result measured points should have the correct offsets as they are measured. However, if you forget to set the proper offsets when measuring, you can always correct them after the fact by changing the offsets on measured points. (More information on this can be found in the Measurement section).

Surface Normal Direction

In addition to having the proper stored offsets, one additional factor is required to be correct in order to perform a properly compensated analysis: it is critical that the direction of the surface normals being used in any comparison are facing the proper direction. Consider the example in [Figure 3-6](#), in which four measurements were taken with an SMR at four positions on a measured surface (the blue dashed surface in the illustration). Note that, for clarity, the as-built error in the measured surface is greatly exaggerated here.

Figure 3-6. Proper offset compensation is applied when the surface normal faces the side to be measured.

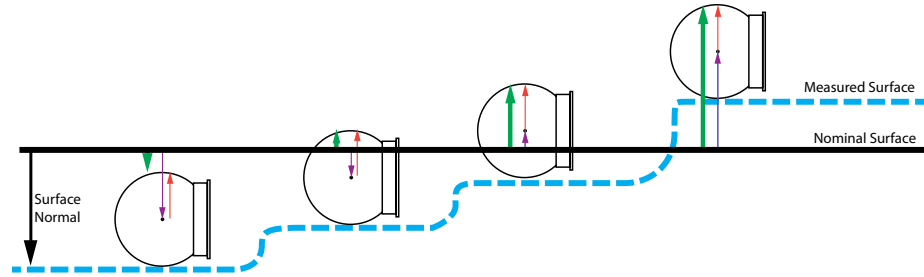


Suppose we wish to compare our measurements to the nominal CAD part in order to determine whether the as-built part is “high” or “low” of the ideal design. When comparing the measured point to the nominal surface, SA first calculates the deviation from the nominal surface to the recorded point (the purple vector), and then adds the probe offset in the direction opposite to the surface normal (the red vector in the illustration). The resulting vector (green) indicates the deviation between the measured surface and the nominal surface. In this example, the first three measurements are low of nominal, and the last measurement is high of nominal.

To visualize how an incorrect surface normal direction can result in

incorrect results, suppose that the surface normal in [Figure 3-6](#) is accidentally reversed and is now facing the wrong direction ([Figure 3-7](#)), with all else being the same. The deviation from the nominal surface to the measured point stays the same (purple vector). Again, SA adds the probe offset in the direction opposite to the surface normal, but since the surface normal is facing the wrong direction, the offset compensation is performed in the wrong direction. As a result (green vectors), the reported deviation is off by exactly the probe diameter—a very large error!

Figure 3-7. Offset compensation applied when a surface's normal is facing the wrong direction. This will result in incorrect results!



This behavior leads to a golden rule in SA—one that is referred to in various places in the SA dialogs. The rule is that in order for deviations to be reported properly, all surface normals must face the side being measured. Put another way, the surface normal direction for the surfaces being compared should point to the side from which you're approaching the part with the probe (the other direction should point into the part's "material").

Warning: If surface normals are facing the wrong direction, the calculated deviation will typically be in error by exactly the probe diameter.

Warning: If stored target offsets are incorrect, the calculated deviation (or the location of geometry resulting from a geometry fit) will be incorrect.

There is one simple way to ensure that you comply with this rule: When working with surfaces, you want to ensure that the normals all point toward the "outside" of the object—that is, they should point away from the material of the part. If you imagine that your CAD part is submerged in the ocean, the surface normals should point to the side where the water is touching the surface. Upon importing CAD to be used in analysis, one of your first steps should be to ensure that the surface normal directions are all correct, and to correct any normals that point inward.

For information on identifying surfaces whose normals are facing the wrong direction, see the Surface Backsides section in "The Graphical View".

Comparing Measurements to Points and Objects

The fact that SA automatically compensates for probe offsets results in a few rules that should be kept in mind when performing comparisons between measured points, objects, and surfaces:

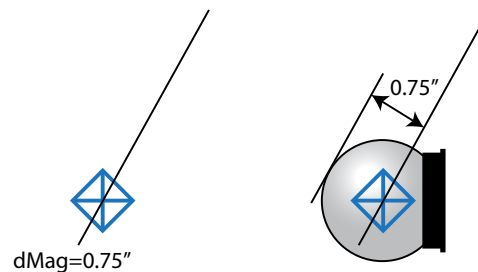
- When comparisons between points are performed, the comparison is always between centers. Point to point deltas and

best-fits, for example, always compare center to center. This leads to a corollary that if two separate points are to represent the same point in space (such as for a reference monument), their centers should be coincident. This might seem obvious, but if the same point is measured twice with probes that have different offsets, a physical target adapter is required to make one of the offsets match the other.

- When comparisons between points and objects or surfaces are performed, the comparison automatically compensates for probe offset.

There are a few cases where you will want to override this behavior and ignore offsets, however.

Figure 3-8. Comparing a measured point with 3/4" offset to a line.



For example, suppose a measured point with a 3/4" offset lies directly on a line. This would appear in SA as indicated at left in Figure 3-8. A normal comparison between the point and the line will indicate a deviation of 0.75" as depicted at right, because offset compensation is being applied. That is, you are 0.75" from touching the line. Typically, however, in this situation you want to compare the center of the probe to the axis. For instance, perhaps you've set the probe in a hole to take a measurement, and you want to compare the measurement to the nominal hole center. In that case, you'll want to ignore the offset so that this situation would indicate a deviation of zero. There are several ways to do this in SA, including overriding target offsets to zero or using commands that specifically ignore offsets.

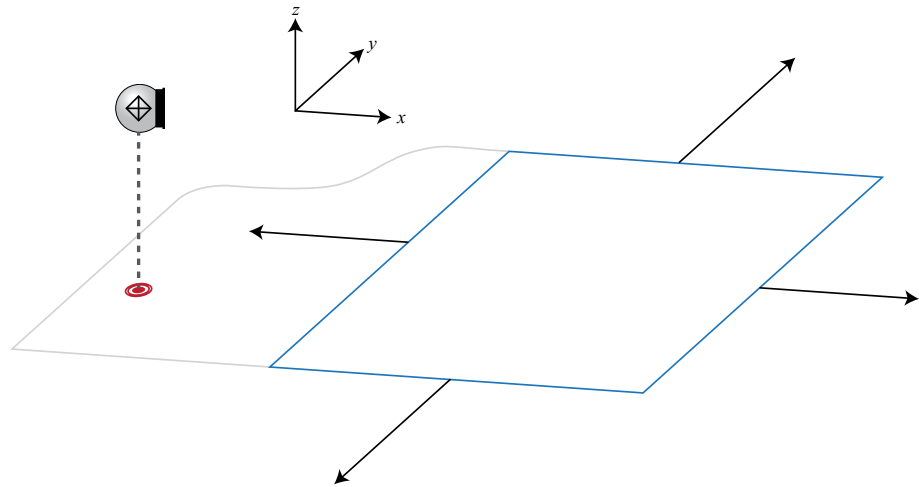
Unbounded Geometry

There are several primitive geometry types in SA that are unbounded. This means that the geometry extends infinitely without bound. For an example of this, refer to Figure 3-9.

Although a plane in SA is depicted in the graphical view with a boundary (blue line), this is only so that the plane is visible in the graphical view. From an analysis perspective, the plane extends infinitely in two dimensions, just as it might be defined in an academic math class. Keep in mind that any time you compare a point to unbounded primitive geometry, the geometry will behave as if it extends infinitely

along its unbounded edges. The following primitive geometry types are unbounded:

Figure 3-9. A plane in SA has a graphical depiction (blue lines) so that it can be seen, but from an analysis perspective it extends infinitely in two dimensions. When compared to a point beyond its graphical depiction, a plane is extended infinitely. The closest point to this measurement on the depicted plane is shown as a red dot above.



- Planes
- Lines
- Cylinders (lengthwise)
- Paraboloids
- Cones

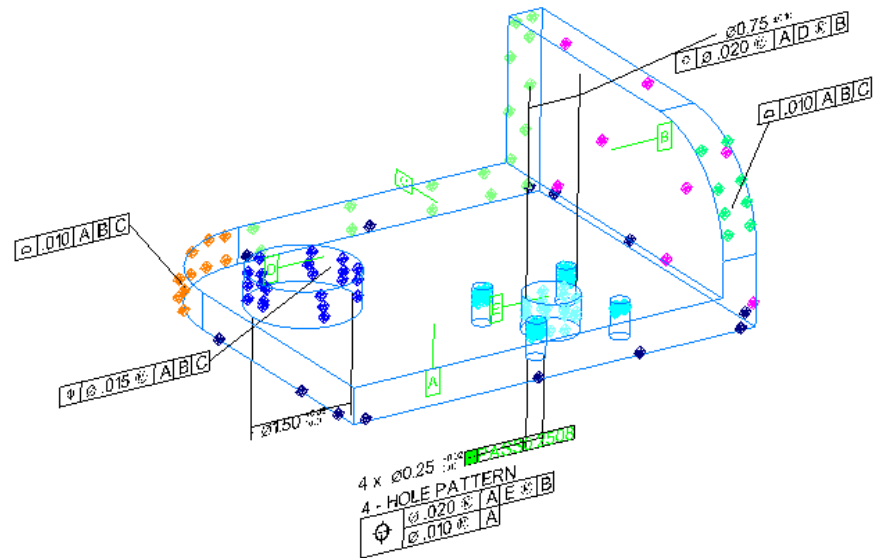
If you would like the geometry to be bounded for analysis purposes, it must be converted to a surface (see [“Working With Surfaces”](#) on page 821).

Feature Measurement

One important general rule to follow in SA is that each feature should be measured into its own point group when possible (see [Figure 3-10](#)). One intended meaning for the term *feature* in this case is a specific portion of a measured part. For instance, measurements on the inside of a hole should go into one group, measurements of the top surface of the part should go into another, and measurements of the side of the part should go into a third. When measuring features that represent geometric shapes such as planes, circles, cylinders or cones, consider using the dynamic features from the Feature tab of the ribbon (see [“Why Feature Based Inspection”](#) on page 489).

But another intended meaning of feature in this context is to refer to any set of related points. So, for instance, all points in a drift check should be measured into their own point group, and the reference points (fiducials) should be measured into another.

Figure 3-10. Proper feature-based organization of point groups. Measurements for each individual feature on this part are organized into separate point groups.



SA is designed around the assumption that features are measured into their own individual point groups. If you follow this rule, you will find that working in SA becomes much easier and more intuitive. The only time when this rule does not apply is when measuring with theodolites. Since theodolite measurement requires at least two observations to calculate a target position, each observation of a given target should be measured under the same point.

Traceability

One of the unique features of SA is its traceability. Traceability refers to maintaining an unbroken chain of comparisons relating measurements back to a defined standard. National measurement institutes such as the National Institute of Standards and Technology (NIST-USA), the National Physical Laboratory (NPL-UK), and the Physikalisch-Technische Bundesanstalt (PTB-Germany) all maintain national standards for weights and measures for their respective countries. Properly calibrated measurement instruments are evaluated and certified to this standard, which establishes the first link in this chain.

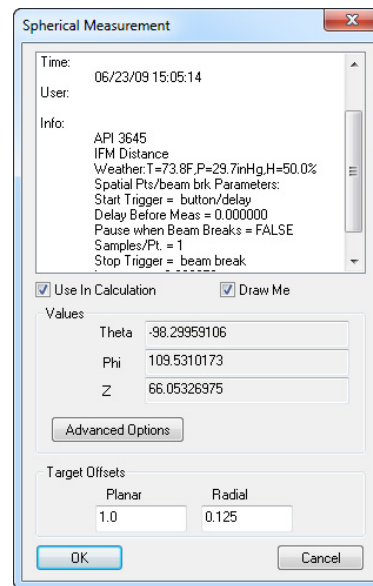
The chain is continued through measurement in SA. Each measured point in SA contains an array of metadata that describes auxiliary information about the measurement, such as the instrument (and serial number) that measured it, the time and date it was measured, who measured it, weather factors such as temperature/pressure/humidity, and more (see [Figure 3-11](#)). This information all helps maintain a complete record of the data.

In addition, actions performed in a given job file are logged, and enti-

ties themselves have historical logs that indicate how they have been modified throughout their existence. Entities even maintain information about how they were created in the first place. All of this information helps provide a complete picture of traceability for your measurements. If a specific instrument begins to show anomalies, you can look at any previous measurements, perhaps from weeks or months earlier, and identify which instrument measured it. This might give you cause to question those measurements as well—but without this traceability metadata, such a correlation may not have been possible.

NIST further maintains that “Traceability of measurement requires the establishment of an unbroken chain of comparisons to stated references, each with a stated uncertainty”.¹ Measurement uncertainty information can also be calculated for any measurement in SA, providing compliance with this policy but also providing a number of additional benefits as well (such as USMN—(see “[Unified Spatial Metrology Network](#)” on page 785).

Figure 3-11. Some of the traceability metadata stored with a laser tracker measurement.

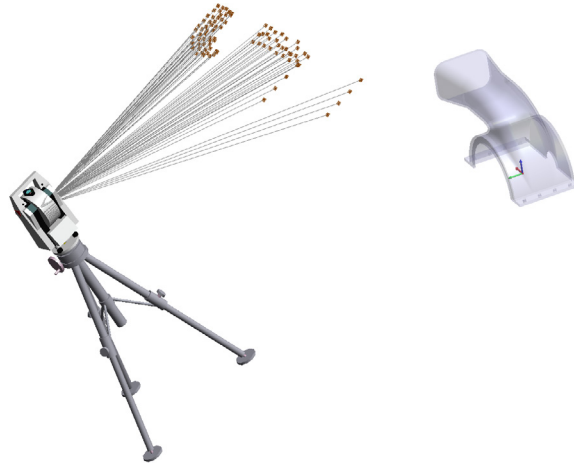


The Wire Model

One of the most important things to keep in mind when working in SA is that measurements are linked and tied directly to the instrument that measured them. If you move an instrument (using **Instrument>Drag Instruments Graphically**) from one area of the SA work-

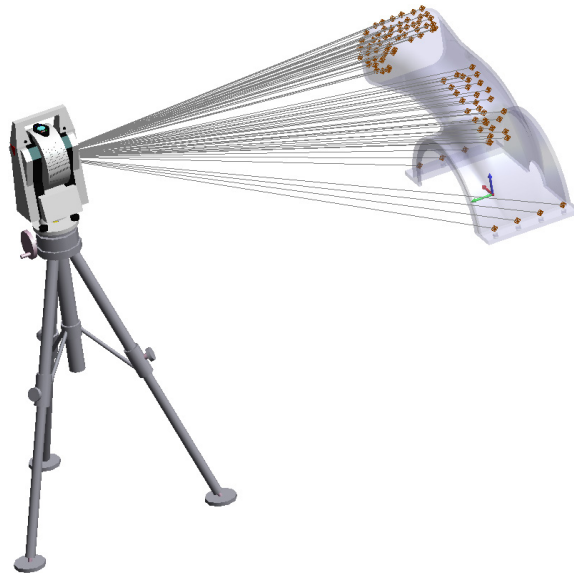
¹ NIST Policy on Metrological Traceability.

Figure 3-12. An instrument has measured the surface of this part. Prior to alignment, the measured points are not yet located to the CAD surfaces.



space (Figure 3-12) to another, you will find that all of the targets that were measured by that instrument also move along with it. You can think of each of the measured points as having a straight piece of wire between the instrument and the measurement—the instrument and its measurements move together as a rigid body.

Figure 3-13. After alignment, the points are aligned to the CAD surface. Here, the instrument was moved. The points just “came along for the ride”.




This behavior has a number of important benefits, and if you stop to think about it, the behavior most accurately reflects reality. When a measurement is recorded, the instrument provides to SA the position of the probe relative to the instrument’s reference system. For spherical measurement devices such as total stations, laser trackers, and laser radars, this consists of a horizontal and vertical angle and a distance component. These three values indicate where the point is located relative to the instrument’s reference—usually its optical center. The only thing the instrument knows about the measurement is the location of that measurement relative to itself—it (typically) knows nothing about where the measurement is located in the real world.

The link between the position of the instrument and the position of the measurement (relative to the instrument) should really never be broken, because if the two were to become disassociated relative to each other, then the value of the measurement is lost. That is, its position (and therefore the measurement) becomes meaningless to us. Therefore, it makes sense that SA maintains and keeps track of this relationship.

The Alignment Process

Tied to the association between the instrument and its measurements is the idea of alignment. Alignment is really the process of positioning the digital instrument in the SA file until it ends up in the same corresponding position as the instrument in the real world. For example, [Figure 3-12](#) shows an instrument and its measurements prior to alignment, and [Figure 3-13](#) shows that same instrument after alignment. In both cases, the only thing that was moved was the instrument—the measurements merely moved as a result of being rigidly tied to the instrument. Put another way, the measurements “came along for the ride”.

As long as a reference system is measured to properly perform an alignment at some later time, alignment is not required to be performed as one of the first steps in the measurement process. If alignment to CAD surfaces or other instruments is not required, then there is no need to perform any alignment at all.

 **Note:** If you are performing a build operation or need on-the-spot comparison of measurements to CAD or to your reference system, then the alignment must be performed at that time.

You can measure features on a part, measure a reference system, move the instrument, measure more features and the reference system, and then measure again from a third perspective, all without once performing an alignment. Only when the analysis is performed are the instruments required to be aligned (so that you can extract meaningful information from the data). As a result, you can measure everything on the shop floor and perform the alignment at your desk at the end of the day.

For precise point based alignment options refer to “[Review of Available Alignment Options](#)” on page 346, and when working with point clouds refer to “[Point Clouds](#)” on page 545.

Selection Filtering

In any SA file, there are four main types of entities. Three of them have a represented position in space (that is—they have a position and, with the exception of points, an orientation), and the fourth encompasses everything else. These entity types are described below:

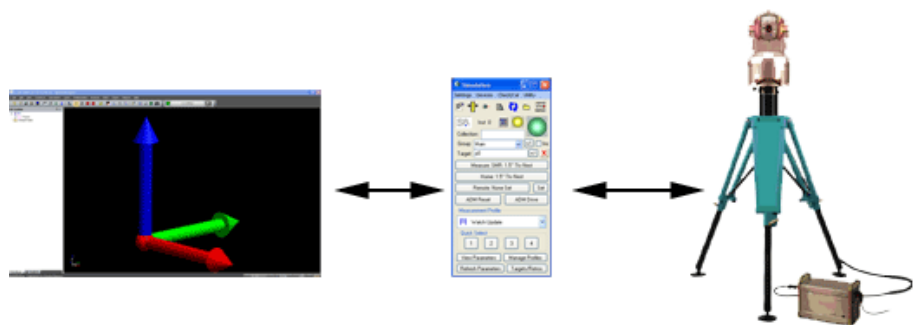
- **Points.** This includes measured points (recorded by an instrument) and constructed points (created via any other method).
- **Instruments.** This includes only measurement instruments.
- **Objects.** Anything that has a position and orientation in space that is not a point or instrument is an object. This includes entities such as lines, point groups, vector groups, cylinders, spheres, planes, surfaces, and point clouds.
- **Other.** Anything that is not a point, object, or instrument. For example, relationships, Measurement Plans, scale bars, charts, and vector groups.

Knowing these four entity types is important, because most commands in SA are filtered for the proper entity type. For instance, the **Edit>Delete Objects** command only allows selection of objects—attempting to select a point will choose its parent point group instead (because a point is not an object, but a point group is).

Architecture

SA has been designed from the ground up with a very decentralized concept of operation. That is, the software's various parts do not necessarily need to be running on the same computer. Because of its network architecture, SA is able to communicate with its various instrument interfaces via the TCP/IP network protocol, which permits a number of interesting capabilities. In addition, instruments themselves can communicate to SA through a wide array of protocols/interfaces, including TCP/IP, Serial RS-232, Bluetooth, and USB. Each method requires that the settings be configured to allow proper communication between SA and the instrument(s).

Figure 3-14. The components involved in measuring with an instrument in SA. From left to right: the Host (SpatialAnalyzer itself), the Interface, and the Instrument.

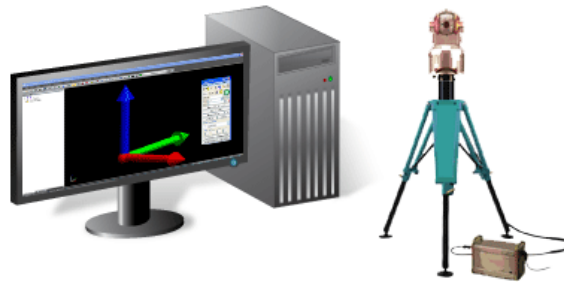


There are three main components to an SA configuration: The host SpatialAnalyzer session, the instrument interface, and the instrument (see [Figure 3-14](#)).

Host SpatialAnalyzer Session

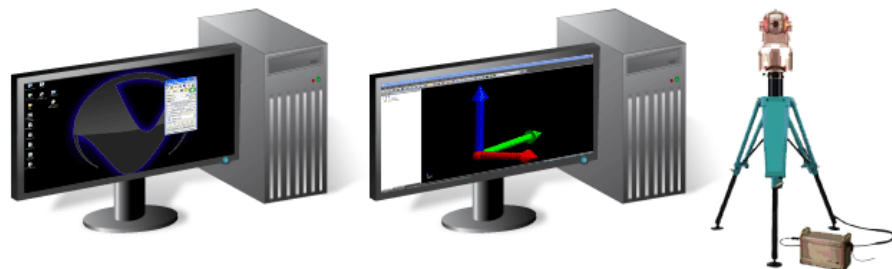
The host SpatialAnalyzer session is an instance of SA running on a computer that will be the destination for measured data. While it is usually the same computer as the one running the instrument interface (see [Figure 3-15](#)), it can be any computer that is connected via TCP/IP on the local network to the computer running the instrument interface (see [Figure 3-16](#)).

Figure 3-15. A host SA session and an instrument interface running on the same computer.



The flexibility provided by the separation of host, interface, and instrument means that you can run multiple instruments and interfaces all in the same SA job.

Figure 3-16. A host SA session and an instrument interface running on different computers.



Instrument Interface

The instrument interface is the application that directly connects with an instrument, “speaks the instrument’s language”, and directs measured data to the host SA session. When starting the instrument interface, you must choose which SA session will be the destination for measured data—if multiple computers on a network are all running SA, any one of them could have measured data sent to it. If not on the same computer, the host SA session should be located on the same network as the instrument interface. You must also define how the interface should connect with the instrument. (For example, which COM port, or using which IP address?)

There are two primary requirements in order to start up an instrument interface:

- In order for a host SA session to be a destination for measured data, you must add the appropriate instrument to the job file on the SA host, and it must be discoverable on the network.

- In order to connect to the instrument, the interface must be able to communicate with it: Either directly via an interface such as USB or Bluetooth, or via a network using TCP/IP.

A single instrument interface can connect to only one SA host session and only one instrument at a time. However, any one computer may run multiple instrument interfaces (connected to different instruments) at the same time.

The instrument interface is the mediator between the SA host and the instrument. The interface sends commands to the instrument to control its behavior, and the instrument sends data and state information back to the interface. The interface then grooms this data as necessary and sends it on to the SA host.

Instruments

Each instrument must be able to communicate with its respective instrument interface, and therefore must be set up correctly to talk to the computer running the interface. This is handled in a number of ways depending on the method that the instrument uses to connect to the computer. Detailed information on the proper setup for instruments using each communication protocol is found in the *SA Instruments Manual*.

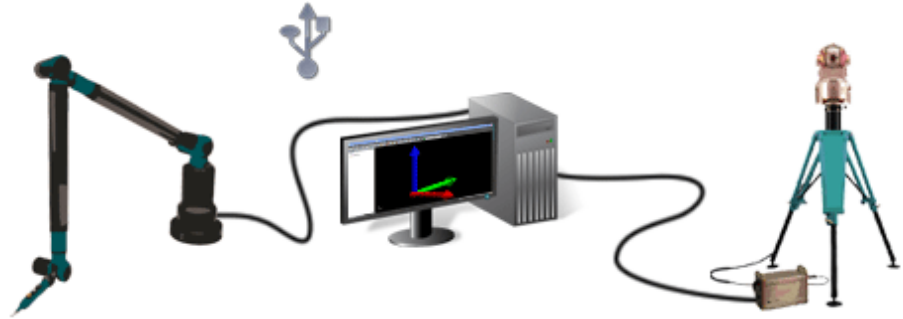
The following graphics illustrate just a few of the many possible ways to connect instruments with SpatialAnalyzer.

Figure 3-17. A single computer connected directly to an instrument via TCP/IP. This is the most common setup.



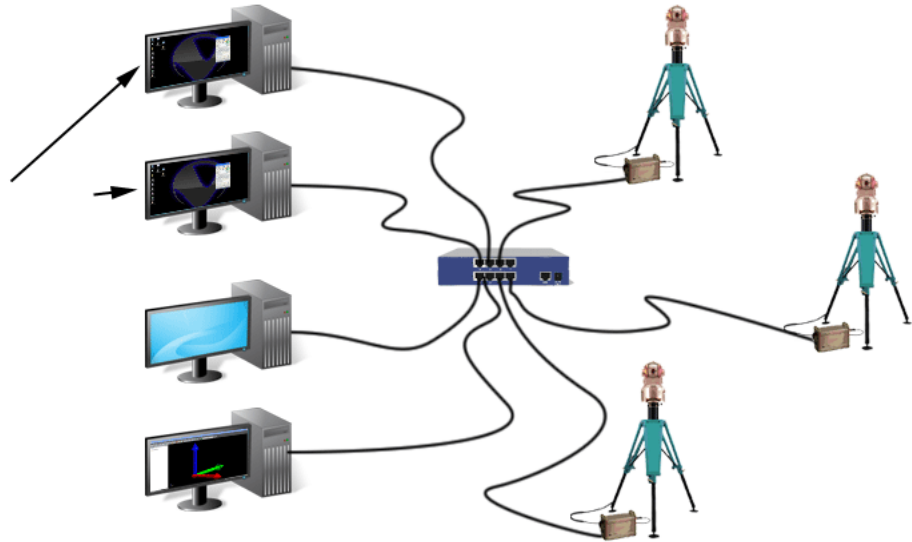
In [Figure 3-17](#), since the instrument is connected directly to the host computer via a network cable and the TCP/IP protocol, no other communication hardware is required. The interface runs on the same computer as the host SA session. This is the most common way to connect in SA.

Figure 3-18. Two computers connected directly to the host computer: One via USB, the other via TCP/IP.



In [Figure 3-18](#), one instrument is connected via USB, and the other through TCP/IP. The instruments are connected directly to the same computer on which the instrument interfaces and the host SA session are running. Only one instrument might be running at a time, or two instrument interfaces might be running simultaneously in a single session of SA.

Figure 3-19. Multiple instruments and computers on a local network connected via TCP/IP and an ethernet switch. The interfaces may be running on a different computer than the host.

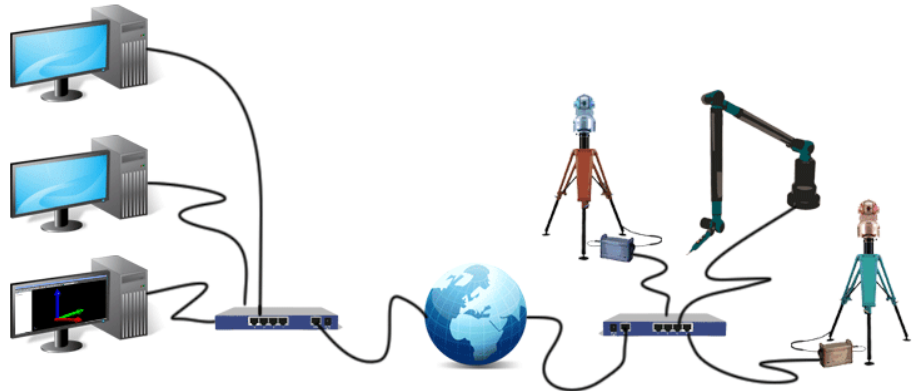


With this configuration ([Figure 3-19](#)), a connection to the internet is not used. Instead, one or more instruments and one or more computers are connected to a local network through a network hub or switch (and the TCP/IP protocol). One SA host computer may be running instrument interfaces for all instruments, or the interfaces may be run on separate networked computers. Or, three instances of SA might be running on three different computers to control the three different instruments.

While the configuration in [Figure 3-20](#) is rare, it is technically possible. In this case, an instrument is connected via TCP/IP over the internet to an instrument interface, perhaps on the other side of the world. The interface then connects to the SA host via the local network. As long as you can ping the desired instrument from the computer running the instrument interface, you can connect to it.

As you can see, the networked architecture of SA lends immense flexibility to the way that you measure and operate on data. With this set-up, it is possible for an operator in a metrology office to control a laser radar hanging from a gantry crane in a facility on the other side of the world. Or, one workstation might be set up to receive measurement data, and three operators with laptops might measure data with their own instruments and send it to the receiving workstation.

Figure 3-20. Networked configuration. Instrument and interface connected via TCP/IP over the internet, and the host SA session is connected to the interface via a local network.



While there are many different configurations and possibilities, most users end up with one of the first three configurations illustrated above.

This Page Intentionally Left Blank.

Exploring The User Interface

CHAPTER

4

SpatialAnalyzer provides a user-friendly interface to access all of your measurement data. There are often several ways to perform a given task, enabling you to choose a preferred method of interacting with the application.

Application Components

The SpatialAnalyzer software is fundamentally a three-dimensional rendered view of a measurement job, which is displayed in a *Graphical View* window. Around the sides of this window are a set of essential tools. The *Treebar* is a list the points and objects in the measurement job and a *Report Bar* displays a detailed summary of information on the selected item in the job. Finally, across the top is set of Ribbon menus or a combination of classic menu bars.

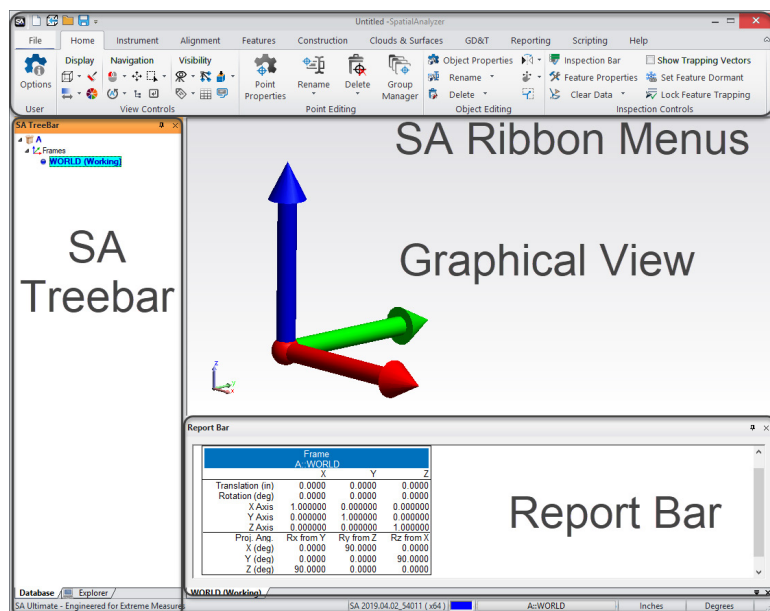


Figure 4-1. Core Components of the SA Application.

- **Graphical View.** A three-dimensional rendered view of the

current job.

- **Treebar.** A hierarchical view of all entities in the open job. For brevity this will typically be referred to simply as the tree.
- **Report Bar.** Displays important information about the selected item and a convenient way to generate reports.
- **SA Ribbon Menu.** The ribbon menu or the classic set of menus provide access to all of the tools SA offers for analysis and construction purposes.

Each component comes in a separate window which can be docked to the edge of the screen or to the edge of another window. The components can be closed independently to allow you to maximize your screen space and to enable you to customize the user interface for the specific task at hand.

A series of addition controls can also be added as needed for applications such as controlling an instrument or running a measurement script. At any point if you wish to display an additional toolbar you can right-click on any of the toolbar headers and enable additional toolbars (see [Figure 4-2](#)).

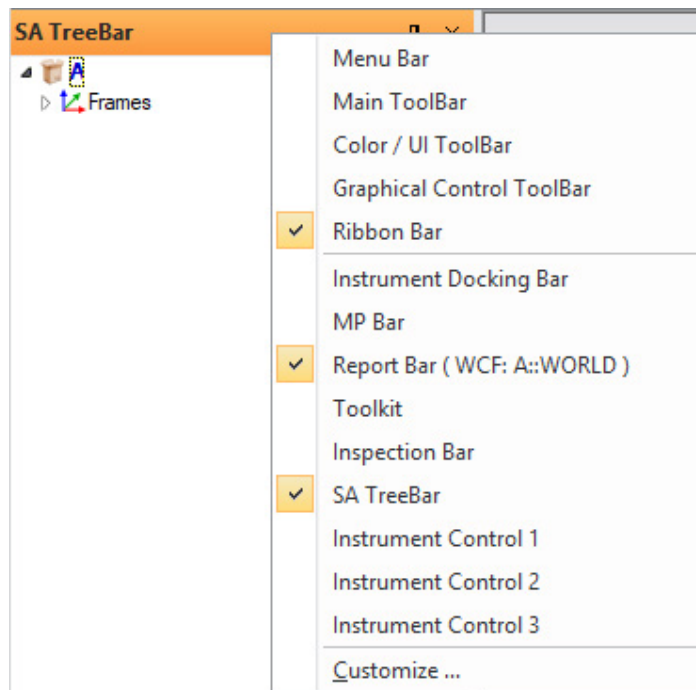



Figure 4-2. Right-Click Menu allowing access to additional or hidden toolbars accessed from any toolbar header.

SA Ribbon Menus

SA can be displayed in either of two modes. It can be shown with the classic SA Toolbars (see “[Classic Toolbars](#)” on page 58) or with the SA Ribbon Menu. To enable or disable the ribbon menu use the check box on the Display tab of the User Options . Enabling the ribbon display will require you to restart SA but will persist from that point on (see [Figure 4-3](#)).

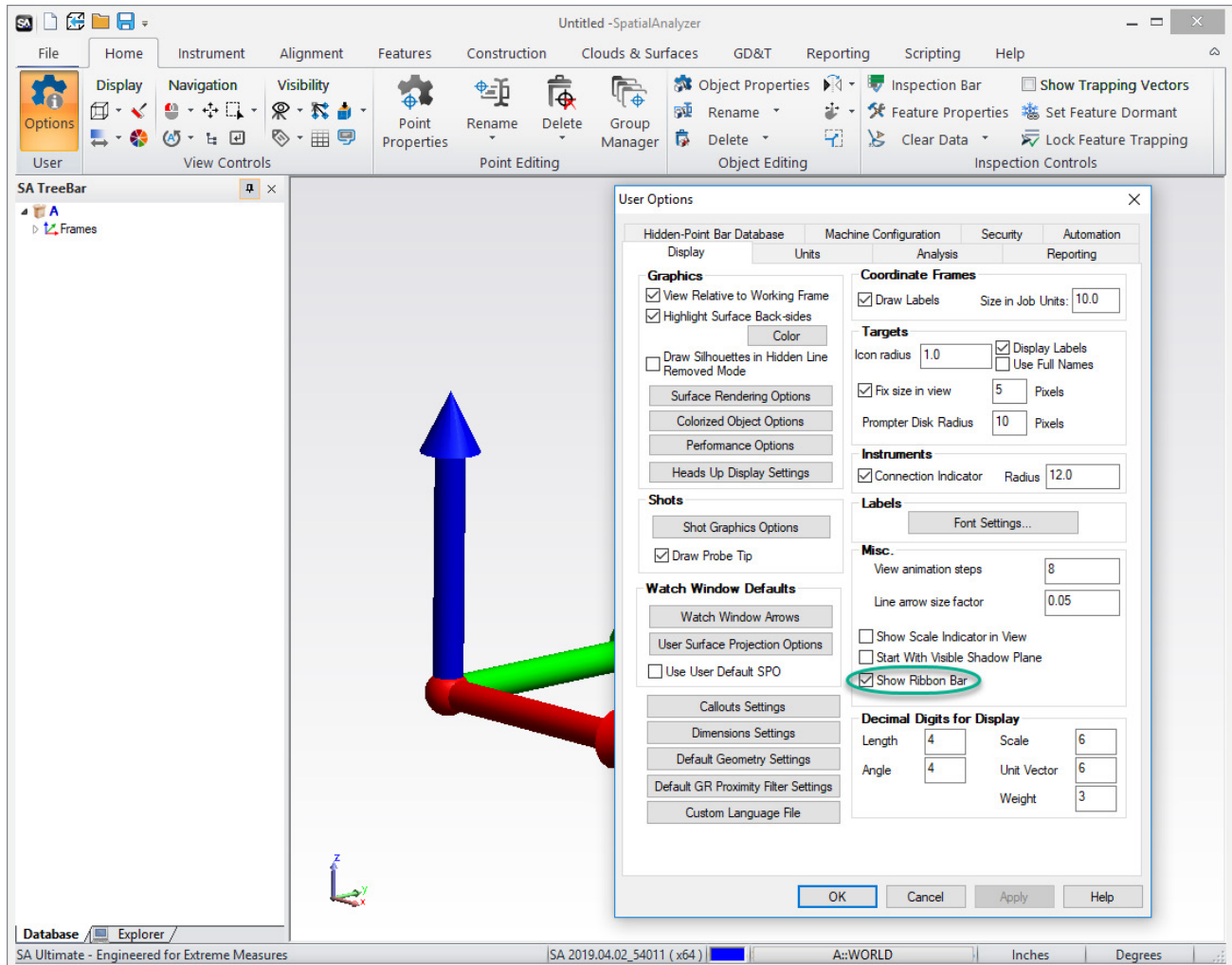


Figure 4-3. SA Ribbon Menu.

The SA Ribbon Menu is a full replacement for the [Main Toolbar](#), [Graphical Control Toolbar](#), [Color/WCF toolbar](#) and [SA Toolkit](#).

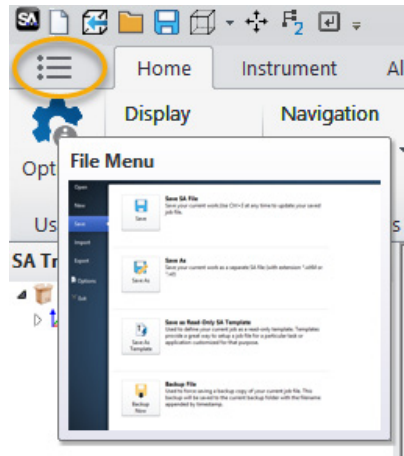
Ribbon Bar Customization

The biggest advantage of the ribbon menu is its ability to be customized to fit a user’s needs. It can be streamlined for a user or even for a specific task.

To access the customization dialog. First press the [File Menu](#) button

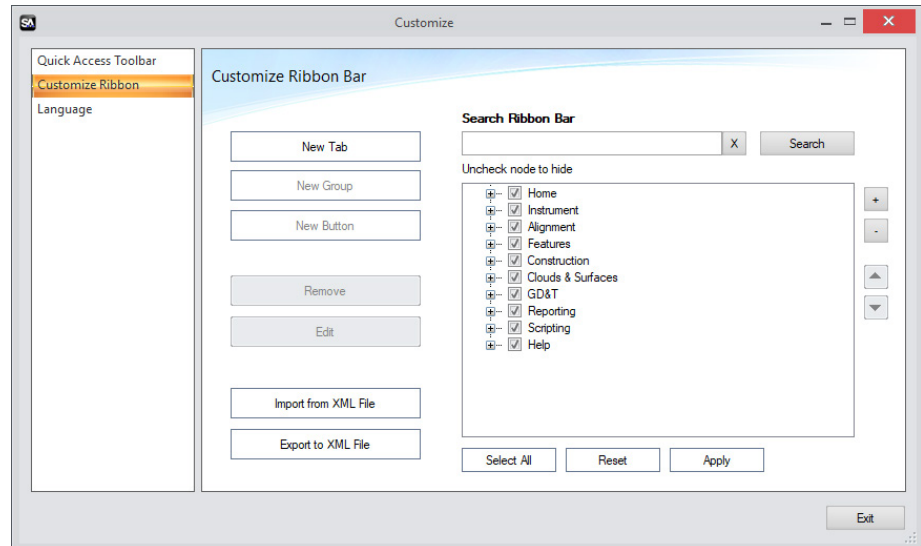
that takes you to the back Stage (Figure 4-5). Then Select **Options**.

Figure 4-4. File Menu



The “Customize Ribbon” option has been added to a resizable Customize Options dialog. It presents the ribbon bar structure in a tree format and allows to search and locate needed command, add new tabs/group/buttons, define own MP-based command, control button size and icon, rearrange tabs, hide/unhide tabs/group/buttons, import/export ribbon bar custom configuration to XML file, customize keyboard shortcuts, and etc. (Figure 4-5).

Figure 4-5. Customize Ribbon Dialog.



- **Search Ribbon Bar.** The “Search Ribbon Bar” operation allows to find a command location on SA Ribbon Bar. User needs to enter a keyword and press “Search” button. The found entries are expanded and highlighted in the tree. Additionally, there are two buttons “+” (Expand All) and “-” (Collapse All) may help to control the ribbon tree presentation to locate needed item.
- **Ribbon Bar Node Visibility.** To hide/unhide ribbon bar node (tab, group, button, or menu entry), user needs to check/uncheck a corresponding node in the ribbon tree and save the

changes. A “**Select All**” button allows quickly check all nodes in a tree. If saved, it will make all nodes in the Ribbon Bar visible. The visibility of all default and custom tabs, groups, and buttons can be controlled.

- **Add New Tab.** User can add a new tab to the SA Ribbon Bar. Once added, new tab is considered as a Custom Tab.
- **Add New Group.** User can add new group to any current tab. Once added, new group is considered as a Custom Group. User needs to select a tab in the tree and press “**New Group**” button. The application automatically adds new groups after the last default group of the tab.
- **Add New Button.** User can add a New Button to a *Custom Group* only. Once added, a new button is considered a Custom Button.

New Button Options

User needs to select a custom group in a tree and press “**New Button**” button. The application automatically adds a new button to a custom group and popup “Ribbon Bar Custom Button Settings” dialog where user can define the custom button properties and operation [Figure 4-6](#).

Figure 4-6. Custom Button Controls.

- **Button Location.** The button location is defined by Tab and Group set. The dialog presents the current button's location. User can change the button location by picking new tab or group from the combo boxes. The Tab combo box lists all ribbon pages that have at least one custom group. The Group combo box lists all custom groups of the selected page.
- **Button Name.** User can define or edit button's name. Small buttons have a single line name. Large buttons may have a name presented as two lines.
- **Ribbon Command Based Custom Button.** User can select and assign a command from default ribbon Tabs. Once selected, the application automatically sets all known information to a new custom button including button name, command id, screen tips, default icon and button size. User can modify any of these properties through Ribbon Bar Custom Button Settings dialog.
- **MP Script Based Custom Button.** A custom button can launch

a pre-defined MP script. The Ribbon Bar Custom Button Settings dialog allows user to define MP script location (embedded or external file)

- **Button Size and Icon.** Custom button's size can be configured as large with 32x32 bmp icon or small with 16x16 bmp icon. Please note that currently single button launches a single command. No drop-down menu items for a button are supported. When configuring a custom tab, please keep in mind the button's size and the space they take.

For example, the small buttons configuration has the following advantages:

- Allow long button name (self-explanatory)
- Three small buttons can create a column on the ribbon tab. In the same space only two large buttons can be created. By default, the application uses a default button icon. However, user can specify location of custom *.bmp file to use instead of default.
- **Button Screen Tip.** User can define screen tips for button operation.

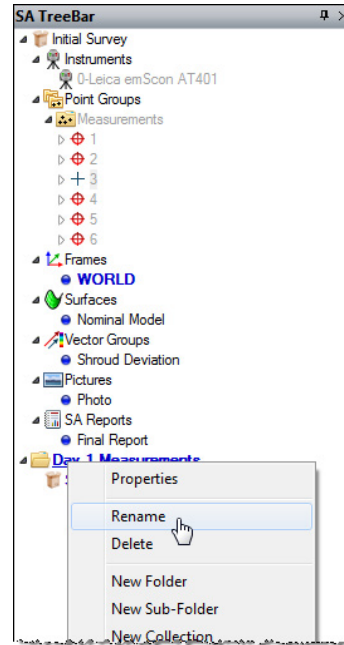
Editing a Custom Ribbon Configuration




- **Remove Custom Node.** User can remove any created custom node (Tab/Group/Button).
- **Edit Custom Node.** User can edit any custom Node. Editing of custom Tab or Group means renaming. For Custom Button, the application automatically opens "Ribbon Bar Custom Button Settings" dialog to allow user to change any button's property.
- **Apply.** When updating an existing configuration the apply button should be used in order to ensure the current settings are saved for export.
- **Import and Export from XML File.** The SA Ribbon Custom Configuration can be imported/ exported from XML file.
- **Reset to Defaults.** The Reset to Defaults operation restore the original SA Ribbon Bar configuration. It removes all custom nodes, makes all default nodes visible, and restore the original order of tabs.

SA TreeBar

Database Tab

Figure 4-7. The database tab.





The Database tab of the tree displays a hierarchical view of all entities in the current job (see [Figure 4-7](#)). It's called the Database tab because it reveals all of the entities in SA's underlying measurement database. This provides quick access to items of interest and enables simple but effective organization and management of properties. The tree contains items such as instruments, point groups, reports, embedded files, and geometric objects (lines, planes, etc). Headings or parent items within the tree can be expanded by clicking on the  button to expose child data (on some machines, this appears as an  icon). Clicking the  button will collapse these items again. Right-clicking on an entry will display a context menu of commands that relate to the selected item.

The hierarchical tree displays a number of different entities depending on what is present in the current job file:


SA job organizational levels

SA's database structure and naming is composed of 2 levels: Collection and item. Some special items like point groups also have subcategory within them (points) which are exposed in the tree. While others like point clouds or vectors also include a large number of entries (cloud points or vectors) that are not directly accessible from the tree.


 **Folders.** Like folders in Windows Explorer, you can sort collections into folders to organize the SA file.


 **Collections.** An organized collection of entities (objects, instruments, reports, etc) that can be managed as a set in the tree. Collec-

tions can be moved into a folder or stand alone in the tree.

 **Point Groups.** A series of points (measured or constructed) that are conveniently treated as a single object. Each group resides within a single collection.


Additional important Item types:

 **Frames.** Coordinate frames. The *World* frame is the base coordinate frame for the job file and cannot be moved or edited. A single job file may have many frames within it any of which can be designated as the *Working* frame for the job. The xyz position of any object in the job is reported with respect to the Working frame coordinate position.

 **Instruments.** Measurement instruments—laser trackers, portable CMMs, etc.

Additional Items/Objects in the tree:


In SA the term “Object” is used for items that are drawn in the SA graphics such as:


 **Planes.** Planar geometry that, while drawn with a finite size in the graphical view, extends infinitely in two dimensions.


 **Circles.** Circular geometry.


 **Cylinders.** Cylindrical geometry.

 **Spheres.** Spherical geometry.


 **B-Splines.** Curves that have a mathematical definition and a defined direction along the curve.


 **Surfaces.** Surfaces are parametrically defined and have a complete mathematical definition for every point on the surface. They are exact representations of a given surface.

 **Vector Groups.** Vector groups represent the deviation between two entities as one or more vectors.

 **Point Clouds.** Groups of points typically gathered from non-contact measurement devices (such as a laser line scanner or coherent laser radar).


The term Item is used more generally and applies to additional database items that are in the tree but not part of the 3D world of the SA graphics view. These include things like:


 **Pictures.** Imported images or graphical view screenshots that can be dragged into SA Reports.

 **SA Reports.** One of several types of reports that can be created in SA.

 **Callouts.** Callout views enable you to add captions, labels, and call-


outs in the graphical view.

 **Charts.** A number of different statistical charts can be created as a result of various operations in SA.

 **Events.** Easily-reportable records of significant events (such as fitting geometry, minimizing relationships, or importing a CAD file) that have taken place in the file.


Relationships in SA


SA uses the concept of a *Relationship* as a entity that defines the connection between two or more objects. It can either be used for reporting or computing dynamic changes in the job. For example a vector group can be built to define the distance between points and an object. If this vector group is build by a relationship it will dynamically update if the position of the points and/or object are moved.


 **Relationships.** Dynamic (continuously updated) comparisons between entities.

Embedded Files

It can be helpful to use the SA job file as a resource to store additional reference materials. This can include scripts and reference files such as:

 **Measurement Plans.** Scripted programs that can automate measurement, analysis, and reporting operations.

 **Embedded Files.** Any file can be embedded into an SA file for quick access or to conveniently link a file with the SA job.

 **Graphical Entities.** Geometric shapes (such as boxes and pyramids) that act as purely visual elements and cannot participate in any analysis.

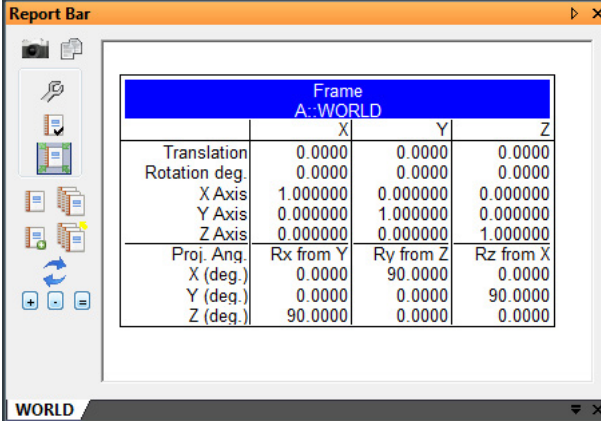
All of the above items (with the exception of Folders and Collections) are actually categories of items. The tree organizes items in a collection by their category--the individual items of that category are listed under the category. For instance, in the Instruments category, there might be three actual instruments.

Explorer Tab

The Explorer tab displays the file and directory structure of the local computer. This is a convenient way of navigating the disk to find files to import into the current SA job.

Report Bar

Figure 4-8. The Report Bar showing properties for a coordinate frame.



The Report Bar window displays a table of properties for the selected coordinate frame 'A::WORLD'. The table is organized into columns for X, Y, and Z axes, with rows for Translation, Rotation, and Projection Angles.

Frame A::WORLD			
	X	Y	Z
Translation	0.0000	0.0000	0.0000
Rotation deg.	0.0000	0.0000	0.0000
X Axis	1.000000	0.000000	0.000000
Y Axis	0.000000	1.000000	0.000000
Z Axis	0.000000	0.000000	1.000000
Proj. Ang.	Rx from Y	Ry from Z	Rz from X
X (deg.)	0.0000	90.0000	0.0000
Y (deg.)	0.0000	0.0000	90.0000
Z (deg.)	90.0000	0.0000	0.0000

The Report Bar (Figure 4-8) displays the detailed properties of the currently selected item in the tree. It can also serve as a report generator by allowing multiple tabs of information to be compiled into a single report. For details on the report bar and reporting, refer to the Reporting chapter.

To Toggle Display of the Report Bar:

- Right-click in the menus, toolbars, or status bar area of the interface and select **Report Bar** from the context menu, or
- From the menus, select **Reports>Report Bar Visible**.

Status Bar

The Status Bar (Figure 4-9) is a small bar at the bottom of the interface that has a number of minor, auxiliary displays. The left side of the bar will display any minor, low-importance messages that may arise during the course of your work (for example, when copying data to the clipboard from the Report Bar).



Figure 4-9. The Status Bar.

To the right of this message area is the SA version. In Figure 4-9, the SA version is a date followed by a build number: 2020.07.20_67800.

Next to the version number are a set of status displays:

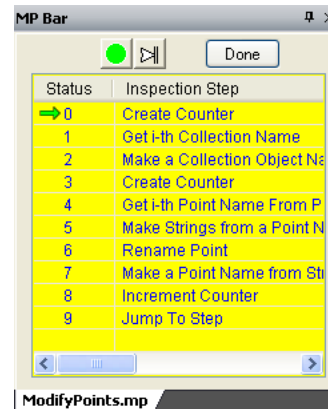
- The current active color used if a new object is created.
- The working frame in the job file, which can be a button allowing you to edit the working frame.
- Current job unit settings
- The three slots to the right of the units display indicate wheth-



er **Caps Lock**, **Num Lock**, or **Scroll Lock** are active on the keyboard (respectively).

- Two small processor icons on the right side of the status bar light up when SA is in a multi-threaded operation.

MP Bar

Figure 4-10. The MP Bar with an MP loaded.




The MP Bar (Figure 4-10) is normally only visible when a scripted Measurement Plan has been loaded for execution. The main window lists each command of the MP, as well as a pointer indicating the command which will be executed next, and the status (success, partial success, or failure) of each step. The MP Bar has a button to start executing the current MP , execute the next step in the MP , and terminate/close the current MP . The tabs at the bottom of the MP Bar display the primary MP and any MP subroutines currently executing.

Classic Toolbars

Toolbars provide icons for quick access to frequently-used functionality and are organized into related groups. There are three factory-default toolbars that ship with SpatialAnalyzer: the *Main Toolbar*, *Graphical Control Toolbar* and the *Color/WCF Toolbar*.

To Hide an Interface Component:

- Click the  icon in the corner of the window, or
- Right-click on the menu bar, toolbar, or a window title bar and uncheck the component you wish to hide. Check it to show it again.

Interface components can be set to auto-hide, so that they collapse into a tab at the edge of the screen when not active. This provides for more space to work in other interface windows.

To Set an Interface Component to Auto-Hide:



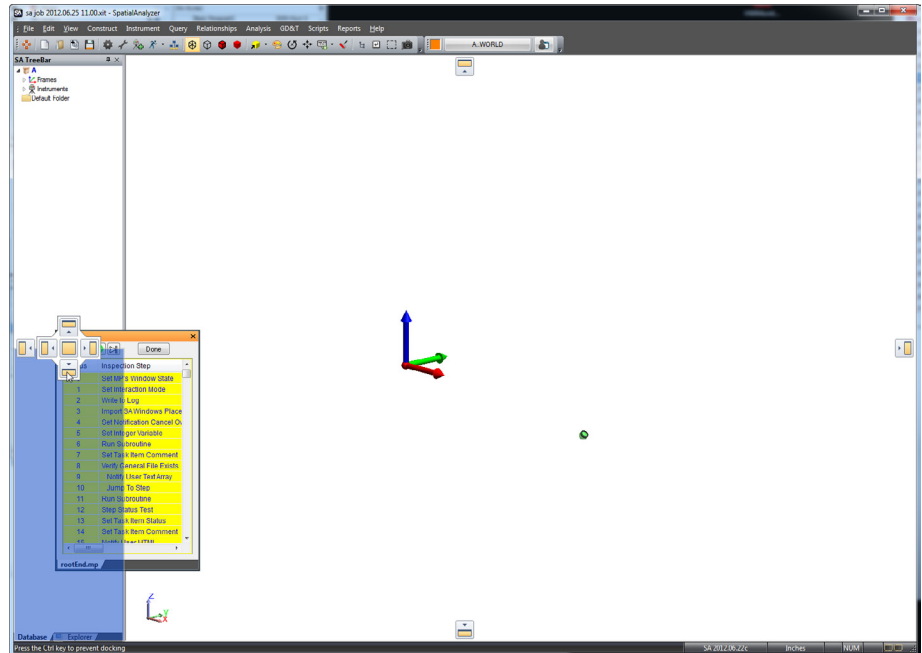
Click the  icon. When the mouse is moved away from the window, it will collapse into a tab (after a short delay). Roll over the tab and click the  icon to disable the auto-hide behavior.

Figure 4-11. Docking the MP Bar to the bottom of the tree.



You can rearrange the positioning of the different user interface components using docking. Interface components can be undocked, docked to the edge of the application screen, or docked along the edge of another interface component.

To Undock an Interface Component:

- Click and drag the title bar for the component away from its current position.

To Dock an Interface Component to the Edge of the Screen or Another Component:


- Click and drag the title bar for the component. As you drag, a series of boxes representing locations to dock the window will appear. Drag over an icon to dock to a specific position (see [Figure 4-11](#)).



















You can customize the user interface, and then save this customization to a User Interface Profile. For more information, see “[Custom UI Profiles](#)” on page 63.

Main Toolbar

Figure 4-12. The Main Toolbar.







 User Manual. Opens the manual (this document) for SpatialAnalyzer.

-  New. Starts a new job file.
-  Open. Opens a saved job file.
-  Auto Import. Imports a file automatically based on its file type.
-  Save. Saves the current job file.
-  User Options. Opens the user options dialog.
-  Object Properties. Displays the properties of an object after clicking the object in the graphical view.
-  Add Instrument. Adds an instrument to the current job.
-  Run Instrument Interface. Starts an instrument's interface.
-  SA Network. Opens the SA Network Browser.
-      Rendering Settings. Switches between Wireframe, Hidden Line Removed, Solid+Edges, and Solid rendering modes.
-  Views. Selects, edits, and saves preset views.
-  Background. Changes the background and highlight colors.
-  Callout View. Adds callout views to the active collection.
-  Capture Graphics Picture. Captures the current graphics view as an image for use in reports or other documents.


Graphical Control Toolbar


Figure 4-13. The Graphical Control Toolbar




-  Autoscale. Automatically scales the graphical view to fit all visible entities.
-  View Rotation Center. Defines a pivot point for view rotations when enabled. Click to enable the mode and select a rotation point. Click again to change the rotation point, and double click to exit the mode.
-  Instrument View. Provides a means to easily enable the instrument view option also available through the instrument's right-click menu. Single click to enable or turn off the mode and double click while active to access the settings dialog see "[Set Viewpoint From Instrument Updates](#)" on page 78).
-  Clipping Plane Mode. This button provides a convenient way to both define and check the status of the cloud default clipping plane for the job file. Left-Click to build a plane feature and set that plane feature as the jobs default clipping plane, setting you to measure it. Right-click to select an existing plane. Once set, right-click again to

directly access the clipping plane's offset definitions see “[Clipping Planes](#)” on page 565).

 Graphical Selection Mode. Changes the method for graphically selecting items with the mouse. Single click to toggle between rectangular and polygon selection modes. Double click to open the properties dialog.

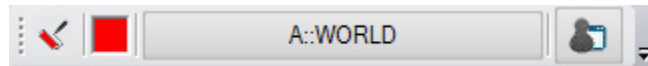
 Left Mouse Button Overrides. Provides options to control the function of the Left mouse button for computers or tablets with only 1 button. Choose between Left, Right, and middle mouse button operation. R-click (one-shot) provides a single right-button press following button selection followed by normal left button operation.


 F2 - Select from a List. Opens the F2 selection dialog.


 Enter - End Multiple Selection. Equivalent to pressing the **Enter** key. Ends multiple-select mode.

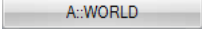
Color/WCF Toolbar


Figure 4-14. The Color/WCF Toolbar.



 Color Objects. Changes the color of selected objects in the graphical view.

 Working Color. Changes the working color.

 Working Frame. Displays the current working frame and allows you to pick a new one.

 User Interface Profiles. Used to define and activate custom user interface profiles. For more information, see “[Custom UI Profiles](#)” on page 63.

Toolbars can be rearranged by dragging on their toolbar grips(see [Figure 4-15](#)). You can completely undock a toolbar from the menus, or simply rearrange them. To dock a toolbar back into its original location, drag the grip to slide the toolbar back into place.

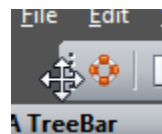
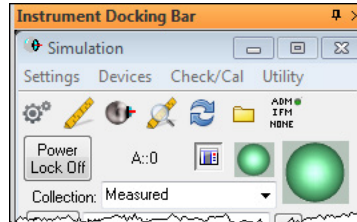


Figure 4-15. Rearranging a toolbar by grabbing its grip.

You can create your own custom toolbars. For more information, see “[Custom UI Profiles](#)” on page 63.

Instrument Docking Bar

Figure 4-16. The tracker interface inside the Instrument Docking Bar.

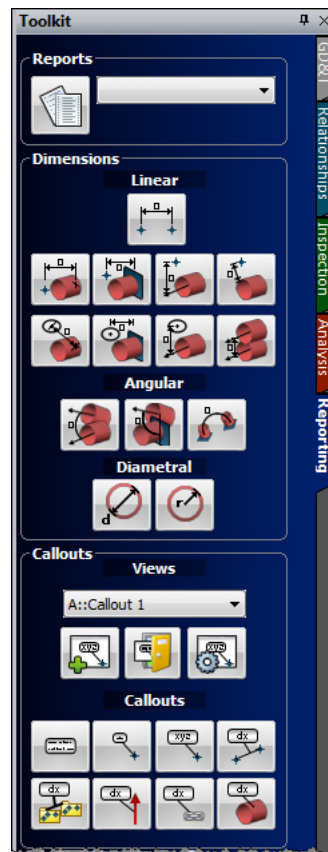


The Instrument Docking Bar (Figure 4-16) is a window that houses the currently docked instrument interface. It enables the instrument interface, which is actually a separate application from SA, to be docked onto one edge of the SpatialAnalyzer interface.

SA Toolkit

The SA Toolkit (Figure 4-17) contains five tabs that provide convenient access to a number of useful SA tools. These tabs include GD&T, Relationships, Inspection, Analysis, and Reporting.

Figure 4-17. The SA Toolkit.



- **GD&T.** Tools for creating GD&T checks from CAD surfaces, primitive objects, or directly from points.
- **Relationships.** Tools for quickly creating various types of rela-

tionships.

- **Inspection.** Provides access to a list of inspection-ready items with buttons for advancing through the inspection.
- **Analysis.** One-click access to many analysis tools.
- **Reporting.** Buttons for creating reports, dimensions, and callouts.

UI Themes

A number of standard user interface themes are available to change the look of the SA interface. Three of the 14 interface styles are pictured below:

Figure 4-18. Top to bottom: MS Office XP, MS Office 2007 (black), and MS Office 2003 themes.



To Change the Look of the SA Interface:

Note: The user interface theme is saved as a machine setting.

- In the **View>User Interface Features>UI Theme** menu, choose a theme from the list. The interface will update immediately.

Custom UI Profiles

Custom User Interface Profiles can be used to customize the Classic SA menus. They are not compatible with the ribbon bar and setting a custom user interface profile will prevent switching to the ribbon display.

Using Custom User Interface Profiles

A large portion of the SA user interface can be customized. With user interface customization you can:

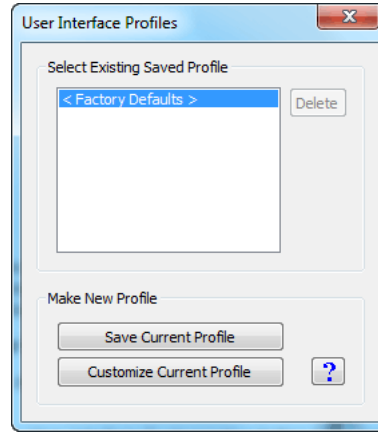
- Add, rearrange, delete, and rename menu items,
- Create custom toolbars and menus,
- Save menu and toolbar docking locations,
- Create custom keyboard shortcuts, and
- Create custom icons.

This allows you to modify an interface to display only the commands you use while hiding unused features and leaving more room to work.


User interface settings are saved as profiles, with an `.saprofile` ex-

tension in the Analyzer Data\Templates directory. Once saved, they can later be recalled, or easily transferred to another computer by copying the file into the same directory on another machine. The selected profile becomes the default used when SA starts, and is saved as a machine setting.

Figure 4-19. The User Interface Profiles dialog.




To Create a Custom User Interface Profile:

1. Do one of the following:
 - Click the **User Interface Customization** button  in the *Color/WCF Toolbar*, or
 - Select **View>User Interface Customization**.
2. Click the **Customize Current Profile** button (Figure 4-19). This places you into Custom mode, which allows you to edit the properties of toolbar and menu items instead of performing their equivalent operations. (For details on customization, read the following sections). Custom mode ends when you return to the **User Interface Profiles** dialog.
3. When finished customizing the profile, click the **Save Current Profile** button. Enter a profile name, and click **OK**.

Warning: User Interface profiles are not forward-compatible. Once a profile file is saved, it cannot be opened by a previous version of SA.

To Delete a Custom User Interface Profile:

1. Do one of the following:
 - Click the **User Interface Customization** button  in the *Color/WCF Toolbar*, or
 - Select **View>User Interface Customization**.
2. Select the profile in the list that you'd like to delete (Figure 4-19).
3. Click the **Delete** button. The profile will be deleted from the list and from the Analyzer Data\Templates directory.

Warning: User Interface Profiles cannot be used with the Ribbon Menu.

To Reset to the Factory Default User Interface Profile:

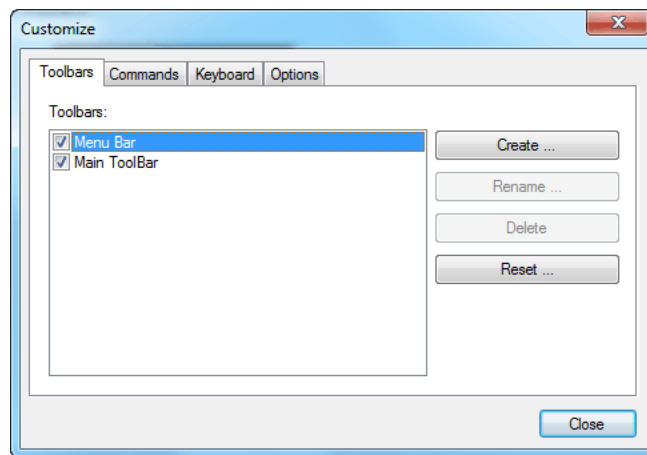
- In the **User Interface Profiles** dialog (Figure 4-19), choose **<Factory Defaults>** from the list, or

- From the menus, choose **View>User Interface Features>Reset GUI Toolbar Layout**.

Customizing Toolbars & Menus

The Toolbars tab (Figure 4-20) allows you to show and hide toolbars, as well as create your own. Custom toolbars are treated just like standard toolbars and act identically. Checked toolbars are displayed, while unchecked toolbars are hidden. The following assumes you are in customize mode (see “To Create a Custom User Interface Profile:” on page 64).

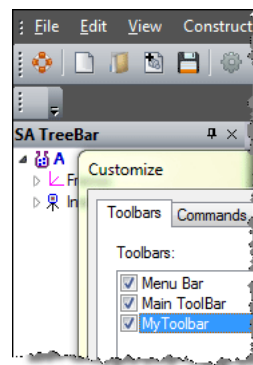
Figure 4-20. The Toolbars tab of the Customize dialog.



To Create a Custom Toolbar:

1. In the *Toolbars* tab, click the **Create** button.
2. Give the toolbar a unique, descriptive name.
3. The new toolbar will appear in the list, and an empty toolbar will appear in the user interface (Figure 4-21).

Figure 4-21. A new custom toolbar named MyToolbar has been added. Notice the empty toolbar below the main toolbar.



While in Customize mode (see “To Create a Custom User Interface Profile:” on page 64), you can modify toolbars as well as menu items. You can drag items to create new menus, drag them into current menus, or drag them into toolbars.

To Move an Item:


- Drag a menu, menu item, or toolbar button to the desired position. Drag it back to reverse the process.

To Add a Separator Between Items in a Custom Toolbar or Menu:

- Right-click an item and choose **Begin a Group**. The separator will be placed before the selected item.

Figure 4-22. A group separator has been added after the Scripts menu.

**To Delete an Item from a Custom Toolbar or Menu:**

- Drag the item into an “empty” area of the screen (not over any menu or toolbar). The mouse cursor will change to , signifying that the item will be deleted, or
- Right-click the item and choose **Delete**.

To Add a New Item to a Custom Toolbar or Menu:

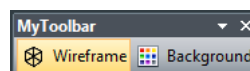
- From the Commands tab, choose a menu category from the list at left. Then drag a command from the list at right to the toolbar or menu.

The appearance of items in a toolbar or menu can be changed. You can view an item as an icon only, text only, or both. You can even modify the icon for an item.

To Change the Appearance of a Toolbar Item:

- Right-click the item and select from either **Default Style, Text Only (Always)**, **Text Only (In Menus)**, or **Image and Text**.
- **Default Style.** This will display the icon if one is available. Otherwise, it will display text.
- **Text Only (Always).** This will display the item as text, regardless of whether it is on a toolbar or in a menu.
- **Text Only (In Menus).** This will display the item as an icon when in a toolbar (if an icon is available), and as text when in a menu.
- **Image and Text.** This will display both the icon (if available) and the text for the item, regardless of whether it is in a menu or a custom toolbar.

Figure 4-23. Two icons displayed with the *Image and Text* option.

**To Change the Text Associated with an Item:**

- In Customize mode, right-click the item and click in the *Name*

text field. Enter a new name for the item. Note that this will only be displayed if the current appearance setting displays text.

To Reset an Item to its Default Appearance:

- Right-click the item and choose **Reset**.

To Change a Toolbar Icon's Appearance:

- In Customize mode, right-click the item and choose **Edit Button Image**. This will open a simple editor to allow you to modify the icon, as depicted in [Figure 4-24](#):

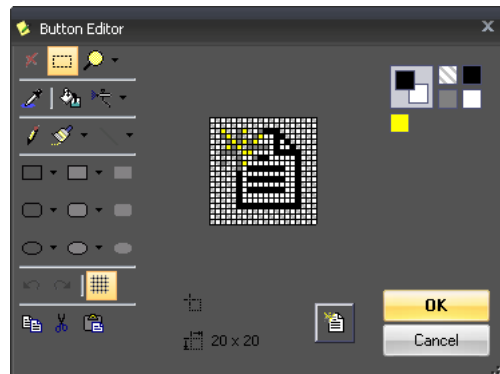


Figure 4-24. The Button Editor.

To Reset a Button's Image Back to Defaults:

- In Customize mode, right-click the item and choose **Reset Button Image**.

To Copy or Paste a Button Image:

- In Customize mode, right-click the item and choose **Copy Button Image** or **Paste Button Image**.

To Assign a Supplied Image to an Icon:

Note: If a command does not come with a default icon, a custom icon cannot be added.

- In Customize mode, right-click the item and choose **Change Button Image**, then choose an image from the list.

Keyboard Shortcuts

You can create custom keyboard shortcuts for virtually every command available to you in SA. Keyboard shortcuts are defined in the *Keyboard* tab of the Customization window. These keyboard shortcuts are saved as part of the custom user interface profile.

To Define a Keyboard Shortcut for a Command:

1. In the Keyboard tab (Figure 4-25), choose the desired category and command from the list.

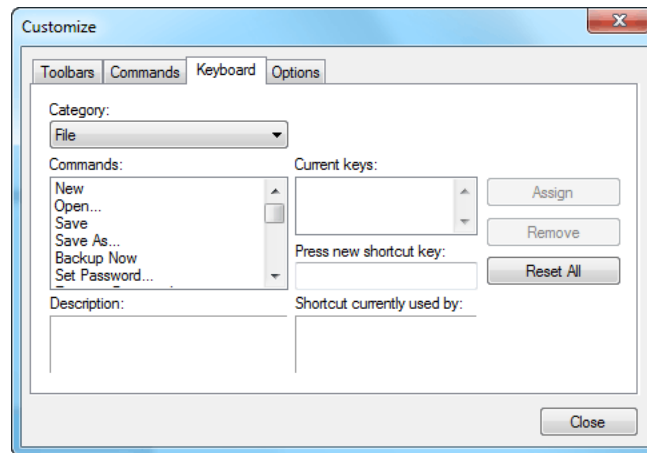


Figure 4-25. Assigning custom keyboard shortcuts.

Tip: You can assign a keyboard shortcut to a main menu item by placing an ampersand before the letter that you'd like to become the shortcut—for example, &GD&&T. This will allow you to open the GD&T menu by typing Alt+G. To place a single ampersand in a menu item, the name field must have two consecutive ampersands (to distinguish it from the keyboard shortcut assignment), as illustrated in this example.

2. Click in the Press new shortcut key field, then enter the desired shortcut.
3. If the shortcut is currently assigned to another command, it will appear in the *Shortcut currently used by:* field as a warning.
4. Click the **Assign** button to assign the new shortcut.

To Remove a Keyboard Shortcut:

- Select the Category and Command from the *Keyboard* tab, then choose the shortcut from the *Current keys:* box and click **Remove**.

To Reset Keyboard Shortcuts to the Defaults:

- In the Keyboard tab, click the **Reset All** button.

Options

The *Customize* dialog provides several additional options for controlling the display and behavior of the interface (Figure 4-26).

- **Always Show Full Menus.** If disabled, menu items which are used rarely are not displayed. As a result, only an abbreviated menu is displayed. Click the down-arrow in the menu listing to view all commands for that menu.
- **Show Full Menus After Short Delay.** If selected, the abbreviated menu will be displayed for a moment, followed by the full menu. (Disabled if *Always Show Full Menus* is selected).
- **Animate Menu When Expanding.** Animates the menu as it expands from the abbreviated menu to the full menu. (Disabled if *Always Show Full Menus* is selected).
- **Highlight Rarely Used Menu Items.** Menu items which are not used often will be highlighted.

- **Display Menus With Shadows.** Renders a shadow around menus.
- **Large Icons in Toolbars.** When selected, large icons will be rendered in the toolbars. This is useful for very high resolution displays (or poor resolution eyeballs!)
- **Large Icons in Menus.** When selected, icons displayed along menu items will be drawn in large size.
- **Show Screen Tips on Toolbars.** When selected, displays tooltip information for toolbar icons.
- **Show Screen Tips on Menus.** When selected, displays tooltip information for menu items.
- **Show Shortcut Keys in Screen Tips.** When selected, displays the assigned shortcut key for a toolbar item as part of its tooltip information.
- **Menu Animations.** Controls the transition used to display menus as they expand.

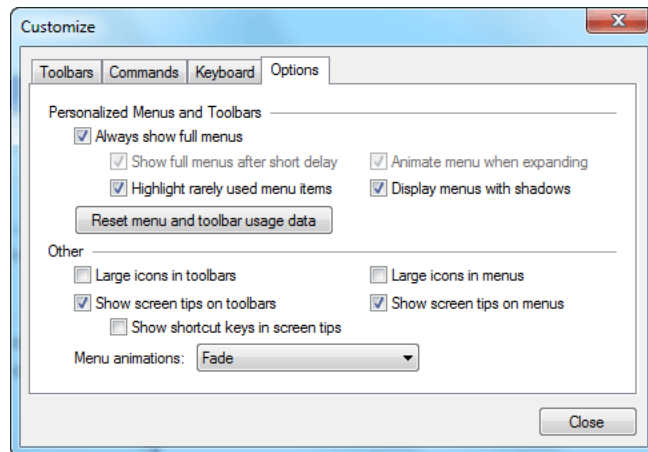


Figure 4-26. Changing UI options in the Options tab.

To Reset the Historical Usage Information for Menus and Toolbars:

- Click the **Reset menu and toolbar usage data** button.

Language Translation

SpatialAnalyzer comes with seven stock languages:

- English
- Chinese
- French
- German
- Italian
- Portuguese

- Russian
- Spanish

If you would like to add an additional language, SA has a language translation tool that can be used to translate menu items, dialogs, and other user interface elements. Language translation files are stored as .lan files which can be transferred to different computers and loaded into the interface. Several SA resellers also offer ready-made language translation files to customers.

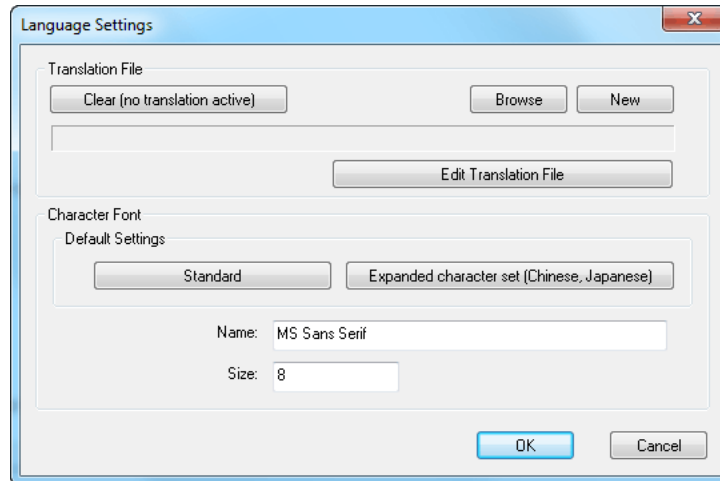


Figure 4-27. The Language Settings dialog.

Note: Language settings are saved with custom user profiles. If you attempt to create a new language translation file while a custom user profile is active, the user interface will be returned to the default.

The language translation interface (Figure 4-27) works by storing translations from English to a specified language. If a certain phrase has not been translated, it will remain in the default English language.


To Make A Stock Language Translation Active:

Note: After changing the active language, you may need to restart any running instrument interfaces for the new language to take effect.

1. Select **File>Language Translation**.
2. From the **Language Translation** dialog, select the desired language, then click the **Close** button. If the window indicates that a custom translation is active, it must be deactivated first before choosing a stock language.

The active stock language will also apply to instrument interfaces unless a specific language has been loaded into that interface.

You cannot modify a stock language—it is read-only. However, you can create a new translation based off of a stock language, as described below.

All custom translation is performed in the **Language Settings** dialog. To access this dialog, click the User Options  icon, and in the **Display** tab, click the **Custom Language File** button.

To Create a New Language Translation:

1. In the **Language Settings** dialog, click the **New** button.
2. In the dropdown, choose **Empty Language File** to create a new language file, or **Language File based on** to create a custom lan-

guage file based on one of the stock languages.

3. Assign a filename for the translation file and click **OK**.

SA will build a database of phrases to translate and present the NRK Language Translation Interface. If the translation is based off of a stock language file, you will find that many of the phrases are already translated.

To Load an Existing Language Translation File:

- In the **Language Settings** dialog, click the **Browse** button, and select a language translation file. The translations will immediately be applied to the user interface.

To Remove All Translations and Return to the Default Language of English:

- Click the **Clear (no translation active)** button.

To Edit the Active Translation File:

- Click the **Edit Translation File** button.

Character Sets

Certain languages (Asian languages in particular) do not use a traditional Latin character set (standard). To account for this, they may require an expanded character set in order to be displayed properly.

To Modify the Character Set Used for Dialogs:

- Click the **Standard** or **Expanded character set (Chinese, Japanese)** buttons.

To Modify the Font Type and Size Used for Dialogs:

- Enter a new font name or size into the dialogs.


Creating a Translation

The language translation interface ([Figure 4-28](#)) allows any phrase that is used in SA to be selected and translated. Once a translation has been added for a particular phrase, it will appear translated in the interface. The top list contains phrases that have not been translated. The bottom list contains phrases that have been translated, and lists both the original English and the translated phrase.

To Assign a Name for the Language:

- Type a language name in to the **Language Name** field, then click **Apply**.

To Search for an Untranslated Phrase:

 **Note:** The search field allows you to enter phrases with wildcard characters (*, ? and []). For example, to search for the phrases "Password Protection" and "Projection", enter **Pro?ection** in the search field.

1. In the top search field, type in a phrase. As you type, the first matching item will be highlighted.
2. To find the next result, click the **Next Missing Phrase** button again.

To Save the Current State of the Language File and Apply Changes:

- Click the **Save** button.

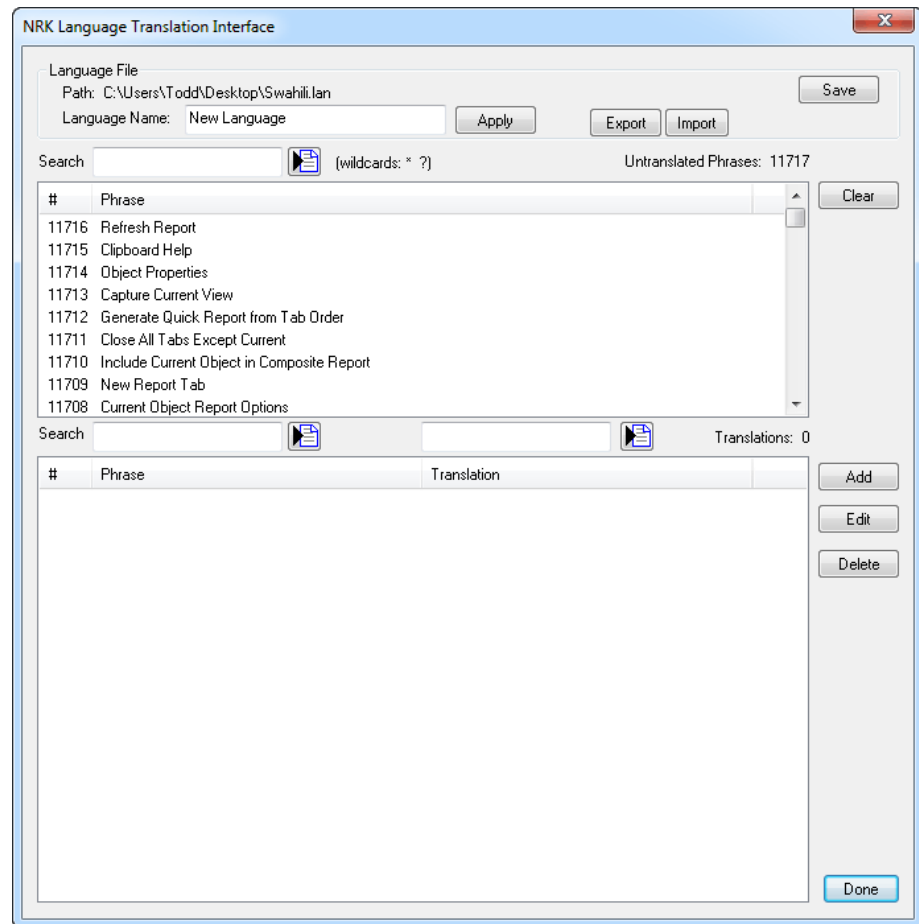


Figure 4-28. The Language Translation Interface.

To Apply a Translation to a Phrase:

- Select the phrase from the top list of untranslated phrases.
- Click the **Add** button.
- In the Translation Entry window, enter the translated phrase. To automatically assign an **ALT** keyboard shortcut to a menu item, precede the desired letter with an ampersand. To enter an ampersand in the phrase, use two consecutive ampersands (&&).
- When finished, click **OK**. The translated phrase will appear in the list at the bottom, and the phrase will disappear from the top list of untranslated phrases.

To Edit or Delete a Translated Phrase:

- Select the phrase in the bottom list of translated phrases and click **Edit** or **Delete**.

To Search for an Original Phrase (English):

1. Enter the search text in the lower-left search field. As you type, the first matching phrase will be highlighted. Wildcard characters (*, ? and []) may be used in this search field.
2. To find the next matching phrase, click the **Next English Phrase** button.

To Search for a Translated Phrase (Translated Language):

1. Enter the search text in the lower-right search field. As you type, the first matching phrase will be highlighted. Wildcard characters (*, ? and []) may be used in this search field.
2. To find the next matching phrase, click the **Next Foreign Phrase** button.

To Clear Untranslated Phrases From The Dialog:

- Click the **Clear** button. All untranslated phrases will be removed from the list.

Merging Language Files

Multiple language files can be merged into a single language file. Consider translating an instrument interface, which is performed in the instrument interface itself. This translation is saved into a separate translation file. You can merge an instrument interface translation file into the SA translation file so that a separate translation does not need to be applied to the instrument interface. (This is how the stock language files work).

To Merge Language Files:

1. From the **NRK Language Translation Interface** dialog, click the **Import** button.
2. In the file filter dropdown, change the file type to **Language File (*.lan)**.
3. Select the desired language file to merge, then click **Open**.
4. Confirm the merge procedure. The translated phrases from the language file will be merged into the current file.

Using .CSV Files For Translation

Once translations have been applied, converting one language to another is made easier with the ability to use .CSV files. Translated phrases (those appearing in the bottom list) may be exported to a two-column .CSV format, which allows you to simply change a translation to another language using a spreadsheet program such as Excel. This way, you can avoid having to click the **Edit** button for each

translated phrase.

To Export Translated Phrases to .CSV Format

1. Ensure the desired phrases to translate already have some sort of translation applied to them (so that they appear in the bottom list).
2. Click the **Export** button, and choose a file name for the .CSV file.

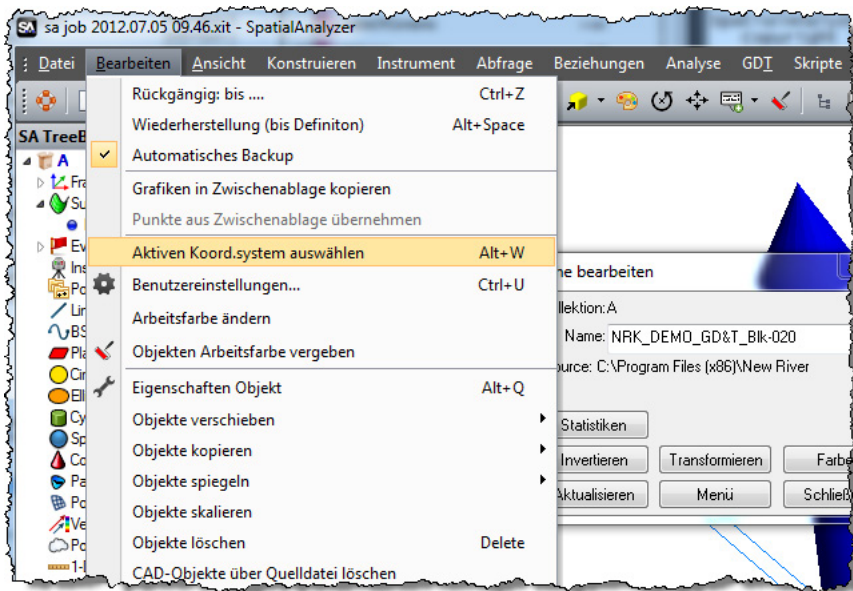
To Edit an Exported .CSV Translation File

1. In a program capable of working with the CSV format (such as Microsoft Excel or Notepad), open the .CSV file.
2. Leave the first column (*English Phrases*) alone, but change the second column (*Translated phrases*) to the desired language.
3. Save the .CSV file without changing its format.

To Import a .CSV Translation File

1. In the Language Translation Interface, click the **Import** button.
2. Choose the file to import and click **OK**. The new phrases will load into the translated phrases section of the window.

Figure 4-29. SA with a German language translation active.



The Graphical View

CHAPTER

5

The graphical view is arguably the most important single display in all of SpatialAnalyzer. It allows for graphical selection, provides visual feedback, and allows you to visualize your measurements and data.

The graphical view is drawn as a parallel projection—that is, lines of perspective do not converge in the distance but are instead parallel. This is done for a very good reason: there is no “vanishing point”, therefore objects appear the same size regardless of their distance from the viewpoint. From a practical perspective, this allows you to visually estimate size and position differences among objects without worrying about whether their distance from the viewpoint is affecting their apparent size.

Navigating the Graphical View

The graphical view provides a graphics-accelerated, three-dimensional rendered view of everything in your job. It allows you to inspect, select, and operate on objects graphically. In order to take advantage of this realistic view of your data, it’s important to be able to navigate the view with ease.

Probably the easiest way to get started is with a tutorial:

Graphical View Controls

Zoom

To Zoom In and Out:

- Hold down **Ctrl** and click-drag the mouse up or down, or
- Scroll the mouse wheel up/down, or
- Press the **PgUp/PgDn** key, or

- From the menus, choose **View>View Control>Zoom>In [Out]**, or
- Use the controls in the *Adjust View Components* window (see “[Adjust View Components](#)” on page 77).

To Zoom Into a Region of the View:

Hold down the left mouse button, drag the mouse to define a rectangle, and release the mouse button. The view will zoom into the defined region.

Autoscale

Autoscale will zoom and center the view so that all visible objects are displayed in the graphical view at once.

To Autoscale the View:

- In the main toolbar, click the **↕ Autoscale** button, or
- In the menu, select **View>View Control>Zoom>Autoscale**, or
- Press **Alt+A**, the default keyboard shortcut.

Pan

To Pan the View:

- Hold down the middle mouse button.
- Hold down the **Shift** key and drag the mouse in the graphical view, or
- Use the keyboard arrow keys (**Up**, **Down**, **Left**, and **Right**), or
- Select **View>View Control>Pan>Up/Down/Left/Right**, or
- Use the controls in the *Adjust View Components* window (see “[Adjust View Components](#)” on page 77).

To Center the View on a Point:

- Middle-click anywhere in the graphical view.

Rotate

To Rotate the View (Freeform):


- Hold down the right mouse button and drag, or
- Use the controls in the *Adjust View Components* window (see “[Adjust View Components](#)” on page 77).

The viewpoint can be rotated about an axis perpendicular to the screen (this is referred to as spinning the view).

To Spin the View:

- Hold down the **Ctrl** key and right-click-drag the mouse.

To Set a Specific Point as the Rotation Pivot Point:

1. On the Main Toolbar, click the  **View Rotation Center** button.
2. Click a point to set the pivot point.

Normally, SA chooses the entity closest to the center of the graphical view to become the pivot point for rotation. However, if desired you can select a specific point to serve as the rotation pivot point.

To Reverse the View Rotation Sense:

By default, SA uses the *Front of Sphere* view rotation mode. When rotating in this mode, the rotation center is considered to be at the center of a sphere, while the cursor is “stuck to” the front of the sphere and spinning it. This rotation sense can be reversed to *Back of Sphere* mode in the User options *Machine Configuration* tab. Change the *Front of Sphere Mode* setting.

Center In View

Tip: Centering the view on a point is an easy way to make that point the center of rotation.

If you have an item in the tree and are having trouble finding it in the graphical view, you can easily center the view on an object, point, instrument. (Does not apply to point groups, point clouds, vector groups, annotations, datums, feature checks, or dimensions).

To Center the View on an Item:

- Right-click the item in the tree or graphical view and select **Center In View** from the context menu.
- Alternatively you can use the *Ctrl+Middle* Mouse button.

Adjust View Components

If desired, you can control the viewpoint (zoom, rotation, and panning) through the *Point of View* dialog. To open the *Point of View* dialog, choose **View>View Control>Adjust View Components**.

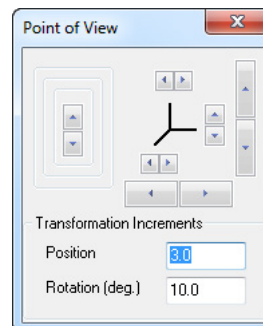


Figure 5-1. The *Point of View* dialog has controls for zoom, rotation, and panning, as well as transformation dialog increment fields.

Each of these buttons works similar to the manual view operations, except that the rotation buttons rotate about axes that are horizontal, vertical, or perpendicular to the screen.

This dialog also contains fields for controlling how much an object’s

position or orientation changes when an up/down arrow is clicked in a transformation dialog.


- **Position.** Controls the increment applied to positional values.
- **Rotation (deg).** Controls the angular increment applied to angular values.

Jumping to Previous/Next Views

It is often useful to jump back and forth among various recent views. As you navigate throughout the graphical view, if you remain at a given viewpoint for at least one second, SA will store the viewpoint. You can then quickly navigate back and forth between previous and next views.

These views are not saved with the SA file. If you manually change views while at an earlier view in the list, later views are removed from the list.

To Navigate Among Previous Views:

 **Note:** The smoothness of the transition between views is controlled in the User Options>Display tab with the *View Animation Steps* option.

- To jump to a previous view, press **Ctrl+Alt+←** or select **View>Viewpoint Control>Go to Prev View** from the menus.
- To jump to the next view, press **Ctrl+Alt+→** or select **View>Viewpoint Control>Go to Next View** from the menus.

Set Viewpoint From Frame

The viewpoint can be set from any existing frame using the **View>View Control>Set Viewpoint From Frame** command. The resulting view will look down the +Z axis.

To Set a Viewpoint From a Frame:

1. Select **View>View Control>Set Viewpoint From Frame**.
2. At the prompt, select the frame you'd like to use for orienting the view.

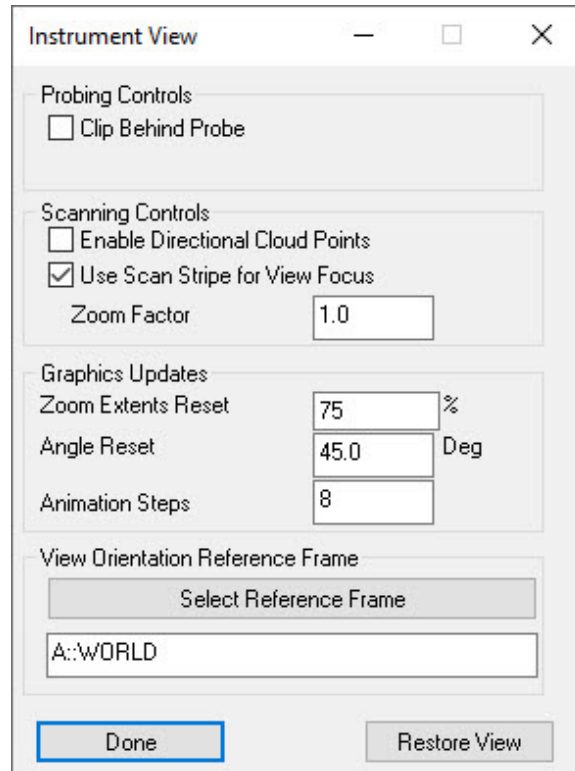
Set Viewpoint From Instrument Updates

The viewpoint can be updated dynamically based upon the current probe position and orientation (for instruments that provide 6-DOF information such as portable CMM arms). This enables the capability to “fly around” a part using an instrument, or to update the viewpoint while measuring such that the probe is always visible in the graphics.

This mode can also be extremely helpful when scanning a part.

Several configuration options are available (see [Figure 5-2](#)).

Figure 5-2. View from Instrument Updates Configuration page



Probing Controls

- **Clip Behind Probe.** This check box enables graphical clipping (use of a clipping plane) behind the probe position. This provides visibility even when objects are blocking the view.
- **Automatic Zoom When Trapping.** This option adds zooming to the graphic control. This is linked to feature trapping such that the view zoom depth is set to fit the nominal feature and the current probe position. As the probe is moved toward the nominal feature the view will zoom in on the operation.

Scanning Controls

- **Enable Directional Cloud Points.** With this option enabled the directional flag for cloud points will be set such that cloud points are only viewed from their probing direction. For more information see “[Directional View Control](#)” on page 550.
- **Use Scan Stripe for View Focus.** With this option enabled the camera orients differently. Rather than directly looking down the probing axis of the scanner, it centers the view on the scan strip as you measure. When measurement stops for 3 seconds it then zooms to fit the entire point cloud. Use the **Zoom Factor** control box to adjust the how tightly the view tracks the scanning process.

Graphic Updates

- **Zoom Extents Reset.** When the probe or scan line approaches the edge of the screen the view will adjust to re-cent on the data. This value controls how close you can get to the edge of the graphics before the view will adjust.
- **Angle Rest Threshold.** This defines how often the view is updated and is the key control when using a 6D probe. The current orientation of the probing vector is compared with the last saved position and when the angle is greater than the specified value the graphics will update orientation. Otherwise they will remain fixed.
- **Animation Steps.** The transition from one view orientation to another can be disorienting without some degree of animation. But too much slow performance. Start low and raise this threshold until you find an optimum for your application.

View Orientation Reference Frame. The view tries not to tip the graphics. This maintains the expectation that gravity is always up. We use the convention that the WORLD frame's Z-axis is most likely oriented with gravity and that the view should not rotate significantly from this vertical orientation. However, that might not be the case and a different frame can be selected for this purpose.

Example Use Cases:

1. *Measuring with a Laser Tracker and an 1.5" SMR.* In this scenario the only control you need is an ability to keep the reflector within the graphics. You may also want to check the box to Automatic Zoom when Trapping. This allows the view to zoom out to encompass the nominal target and the allows the zoom to increase as you move the probe closer to it. It also helps to more precisely measure that feature.
2. *Measuring with a 6D probe, such as a T-probe.* In this case the view is set by the probe such that the user is looking straight down the probe tip. A 6D probe can also be used as a navigation device that cuts through features in the graphics showing those objects behind. To do so use the Click Behind Probe option.
3. *When Point Cloud Scanning.* When scanning a point cloud a slightly different visual focus is needed. The camera does not track exactly with the direction the probe is pointed but rather follows the current scan line and center the graphics based upon it. To do so use the Use Scan Stripe for View Focus option.

Set Viewpoint From Object

Any object's orientation can be used as the basis for a viewpoint. The resulting view will look down the -Z axis of the object.

To Set a Viewpoint From an Object:

1. From the menu, select **View>View Control>Set Viewpoint From Object**.
2. At the prompt, select the object to use for establishing the viewpoint.

Set Viewpoint From Surface Point

You can also set the viewpoint to look at a surface along its normal.

To Set a Viewpoint From a Surface Point:

1. From the menus, select **View>View Control>Set Viewpoint From Surface Point**.
2. At the prompt, click on the surface where you'd like to look.

View Clipping

Circumstances might arise in which you want to clip portions of the graphical view. Two major examples include:

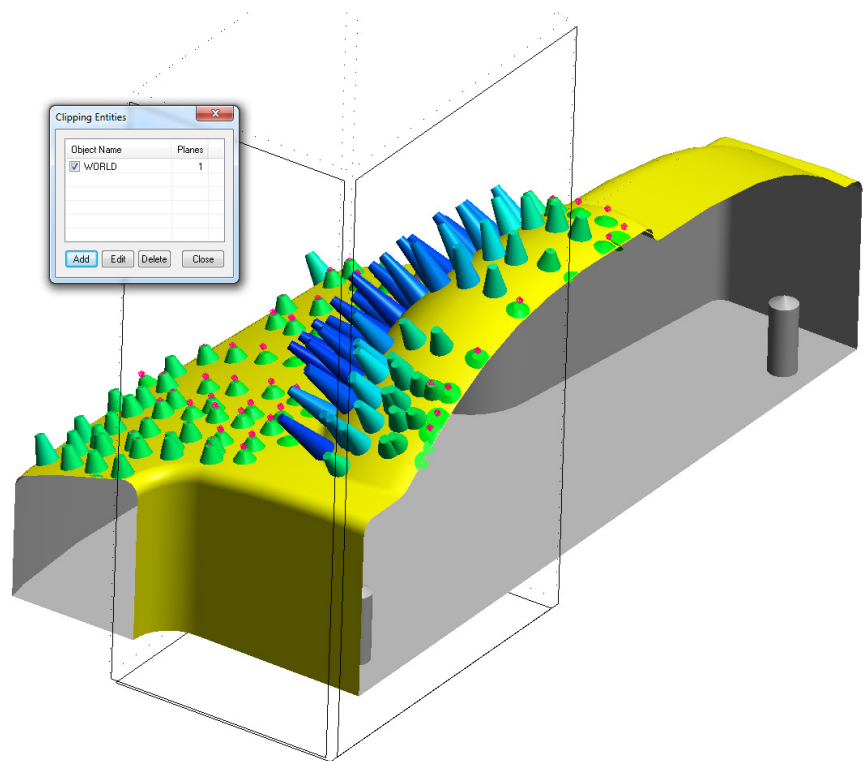


Figure 5-3. Viewing a model, measurements, and a vector group with an active clipping region.

- Removing data (such as point cloud data) outside of a region in order to simplify the view. This may be done for reporting rea-


sons (to obtain a clear screenshot for a report) or just to be able to more easily work in the view. For example, if you have point cloud scans of the inside and outside of a building, you might clip away the near “wall” so that you can view the data on the inside of the building without also seeing the wall data.

- Displaying a “section cut”. You might do this to “cut away” portions of your CAD or data in order to display only a region of interest, as in [Figure 5-3](#).

A clipping region is defined by a box that can be moved or have any of its sides scaled along a coordinate direction. The orientation of the region (its rotation in space) can be defined by any object, and each of the six faces of the box can have clipping enabled or disabled.

To Set A Clipping Region:

1. From the menus, select **View>View Control>Clipping Planes** or press **Alt-X**.
2. In the Clipping Entities dialog, click the **Add** button.
3. Select any object to define the initial position and orientation of the clipping box.
4. Mouse over the center of a face (it will turn black) and click it to toggle that face’s clipping plane.
5. Mouse over the center of any edge (an icon will appear) and drag to move the clipping region along that edge’s direction.
6. Mouse over the end of any edge (an icon will appear) and drag to grow or shrink the region along that edge’s direction.
7. When you’re happy with your region, click the **Close** button, or leave the dialog open if desired.

 **Note:** Clipping regions are tied to the object used to define the position and orientation of the clipping cube. If the object moves or rotates, the clipping planes will move or rotate along with it.

To Assign Numeric Boundaries to a Clipping Region:

1. Create the clipping region as described above in “[To Set A Clipping Region:](#)” on page 82.
2. In the **Clipping Entities** dialog, double-click on a clipping region or select it and click the **Edit** button.
3. In the resulting **Clipping Volume Properties** dialog, check the regions you’d like to be active, and enter desired values for each clipping plane.

To Delete a Clipping Region:

- In the **Clipping Entities** dialog, select the clipping region to delete, then click the **Delete** button.

You can define as many clipping regions as you’d like. Any of them can be activated or deactivated.

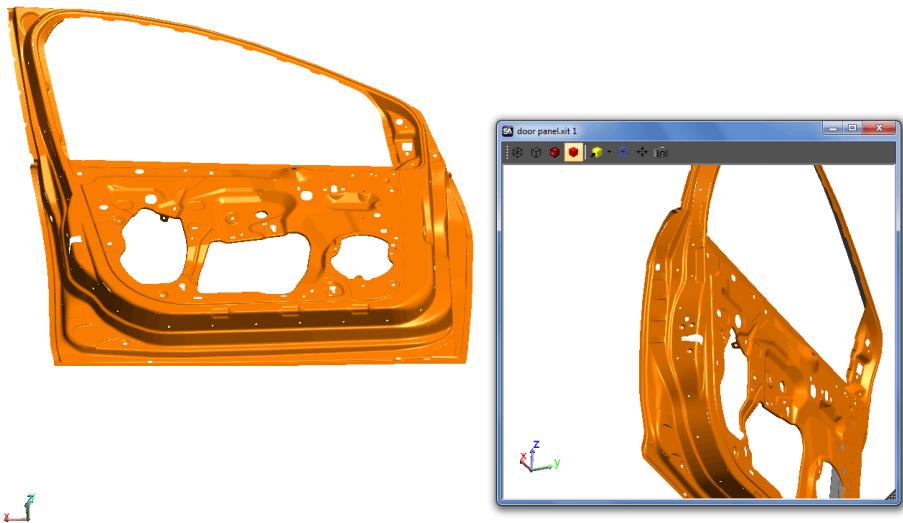
To Activate or Deactivate a Clipping Region:

- In the *Clipping Entities* dialog, check (to activate) or uncheck (to deactivate) the clipping region's checkbox.

Multiple Graphical Views

Multiple graphical views (see [Figure 5-4](#)) can be displayed by selecting **View>New Graphical View Window**. This allows you to simultaneously view and interact with your job from multiple perspectives.

Figure 5-4. An auxiliary view has been opened to view and interact with this door panel from a different perspective.



Auxiliary graphical views can be rotated, positioned, and rendered independent of the primary graphical view, and they also support selection. Hold the **Ctrl** key while changing the render mode, auto-scaling, or creating a snapshot of the screen to have the setting apply across all open graphical view windows.

You can add as many auxiliary views as you'd like, although more than two auxiliary views begins to become unmanageable without multiple computer displays.

Heads-Up Display

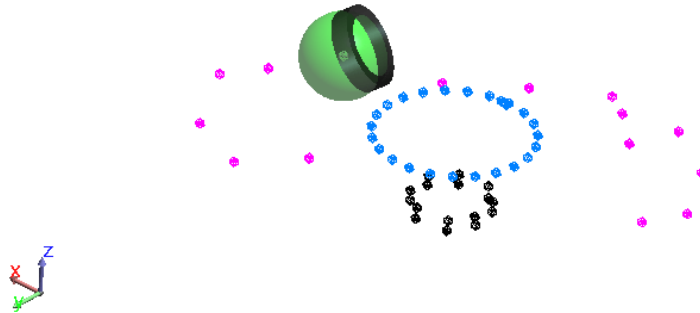
The graphical view has a Heads-Up Display (HUD) that can display a variety of measurement information (see [Figure 5-5](#)), including:

- Feature Name (if trapping measurements to a feature)
- Last computed Feature RMS (for geometry relationships only)
- Current Point Number (if trapping to a feature with a prescribed number of measurements)
- Point coordinates
- Point name (when not trapping)

- RMS error (if the RMS error of a sampled measurement exceeds the warning threshold)


> Top Bore Face **RMS: 0.0019**
 [2 of 6]
 X = 85.9560
 Y = 129.4121
 Z = -13.3793

Figure 5-5. The Heads-Up Display.



User Options Controls

To modify or enable the Heads-Up Display you can access the User Option controls as follows:

1. From the SA Toolbar, click the User Options  icon.
2. In the *Display* tab, click the **Heads Up Display Settings** button (Figure 5-6).

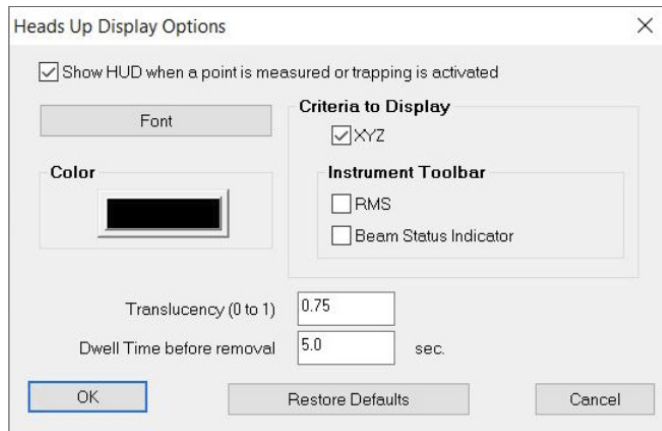


Figure 5-6. The *Heads Up Display Options* dialog.

- In the HUD options dialog, enable or disable the *Show HUD when a point is measured or trapping is activated* checkbox.

Font Size and Color:

The Font, Font Color, and text transparency can also be set as needed.

- In the HUD options dialog, click the **Font** button, then choose a new font from the dialog.

- In the HUD options dialog, click the color and choose a new color.
- In the HUD options dialog, set the *Translucency (0 to 1)* field to a value between 0 (fully transparent) and one (fully opaque).

Display Duration

In the HUD options dialog, set the *Dwell Time before removal* option (in seconds) to control how long the text is displayed graphically after a measurement is taken.

The *Beam Status Indicator*, displayed as a colored rectangle around the graphic view, reflects the beam status of the live instrument in the job file. Its used to supliment the toolbar status indicator light for greater visibility when measuring away from the computer. The Beam Status Indicator is available only in Toolbar Mode with a single live tracker in the job file.

XYZ

By default the XYZ position of each newly measured point will be displayed along with the new point name.

An RMS warning threshold can also be set to display for this point through the **alarm** button in the tracker interface (Figure 5-7).

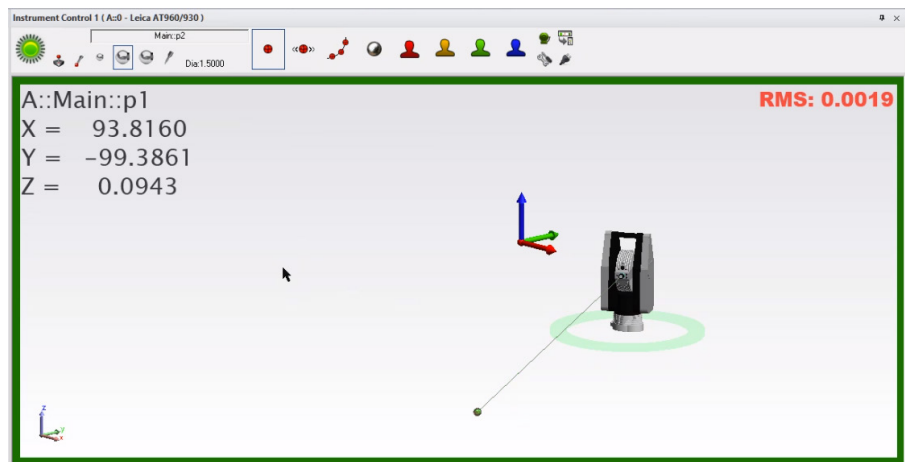


Figure 5-7. Default HUD display showing a newly measured point's name, XYZ position, and RMS error

GR- Feature Trapping

When Trapping is initiated for a relationship, or GD&T datum or feature check, the name of the active feature will be displayed in the HUD along with the number of measurements associated with it in brackets [#].


The *RMS* selection in the HUD options dialog is used to control the display of the current RMS of the active Relationship and will be displayed below the feature point count. Note that the individual point RMS warning is displayed on the right hand side of the graphics and

is set within the instrument toolbar controls.

Saved Viewpoints

You can save custom viewpoints in order to quickly return to a preset view orientation.

To Save a Custom Viewpoint:

1. Orient the view to the desired position.
2. Click the  **Views** icon in the Main Toolbar, or select **View>View Control>Set Viewpoint**.
3. In the **View List** dialog, click **Add Current**, as depicted in [Figure 5-8](#).

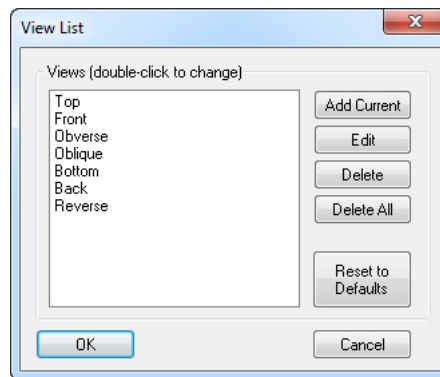


Figure 5-8. The View List dialog.

4. In the **Preset View** dialog, type a name for your custom view ([Figure 5-9](#)).

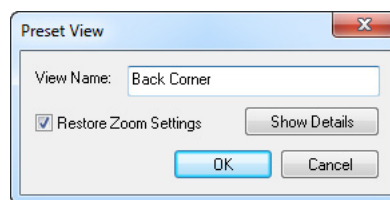
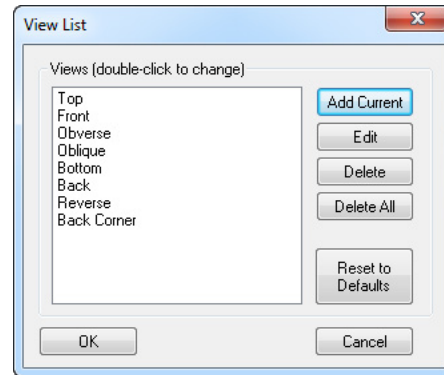



Figure 5-9. Entering the name for a custom view.

5. If you would like the view to restore the current zoom settings, select **Restore Zoom Settings**.
6. To view position and orientation information about the view, click the **Show Details** button.
7. Click OK. Your custom view will now be available in the **View List**:

Figure 5-10. The custom view has been added to the view list.



To Restore a Custom Viewpoint:

- Click the down-arrow next to the  **Views** icon and pick the view from the list, or
- In the **View List** dialog, double-click the desired view or select it and choose **OK**.

To Edit a Custom Viewpoint:

- In the **View List** dialog, select the view and choose **Edit**.

To Delete a Custom Viewpoint:

- In the **View List** dialog, select the view to delete and click **Delete**.

To Delete All Viewpoints:

- In the **View List** dialog, click **Delete All**.

To Restore the Default Viewpoints:

In the **View List** dialog, click **Reset to Defaults**.


By default, saved view directions are based on the orientation of the world coordinate frame. For example, if a view is saved looking along the world frame's Z-axis, then when the view is restored, it will again be looking along the Z-axis of the world frame, regardless of the active frame. However, you can choose to restore saved view directions relative to the working frame instead. By doing so, restoring the view described above would look along the Z-axis of the active frame.

To Restore a Viewpoint Based on the Working Frame:

- In the *User Options>Display* tab, check the *View Relative to Working Frame* option.

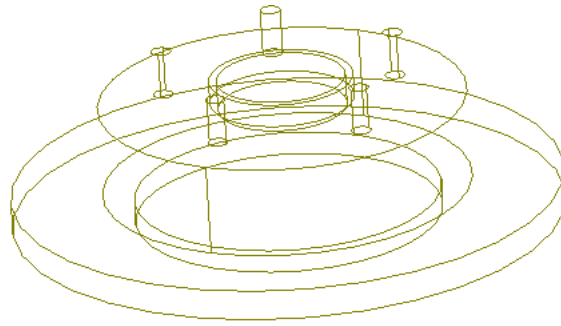
Rendering Modes

SA provides four rendering modes for viewing geometry: Wireframe, Hidden Line Removed, Solid+Edges, and Solid. The rendering modes are selected from the Main Toolbar.

-  **Wireframe.** Geometry is displayed as edges only, and hid-

den edges are visible, as depicted in [Figure 5-11](#).

Figure 5-11. Wireframe rendering mode.



Note: By default, silhouette edges are not rendered in Hidden Line Removed mode. To draw silhouettes, in the *User Options>Display* tab, check *Draw Silhouettes in Hidden Line Removed Mode*.


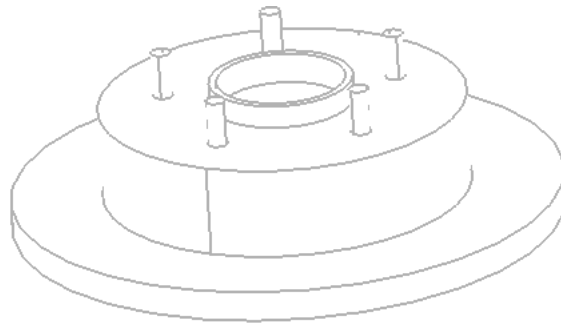
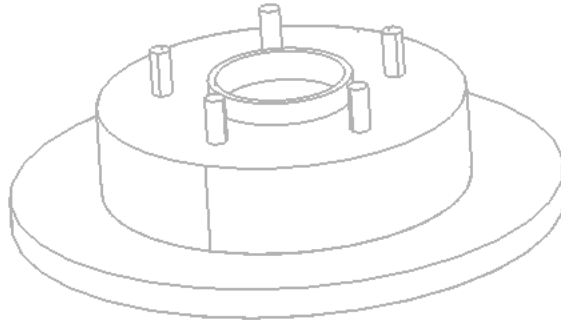
-  **Hidden Line Removed.** Geometry is displayed as edges only, but hidden edges are not displayed, as depicted in [Figure 5-12](#).

Figure 5-12. Hidden Line Removed rendering mode.



Here is the same view with silhouettes turned on:

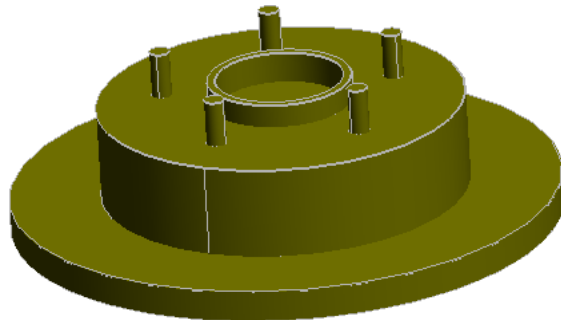
Figure 5-13. Silhouettes have been drawn in this hidden line view.



Note: Because of the necessary computation time, each time the view is rotated, you need to click the **Hidden Line Removed** icon to redraw silhouettes if you want to see them. The *Draw Silhouettes in Hidden Line Removed Mode* setting is saved with the SA file.

-  **Solid+Edges.** Solid surfaces are rendered in addition to edges, as shown in [Figure 5-14](#).

Figure 5-14. Solid+Edges rendering mode.



Note: When a file is first opened, the rendering method always defaults to wireframe mode. This prevents a potentially lengthy rendering process from starting immediately after a file is loaded.


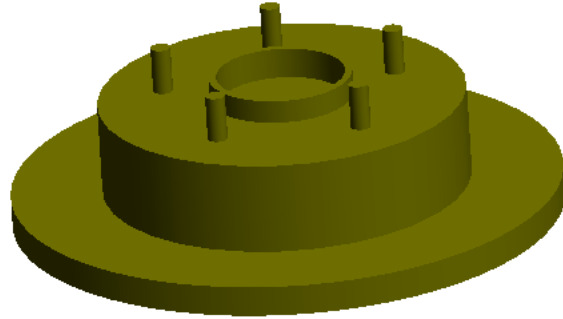
-  **Solid.** Solid surfaces are rendered without edges, as depicted in [Figure 5-15](#).

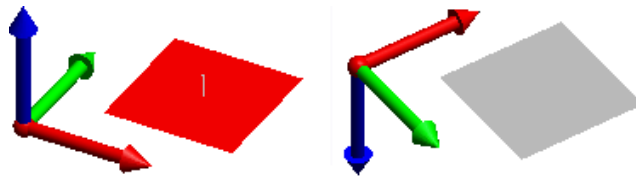
Figure 5-15. Solid rendered mode.



Surface Backsides

Every surface in SA (in addition to other entities, such as planes) has a positive and a negative side (see [Figure 5-16](#)). In order to properly account for a probe's target offsets, it is important that all comparisons be performed against the positive side of a surface. In order for you to determine the positive side, SA renders the negative side of surfaces in a special, user-definable color (a light gray by default).

Figure 5-16. The same surface as viewed from the front (left) and back (right) with *Highlight Surface Back-sides* enabled.



Although the default gray is usually sufficient, you can choose a custom color to denote surface backsides. It is best to choose an unusual color which is not one of the typical working colors.

If desired, you can disable surface backside highlighting so that both the front and back side of surfaces appear in the same color.

To Disable Surface Back-Side Highlighting:

Note: The *Highlight Surface Back-sides* setting is saved as a machine setting.

- In the *User Options>Display* tab, uncheck the *Highlight Surface Back-sides* option.

To Change the Color of Surface Back-sides:

1. Ensure the *Highlight Surface Back-sides* option is selected.
2. Click the **Color** button, then choose a new color.

Background/Highlight Color

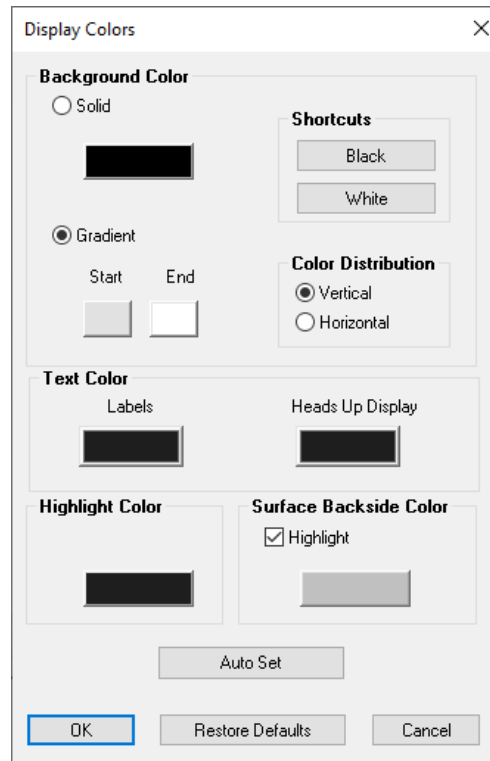


Figure 5-17. The Display Colors dialog.

The background of the graphical view can be set to a solid color or a gradient, and you can also adjust the highlight color used in SA.

To Adjust Background Color Settings:

- Click the  Background icon in the Main Toolbar.

To Change the Background Color to a Solid Color:

- Click the *Solid* radio button, then click the color swatch to change the color.
- Click the **Black** button to change the background color to black.
- Click the **White** button to change the background color to white.

To Change the Background Color to a Gradient:

Tip: For fastest rendering, set your background to a solid color. Solid backgrounds redraw faster than gradient backgrounds.

- Click the *Gradient* radio button, then choose a start and end color. Specify whether you'd like a horizontal or vertical gradient.

The highlight color is used when an object is selected or when it is explicitly highlighted using the **Highlight** command.

To Change the Highlight Color:

- Under the *Highlight Color* section, click the color swatch and choose a new color, or
- Click the **Auto Set** button, which will automatically select a highlight color that has high contrast when compared to the cur-

rent background color.


To Restore the Factory Default Background/Highlight Colors:

- Click the **Restore Defaults** button.

Hardware Acceleration

If your graphics card supports it, SpatialAnalyzer provides OpenGL™ hardware acceleration for the graphical view. This option is enabled by default, and provides a significant performance increase—particularly with complex models and large amounts of geometry.

To Enable or Disable Hardware Acceleration:

1. Open the User Options dialog by clicking the  icon.
2. On the *Display* tab, click the **Performance Options** button.
3. Check or uncheck the *Use Hardware Accelerated Rendering if Available* option (Figure 5-18).

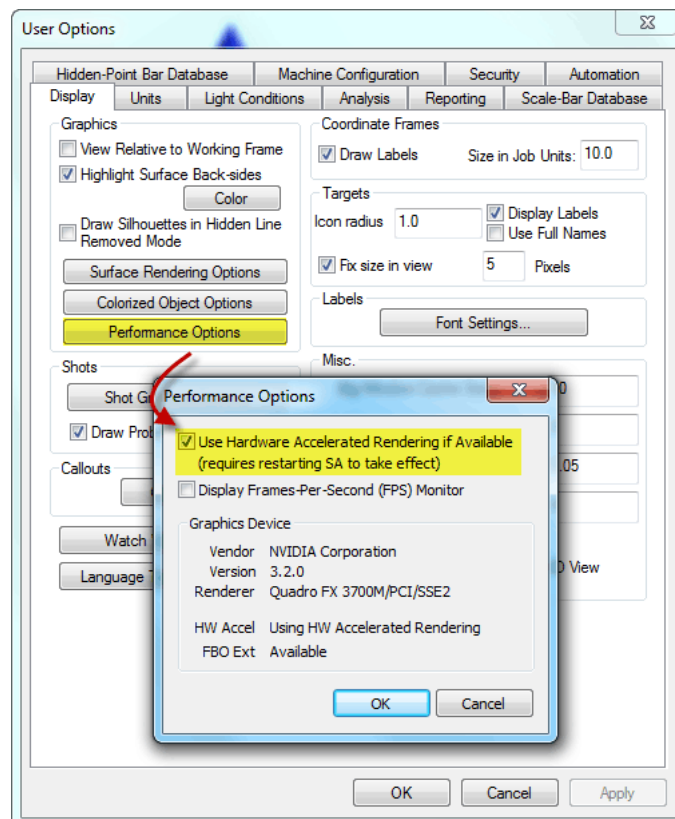



Figure 5-18. Enabling hardware acceleration for faster graphics performance.

 **Note:** Some graphics cards may cause visual artifacts with hardware-accelerated rendering enabled. If you see these artifacts, first ensure that you have the latest drivers for your graphics card. If the artifacts persist, disable hardware acceleration.


After changing the hardware acceleration option, it is necessary to exit SA and restart for the new setting to take effect.

SA has a built-in Frames-per-second (FPS) monitor that you can use to determine whether hardware acceleration is improving rendering performance in your job (see Figure 5-19). The FPS monitor displays a running average of the rendered frame rate. The progress bar under

the frame rate shows how “full” the measurement buffer has become. For the most accurate value, continue rotating the view until this buffer is nearly full.

To Test the Effectiveness of Hardware Acceleration on Your Machine:

Note: You will not see a framerate that exceeds the refresh rate of your display—typically around 60 fps.

1. Open the User Options dialog by clicking the  icon.
2. On the *Display* tab, click the *Performance Options* button.
3. Select the *Display Frames-Per-Second (FPS) Monitor* option.

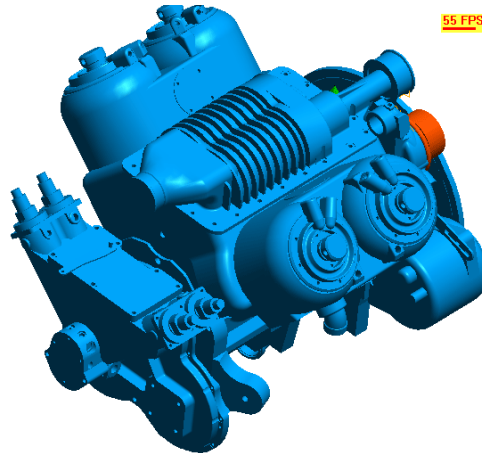


Figure 5-19. The Frames-per-second (FPS) monitor indicates how quickly the graphical view is being redrawn.

Surface Rendering Options

Note: Surface rendering options only affect the depiction of the CAD in the graphical view. In calculations, the exact surface is always used.

CAD surfaces (sometimes referred to as B-Reps or NURBS—Non-rational Uniform B-Splines) are mathematically defined and therefore have an “exact” representation. However, curved surfaces must be converted into a series of approximated flat faces in order to be rendered in the graphical view. The quality of this approximation is determined by the Surface Rendering Options (see [Figure 5-20](#)).

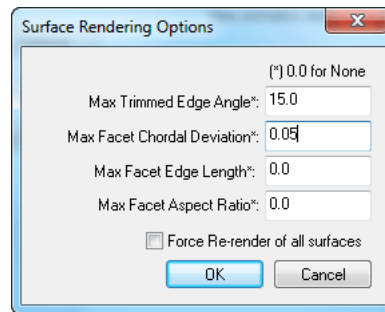


Figure 5-20. The Surface Rendering Options dialog.

If you find that your curves and surfaces are not being represented accurately enough for your liking, or if you see “tearing” in your CAD surfaces, you can modify the surface rendering options.

To Change the Surface Rendering Options:

- In the User Options>Display tab, click the *Surface Rendering Options* button. Choose the desired settings, then click *OK*.

The adjustable options that affect surface rendering are as follows:

- **Max Trimmed Edge Angle.** Specifies the maximum length of edge segments for trimmed edges. A lower value gives a higher quality representation. Use a value of 0 to default to a low-quality setting.

Figure 5-21. Max trimmed edge angle: 65 (left) and 15 (right).



Note that this setting also applies to spline curves:

Figure 5-22. Max trimmed edge angle: 125 (left) and 5 (right).



- **Max Facet Chordal Deviation.** Specifies the maximum allowable distance between a rendered edge and the surface with which it is associated. Use a value of 0 to default to a low-quality setting.

Figure 5-23. Max facet chordal deviation of 0.000 (left) and 0.001 (right).

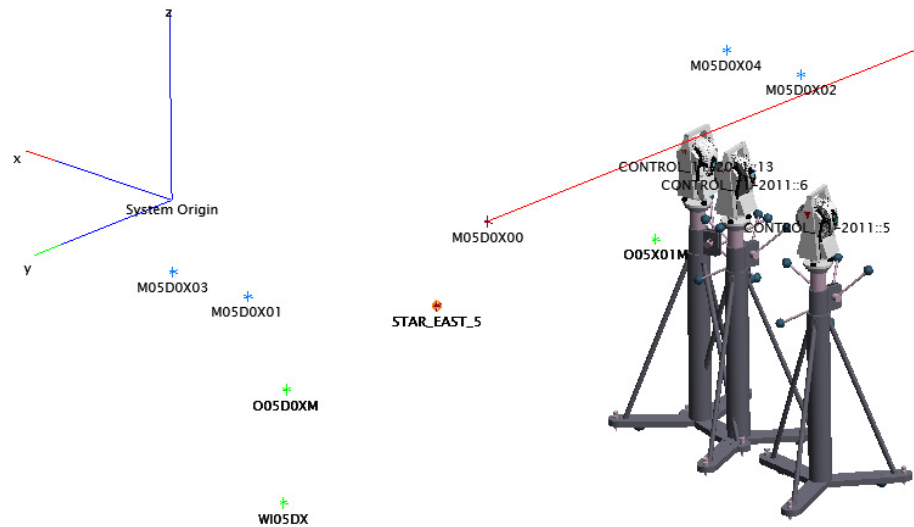


When laying out points on an object's vertices, this setting affects how many points are constructed.

- **Max Facet Edge Length.** Specifies the maximum allowable length of each edge segment.
- **Max Facet Aspect Ratio.** Specifies the maximum aspect ratio (length vs. width) of each rendered facet.
- **Force Re-render of all surfaces.** Checking this option will force SA to recalculate surfaces using the latest settings in the window. If unchecked, and surfaces have already been rendered, then the settings will not take effect until the file is reopened.

Labels


Figure 5-24. Point and object labels are visible in this view.



Labels can be turned on in the graphical view for displaying names of points, objects, and instruments (see [Figure 5-24](#)). This is often useful for visually distinguishing different objects, but can become overwhelming in complicated files—so you can toggle label visibility.

You can specify the font with which to display labels, as well as the text size.

To Change the Font Used for Labels:

1. Click the User Options icon  in the toolbar.
2. In the *Display* tab, click the **Font Settings...** button in the *Labels* box.
3. Select the desired font, style, size, and color (the other options will not apply).

Labels can be controlled individually for points (which also displays instrument labels) or objects.


To Toggle the Display of Labels:

Tip: Displaying a large number of labels simultaneously in the graphical view can significantly slow down display performance. Hide items not in use or turn labels off to speed up display speed.

- **Points and Instruments.** Toggle **View>Show Point Labels** or press **Alt+L** (default keyboard shortcut).
- **Objects.** Toggle **View>Show Object Labels**.


Point labels can display just the target name (such as **STAR_EAST_5** in [Figure 5-24](#)) or can display the entire collection, group, and target name (such as **Controls::Network::STAR_EAST_5**).

To Toggle the Display of Full Point Names:

1. Click the User Options icon  in the toolbar.
2. In the *Display* tab, check or uncheck the *Use Full Names* checkbox.

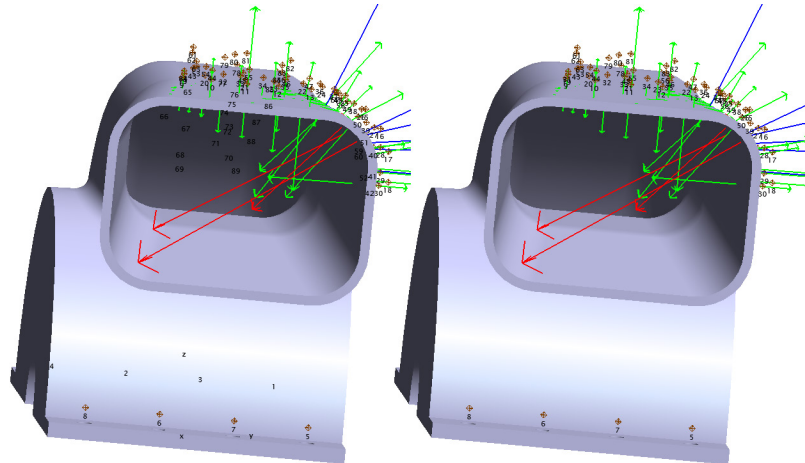
To Display Instrument Labels Without Point Labels:

Note: Hiding obstructed labels is computationally intensive and can slow down the graphical view considerably. For a more efficient method, simply hide the points.

1. Click the User Options icon  in the toolbar.
2. In the *Display* tab, uncheck the *Display Labels* checkbox.
3. Ensure that **View>Show Point Labels** is turned on.

When point labels are enabled, by default they are displayed even when the corresponding point is hidden behind a surface. To avoid confusion, you can set the graphical view to hide labels for points that are not visible from the current point of view.

Figure 5-25. Hide obstructed labels disabled (left) and enabled (right).




To Hide Labels for Hidden (Obstructed) Points:

- Within the Visibility section select **Hide Obstructed Labels**.

When point labels are displayed, by default SA will label the axes of coordinate frames as X, Y, and Z. This can be disabled.

To Disable Labeling of Frame Axes:

1. Click the User Options icon  in the toolbar.
2. In the *Display* tab, uncheck the *Draw Labels* checkbox in the *Coordinate Frames* box.

The Hover Window

If you are looking at an item in the graphical view, finding that item's name—or where it is located in the tree—can be a difficult task, particularly if the file is complicated and has lots of objects. The **Hover** window is a floating dialog, so it can be left up and “floating” over the graphical view. When you hover over an item in the graphical view, the Hover window will indicate the collection, name, type, and location of the item in the tree, the coordinates of a point (in Cartesian, Cylindrical, or Spherical coordinates, the deltas for a vector, or the number of measurements associated with an instrument.

To Toggle Display of the Hover Window:

Do one of the following:


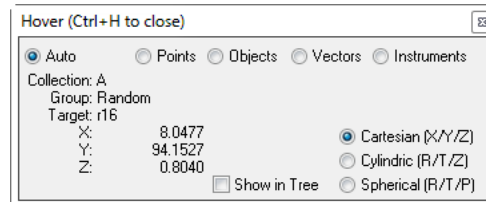
- From the SA Ribbon Home tab (select **Visibility>Hover (nearest item)**), or
- Press **Ctrl+H** (the default keyboard shortcut), or
- From the W's *Analysis* tab, click the Hover Window 

Figure 5-26. The *Hover* window showing the name and coordinates of a point.



The *Hover* window has a filter that you can apply to indicate what can be “hovered”.

- **Auto.** No filter is applied—SA will provide information for points, objects, vectors, or instruments—whatever you roll over.
- **Points.** Information will only be displayed when the cursor is rolled over a point.
- **Objects.** Information will only be displayed when the cursor is rolled over an object.
- **Vectors.** Information will only be displayed when the cursor is rolled over a vector.
- **Instruments.** Information will only be displayed when the cursor is rolled over an instrument.

To Set the Hover Window Filter:

- With the *Hover* window displayed, choose either the *Auto*, *Points*, *Objects*, *Vectors*, or *Instruments* option.

To Find an Item in the Tree:

1. Activate the *Hover* window and ensure that the *Show In Tree* option is selected.
2. Hover the mouse cursor of the item of interest. The tree will be expanded and auto-scrolled to reveal the item, and it will be selected.
3. To keep the selection active and the tree expanded, turn off the hover window while hovering over an item by pressing **Ctrl+H**.

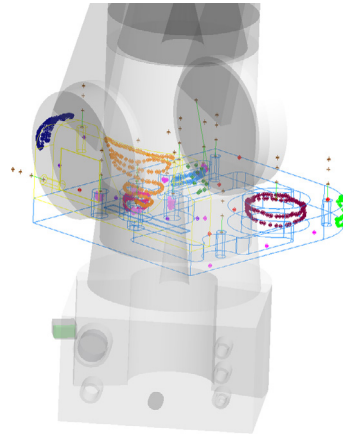
To Change the Coordinate System for Viewing Point Coordinates:

- In the *Hover* window, choose the *Cartesian*, *Cylindric*, or *Spherical* option.

Object-Specific Rendering Modes

It is often useful to make a specific object or instrument in the graphical view partially transparent (translucent) or to render it differently than other objects. For example, if a measured part has an area of insufficient material, its corresponding vector group will point into the CAD model. As a result, this vector will be nearly invisible when the CAD is rendered in Solid, Solid+Edges, or Hidden Line Removed mode. As another example, depending on viewing angle, sometimes the instrument's graphical representation, a surface, or the active coordinate frame will obstruct the work area that you're most interested in (see [Figure 5-27](#)). A third case might involve a surface that you want to see through. While there is always the option of hiding the obstructing items, often times it is desirable to keep them visible. In such circumstances, translucency allows you to avoid hiding the objects, but still be able to see through them to the work area.

Figure 5-27. Viewing the work area through a translucent instrument.



To Toggle Instrument Translucency:

- Select **View>Translucent Instruments**.

To Toggle Frame Translucency:

- Select **View>Translucent Working Frame**.

Surfaces and primitive objects can be set to a specific level of translucency or can be independently set to render in solid or wireframe mode.

To Set the Rendering Mode of an Individual Surface or Primitive Object:

1. Right-click the object in the tree or graphical view.
2. Select **Translucency**.
3. Select a desired rendering mode. If *Translucent* is selected, choose an opacity level with the slider.

To Set the Rendering Mode of Multiple Surfaces or Primitive Objects:

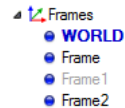
1. Select **View>Object Visibility>Translucency>Individual Objects** or **>All**

Objects as desired.

2. If **>Individual Objects** was selected, select the objects of interest and press **Enter**.
3. Select a desired rendering mode. If *Translucent* is selected, choose an opacity level with the slider.

Visibility

Figure 5-28. The frame named **Frame1** in this collection is hidden.



All of the items listed above can be shown or hidden. Hiding an item will remove it from the graphical view, and (unless it is a folder or collection) will dim the item's name in the tree to gray (see [Figure 5-29](#)).

Hiding items not in use has a number of benefits, chief of which are:

- The graphical view becomes less cluttered, and it's easier to work with the items of interest.
- If many items are being displayed at once, hiding those not in use will speed up the graphics considerably.
- Graphical selection becomes easier, because there are less items to pick incorrectly.

To Toggle An Individual Item's Visibility:

- Right-click the item in the tree or graphical view and toggle the **Show** option.

To Hide or Show All Items In a Container or Category:

- Right-click the container (collection or folder) or category in the tree and select **Hide All** or **Show All**.

To Hide or Show Individual Points:

- Right-click the point in the tree or graphical view and toggle the **Show** setting to toggle visibility of individual points, or

Select **View>Show Points** or **>Hide Points**, then select the points to show or hide.

To Hide or Show Multiple Objects:

- Select **View>Object Visibility>Hide Objects** or **>Show Objects**.

To Show All Objects:

- Select **View>Object Visibility>Display All**.

You can also filter the graphical view to display items only from a specific source. This source can be either SA itself, or an imported CAD file. This allows you to immediately hide objects imported from one

or more CAD files, or to immediately hide native SA objects.

To Filter The View By CAD Source:


1. Select **View>Object Visibility>Filter By CAD Source**.
2. In the *Show Selected Sources* dialog, select the sources you'd like to show, then click **OK**.
3. The filter will apply until you change it again.

Miscellaneous Settings & Commands

View Animation Steps

Sometimes, switching from one saved viewpoint to another can be disorienting. If you find that switching from one saved view to another is too jarring, you can increase the number of animation steps as the view transitions from one saved viewpoint to another. This has the effect of smoothing out (and slowing down) the transition.

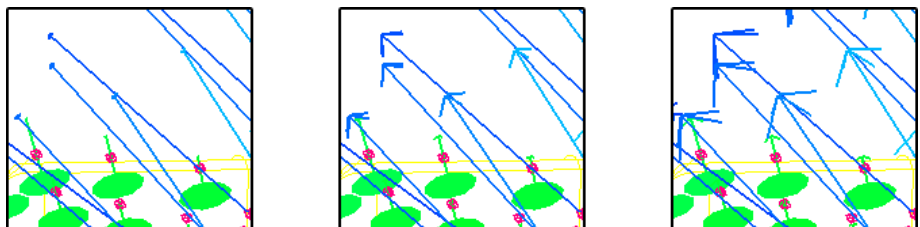
To Change the Number of View Animation Steps:

1. Click the User Options icon  in the toolbar.
2. In the *Display* tab, change the *View animation steps* value. The value is the number of frames used to animate from one view to the other.


Arrowhead Sizes


You can change the size of arrowheads on vectors with the Line arrow size factor setting (see [Figure 5-29](#)).

Figure 5-29. Line arrow size factors of 0.01 (left), 0.05 (middle), and 0.1 (right).



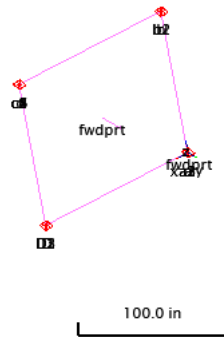
To Change the Size of Arrowheads on Vectors:

 **Note:** Vector arrowheads also get larger as the vector gets longer.

1. Click the User Options icon  in the toolbar.
2. In the *Display* tab, change the *Line arrow size factor* setting to a different value. A small change goes a long way!


Scale Indicator

Figure 5-30. The scale indicator. These points are approximately 100 inches apart.



To get a better grasp on the scale of the current zoom setting, you can display a scale indicator—somewhat like a distance legend on a map—that indicates the size of distances in the graphical view (Figure 5-30). The scale indicator automatically updates and adjusts itself as you zoom in and out in the view.

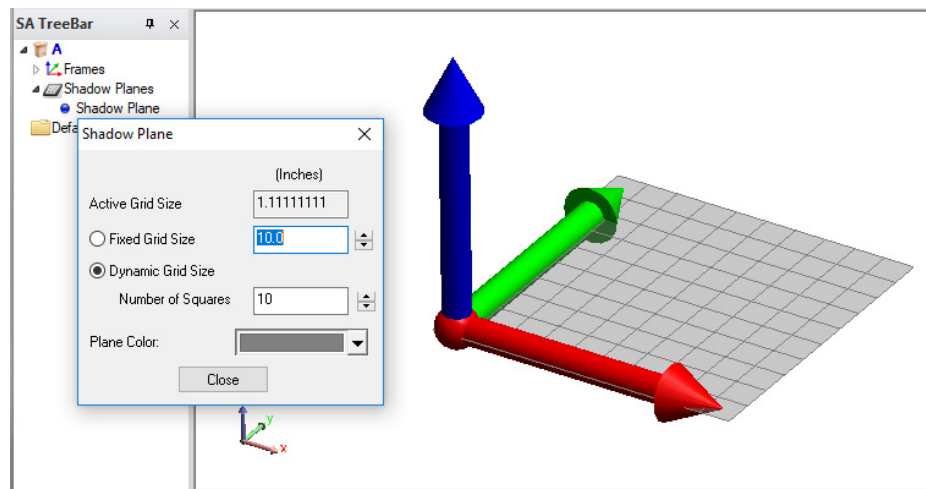
To Display the Scale Indicator:

1. Click the User Options icon  in the toolbar.
2. In the *Display* tab, check the *Show Scale Indicator in View* checkbox.

Shadow Plane

A shadow plane provides a more advanced way to establish relative size and orientation in a 3D space. The grid of the shadow plane will adjust dynamically to display the extents of the visible objects in the job. It is always oriented parallel to the XY plane of the working frame, but will shift vertically to the lowest (-Z) extent of the lowest object. This provides a ground level grid in the 3D world of the job file (Figure 5-31).

Figure 5-31. Shadow Plane



The properties of a shadow plane control the grid scale and color. The grid can be set to either dynamically adjust to always display a fixed

number of squares or can be set to a fixed grid size, using job units. The length and width of each square in the grid is defined by the “Actual Grid Size” value.

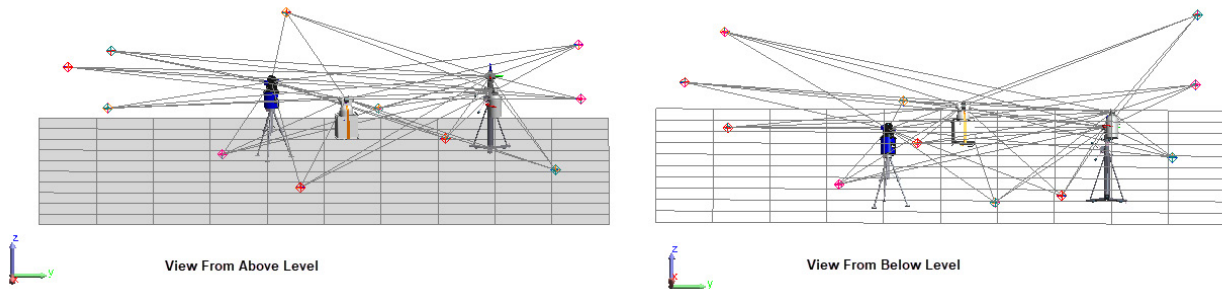


Figure 5-32. Shadow Plane
Viewed from Above and Below

To help interpret up from down, the shadow plane is colored in solid mode from viewed from above and shown only in wire frame mode when viewed from below (see Figure 5-32).

A new job will add a shadow plane automatically, although this can be turned off in the **User Options > Display tab**. It behaves like any other object in the tree in that it can be moved between collections and has properties, but only one can exist at any time in a job file. If it has been deleted it can be re-added to the job using **View > Show Shadow Plane**.

Connection Indicator

If you have more than one instrument in your job file, it may begin to get confusing as to which instrument is currently connected to the instrument interface (that is, which instrument in the SA file is currently live). The connection indicator (Figure 5-33) is designed to highlight the connected instrument and prevent the confusion.




Figure 5-33. The connection indicator.

You can control whether the indicator is displayed, and if so, how

large it is drawn.

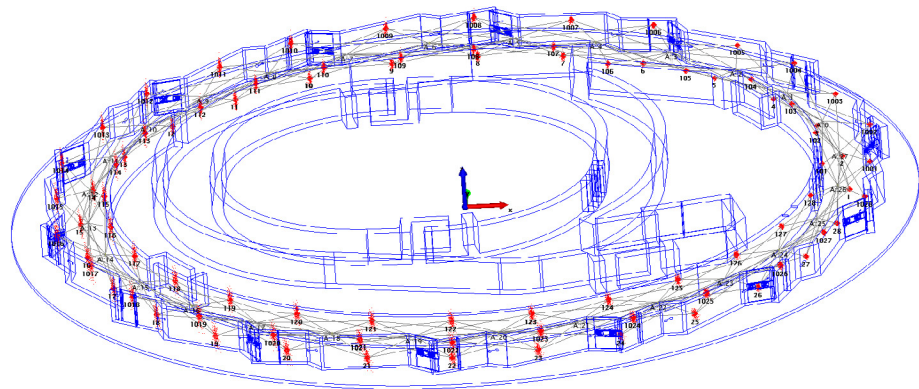
To Adjust the Connection Indicator:

1. Click the User Options icon  in the toolbar.
2. In the *Display* tab, enable or disable the *Connection Indicator* checkbox.
3. To change the size of the indicator, change the *Radius* value.

Uncertainty Fields

As discussed in “Measurement Uncertainty”, uncertainty fields allow you to visualize the uncertainty of a measurement as a cloud of points (Figure 5-34). Uncertainty clouds provide a reference for determining relative uncertainty magnitude, shape, and distribution when comparing measurements.

Figure 5-34. Viewing uncertainty fields for a set of measurements.




You can choose to display these uncertainty fields in the graphical view, or to hide them.

To Toggle Display of Uncertainty Fields:

- From the menu, select **View>Show Uncertainty Fields**.

Uncertainty fields are rendered in the graphical view at a certain magnification level so that they can be easily seen.

To Change the Size of Uncertainty Fields:

1. Click the User Options icon  in the toolbar.
2. In the *Analysis* tab, change the *Cloud Magnification* value to the desired amount.

By default, uncertainty clouds are drawn as individual dots on the screen, each representing a sample of the overall uncertainty cloud. For ease of viewing, you may prefer to draw the cloud with lines instead of dots (Figure 5-35).

To Draw Uncertainty Clouds as Lines:

1. Click the User Options icon  in the toolbar.

2. In the *Analysis* tab, enable the *Draw using lines* option.

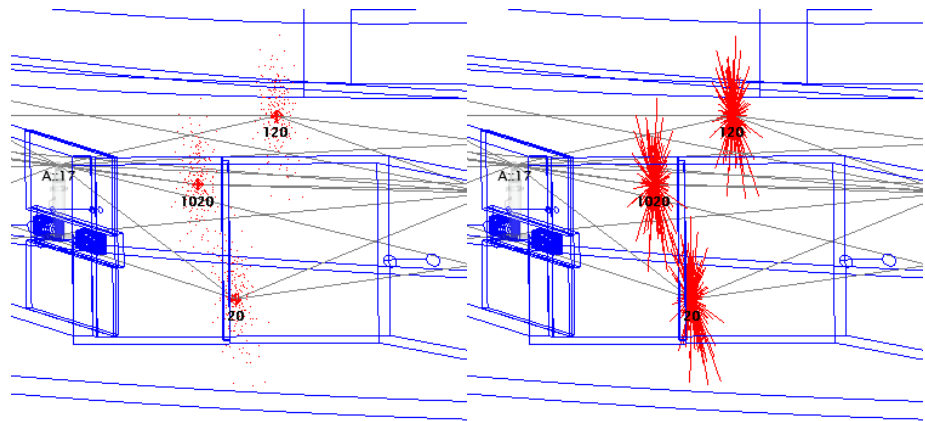


Figure 5-35. Uncertainty clouds drawn as dots (left) and lines (right).

Highlighting

As a job file grows to hold more objects and measurements, it may become somewhat difficult to distinguish objects in the graphical view. While hiding items not of concern is probably the most common solution to this, often times it is faster to simply select an object by name in the tree and highlight it. Highlighting an item does two things: it changes the item in the graphical view to the current highlight color, and it also colors the item in the tree a bold blue. Doing this enables you to pinpoint the selected object while keeping all objects of interest visible. In addition to individually highlighting objects, you can also highlight points, collections, instruments, and folders. (Highlighting a collection or folder highlights everything inside that collection or folder).

There are also cases in which simply hiding items does not provide sufficient information. For instance, you can right-click a relationship and ask it to highlight its associated points or objects in order to identify which entities are associated with that relationship. This is information that cannot be revealed by simply hiding objects.

To Highlight a Point, Object, Instrument, Collection, or Folder:

- Right-click the item in the tree or graphical view and select **Highlight**.

To Highlight the Points or Objects Associated With a Relationship:

- Right-click the relationship in the tree and select **Highlight Entities**.

To Clear Highlights:

- Right-click the item(s) in the tree or graphical view and uncheck the **Highlight** setting, or
- Select **View>Clear all Highlights** to clear all highlighted items.

View Refresh

Rarely, you might occasionally find the need to manually refresh the view. If you are uncertain that the view updated after performing an action, you can refresh the view to ensure that you're looking at the latest rendering of the graphical view.

To Refresh the Graphical View:

- Select **View>Refresh** from the menu, or
- Press **F5**, the default keyboard shortcut.

Copying the Graphical View to the Clipboard

You can quickly and conveniently copy just the current graphical view to the clipboard for pasting in other applications (such as into an Excel or Word document).

To Copy the Graphical View to the Clipboard:

- From the menus, select **Edit>Copy graphics to clipboard**.

Creating a Movie Capture of the Screen

SA comes with a tool for capturing video of the screen (or certain windows) to a video file. This may be useful for sharing your work with others or communicating an important feature to a customer.

To Create a Movie Capture of the Screen:

1. From the menus, select **View>Create AVI Movie>New File**.
2. When prompted, specify a filename for the resulting video.
3. The *Image Capture Options* dialog appears (Figure 5-36).

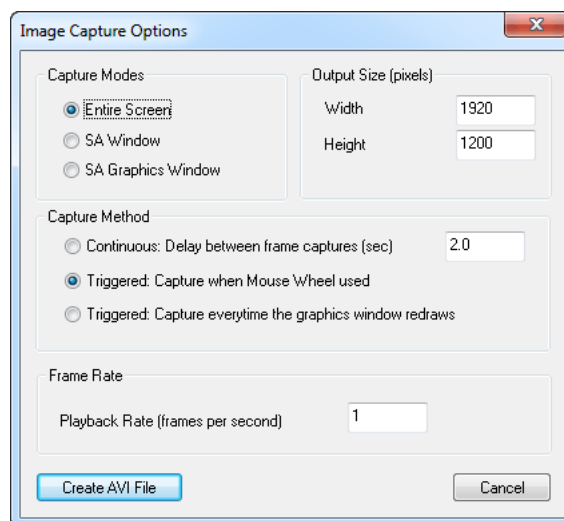


Figure 5-36. Options for an video capture.

4. Select the desired settings, then click **Create AVI File**.

Capture Modes

- **Entire Screen.** The entire screen will be captured.
- **SA Window.** Only the boundary of the SA window will be captured.
- **SA Graphics Window.** Only the graphical view will be captured.

Output Size (pixels)

Controls the resolution of the resulting video file.

Capture Method

- **Continuous.** Captures frames with a specified delay between frames.
- **Triggered: Capture when mouse wheel used.** Captures a frame only when the mouse wheel is used.
- **Triggered: Capture everytime the graphics window redraws.** Captures a new frame whenever the graphical view is redrawn.

Frame Rate

- **Playback Rate (frames per second).** Indicates the playback rate of the video—the number of video frames displayed per second.
5. If applicable, choose the appropriate video compression settings.
 6. Start recording by selecting **View>Create AVI Movie>Start Recording** or press **Alt+1**.

To Pause or Resume Video Recording:

- To pause, select **View>Create AVI Movie>Stop Recording** or press **Alt+2**.
- To resume, select **View>Create AVI Movie>Start Recording** or press **Alt+1**.

To Finish Recording:

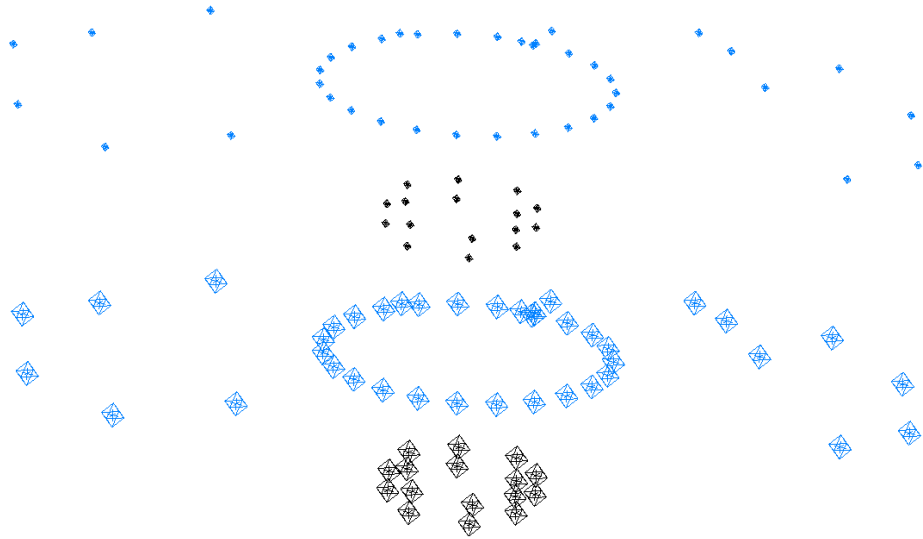
1. First stop recording using the **View>Create AVI Movie>Stop Recording** command or press **Alt+2**.
2. Select **View>Create AVI Movie>Close File**.

Controlling Target Icon Size


You can control the size of constructed and measured points (targets) as displayed in the graphical view. Targets can be drawn at a fixed size—which does not change regardless of zoom level—or a varied size, in which the points get larger as you zoom in. The former avoids

clutter when multiple points are spaced closely together, whereas the latter gives a visual indication of scale on the current view.


Figure 5-37. Targets displayed at 5 pixel size (top) and 15 pixel size (bottom).



To Use a Fixed Target Size:

1. Click the User Options icon  in the toolbar.
2. In the *Targets* area of the *Display* tab, ensure that *Fix size in view* is checked, and enter the desired pixel size for the targets.

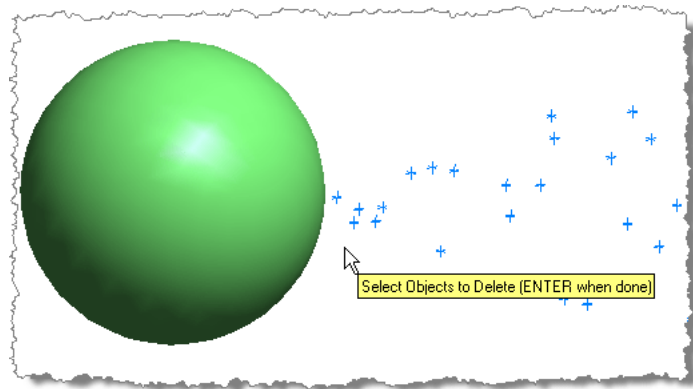
To Use a Variable Target Size:

1. Click the User Options icon  in the toolbar.
2. In the *Targets* area of the *Display* tab, ensure that *Fix size in view* is unchecked. Set the *Icon radius* to the desired size, in job units.

This chapter will help you to learn the ins and outs of working in SA on a day-to-day basis, which will speed up your workflow and allow you to complete your work as quickly as possible.

Selection

Figure 6-1. A selection prompt displayed in the graphical view.



SA generally follows the *Action-Object* selection paradigm when it comes to issuing menu commands. This implies that you first choose a command or operation, and then select the items to which the operation should be applied. Depending on the operation, multiple selection steps may be required. When an object is selected, it will change to the selected highlight color and its listing in the tree will become **bold blue**.

You can select entities visually in the graphical view. When an operation requires a user to make a selection, a selection prompt will appear as in [Figure 6-1](#).

The entities that you will be prompted to select will depend on the selected command (one of the benefits of the *Action-Object* paradigm). SA will filter entity types for you, so that if you are prompted to select points, you will only be allowed to select points—not instruments or objects.

To Abort a Command and Exit the Selection Mode:

- Press the **Esc** key. The selection prompt will disappear and the command will be cancelled.

In many commands, the prompt permits you to select more than one item—usually as many items as desired. You can continue to select items using the selection methods below until satisfied, at which time you must indicate to the prompt that you are finished selecting items.

To Complete Selection:

- Press the **Enter** key.

There are several ways to make selections in SA:

- **Graphically.** Pick items directly in the graphical view.
- **Treebar Selection.** Pick items using the tree, using a double click to select the item.
- **F2 Selection.** Pick items from a filtered list.

The F2 selection also provides more advanced filter options:

- **Wildcard Selection.** Pick items based on naming criteria using wildcards.
- **By Color.** Pick items based on their color.
- **By Type.** Pick items based on their type.

Graphical Selection Process


One way to graphically select items is to simply click them in the graphical view.

To Graphically Select Items One at a Time:

- With a prompt displayed, single-click the item in the graphical view. The item's selection state will be toggled—once selected, clicking an item again will remove it from the selection.


You can also graphically select multiple items at one time. To do this, SA provides graphical selection modes.

Graphical Selection Options

 **Note:** The graphical selection mode options will persist until SA is closed.

The *Graphical Selection Mode* dialog (Figure 6-2) provides a number of options to allow you to control graphical selection. You can pick the selection mode as well as define whether entities inside, outside, or crossing the selection boundary should be selected.

To Open the Graphical Selection Mode Dialog:

- Choose **View>Graphical Selection Mode**, or
- Select the Graphical Selection Mode icon  from the Main Toolbar (or Home Tab of the ribbon).

To Change Graphical Selection Modes:

- Choose the *Rectangle* or *Polygon* radio button.

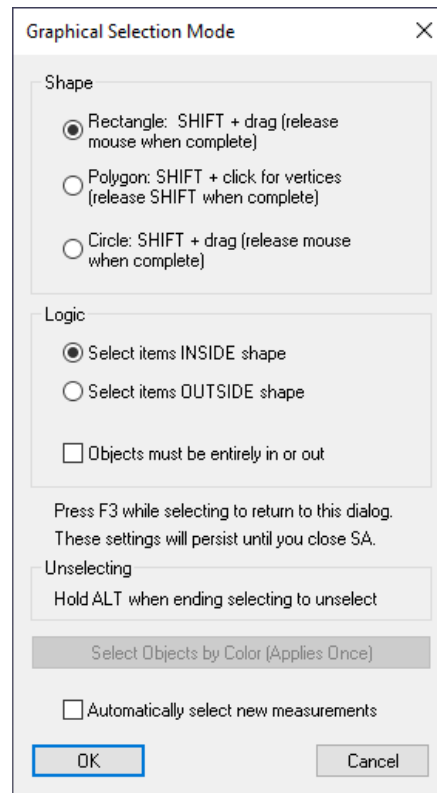


Figure 6-2. The Graphical Selection Mode dialog allows you to define the behavior of the selection marquee.

To Select Whether Entities Inside or Outside the Selection Region Should Be Selected:

- Choose the *Select items INSIDE shape* or *Select items OUTSIDE shape* radio button.

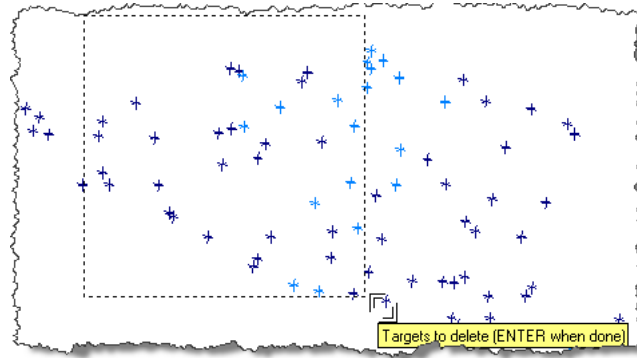
To Define Whether Entities Crossing a Selection Region are Selected:

- Check or uncheck the *Objects must be entirely in or out* checkbox.

Graphical Selection Modes

There are three graphical selection modes in SA: rectangle, circle and polygon.

Figure 6-3. Selecting a set of points using a rectangular selection marquee.



Tip: If you've started to define a selection in the wrong place, drag the mouse back over the starting corner of the rectangle. When the cursor changes back to an arrow, release the mouse button. You will remain in the selection mode but nothing will be selected.

Rectangle mode allows you to draw a rectangle around the entities of interest, as illustrated in [Figure 6-3](#). Everything inside the rectangle is either selected (or deselected), whereas the selection states of items outside of the rectangle remain unchanged.

Ensure you are in rectangle graphical selection mode (see "[Graphical Selection Options](#)" on page 108). Graphical selection requires a combination of the mouse and keyboard as follows:

To Select Multiple Items Using Rectangular Selection:

Note: If you release the Shift key prior to releasing the mouse button, you will define an autoscale region, not an entity selection.

1. Press and hold down the **Shift** key.
2. Click with the left mouse button to define one corner of the rectangular selection region and hold the mouse button down.
3. Drag the mouse to the opposite corner of the rectangle, then release the mouse button to complete selection. The items selected should highlight.
4. Press and hold **Shift+Alt** key and repeat the process to remove selection of a sub group of items that were previously selected.

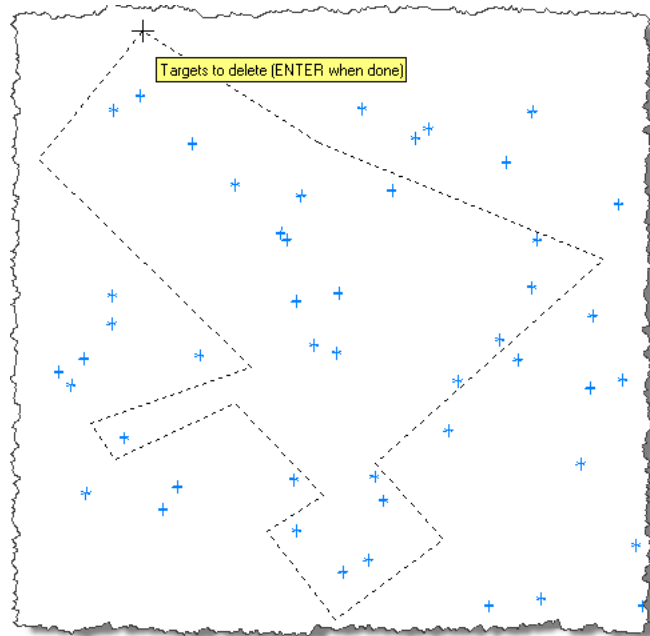
To Select Multiple Items Using Circular Selection:

Circle selection mode is performed in the same manner, although rather than a rectangle, the two click points define either extent of the diameter line of a circle.

To Select Multiple Items Using Rectangular Selection:

Polygon selection mode allows you to draw a polygon of any shape to define the selection, as depicted in [Figure 6-4](#).

Figure 6-4. Points are being selected through the use of a polygon selection marquee.



Items inside the polygon are either selected (or deselected), whereas items outside of the polygon are not affected.

To Select Multiple Entities Using Polygon Selection:

Note: To remove entities from the selection, hold down **Shift**, then hold down **Alt** and make your selection. When finished defining the polygon, release the **Shift** key, but keep the **Alt** key depressed.

1. Ensure you are in polygon graphical selection mode (see “Graphical Selection Options” on page 108).
2. While pressing the **Shift** key, single-click to define each vertex of the polygon.
3. For the last vertex, locate the mouse cursor where you’d like the vertex to be and release the **Shift** key.

To Change the Selection Mode During a Selection:

- Press the **F3** key.

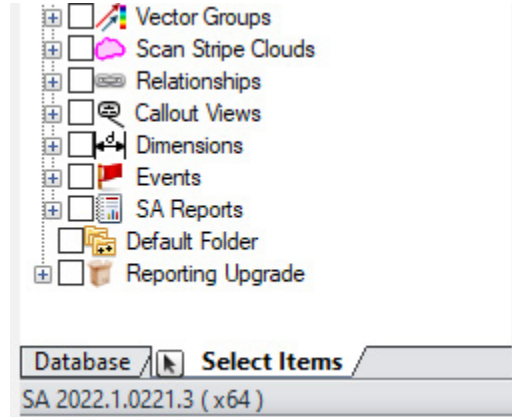
Treebar Selection

Any item can also be selected in the tree. Selecting items in the tree provides an advantage over the graphical view in cases where there are many entities overlapping, when the item is not visible, does not have a representation in space, or when the geometry of the view makes it difficult to determine which items are actually selected.

Select items tab of the tree

Starting in 2022, an embedded *Select Items* tab has been added to the tree to facilitate selection, and opens automatically. This tab offers a filtered list of items to facilitate selection and provides the ability to Shift+click to multi-select. At any time you can switch between the Selected Items tab and the Database to see the full tree display.

Figure 6-5. Select Items Tab within the SA Tree.



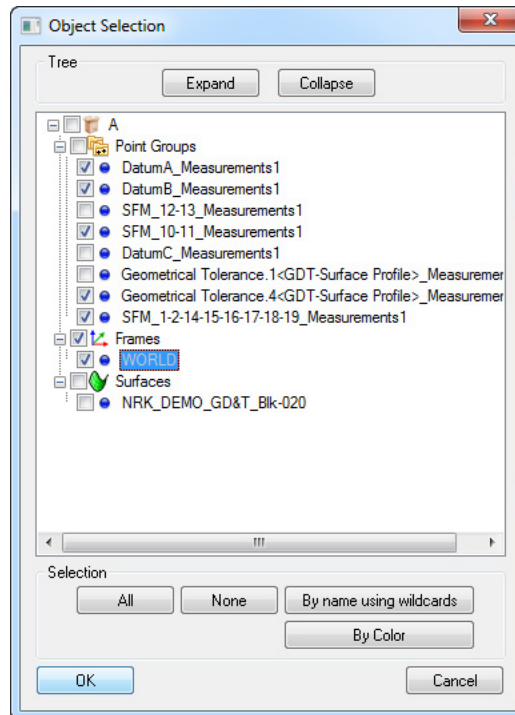
To Select Items in the Tree:

Selection from the Database tab of the SA tree requires a double-click in order to select an item, but multiple items can be selected at one time. Double-click the entity again to toggle its selection state.

F2 Selection

The F2 selection window (Figure 6-6) provides significant advantages for selection in cases where multiple items need to be selected. F2 selection filters out entity types that don't apply to the current selection and lists the remaining items in a hierarchical view similar to the treebar. It also offers additional tools such as search by Wildcard and selection by color.


Figure 6-6. The F2 Selection dialog.



There are three variants of the F2 selection window: there is a window specific to point selection mode, object selection mode, and instru-

ment selection mode. Each is slightly different and has options particular to the specific mode.

To Use the F2 Selection Window:

1. During selection (while the prompt is displayed), do one of the following:
 - Press **F2**,
 - Click the F2 - Select from a list  icon in the toolbar, or
 - From the menu, select **View>Point Selection Tree-View**.
2. Select the desired items, then click the **OK** button.
3. Continue selecting items using other methods (if desired), then confirm your selection by pressing the **Enter** key.

To Expand or Collapse the Entire Tree of Items (Point/Object Selection Modes):

- Click the **Expand** or **Collapse** button.

To Select All Items in a Folder or Collection:

- Check the box next to the folder or collection name.

To Select an Entire Category or an Entire Point Group (Point/Object Selection Mode):

- Check the box next to the category/point group (for example, Spheres, Point Groups, or Frames).

To Select All But a Few Items in a Category or Point Group (Object/Point Selection Modes):

- Check the box next to the category type, then uncheck individual items within that category.

To Select Contiguous Rows of Items (Object/Point Selection Modes):

1. Check the box next to the first item in the list.
2. Hold down **Shift** and check the box next to the last item in the list.

To Select All Items in the List:

- Click the **All** button.

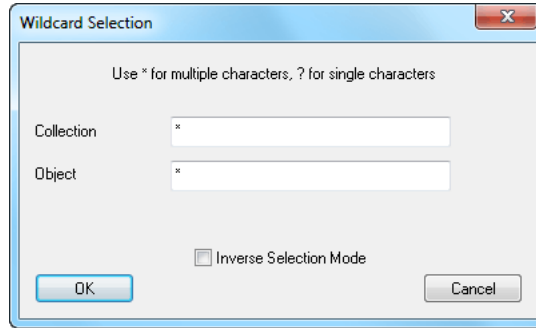
To Deselect All Items in the List:

- Click the **None** button.

Wildcard Selection

Wildcard selection is a powerful way to select items by name from the F2 window. To use it, click the **By name using wildcards** button, which opens the dialog in [Figure 6-7](#).

Figure 6-7. The Wildcard Selection dialog.



Warning: Wildcard selection automatically adds an asterisk (*) at the beginning and end of your search text. This may result in selection of more items than desired.

This dialog may appear slightly different depending on whether the current command allows you to select points or objects. In this dialog, asterisks (*) represent a set of contiguous characters, and question marks (?) represent a single character. You can also use [] to select a specific character.

This dialog allows you to enter selection criteria based on collection, object, and point names. In each criteria field, enter a search string. Any entities that match all of the entered criteria will be selected.

Note: The wildcard selection criteria, like all names in SpatialAnalyzer, are not case sensitive.

Examples (all based on point selection):

To Select:	Enter the following into the fields:		
	Collection	Group	Target
All points in all collections	*	*	*
All points in all collections containing "Survey"	SURVEY	*	*
All points in the Survey collection from point groups whose names contain "Instrument", followed by any string, followed by "Measured"	SURVEY	INSTRUMENT*MEASURED	*
Selects all points in all collections with a "*" at the end of the name	*	*	*[*]

To select everything except the items selected by the search criteria, check the Inverse Selection Mode option. After completing a wildcard selection, the items are selected (signified by the checked checkbox next to their entry in the list) but the F2 dialog remains open so that you can modify your selection or add to it. You can even add the result of another wildcard selection to your currently selected items.

To Select Based on Wildcards:

1. At a selection prompt, open the F2 Selection Window.
2. Click the **By Name Using Wildcards** button.
3. Type in your criteria, then click **OK**.
4. Modify the selection as needed, or perform another wildcard selection.
5. Click **OK** to select the items.

Selection By Object Type

The F2 selection window enables you to select items by their type (see Figure 6-8). Although this is not particularly useful when selecting items in a single collection (because those items are already organized by type and can be easily selected by clicking one checkbox), if you have multiple collections of items that you want to select by type, this can save you some time.

To Select Objects By Type:

1. In an object selection mode, call up the F2 selection dialog.
2. Click on the **By Object Type** button.
3. In the **Select Object Types** dialog, select the object types you'd like to select, then click the **OK** button.

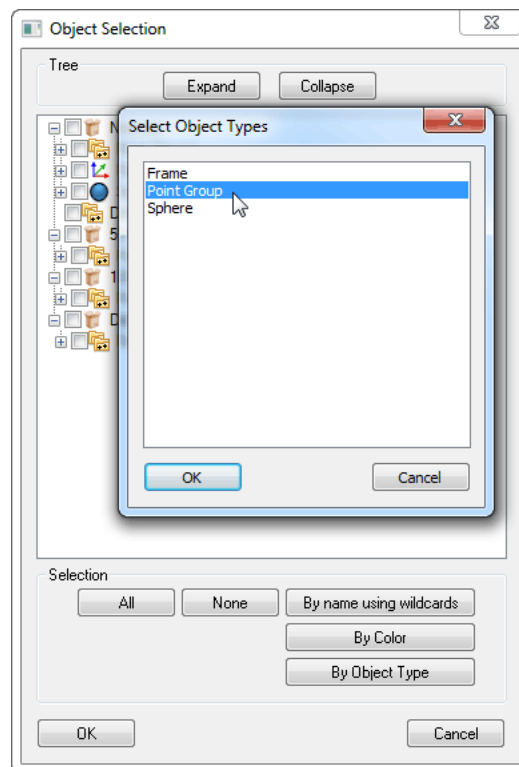


Figure 6-8. Selecting all point groups by their object type.


Selection By Color

It may be desirable to use object color to characterize different entities in the job file. For instance, you may decide to color all nominal objects yellow, and all as-built or measured objects blue. If you follow this colorization scheme, then you can easily select all nominal or measured objects using selection by color.


There are two ways to select objects by color. You can select graphically by color, or you can F2-select by color. Both color selection modes apply to object selection only, because only objects can be assigned a color. When selecting graphically by color, you choose a

single item whose color defines the color to select. SA will then automatically also select all objects that match that color. F2 selection by color allows you to directly pick a color (not an object) and SA selects all objects of that color.

To Graphically Select Objects By Color

1. While in object selection mode, press the **F3** key or click on the Graphical Selection Mode  icon.
2. In the *Graphical Selection Mode* dialog, click the **Select Objects by Color (Applies Once)** button.
3. Click on an object whose color defines the color you'd like to select.

To F2 Select Objects By Color

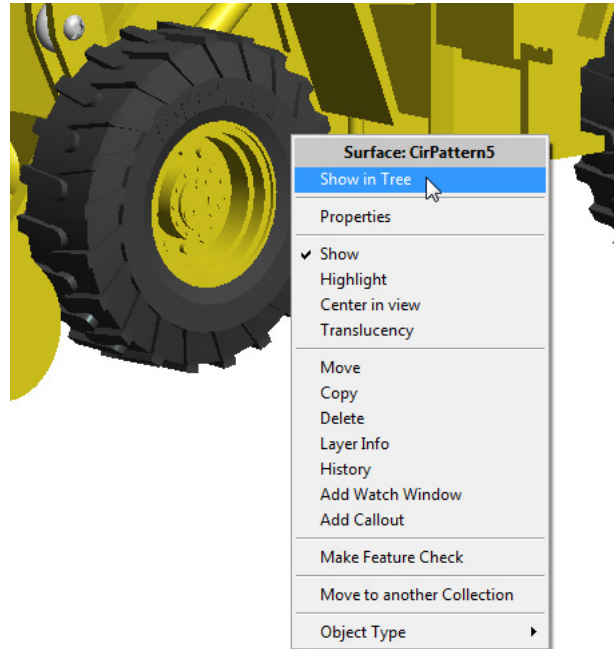
1. With a selection prompt displayed, press the **F2** key or click the list selection  icon.
2. In the resulting *Object Selection* dialog, click the **By Color** button.
3. In the *Select Color* dialog, check one of the colors in the list. (Only the colors in use in the file will be displayed).
4. Click **OK**. You will be notified how many objects matched the color criteria, and they will be added to your selection.

Context Menus

Context menus play an important part in the SpatialAnalyzer workflow. They allow you to select commands directly related to an item by selecting that item instead of looking for a command in a menu. In some cases, commands are available in context menus and nowhere else. To access a context menu, simply right-click an item or object. If it has a context menu, it will be displayed adjacent to the mouse cursor.

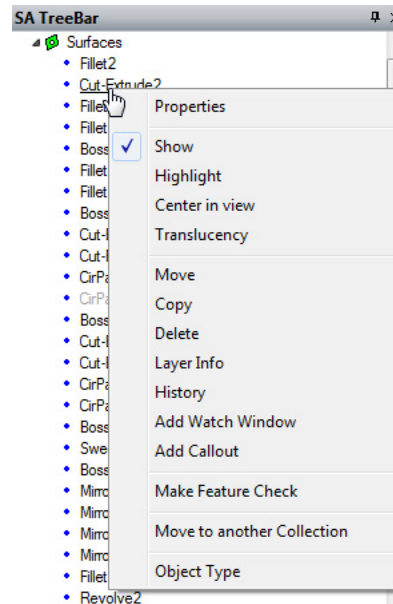
Context menus are frequently used in the graphical view. Right-clicking an entity in the graphical view reveals its context menu, allowing quick access to commands related to that entity (see [Figure 6-9](#)). There are several commands that appear in the graphical view context menus that are not found anywhere else.

Figure 6-9. The context menu for this tire contains a command to reveal the surface in the tree.



Right-clicking an entity in the SA tree will also present a context menu and is one of the most common actions performed in the tree (Figure 6-10).

Figure 6-10. The context menu in the tree.



Organizing Your Job

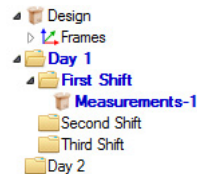
In a typical measurement job, you'll be working with many different types of objects in a file: instruments, coordinate frames, measured points, geometry, CAD data, reports, and more. Without a way to organize this data, it would be easy to quickly get lost amongst the variety of entities in the file. Fortunately, SA provides a flexible way to

organize this data so that it can be accessed conveniently and without confusion. The primary player in all of this organization is the tree, in which the items in your job are displayed in a hierarchical fashion.

Folders and *Collections* are two primary constructs that enable this organizational capability.

Folders

Figure 6-11. Multiple folders and collections in a hierarchy in the tree.



Folders allow you to sort and arrange your job information in the tree in a way that makes the most sense based on the way you work (Figure 6-11). You can create nested folders and move collections from one folder to another. You can expand and collapse a folder hierarchy, hide folders, rename and move folders, and change their order in the tree. Folders can only contain other folders or collections. You cannot put an instrument, point group, coordinate frame, or any other entity directly inside of a folder.

To Create a New Folder:

- Right-click an empty area of the tree or on any folder and choose **New Folder**.

To Create a New Sub-Folder:

- Right-click the folder you'd like to be the parent of the new sub-folder, and choose **New Sub-Folder**.

To Delete a Folder:

- Right-click a folder in the tree and choose **Delete**. All items within the selected folder will be deleted.

To Delete Multiple Folders by Name:

- From the menu, select **Edit>Delete>Delete Folders By Wildcard...**
- Type a wildcard search criteria (*, ? and []) are allowable wildcard characters) into the *Search Criteria* field.
- Click the **OK** button and confirm the deletion action.

To Rename a Folder:

- Right-click the folder and choose **Rename**.

To Expand or Collapse an Entire Folder Hierarchy:


- Right-click the folder and choose **Expand All** or **Collapse All**.


To Change the Order of Folders in the Treebar:

- Right-click a folder and choose **Change Order In List>Move Up (Move**

Down).

Collections

 **Note:** If you measure points (through an instrument interface) into a collection that does not already exist, the collection will be created automatically for you.

A *collection* refers to a group of objects, measured data, reports, and other entities that can be managed as a group from the tree, and as depicted with an open box icon . This “collection” of objects serves to group related data in a logical manner, making measurement and analysis more intuitive.

A job may have any number of collections. However, only one of these collections may be the *active* collection at any one time. When a collection is active, it is considered to be the *default* collection. Any newly-created entities (e.g., Points, Instruments, Planes, Lines, Reports, Relationships, etc) are created in the active collection unless a different collection is explicitly specified. The active collection is depicted in the tree by a *bold blue font*. Notice in [Figure 6-11](#) that any folders enclosing the active collection are also highlighted with a *bold blue font* so that you can find the active collection if the folder hierarchy is collapsed.

To Create a New Collection:

- Right-click an empty area of the tree, any collection, or any folder and choose **New Collection**, or
- Measure a point or object into a collection that does not already exist (see Measurement).

To Rename a Collection:

- Right-click the collection in the tree and choose **Rename**.

To Delete a Collection:

- Right-click a collection in the tree and choose **Delete**.

To Delete Multiple Collections:

- Right-click a collection in the tree and choose **Delete Multiple Collections**, or
- Right-click an empty area of the tree and choose **Select Collections to Delete**.

To Delete Multiple Collections By Name:

1. From the menu, select **Edit>Delete>Delete Collections By Wildcard...**
2. Type a wildcard search criteria (*, ? and []) are allowable wildcard characters) into the *Search Criteria* field.
3. Click the **OK** button and confirm the deletion action.

To Copy a Collection:

- Right-click a collection and choose **Copy**.


To Change the Order of Collections in the Treebar:

- Right-click the collection and choose **Change Order In List>Move Up (Move Down)**.


To Make a Collection Active:

- Right-click the collection and choose **Make active default collection**.

To Move a Collection to a Folder:

 **Note:** To move a collection out of all folders, choose the Root folder as the destination.

1. Right-click the collection and choose **Move Collection to Folder**.
2. Choose the destination folder from the resulting dialog.

 **Note:** A common mistake for new users is to try to double-click a category in the tree to select it. You must expand the category to reveal the actual items inside it.

When objects are created inside of a collection (either explicitly or as the result of an operation), SA automatically organizes those objects by their type inside what is referred to as a *category*. A category is simply a container that holds all like items in a given collection. For instance, all point groups in a given collection are placed into the **Point Groups** category, and all instruments in a collection are placed into the **Instruments** category. This behavior makes it easy to keep your file's elements organized by type.

Relocating Items to Another Collection

A common use for collections is to organize nominal and measured data separately. Another use is for organizing multiple states of measured data (for instance, measuring an assembly at various stages in the production process). You may occasionally come across the need to move items from one collection to another.

To Relocate an Item to Another Collection:

1. Right-click the item in the tree or graphical view and select **Move to Another Collection** from the context menu.
2. In the **Select Collection** dialog, double-click the destination collection (or select it and click **OK**). The item will be moved into the destination collection.

To Relocate Multiple Objects to Another Collection:

1. From the menu, select **Edit>Relocate Objects to another Collection**.
2. Select the objects you'd like to relocate.
3. In the **Select Collection** dialog, double-click the destination collection (or select it and click **OK**). The objects will be moved into the destination collection.

To Relocate Multiple Instruments to Another Collection:

1. From the menu, select **Edit>Relocate Instruments to another Collection**.
2. Select the instruments you'd like to relocate.

3. In the *Select Collection* dialog, double-click the destination collection (or select it and click **OK**). The instruments will be moved into the destination collection.

To Relocate Multiple MPs to Another Collection:

1. From the menu, select **Edit>Relocate MPs to another Collection**.
2. Select the scripts you'd like to relocate.
3. In the *Select Collection* dialog, double-click the destination collection (or select it and click **OK**). The MPs will be moved into the destination collection.

Rearranging Points

At some point while working in a file, you will likely want to rearrange your points by moving them from one point group to another. There are several ways to do this.

To Relocate Points into a Separate Group:

1. From the menu, select **Construct>Points>From Existing Points>Move**.
2. Select the points to relocate.
3. In the dialog, enter a name for a new group into which to place the points. This group must not already exist.
4. The points will be removed from the current group and placed into the new group. The point names will be prepended with the source group name.

To Relocate a Single Point into a Different Group:

Renaming a point and giving the point a different collection or group will move it into that collection/group. If the collection or group does not already exist, it will be created for you.

1. Right-click the point to relocate and choose **Rename** from the context menu.
2. In the *Rename Point* dialog, enter a new collection or group name, then click **OK**.

The Group Manager

You can rearrange points in an intuitive way using the *Group Manager*. The *Group Manager* is a Windows Explorer-like dialog that allows you to drag and drop points from one group to another, and also provides the capability for renaming, group creation/deletion, search and replace, sorting, and more.

To Call Up The Group Manager:

- From the menu, select **Edit>Group Manager...**

The Group Manager is shown in [Figure 6-12](#).

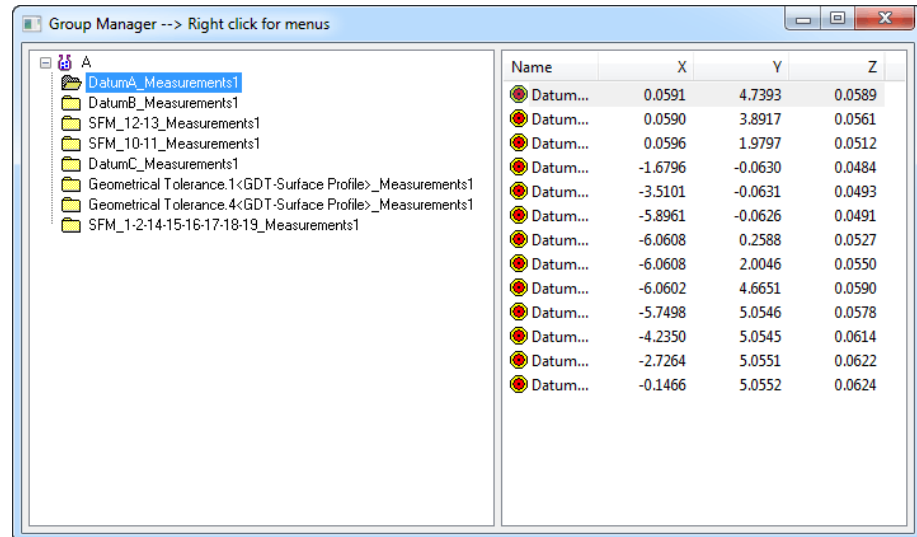


Figure 6-12. The Group Manager.

On the left side of the group manager is a list of all groups in your job, organized by collection. At right is a list of the points in the selected group, as well as their coordinates.

To Create a New Group:

1. Right-click anywhere on the left side of the Group Manager and select **New** from the context menu.
2. Type a new group name, then click **OK**.

To Delete a Group:

- Right-click a group on the left side of the *Group Manager* and select **Delete** from the context menu.

To Rename a Group:

1. Right-click a group on the left side of the *Group Manager* and select **Rename** from the context menu.
2. Enter a new name for the group, then click **OK**.

To Relocate Points to a Different Group:

1. Select the points to relocate from the right side of the window. You can only select points from one group at a time, although multi-selection is allowed.
2. Do one of the following:
 - Drag the selected points to the destination group, or
 - Right-click the selection and select **Cut** from the context menu, then right-click the destination group and select **Paste** from the context menu.

To Copy Points To Another Group:

1. Select the points to copy on the right side of the window. You can only select points from one group at a time, although

multi-selection is allowed.

2. Do one of the following:
 - Right-click the selection and choose **Copy** from the context menu, then right-click the destination group and select **Paste** from the context menu, or
 - Right-click the selection and choose **Copy selections into a New Group** from the context menu. Type a new group name, then click **OK**.

To Create New Points:

1. Select the group to contain the new points on the left side of the *Group Manager*.
2. Right-click the right side of the window and select **New** from the context menu.
3. Enter the Cartesian coordinates for the point, then click **OK**.
4. Enter the name for the new point, then click **OK**.

To Delete Points:

- Select one or more points, then right-click the selection and choose **Delete** from the context menu.

To Sort Points by Name or Coordinate Value:

- In the *Group Manager*, click a column header to sort by that value. Click again to sort in the reverse direction.

To Find Points:

1. Right-click a point or selection (select points to search amongst first, if desired) and select **Find...** from the context menu.
2. Type a search string, and choose either the *Selection* option to search the selection, or the *Group* option to search the entire group.
3. Click the **OK** button. The list of points will be filtered to show only the matching points. (Click the group name to reset the list).

To Perform Search and Replace in Point Names:

1. Select points (if desired), then right click the selection and choose **Replace...** from the context menu.
2. In the *Replace* dialog, enter text to search for, and the text to replace it with. Choose *Selection*, *Group*, or *All* to indicate whether you want to search among the selection, entire group, or all points.
3. Click the **OK** button. All matching text substrings are replaced.

To Access Point Properties:

- Double-click a point in the *Group Manager*, or
- Right-click a point and select **Properties** from the context menu.

To Access Group Properties:

- Right-click a group and select **Properties** from the context menu.

Naming & Renaming Items

If you create objects in SA without properly naming them—that is, by using the default names provided by SA—you will soon find that your job file becomes completely unmanageable. In all but the simplest of measurement files, there are enough entities created that you can quickly lose track of what each entity represents. Therefore, best practice is to clearly name all items in your file when you create them. Doing so makes the file easier to work with, keeps it organized, and also allows you (or another person) to look at the file perhaps years later and still understand the data inside it.

Choosing a name is entirely up to you, although there are a few rules that should be followed.

- Names are not case-sensitive. A group named **TEST** is not distinguishable from a group named **test**.
- Spaces are allowed, but depending on the letters surrounding the space, it is often difficult to determine if a name contains a space at all (or perhaps multiple spaces). If you're worried about potential confusion, avoid the use of spaces.
- Multiple points in the same point group cannot have the same name. They must be unique.
- Multiple objects may have the same name in a given collection, however this is not recommended if you expect to use scripting (Measurement Plans), as this leads to potential ambiguity within your script as to which object you are referring.
- Avoid long consecutive sequences of numeric digits at the end of a point or group name. Keep them to less than 20 digits. (Otherwise, incrementing and decrementing of point and group names will not function correctly).

To Rename a Group of Points:

Do one of the following:

1. Do one of the following:
 - From the menu, select **Edit>Rename>Group of Points**, or
 - Right-click a group in the tree and select **Rename** from the context menu.
2. At the prompt, enter a new name for the group, then click **OK**.

To Rename a Point:

1. Right-click a point and select **Rename** from the context menu, or select **Edit>Rename>Single Point**.
2. At the prompt, enter a new Collection/Group/Target name (as desired), then click **OK**.

To Rename Objects:

- Right-click the object and select **Rename** from the context menu. (If this menu item is not available, choose the **Properties** menu item and the name will be found in that dialog).

Renaming Using Unique Names

Figure 6-13. Automatically removing duplicate names.



Occasionally, you will be presented with a file in which several objects have the same name. This is sometimes the case when importing an external CAD file, for instance. You can easily automatically rename a series of objects, ensuring that they all have unique names (Figure 6-13). SA will rename the objects by appending the type of object with a unique number and using that for the new name.

To Rename Objects Using Unique Names:

1. From the menu, select **Edit>Rename>Objects by Assigning Unique Names**.
2. At the prompt, select the objects to rename.

SA will rename each of the objects in turn, ensuring that the names are unique.

Renaming Using A Name Pattern

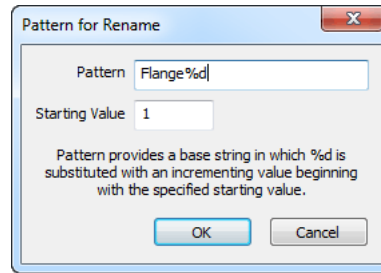
You can rename points and objects using a *name pattern*. Suppose you wish to rename a set of objects and call them **Obj1**, **Obj2**, **Obj3**, etc. This would use a name pattern that looks like this: **Obj%d**. When renaming using a name pattern, you can specify any prefix or suffix for the name, then add the wildcard **%d** to the search field. The **%d** characters will be replaced by an actual number when the command is performed. By default, the first number is 1, but you can specify your own starting value.

To Rename Objects Using a Name Pattern:

1. From the menu, select **Edit>Rename>Objects Using a Name Pattern**.
2. Select the objects you'd like to rename.
3. In the *Pattern for Rename* dialog (Figure 6-14), specify the

pattern and a starting value, then click **OK**.

Figure 6-14. Renaming a group of objects using a name pattern.



To Rename Points Using a Name Pattern:

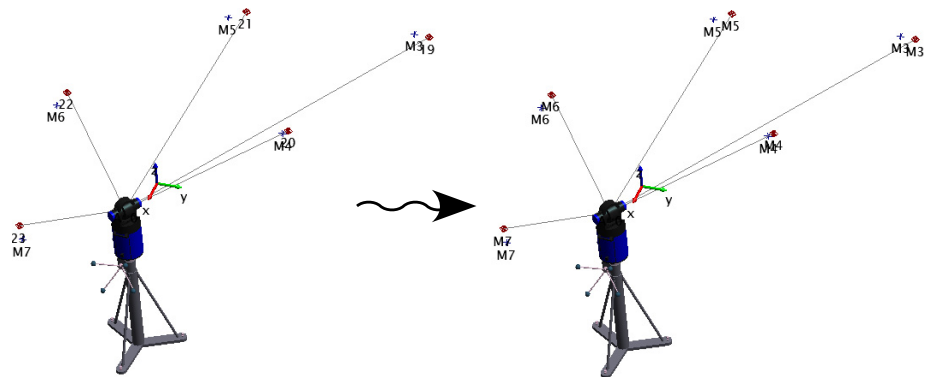
1. From the menu, select **Edit>Rename>Points Using a Name Pattern**.
2. Select the points to rename.
3. In the *Pattern for Rename* dialog (Figure 6-14), specify the pattern and a starting value, then click **OK**.

Renaming By Proximity

Renaming by auto-corresponding to points uses a group of points as a name reference. A different group of points is then renamed based on the reference group’s names.

You can rename points by proximity to a set of reference points. This is extremely useful for quickly renaming points in preparation for a Point to Point Best-Fit operation. When renaming by proximity, SA will search for a point in a reference group that is closest to a point in the group to rename. As long as the reference point is within the specified threshold, the point to rename will be assigned the same name as the reference point (see Figure 6-15). Note that the reference group and group to rename need not have the same number of points.

Figure 6-15. Before renaming points by proximity (left) and after (right).



The renaming operation is controlled by a *Proximity Distance Match Tolerance*. This is the furthest that a reference point can be located relative to the point to rename to be considered a match. If more than 1 point is within the tolerance specified the points will be renamed and incremented numerically using the reference name as a base (for example TB1, TB1-1, TB1-2, etc).

Probably the most common use for this is when remeasuring a set of

tie-in points after an instrument has *drifted* (moved) beyond the desired threshold. After remeasuring the tie-in points, since the instrument has only slightly changed position, they will be relatively close to the nominal tie-ins. In this case, you need not be concerned with the proper naming of the points. Simply measure them, and SA will take care of automatically renaming them for you.

To Automatically Rename Points by Proximity:

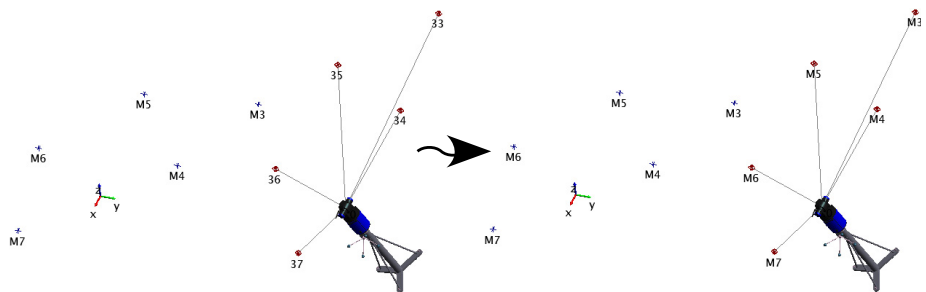
Note: If (due to the threshold) matches are found for some points but not others, points that did not have matches will have **Orig.** prepended to the point name. This is to ensure uniqueness—another point that did have a match may have been renamed to the unmatched point's name.

Note: If the threshold is very large, SA will only use the closest reference point to each measured point.

1. From the menu, select **Edit>Rename>Points by Auto-Corresponding 2 Groups>Proximity**.
2. At the prompt, select the reference group—you may need to select it from the tree. This is the group containing the proper point names that you'd like to assign to the other group.
3. At the second prompt, select the group to be renamed—you may need to select it from the tree.
4. In the **Input** dialog, specify the *Proximity Distance Match Tolerance*.
5. If any matches are found within the specified threshold, you will be presented with the **Residuals** dialog. This dialog lists the original point names, and their proposed new names. Click the **Save to File** button to save this list to a text file, or the **Accept** button to accept the proposed new names.

Renaming By Inter-Point Distance

Figure 6-16. A set of points before renaming by inter-point distance (left) and after (right).





You will often want two sets of points to have matching corresponding point names. Suppose you have a set of nominal reference points in your SA file that you'd like to use for tie-ins. Assuming your instrument hasn't yet been located to the nominal points (as it typically will not be), the measured points will not be close to the reference points. In order for a Point to Point Best-Fit operation to successfully execute, corresponding point names must match up. If you forget to name the points correctly as you measure them, the fit will not succeed. While you could manually rename each individual point, this command performs that process automatically (Figure 6-16), and is a big time-saver in this situation. The command is governed by an *Inter-point distance match tolerance*. This tolerance is the maximum allowable *difference* between the spacing of any proposed pair of renamed points and the corresponding spacing of that same pair of points in the ref-

erence group. If the difference exceeds this value, the set is not considered a match.

Note that the reference group and group to rename need not have the same number of points. Any points that don't match will be ignored.

To Rename Points By Inter-Point Distance:

 **Note:** If (due to the threshold) matches are found for some points but not others, points that did not have matches will have **Orig.** prepended to the point name. This is to ensure uniqueness—another point that did have a match may have been renamed to the unmatched point's name.

 **Note:** If the threshold is very large, SA will only use the closest reference point to each measured point.

1. From the menu, select **Edit>Rename>Points by Auto-Corresponding 2 Groups>Inter-Point Distance**.
2. At the prompt, select the reference group containing the desired point names.
3. Then, select the group containing the points to rename.
4. In the **Input** dialog, enter the desired *Inter-point distance match tolerance*.
5. If any matches are found within the specified threshold, you will be presented with the **Residuals** dialog. This dialog lists the original point names, and their proposed new names. Click the **Save to File** button to save this list to a text file, or the **Accept** button to accept the proposed new names.

Deleting Items

To keep your job file clean, you should delete any items that you no longer need. If you are certain that a measurement was erroneous, for instance, it is best to delete it on the spot and remeasure. This minimizes clutter in the job file and helps make the file more understandable. There are numerous ways to delete items in SA. For a generic delete function choose **Edit>Delete>Delete Database Items**. This will allow you to choose items both from the tree and from within the graphics.

For a filtered List of items to delete select from the following:

To Delete Objects:

Objects within SA include primary level graphical entities such as geometry, surfaces or point groups.

1. From the menu, select **Edit>Delete>Delete Objects**, or press the **Delete** key. You can also use the F2 key to bring up a list.
2. Select the objects to delete.

To Delete an object directly within the graphics you can also:

1. Right-click the item in the tree or graphical view.
2. From the context menu, select **Delete**.

To Delete Points:

Points are not considered Objects because they are included within a

Point Group. For that reason they require a special selection method:

1. From the menu, select **Edit>Delete Points**, or press **Alt+D**.
2. Select the points to delete by holding shift and rectangle selecting in the graphics. You can also use the F2 key to bring up a list of points to select from.

Points can also be deleted within proximity to a reference group. To do so select **Edit>Delete Points>Corresponding to Reference Group by Proximity**. This utility will allow you to select a set of reference points and detect points within a specified proximity to those points. This is a great way to remove duplicate points.

To Delete Cloud Points:

Individual cloud points must also be selected separately.

1. From the menu, select **Edit>Delete Cloud Points>Selected Points** or press **Crt+Alt+D**.
2. Select the cloud points to delete by holding shift and rectangle selecting in the graphics or using one of the other graphical selection tools.

Sometimes you may wish to delete cloud points based on their radial distance from a set of points. For instance, once you've scanned a set of registration spheres and extracted the sphere centers, you may wish to delete the scan data of the spheres themselves (so they do not interfere with other analysis).

To Delete Cloud Points by Their Radial Distance From Reference Points:

1. From the menu, select **Edit>Delete Cloud Points>By Radial Distance from Points**.
2. Select the point cloud(s) of interest.
3. When prompted, pick the points to use as the reference points.
4. Specify a radial distance.
5. At the prompt, specify whether the cloud points inside or outside of the radius should be deleted.

To Delete Vectors:

Vectors, much like cloud points, must be selected separately from within a vector group. To do so:

1. From the menu, select **Edit>Delete Vectors**.
2. Select the vectors to delete by holding shift and rectangle selecting in the graphics or using one of the other graphical selection tools (the vector end points is used for selection).

To Delete Vectors and Point Pairs:

In many cases the deviation vectors can be used to effectively identify bad measurements. When a relationship is displaying Auto-vectors the vectors are directly tied to the points that generate them and can be used for this purpose.

To delete points using vectors:

1. From the menu, select **Edit>Delete Vectors>Auto Vector/Point Pairs**.
2. Select the vectors to delete by holding shift and rectangle selecting in the graphics or using one of the other graphical selection tools (the vector end points is used for selection). This will remove both the vector and the corresponding point.

To delete cloud points using vectors:

1. From the menu, select **Edit>Delete Vectors>Auto Vector/Cloud Point Pairs**.
2. Select the vectors to delete by holding shift and rectangle selecting in the graphics or using one of the other graphical selection tools (the vector end points is used for selection). This will remove both the vector and the corresponding cloud point.

To Delete Objects From a Specific CAD Source

SA keeps track of the source of objects—whether they were created natively in SA or imported from CAD files. As a result, you can delete objects that were imported from a specific CAD file.

1. From the menu, select **Edit>Delete>Delete CAD Objects By Source**.
2. In the *Delete Objects by CAD Source* dialog, select the source CAD file, then click **OK**.

To Delete Pictures:

1. From the menu, select **Edit>Delete>Delete Pictures**.
2. Select the pictures to delete.

To Delete Relationships:

1. From the menu, select **Edit>Delete>Delete Relationships**.
2. Select the pictures to delete.

To Delete Using Layer Manger:

[“The Layer Manager”](#) on page 742.

Double Colon Format

When referring to an entity in your SA file, it is important to know where it is organized in your job file. For instance, in what collection is a specific surface located?

You will notice a specific syntax used in SA in various dialogs to describe the organizational location of an item in SA. A specific point is referred to by listing its containing collection, enclosing point group, and target name separated by pairs of colons, which act as separators:

Measured::Tie-Ins::3

This is an unambiguous reference to the point named **3** in the point group named **Tie-Ins**, which is inside the collection named **Measured**.

Objects and other entities have a similar syntax, but only the collection and entity name are used:

Nominal::Centerline

This name refers to an object named **Centerline** inside the **Nominal** collection.

Instruments are referred to by their instrument index number. An instrument index number is a unique number that is assigned to each instrument in a given collection. The first instrument added to a collection becomes instrument index number **0**, the second is **1**, and so on. When an instrument is moved from one collection to another, its index number changes. If the fourth instrument in collection **A** is moved to collection **B**, and there are already 2 instruments in collection **B**, the instrument will change from index number **3** in the old collection to index number **2** in the new collection.

The syntax for instruments is similar to that for objects, except instead of using an object name, the instrument index number is used instead:

ThisCollection::1

This name refers to the second instrument in the **ThisCollection** collection. Sometimes, you will also see this syntax with the instrument name at the end, such as

ThisCollection::1 - Zeiss ETh 2 Theodolite

The double colon syntax is used in several places in SA—for example, when displaying full point names or instrument names in the graphical view, when viewing an instrument interface, or when working in Measurement Plan scripts.

Configuring SpatialAnalyzer

There are a wide variety of settings in SpatialAnalyzer that provide significant control over the application's configuration. These settings can be stored in one of several different ways, depending on the scope of the setting:

- **Factory Defaults.** Factory default settings are stored as part of SA itself. These are the settings that apply when no other settings override them.
- **Job Files.** These are settings that are stored as part of the job file because it makes the most sense for them to be a property of a specific job file. Opening up a job file will restore its job file settings, regardless of the computer on which file is opened.
- **User Interface Profile.** These are settings that are stored as part of a customized user interface profile.
- **Machine Settings.** These settings are stored in the Windows Registry and apply at the computer level. Machine settings apply on a given computer, regardless of the current job file. (Although a user interface profile is stored in a profile file, the *active* profile is stored as a machine setting).
- **Persistence Files.** Persistence files contain configuration information and can easily be transferred from one computer to another.

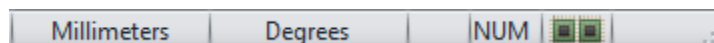
Job File Settings

A number of the more general job file settings are described below—however be aware that this list is not all-inclusive.


Units

Anywhere a number with units is displayed in SA, it is presented in the active unit system, unless indicated otherwise. A job file can be set to one of the different length and angular units. Temperature units can be set to either Fahrenheit or Celsius. The active job's length units are displayed in the status bar (Figure 6-17). Angular units are always presented in decimal degrees, but can be specified to present data in $\pm 180^\circ$ format or 0-360° format.

Figure 6-17. The active length units are displayed in the SA Status Bar.




To Change the Job File Units


1. Click the User Options  icon in the Main toolbar.
2. Click to activate the *Units* tab.

3. Select the desired length or temperature units.

To Change the Representation of Angles

1. Click the User Options  icon in the Main toolbar.
2. Click to activate the *Units* tab.
3. Select the appropriate setting in the *Angle Representation* box.

To Set the Default Job File Units (No Template File In Use)

1. Click the User Options  icon in the Main toolbar.
2. Click to activate the *Machine Configuration* tab.
3. Click the *No Template Default Options* button.
4. In the *New File Defaults* dialog, select the preferred units.

Fractional Unit Display

Fractional unit display options are also available for feet and inch display settings (Figure 6-18).

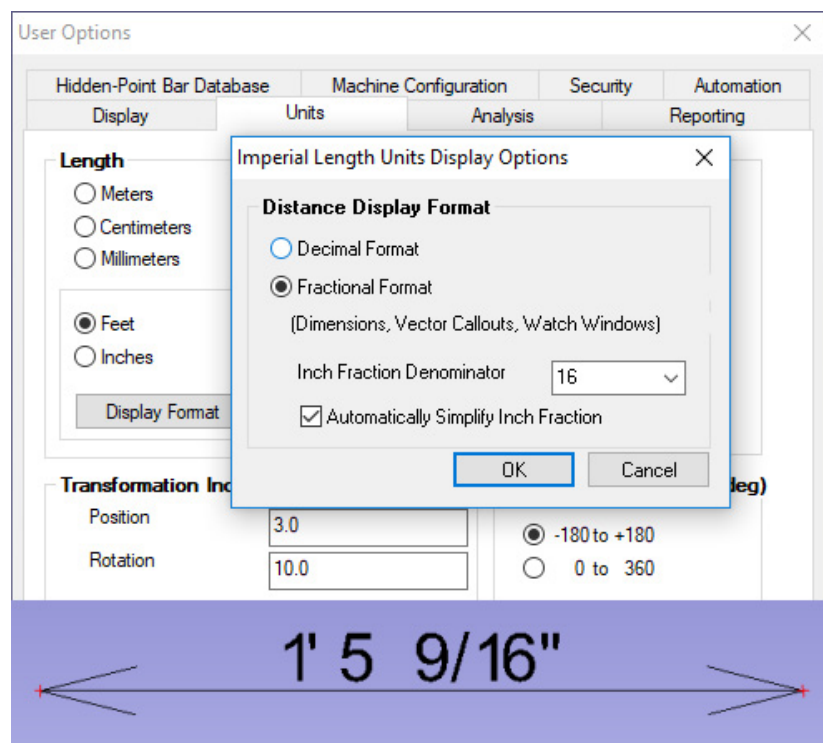


Figure 6-18. Fractional Display Settings

This setting is for display purposes and only impacts a selection of numeric indicators:

- **Dimension.** Both the graphical display of dimensions and their reported values will include the fractional unit designation when set.
- **Callouts.** Vector callouts are converted to display in fractions.
- **Vectors.** Vector Labels will be converted to display in fractions

- **Watch Windows.** The Distance component in a watch window will display in fractions.

US Survey Feet


In addition to a standard (or “international”) foot, SA also supports the US Survey Foot as a unit type (1 US Survey Foot = 1200/3937 meters where as 1 standard foot = .3048 meters). This can be helpful when working with job files saved in State Plane Coordinates and other survey applications.

Decimal Precision


Internally, calculations in SA are performed (and numbers are stored) with a high degree of decimal precision. However, when numeric results are displayed in the user interface, they are typically rounded to a preferred decimal precision in order to simplify the display. You can control the number of decimal places used when displaying different categories of values. Specifically, you can independently control decimal precision for length, angle, scale, unit vector, and weight values.

You can change the decimal precision for a given file at any time without incurring any additional internal numerical error.

To Change the Decimal Precision

1. Click the User Options  icon in the Main toolbar.
2. In the *Decimal Digits for Display* box of the *Display* tab, change the *Length*, *Angle*, *Scale*, *Unit Vector*, and *Weight* decimal precision as desired.

To Set the Default Job File Decimal Precision (No Template File In Use)

1. Click the User Options  icon in the Main toolbar.
2. Click to activate the *Machine Configuration* tab.
3. Click the *No Template Default Options* button.
4. In the *New File Defaults* dialog, enter the preferred display decimal precision.



A Closer Look

Having a different decimal precision for different types of numbers is useful, since different types of numbers are expressed in different units. For example, three digits of decimal precision for an angle (eg. 0.001°) is only accurate to 3.6 arcseconds—not enough for some applications. However, three digits of decimal precision in length when working in millimeters is too much precision for most applications (1 micron).

Reporting Tolerances


Reporting tolerances specify the threshold at which pointing errors or uncertainties are flagged in reports.

Angle-Distance Weights

Angle-distance optimization weighting values provide you with con-

control over the weight of angular measurements compared to distance measurements for bundle adjustments and point calculation. You can weight distance and angular measurement components equally, or you can completely disregard distance or angular measurement components.

To Edit the Angle-Distance Weights

1. Click the User Options  icon in the Main toolbar.
2. In the *Optimization Parameters* box of the *Units* tab, click the **Edit Angle-Distance Weights** button.
3. In the resulting **Optimization Weighting Values** dialog, drag the slider to the left to apply more weight to angular components, or to the right to apply more weight to distance components. You can optionally type relative weights for each into the two text fields.

Optimization Parameters

You have control over a number of parameters involved in certain calculations. These are found in the *Optimization Parameters* section of the *Units* tab in the User Options.

Convergence Tolerance

Optimizations are involved in many different aspects of SpatialAnalyzer's computations. While these optimizations usually have a single "best" answer in theory, in reality often they can only be found through iterative analysis. As a result, finding the "correct" answer would theoretically take an infinite amount of time.


Consider a best fit optimization: once the computer finds what it perceives as the solution that best fits a series of measurements to a CAD surface, it will stop. However, if you were to increase the internal decimal precision and "jiggle" the points around with even more fine movements and rotations, it could find a better solution than the solution initially proposed.

Continuously increasing the decimal precision and making finer adjustments becomes unwieldy after a short time, and it would most certainly try your patience. While a best fit showing 1×10^{-12} inches of RMS error is better than a best fit showing 1×10^{-11} inches of error, we don't particularly care, because the difference is much less than the accuracy of the measurement in the first place. The additional computation required to reach the better solution is just not worth our time.

As a result, SA follows a set of rules that tell it when an optimization is "good enough". When SA notices that the difference between two iterations in a solution is smaller than these thresholds, the solution terminates, because the result is considered to be "good enough". The

default values provided for the convergence tolerances have been found to work well for a wide array of measurement devices and geometric optimizations.


To Edit the Convergence Tolerance Settings

1. Click the User Options  icon in the Main toolbar.
2. In the *Optimization Parameters* box of the *Units* tab, specify new length and angular convergence tolerance values.
 - **Length (object fits).** When the RMS error of a fit (such as a geometry fit, surface fit, or relationship fit) does not change by more than this value, the optimization process terminates. Using a smaller value here will theoretically give a better solution, but will increase the computation time required for the fit.
 - **Angular (bundle adj).** When the RMS pointing error in a bundle adjustment does not change by more than this value from one iteration to the next, the bundle adjustment process terminates. Using a smaller value here will theoretically give a better solution, but will increase the computation time required for the fit.

Perturbations/Damping

For many of the numeric optimizations within SA, the partials matrix is computed using numerical differentiation. A value is changed by a delta (referred to as a perturbation), and the result is computed. The deltas used to offset the values in an optimization are set here. At each step, one portion of the partials matrix is populated. If these values are too small, you can experience sensitivity problems. The perturbation may be so small that the change in the objective is negligible and therefore the optimization proceeds without accurate direction. If the values are too large, it is possible to miss nuances in the solution space by moving over the ideal solutions.

To Adjust the Perturbation/Damping Values

1. Click the User Options  icon in the Main toolbar.
2. In the *Optimization Parameters* box of the *Units* tab, specify new length and angular perturbation values, or specify a new damping factor.
 - **Length Perturbation.** The length perturbation is the amount of perturbation along the X, Y, and Z axes. In the case of a best-fit transformation, for example, the X value is adjusted by the length perturbation. The objective function is recomputed, and this process is repeated with adjustments to the Y and Z values.
 - **Angular Perturbation.** After perturbing X, Y, and Z, the Rx val-

ue is perturbed by the angular perturbation, then Ry, then Rz. At each step, one portion of the partials matrix is populated. If these values are too small, you can experience sensitivity problems. The perturbation may be so small that the change in the objective is negligible and therefore the optimization proceeds without accurate direction. If the values are too large, it is possible to miss nuances in the solution space by moving over the ideal solutions.

- **Damping.** Once the partials matrix is computed for a given step in an optimization, the matrix is inverted, and multiplied by the residual errors. This results in a vector describing how to move the input variables to minimize the objective if the system were truly linear. Since the systems we deal with in coordinate metrology are usually non-linear, the delta vector will most likely not move to the optimal solution. For this reason, we scale the vector by the damping factor to retard its effect. After making the move, the partial determination process repeats until an acceptable solution is reached.

Machine Settings

Although most parameters associated with work in SA are stored as part of the job file, many are instead stored on the computer. These settings are those that are more appropriate to be consistent on a given computer and do not necessarily have anything to do with a given SA file. For instance, the setting defining the file system location for SA backup files is a setting more appropriate to the computer instead of the SA file—regardless of which SA file is opened, it should always be saved to a certain backup folder.


SpatialAnalyzer Directories

SA uses a folder structure for storing certain files, such as measurement backups, templates, and instrument interface settings. You can control where this folder structure is located. By default, the folder structure is placed into the root of the C:\ drive and consists of the following:

- **C:\Analyzer Data\.** The base of the SA folder structure.
- **C:\Analyzer Data\Reports\.** The location for generated reports.
- **C:\Analyzer Data\Templates\.** The location for the default template file, as well as any other template files or custom user interface configuration files.
- **C:\Analyzer Data\Scripts\.** This folder can be used to store Measurement Plan Scripts, although scripts can be located anywhere on the file system.


- **C:\Analyzer Data\Persistence**. This folder contains instrument interface settings files, configuration files, parameter files, and other configuration information.
- **C:\Analyzer Data\Backup**. All automatic backup files are saved to this folder. Measurement backups are saved to a sub-directory of this directory named **Measurements**.

To Change the Location of SA's Entire Folder Structure


1. Click the User Options  icon in the Main toolbar.
2. In the *Machine Configuration* tab, click the **Select Root Directory to contain "Analyzer Data" and its subdirectories** button.
3. Choose a new folder.

The individual folder locations and names can also be changed directly from this dialog.

To Change the Location of an Individual SA Directory


1. Click the User Options  icon in the Main toolbar.
2. In the *Machine Configuration* tab's *SpatialAnalyzer Directories* box, select the desired directory.
3. Click the **Pick new Directory for Selected Entry** button.
4. Select the desired folder from the folder browser.

Parallel Processing

 **Note:** When a multi-threaded operation is underway, the two gears on the right side of the SA status bar will light up.

Many operations in SA are multi-threaded, such as USMN calculations, surface queries and projection, and point cloud filtering. However, in order to reap the substantial performance benefit of multi-threading, SA's parallel processing must be enabled (and your computer must support multi-threading).

To Enable Parallel Processing

1. Click the User Options  icon in the Main toolbar.
2. In the *Machine Configuration* tab, check the *Use multiple processors if available* option.

Fault Handling

Should you ever experience a crash or other serious runtime error while using SA, you can configure SA to automatically collect information on this error and send it on to New River Kinematics for analysis. (This is similar to the Windows error reporting functionality). You can also choose to send a *large diagnostic file*, which is a more complete record and picture of SA's state at the time of the error.

To Enable Fault Handling

1. Click the User Options  icon in the Main toolbar.

2. In the *Fault Handling* section of the *Machine Configuration* tab, enable the *Collect Diagnostic info from faults* option.
3. To also send large diagnostic files, select the *Enable large diagnostic file* option.


Security

You can place security restrictions on SA in order to restrict the access of different classes of users to SA's functionality. For example, you can prevent a user or class of users from saving a job file, measuring, deleting objects, moving objects, or using Measurement Plans.

To enable security restrictions, you first must create at least two *security classes*: One that provides administrator privileges, and another to provide restricted access. A security class defines the rights and privileges a group of people belonging to that class will have. The different rights that can be restricted are listed here:

- **Edit Security Options.** Members can edit security settings (normally this privilege is only granted to an administrator class).
- **Save Jobs.** Members can save files.
- **Print Jobs.** Members can print reports.
- **Measure.** Members can take measurements.
- **Delete Objects.** Members can delete objects.
- **Delete Points.** Members can delete individual points.
- **Delete Collections.** Members can delete entire collections.
- **Import Files.** Members can import files into the current job file.
- **Move Objects.** Members can move (transform) objects.
- **Copy Objects.** Members can make copies of objects.
- **Edit Points.** Members can change point and target properties.
- **Use MP.** Members can run Measurement Plan scripts.
- **Instrument Properties.** Members can modify instrument properties.
- **Bundle Adjust.** Members can perform bundle adjustments.

To Create a Security Class

1. Click the User Options  icon in the Main toolbar.
2. In the *Security Classes* section of the *Security* tab, click the **New** button.
3. Give the class a name, and deselect the privileges you'd like to

restrict. Then click **OK**.

To Remove a Security Class

- In the *Security Classes* section of the *Security* tab, select the class you'd like to delete and click the **Remove** button.

To Edit a Security Class

1. In the *Security Classes* section of the *Security* tab, select the class you'd like to edit and click the **Edit** button.
2. Change the name of the class and/or the associated privileges, then click **OK**.

Once two or more classes have been created, you need to create users. These users need not refer to unique people (although they can)—that is, multiple people can use the same user name.

To Create Users

1. In the *Users* section of the *Security* tab, click the **New** button.
2. Enter a name for the user, then type and confirm a password for the user.

To Remove Users

- In the *Users* section of the *Security* tab, select the user you'd like to delete, then click the **Remove** button.

To Edit Users

1. In the *Users* section of the *Security* tab, select the user you'd like to edit, then click the **Edit** button.
2. Modify the username and/or the password for the user, then click **OK**.

The last step, once classes and users have been created, is to assign users to a class. A single user can belong to one or more classes. When a user belongs to more than one class, the user receives the combined privileges of all classes to which he belongs.

To Assign Classes To a User


1. In the *Users* section of the *Security* tab, select the user you'd like to assign to a security class.
2. Next to the *Belongs to classes* box, click the **Add** button.
3. Select a class to add to the user, then click **OK**.

To Remove a Class From a User

1. In the *Users* section of the *Security* tab, select the user to modify.
2. In the *Belongs to classes* box, select the class to remove from the user.
3. Click the **Remove** button.


Once your users, security classes, and privileges have been set up, you must enable security restrictions in SA, or else no change in behavior will occur.

To Enable Security

1. Click the User Options  icon in the Main toolbar.
2. In the *Security* tab, ensure that all classes and users are set up correctly. Also verify that at least one user has full access to the *Edit Security Options* privilege, and that the user's username and password is known.
3. Uncheck the *Always Auto-login with full access* checkbox.
4. Restart SA. A username and password will be required to start the application.

With the *Always Auto-login with full access* option disabled, you can also use the *Login Now* button to log in as a different user without restarting SA. This is useful for conveniently testing the privileges of different security classes.

To Disable Security

1. Login to SA using a username that has access to the *Edit Security Options* privilege (typically the "administrator").
2. Click the User Options  icon in the Main toolbar.
3. In the *Security* tab, enable the *Always Auto-login with full access* option.

System Diagnostics

SA provides a number of system tools and utilities to monitor configuration and diagnose potential issues with your system or file setup. These tools are listed below.

Memory Monitor

Large files (particularly those containing dense scan data or complex CAD files) can easily consume the available RAM on your system. To monitor your available/used RAM, SA provides a convenient memory monitor utility, which displays SA's memory usage, total system memory usage, total RAM, and operating system information.

To View the Memory Monitor

- From the menu, select **Help>Diagnostics>Memory Monitor**.

Check Job Data Consistency

You can verify the integrity and consistency of any SA file. This tool will examine an SA file and verify that there are no corrupt internal references.

To Verify the Integrity of an SA File

1. Open the SA file.
2. Select **Help>Diagnostics>Advanced>Check Job Data Consistency**.
3. A dialog will be displayed indicating whether any errors were found, and a log of the findings will be saved to the Analyzer Data\Reports_SAJobFileConsistencyReport.txt file.

Regenerate References

If any issues are found with the *Check Job Data Consistency* tool, you can attempt to repair them by regenerating references. This will walk through the measurements and ensure that each measured point has the proper references to its targets.

To Re-Generate Target References



Note: Regenerating references is not normally necessary. It should only be performed if a job file consistency check indicates errors.

1. Open the SA file.
2. From the menu, select **Help>Diagnostics>Advanced>Re-Generate Measurement to Target References**.

Repeating Commands

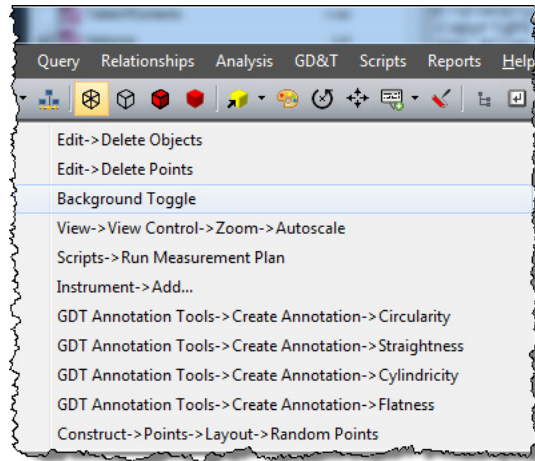
Occasionally, the need comes up to repeat a command, perhaps several times. While you could search through the menus to find the desired command and select it again, such an approach is not efficient. Instead, you can use a keyboard shortcut to repeat the last executed command.

To Repeat the Last Executed Command:

- Press **Ctrl+Tab**, or select **Scripts>Repeat Command**.

In other cases, you may need to repeat a series of non-consecutive commands several times. In such a situation, the **Repeat Command** menu option will not help you, since it only executes the most recent command. Instead, you can ask SA to display a command history of the most recently executed commands (Figure 6-19), with the most recent at the top.

Figure 6-19. The command history.



To Select a Recent Command From the Command History:

- Press **Ctrl+Shift+Tab** or select **Scripts>Command History**, then select the desired command from the resulting list.

Working With Items

Many items in an SA job have some sort of representation in space. In a few cases in which the items are containers (collections, folders, point groups, and categories), the position and orientation of an item is not clearly defined, but still exists. For example, a collection can be moved and rotated (that is—a relative movement can be applied to it), but you would have a difficult time determining where that collection is specifically located in space. The list below indicates the defined positions and orientations of the various items in SA that have a representation in space:

- **Folders.** The specific position and orientation is undefined.
- **Collections.** The specific position and orientation is undefined.
- **Coordinate Frames.** The position is defined by its origin, and its orientation is defined by its axes. The primary direction of a coordinate frame is its Z axis.
- **Point Groups.** The specific position and orientation is undefined.
- **Lines.** The position is defined by its “begin” point. The line itself describes its Z direction.
- **Instruments.** The position is determined by the base, optical center, or other defined origin of the instrument. The standing axis of an instrument is its Z axis, and typically the X axis is defined by the “front” of the instrument (although this is not always the case).

- **B-Splines.** The specific position and orientation is undefined.
- **Planes.** The position is defined by its centroid. The Z axis is defined by its normal, and its X axis is typically defined by the first defined point in the plane.
- **Circles.** The position is defined by its center. The Z axis is defined by its normal, but the X and Y axes are undefined.
- **Ellipses.** The position is defined by its center and the Z axis is defined by its normal. The X axis is defined by its major axis, and the Y axis is defined by its minor axis.
- **Cylinders.** The position is defined by the center of its “begin” end. The Z axis is defined by its axis. The X and Y axes are undefined.
- **Spheres.** The position is defined by its center, and the X axis passes through its poles. The Y and Z axes are undefined.
- **Cones.** The position is defined by its apex, and the Z axis is defined by its axis. The X and Y axes are undefined.
- **Paraboloids.** The position is defined by its focal point, and the Z axis by its axis. The X and Y axes are undefined.
- **Polygonized Surfaces, Surfaces, Vector Groups, Point Clouds, and Perimeters.** The specific positions and orientations are undefined.
- **Graphical Entities.** Each has a defined position and orientation, but graphical entities are only for graphical purposes and are not useful in analysis.

If you’re wondering why this is important, it’s because many different operations in SA that require you to define a position or orientation allow you select most of the items above to describe that position or orientation. For instance, if in the process of constructing a new coordinate frame you’re prompted to specify the X direction for the frame, selecting a circle is a valid selection for that prompt—SA will use the normal direction of the circle to define the X direction.

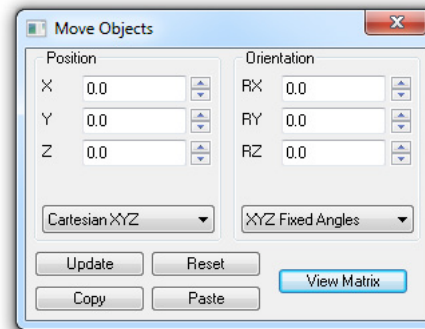
All of the objects listed above have a number of common characteristics, discussed below.

Moving Objects

While there are many commands to automatically apply transformations to objects based on special alignments and fits, all of the items mentioned in “[Working With Items](#)” above can be moved manually through any of several methods. These methods are described below.

Entering a Transform

Figure 6-20. The Move Objects dialog.



The *Move Objects* dialog (Figure 6-20) provides the ability to apply a relative movement to an object by typing in values, expressed in the active coordinate frame. In this dialog, you can explicitly type in the positional and rotational deltas—these values are initially zero because they are relative movements from a starting position. Positional deltas can be entered in Cartesian (xyz), Cylindrical ($r\theta z$), or Spherical ($r\theta\phi$) values. Rotational deltas can be entered in *Fixed notation*, *Euler notation*, *Equivalent Angle Axis notation*, or as *Euler Parameters*.

To Move One or More Objects by Typing In Values:

1. Right-click an object in the tree or graphical view and select **Move**. Or, to move multiple objects simultaneously, select **Edit>Move Objects>Enter Transformation** and select the objects you wish to move.
2. In the *Move Objects* dialog, select the proper format in which to enter the values from the dropdowns in the *Position* and *Orientation* boxes. Usually, *Cartesian XYZ* and *Fixed XYZ* is the intended format.
3. Type in the appropriate values, then press **Enter** or click the **Update** button.

To Reset an Object Back To Its Original Position:

- Click the **Reset** button, or
- Manually type zeroes into all numeric fields, then press **Enter**.

Transforms can be copied from one dialog and pasted in another. This allows you to manually transform an object in the *Move Objects* dialog, then apply that same transform to another object.

To Copy and Paste Transforms Between Instruments:

1. In the *Move Objects* dialog, ensure the transform you want to copy is entered.
2. Click the **Copy** button.
3. Call up the *Move Objects* dialog for the destination object (see “To Move One or More Objects by Typing In Values:” above).

4. Click **Paste**.

If you are performing an analysis of your own, you can copy and paste raw 4×4 transform matrices between objects, and also export transforms to a comma-delimited .CSV file.

To Copy and Paste Transform Matrices Between Objects:

1. In the **Move Objects** dialog, click the **View Matrix** button.
2. In the **Transformation Matrix** dialog, click the **Copy** button to copy the current matrix to the clipboard, or click **Export .csv** to send it to a comma-delimited file.
3. Call up the **Transformation Matrix** dialog for the destination object.
4. Check the **Allow Editing** checkbox, which enables manual editing of an object's 4×4 transform matrix.
5. Click the **Paste** button, or choose **Import .csv** to import the matrix from a comma-delimited file.

To Edit An Object's Transformation Matrix Directly:

1. In the **Move Objects** dialog, click the **View Matrix** button.
2. Enable the **Allow Editing** checkbox.
3. Type the desired values into the 4×4 matrix.

Graphical Dragging

You may wish to move objects by dragging them graphically in the view. This is a great way to visually position objects in an approximate way without being concerned with exact transformation values.

To Drag Objects Graphically:

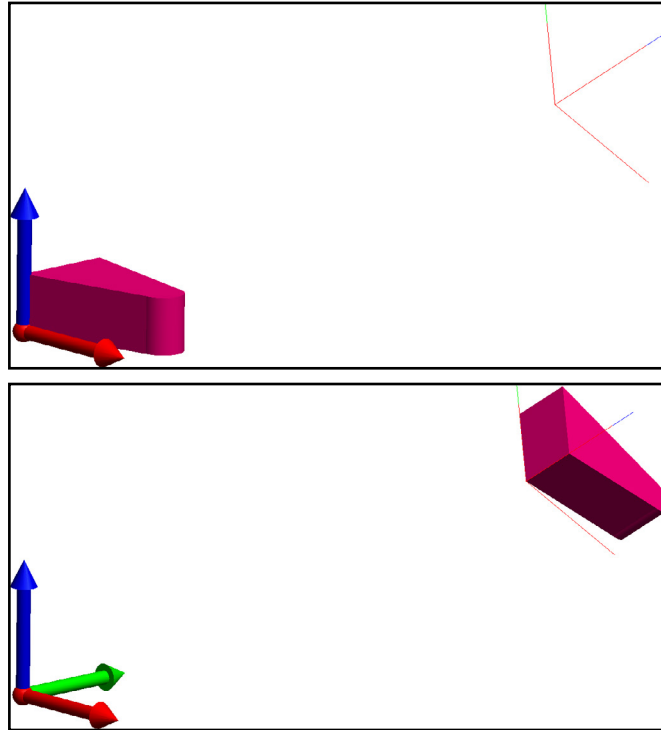
Tip: To rotate the object about different axes, use saved viewpoints to switch to different views without cancelling out of the command.

1. From the menus, select **Edit>Move Objects>Drag Graphically**.
2. At the prompt, select the objects you wish to move, then press **Enter**.
3. Use the left mouse button to move the object, and the right mouse button to rotate it about an axis perpendicular to the view orientation. When finished, press **Enter**.

Frame to Frame Transformations

Frame to frame transformations make note of the transformation required to go from one coordinate frame (the *source frame*) to another (the *destination frame*), and then applies that transformation to selected objects (Figure 6-21).

Figure 6-21. An object prior to (top) and after (bottom) a frame to frame transformation.



An interesting characteristic about coordinate frames is that they store complete information on position and orientation. As a result, they can be used to keep a record of an object's position. As an extension to that idea, two different coordinate frames store a positional and rotational delta (a transformation). Frame to frame transformations take advantage of this fact. If you plan on applying a transformation to an object, you can store the object's initial position by creating a coordinate frame at its location (the **Construct>Frames>On an Object** command is very handy for this). If, after moving the object, you change your mind and would like to put it back to its original position, you can create a frame on the object's current position, then use a frame to frame transformation to move the object back to its original location.

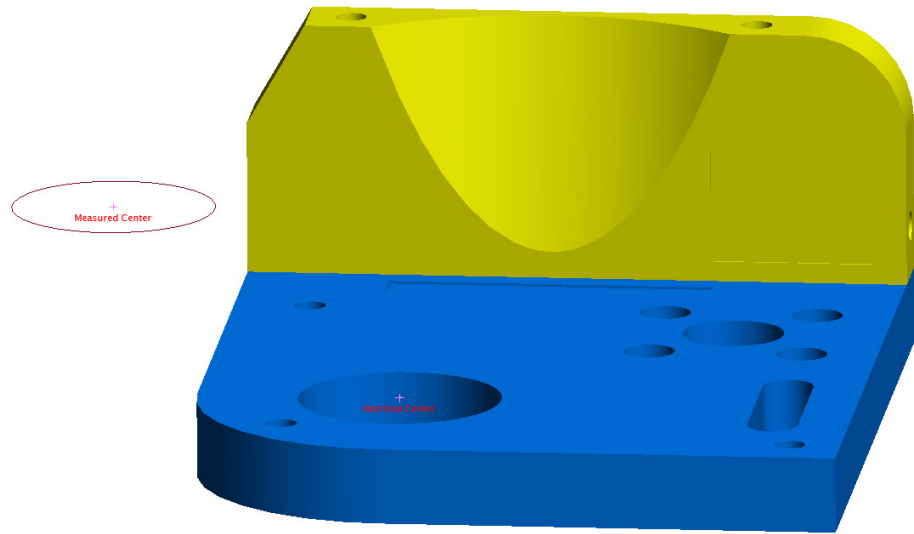
To Apply a Frame to Frame Transformation:

1. From the menu, select **Edit>Move Objects>Frame to Frame Transformation**.
2. At the prompt, select the source frame. This is the coordinate frame that the transformation will be calculated *from*.
3. Select the destination frame. This is the coordinate frame that the transformation will be calculated *to*.
4. Now that the positional and rotational deltas have been calculated, select the objects to which you'd like to apply the transform. You may choose the source or destination frame, if desired.

5. Confirm or reject the transformation at the prompt.

Point to Point Delta Translations

Figure 6-22. A circle is being moved to a nominal position through the use of a point to point translation. In this example, **Measured Center** is the source point, and **Nominal Center** is the destination point.



Often, you will want to apply a pure translation (movement in XYZ but no rotation) to one or more objects. Point to point delta transformations use a source point and a destination point to calculate a positional delta to apply to one or more objects (Figure 6-22). A point to point delta transformation is the 3 degree of freedom analog to a frame to frame transformation (which involves movement in 6 degrees of freedom). Just as frames can record an object's position and orientation in space, points can record an object's position in space. As such, point to point translations are reversible and can be used to "undo" a translation applied to an object—simply reverse the source and destination points.

To Move Objects Using a Point to Point Translation:

1. Select **Edit>Move Objects>Translate by Point to Point Delta**.
2. At the prompt, select the first (source) point.
3. Select the second (destination) point.
4. Choose the objects to translate.

Transform Tracking

Transforms do not need to be one-time events. They can be performed in series and used to dynamically update a model within SA based upon continuous measurements from an instrument or set of instruments. For more information refer to "[Transformation Tracking](#)" on page 400.

Copying Objects

Objects can be copied. The copy is always placed into the active collection. If the collection containing the copied object is the active col-

lection, the object will be renamed as necessary to maintain uniqueness. There are two main methods for copying an object: Copying and then applying a transform (which can be unity to leave the object unmoved), or copying and then applying a point to point translation. Keep in mind that since point groups are objects, copying a point group will duplicate that group of points. However, since the copied points were not actual measured points (after all, they are copies) they will be depicted as constructed points.

It is probably most common for copies to be made without applying any transformation so that the copy is in the same position as the original.

Copy, Then Transform

This method first creates a copy of one or more objects, then permits you to optionally type in a transformation to move the copies.

To Copy then Transform:

1. Right-click an object in the tree or graphical view and select **Copy**; or, to copy multiple objects, select **Edit>Copy Objects>Copy Then Transform** or press **Ctrl+C**.
2. If copying multiple objects, select the objects.
3. The **Copy Objects** dialog will appear, which is identical to the **Move Objects** dialog. Enter the desired transformation to apply, or leave the values zeroed out to leave the copy in place.

Copy, Then Translate Point to Point

If you'd just like to translate the copies only, you can create a copy and apply a point to point translation to them. This is the copy-based version of moving objects using a point to point translation.

To Copy Using Translation Only:

1. In the menus, select **Edit>Copy Objects>Translate by Point to Point Delta**.
2. At the prompt, select the first (source) point.
3. Select the second (destination) point.
4. Specify the objects you'd like to copy, then press **Enter**.

Mirroring Objects

Objects can be mirrored (reflected) about any of a frame's three planes (XY, XZ, or YZ). This is most often used with symmetric parts. For example, if certain nominal features of an aircraft model are only available on the left side, it will need to be mirrored to the right side when it comes time to work on the right side of the aircraft.

You can either create a mirrored copy of selected objects (creates a

duplicate, then reflects the duplicate), or you can mirror the objects directly without creating any copies. Mirrored copies are placed into the active collection and are renamed as necessary to ensure uniqueness. An object moved via a mirroring operation will not be renamed at all. Always use care when mirroring—depending on the object being reflected and the plane being mirrored about, the object's normal may flip directions.

To Mirror One or More Objects:

1. Select **Edit>Mirror Objects>Move** (to mirror the source objects) or **>Copy** (to mirror copies of the source objects).
2. Select the frame about which to mirror.
3. Select the objects to mirror, then press **Enter**.
4. In the *Select Mirror Plane* dialog, choose the plane about which to mirror, then click the **Mirror** button.

Scaling Objects



Warning: The most common situation in which an object is scaled is when you have point groups or other data that is not associated with an instrument and you want to apply a temperature compensation to it. Other than this situation, it is not common and is usually more appropriate to apply a scale factor to an instrument.



Note: The list of materials in the CTE Editor is stored in the SA file. If you'd like a modification to appear in all files, save your SA file as a default template.

Any object may be scaled by an arbitrary amount. This yields an object that has grown or shrunk by the corresponding scale factor relative to the active coordinate frame. Since the scaling is applied relative to the active coordinate frame, scaling generally also results in object translation (except for a point on the origin of the active frame).

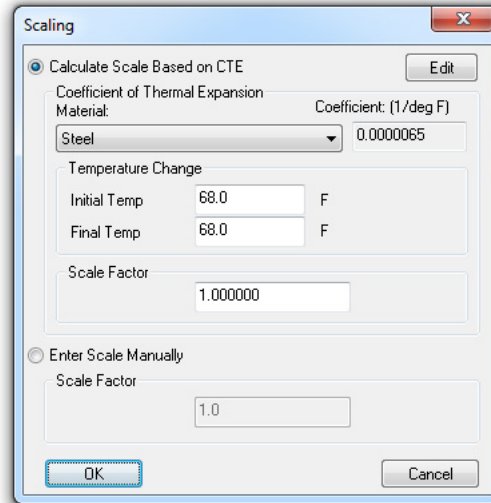
Two different methods can be used for scaling:

- **Calculate Scale Based on CTE.** With this option, you select a material and an initial and final temperature, and SA calculates the scale factor automatically using the linear thermal expansion equation. The material entered should be the material of the objects being scaled. For example, if you are scaling measurements of a steel plate, steel should be used for the material.
- **Enter Scale Manually.** You can enter your own scale factor. As an example, a scale factor of 2.0 will double the size of the object.

To Scale One or More Objects:

1. From the menus, choose **Edit>Scale Objects**.

Figure 6-23. The Scaling dialog.



2. At the prompt, select the objects to scale, then press **Enter**. The *Scaling* dialog will be displayed (see Figure 6-23).
3. Choose the scaling method (either *Calculate Scale Based on CTE* or *Enter Scale Manually*).
4. If scaling manually, enter the desired scale factor. If scaling using CTE, choose the appropriate material and initial/final temperature. Then click **OK**.

If the default list of materials is not sufficient for your needs, you may wish to enter your own material with a custom coefficient of thermal expansion.

To Enter a Custom Material and CTE:

1. In the *Scaling* dialog, click the **Edit** button.
2. In the *CTE Editor*, use the **Add**, **Edit**, and **Remove** buttons to modify the list of materials, then click **Done**.

Finding Items In The Tree

While working in the graphical view (particularly if the job file is complex and has many entities), you may need to find an item in the tree. Searching for the item through the tree is cumbersome in complex files, so SA provides the ability to find any item in the tree.

To Find an Item In the Tree:

- In the graphical view, right-click the item and select **Show In Tree** from the context menu. SA will expand, autoscroll, and select the item in the tree for you.

Network Browser

SA is a very network-capable application. You may be viewing an SA instance running on another computer on your network, or you may be connecting an instrument interface to another computer on the network. The SA Network Browser allows you to view the network and perform a few network-related diagnostic operations.

To Access the SA Network Browser:

- Click the SA Network  icon in the Main Toolbar.

The *SA Network Browser* dialog ([Figure 6-24](#)) displays at the root level all computers running SA that are visible on the network, along with their IP address. In [Figure 6-24](#), there are four computers running SA on the network.

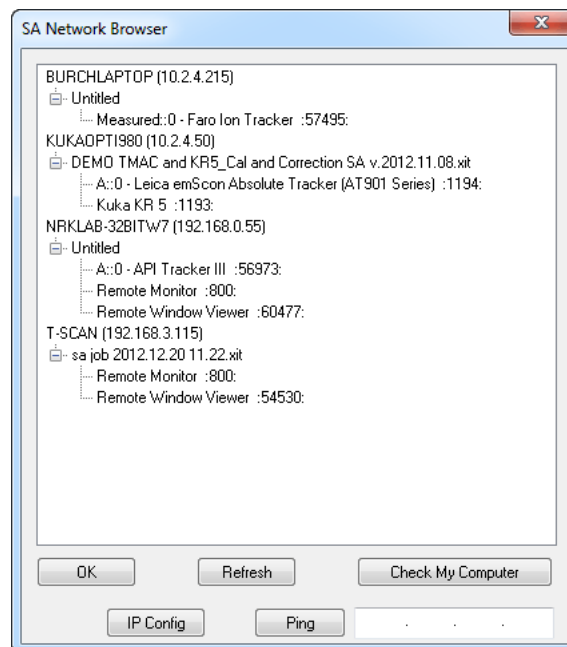


Figure 6-24. This *SA Network Browser* lists a number of instances of SA running on four different computers.

Below the computer name will be listed all open SA files on each computer, and below that all unconnected instruments or other connections that can be made. In [Figure 6-24](#), the computer named **T-SCAN** has two possible connections: A connection to the Remote Monitor and the Remote Window Viewer.

To Refresh the Network Browser List:

- Click the **Refresh** button.

To Run the Check My Computer Tool:

- Click the **Check My Computer** button.

To View the IP Settings of Your Computer:

- Click the **IP Config** button.

To Ping an IP Address:

- Type an IP address into the text field, then click the **Ping** button.

Remote Monitoring

SA provides the capability to pull an open SA file from any computer reachable on the network and work with it. You can also remotely view the SA application, graphical view, or HUD windows from another computer (if applicable).

For privacy and security reasons, both of these remote viewing capabilities require the computer being monitored to have its *Remote Viewable* option set.

To Make a Computer Viewable Via Remote Monitoring:


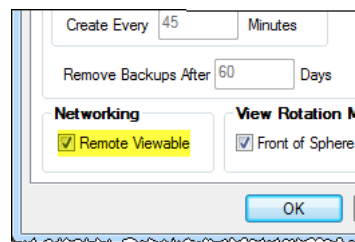
1. Visit the User Options by clicking the User Options icon  on the toolbar.
2. Navigate to the *Machine Configuration* tab and check the *Remote Viewable* option.

Figure 6-25. The *Remote Viewable* option.

**To Work With the Current File From Another Network Computer:**

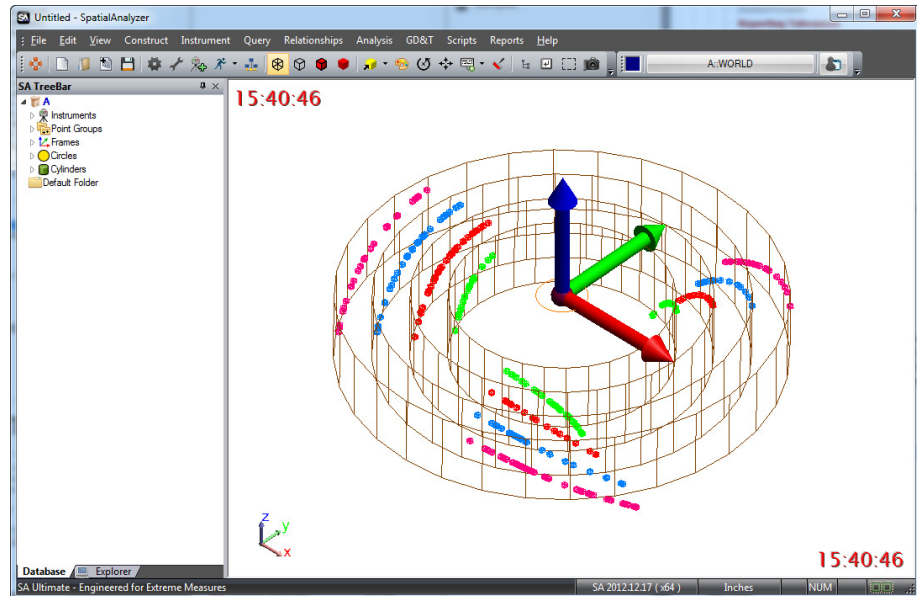
Note: Computers with the *Remote Viewable* option turned off will not be visible in this list.

1. From the menu, select **File>Remote Monitor another SpatialAnalyzer**.
2. When the *Remote Monitor Options* dialog appears, double-click any SA file's **Remote Monitor** option. Or, select the *Use manual entry of IP (or computer) and port* option and manually enter the IP address or computer name and port number to connect to, then click **OK**.

Note: When remote monitoring, SA essentially saves the file from the remote computer and loads it on the local computer. No link is maintained between the two files after that point.

The file will be pulled from the other computer and opened. A timestamp will appear in the corner of the graphical view to indicate the time when the file was pulled ([Figure 6-26](#)).

Figure 6-26. Remotely monitoring another instance of SA on the network.



To View an Active Window From a Remote Instance of SA:

1. From the menu, select **File>Remote Window Viewer**.
2. In the resulting *Remote Window Monitor Options* dialog, double-click the **Remote Window Viewer** entry that you'd like to connect to, or select the *Use manual entry of IP (or computer) and port* option and enter an IP address/computer name and port number, then click **OK**.
3. When the *Select Remote Window* dialog appears, select the desired option, then click **OK**. A separate window will appear, which will periodically get updated. You are now watching the other computer's SA instance, live.

Working With The Clipboard

The Windows clipboard is the fastest way for transferring data within or among applications. It is far better than manually typing data into a different program, and it faster and "cleaner" than exporting data to a file and re-importing in the other application.

You can copy the graphical view to the clipboard as an image for pasting into any program that supports standard image formats.

To Copy the Graphical View to the Clipboard:

- From the menu, select **Edit>Copy graphics to clipboard**.

If you've copied point data from another source—perhaps an Excel file or Notepad text file, you can paste it directly into SA.

To Paste Points From the Clipboard:

1. In the source application (Notepad, Excel, etc.), select the data

and copy it to the clipboard.

2. From the SA menu, select **Edit>Paste Points from Clipboard**.
3. Select the appropriate ASCII import options (see the chapter on File Import for detailed information on these options), then click **Import**.

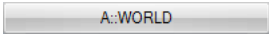
Working Frame and Locked Objects

All data within SpatialAnalyzer is represented relative to the active (working) coordinate frame, unless you have specified a specified a reporting frame for a particular object. When you examine a point's coordinates, for instance, you are seeing those coordinates relative to the working frame. When moving an object, you are entering positional and rotational deltas with respect to the working frame. When performing a best-fit and restricting rotation about the X or Y axis, the X and Y axes are defined by the working frame. As a result, staying mindful of the active frame while working in SA is very important.

You can create as many coordinate frames in your job file as you want or need. You can also freely switch from one active frame to another at any time. This allows you to view deltas, specify rotational values, or look at coordinates with respect to any frame of reference.

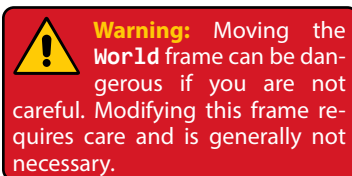
To Change the Working Frame:

Do one of the following:

- Click the Working Frame button , then select the new desired working frame from the list, or
- Right-click any coordinate frame you'd like to activate and select **Make Working Frame**, or from the menu, select **Edit>Pick Working Frame**, and select the new frame.
- Press **Alt+W** (the default keyboard shortcut), and select the new frame.

For more information on frames and frame properties see “[Frames](#)” on page 253.

About the World Frame



The **WORLD** is a unique frame that exists in every SA file and defines the base coordinate system for the job file. Either deleting or moving this World frame can result in both confusion and unforeseen problems. For that reason its properties are locked by default and are not directly accessible. In addition, efforts to move or rename this frame will result in a new World frame being build for you automatically.

If for some reason the World frame must be edited or renamed an

option exists on the Display tab of the Users Options to either disable this function or adjust the name of the locked frame (Figure 10-10).

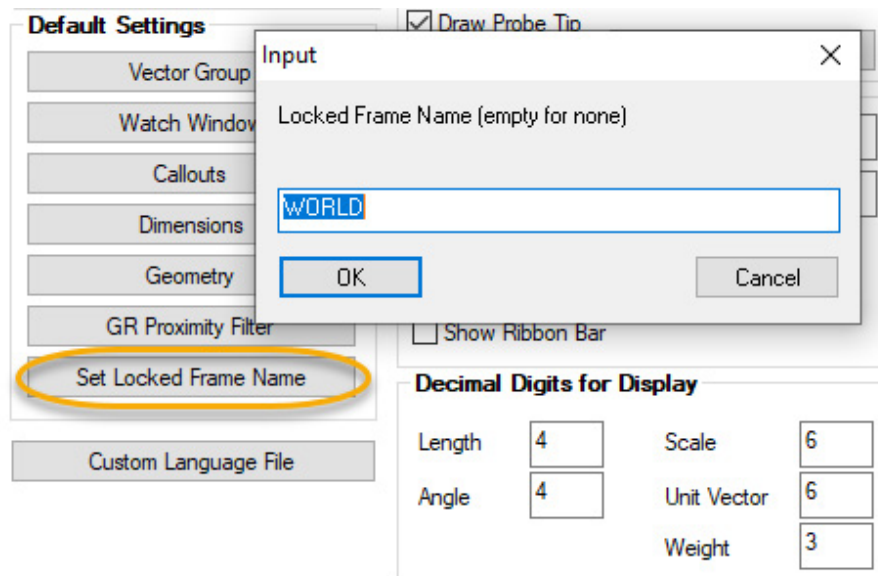


Figure 6-27. Set Locked Frame Name Control in the User Options.

Locked Items

In addition to the *World* frame, many other object types can be locked such that their position within the job cannot be edited. Locked only means entity cannot be spatially moved – all other existing operations allowed. These include:

- Imported Surfaces - which are locked automatically
- Instruments can be locked.
- SA objects not yet associated with instruments can be locked.

The item lock status can be edited using the command **Item Properties > Control Item Lock Status** (or under the *Edit* menu).

Object Color

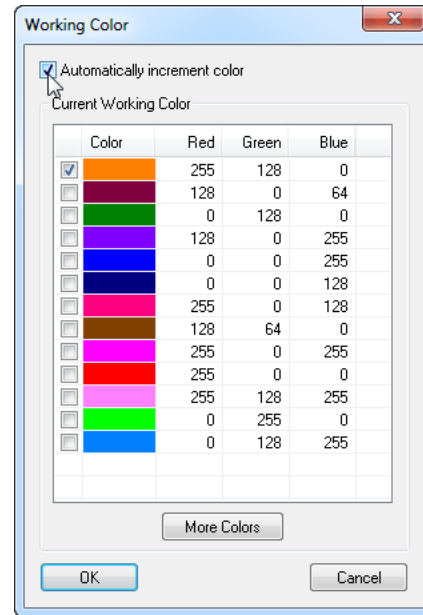
Objects can be given any color, or *painted*, which you can use for any purpose. You can use the color of an object for organizational purposes, to assign special meaning to an object, or simply to make it look good. It's up to you to decide.

SA uses the concept of a *working color*. Every time a new object is created, it is given a color. The color that is assigned is the current—or working—color. Think of the working color as an open can of paint—each object you create is painted, and the working color is used for that painting job.

By default, the working color is set to automatically increment. That

is, when an object is created and assigned a color, the next color in a list of 13 random colors is automatically made the working color. The result of this is that repeated object creation results in objects of different colors so that they are more clearly distinguishable in the graphical view.


Figure 6-28. Toggling the *Automatically increment color* option in the *Working Color* dialog.




Successive creation of multiple objects in different colors may be a behavior that you like—or, depending on what you’re doing, it may not. For instance, you may wish to create a number of nominal objects all of the same color, using the assigned color as a cue that the objects are in their nominal configuration. In this situation, the automatically incrementing working color is not desirable. You can disable working color auto-increment so that the current color remains the working color until you explicitly change it.

To Set the Working Color:

Note: Selecting **Edit>Change Working Color** does not allow selection from a list of existing working colors. On the other hand, selecting the toolbar button does.

1. Do one of the following:
 - In the Color/WCF toolbar, click the color swatch button , which displays the working color, or
 - From the menu, select **Edit>Change Working Color**.
2. Select a new working color, then click **OK**. The next object created will be assigned the working color.


To Turn Working Color Auto-Increment On or Off:

1. In the Color/WCF toolbar, click the color swatch button .
2. In the *Working Color* dialog (Figure 6-28), toggle the *Automatically increment color* option.

To Change the Color of One or More Objects:

1. Choose the desired working color as described above in “To

Set the Working Color:" on page 157.

2. From the toolbar, click the Color Objects button , or select **Edit>Set Objects to Working Color**.
3. Select the objects to paint.

You can also change the color of an object directly through its properties.

To Change the Color of an Object Through Its Properties:

1. Right-click the object in the tree or graphical view and select **Properties** from the context menu.

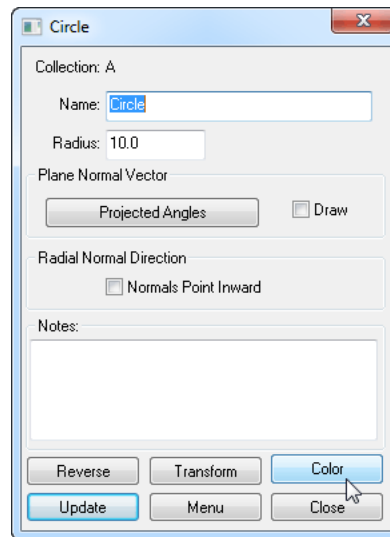


Figure 6-29. Changing the color of a circle through its properties dialog.


2. In the dialog (Figure 6-29), click the **Color** button. Select a new color, then click **OK**.

Accessing Item Properties

Nearly every item in your job has properties associated with it. Some properties are general—such as the name of the item and any notes associated with it—and other properties are more specific and depend on the object type. These more specific properties are discussed in detail later.

To Access the Properties of an Item:

Do one of the following:

- Right-click the item in the tree or graphical view and select **Properties** from the context menu, or
- From the toolbar, click the Object Properties button , then select the object to examine (applies to objects only), or
- From the menu, select **Edit>Object Properties**, then select the object to examine (applies to objects only), or

- Press **Alt+Q** (default keyboard shortcut), then select the object to examine (applies to objects only).


The Calculator

Since you may find the need to perform a quick calculation fairly frequently in SA, a shortcut is provided for calling up the Windows calculator utility.

To Start the Windows Calculator Utility:

- From the menu, select **Edit>Calculator**.

User Notes

Occasionally you may be interested in recording some field notes. You can add user notes to an SA job for later reference or for adding to a report. A user note will be added to the tree under the **Events**  category, and can optionally be added to the job's log file as well.

To Add a User Note:

1. From the menu, select **Edit>Add User Note**, or press **Ctrl+N** (default keyboard shortcut).
2. In the **Enter User Note** field, type a summary and details for the note.
3. If you'd like the note to also be logged in the job's log file, select the **Log File** option.
4. Click **OK**.

Adding Notes to Items

You can also add notes and associate them with specific items. For instance, if you're measuring a cylindrical tank to check for deformation when it's filled, you might want to add a note that a best-fit cylinder was created when the tank was filled, and add the quantity of fluid in the tank for later analysis.

To Add a Note to an Item:



Note: Some items, such as polygonized surfaces, surfaces, and relationships, do not have associated notes.

1. Right-click the item in the tree or graphical view and select **Properties** from the context menu.
2. In the **Notes** field, enter the desired note, then close the dialog.

Item History

Many different items in your job have a *history* associated with them. An item's history shows significant events related to the item's creation or manipulation. For instance, all item histories indicate when the item was added to the job. A circle's history may indicate the details of the geometry fit used to create that circle.

To View an Item's Historical Log:

- Right-click the item in the tree or graphical view and select **History** from the context menu.

To Expand or Collapse all Entries in an Item's History:

- In the item's history dialog, click the + or - button.

To Change the Chronological Order in Which an Item's History is Displayed:

- In the item's history dialog, choose the *Ascending* or *Descending* option.

To Copy an Item's History to the Clipboard:

- In the item's history dialog, select the entries to copy (or select none to copy them all), then click the **Copy to Clipboard** button.

To Export an Item's History to a Text File:

- In the item's history dialog, click the **Export Entire Log as Text File** button. Choose a filename, then click **Save**.

To Search an Item's Historical Log:

- In the item's history dialog, click the **Search** button. In the resulting **Search** dialog, type a string to search for, then click **Find Next**.

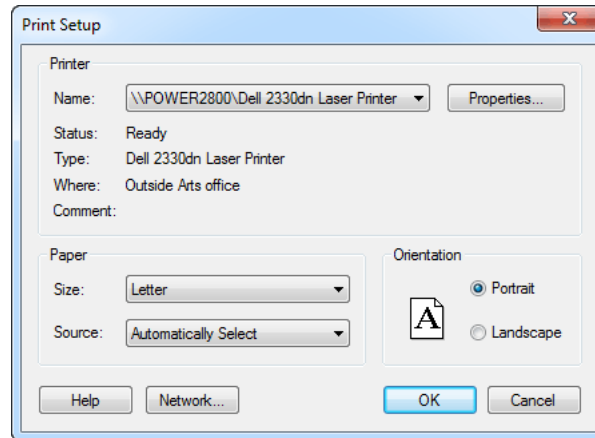
To Trim an Item's Historical Log:

- In the item's history dialog, click the **Trim Log** button. Confirm the action, then specify the number of log entries to keep and click the **OK** button.

Print Setup

Default printing options can be setup through the *Print Setup* dialog (Figure 6-30).

Figure 6-30. The *Print Setup* dialog.



This standard dialog's options depend on the selected printer but define how output from printed reports is handled by the printer. For more information, refer to your printer's documentation.

To Modify Print Settings:

1. From the menu, select **File>Print Setup...**
2. Choose the desired settings, then click **OK**.

Command-Line Arguments

SpatialAnalyzer has a series of command-line arguments that allow you to automate some tasks when launching the program:

Argument	Description
/?	Displays a help window showing all of the available command-line arguments.
{FILENAME.XIT}	Automatically opens the specified SA job file upon launch. Replace {filename.xit} with the path to the desired file.
/MP {FILENAME.MP}	Runs the specified Measurement Plan file upon launch. Replace {filename.mp} with the path to the desired measurement plan file.
/MPE {COLLECTION}:: {MP NAME}	Runs the specified embedded Measurement Plan upon launch. Replace {COLLECTION} with the collection name, and {MP Name} with the name of the embedded MP in the SA tree.
/QUICK {INSTRUMENT NAME}	Launches SA, starts a new file, adds the entered instrument name, and starts its interface. Replace {Instrument Name} with the name of the instrument (as specified in the Add Instrument to SA dialog).
-MINIMIZE	Starts SA in a minimized window state.

Combinations of these arguments may be executed at the same time. Quotes are required around the elements (filenames, instrument names, and object names) if they contain spaces.

Launching SA with Command-Line Arguments

Command-line arguments were traditionally used when running a program from the DOS (command prompt) window. Now that graphical operating systems like Windows have become standard, command-line arguments are usually entered by use of a shortcut icon. Regardless, they can still be executed from the Windows command prompt.

To Use Command-Line Arguments from the Windows Command Prompt:

1. In the windows search box, type **cmd** and press **Enter**.
2. In the resulting window, navigate to the SpatialAnalyzer install directory by typing **cd {PATH}**, where **{PATH}** is the path to the SA install directory. For example, **cd C:\Program Files (x86)\New River Kinematics\SpatialAnalyzer 2021.2.71490.1\x64** Press **Enter**.
3. Type **“Spatial Analyzer64” {ARGUMENTS}**, where **{ARGUMENTS}** includes one or more of the arguments listed above. Press **Enter**.
4. SA should launch, using the parameters that you specified in the arguments.

To Use Command-Line Arguments via a Shortcut Icon:

1. Create a shortcut for the SpatialAnalyzer program (C:\Program Files (x86)\New River Kinematics\SpatialAnalyzer 2019.11.21_62452\x64\Spatial Analyzer64.exe).
2. Right-click on the icon and choose **Properties**.
3. In the **Properties** dialog, locate the **Target** field, add a space at the end of the text in this field, then add the arguments as specified above. When finished, click the **OK** button
4. Double-click the icon to start SA with the specified parameters.

Example Command-Line Arguments

To:	Command
Start SA With A File	“Spatial Analyzer” “C:\MyFolder\My Template File.xit”
	“Spatial Analyzer” FileInThisFolder.xit
View the Command-Line Argument Help	“Spatial Analyzer” /?
Start an External MP in a Specified File	“Spatial Analyzer” /MP C:\MyScripts\MyMP.mp MyFile.xit
Start an Embedded MP in a Specified File	“Spatial Analyzer” /MPE Nominal::MyEmbeddedMP MyFile.xit
Add/Startup a Leica Tracker in a New File	“Spatial Analyzer” /QUICK “Leica emScon Absolute Tracker (AT901 Series)”

To:	Command
Add/Startup an API AXXIS Arm in a Specified File	“Spatial Analyzer” /QUICK “AXXIS 6-200 Arm (3.2m 6 dof)” myFile.xit

Multiple Instances and Exiting SA

SA does not have a “tabbed” interface. If you would like to have multiple copies of SA open on a single computer, simply open the application again.

To Exit SA:

- From the menu, select **File>Exit**, click the “X” in the top-right corner of the application window, or double-click the application icon in the top-left corner of the window.

This Page Intentionally Left Blank.

Working With Files

CHAPTER

7

While working in SA you will likely be working with many types of files—SA job (.XIT) files, CAD files containing nominal designs, and text files containing point coordinates, for example. This chapter explains how to work with external files in SpatialAnalyzer.


SA Files

Creating New Files

Creating a “new” file in SA does not necessarily mean that you’re working with a blank slate, as you would be in some applications. Even a completely new file has at the very least a collection, a coordinate frame, and myriads of settings for anything from colors and units to decimal precision.

When a new file is created without the existence of a default template, SA opens a factory default file—that is, a file that has default settings hardcoded by NRK.

To Create A New File:

- From the menu, select **File>New**, or click the  icon on the toolbar. If you currently have an SA file open, you will be asked if you would like to save it.

If you have an existing file open, you can restore it to the factory default settings (colors, decimal precision settings, etc).

To Revert a File’s Settings to the Factory Default:

- With the file open, select **File>Restore Default Settings**. The settings will return to the defaults. You will of course need to save the file to store the changes.


Opening Files


The Open File command permits opening native SpatialAnalyzer files. These are commonly referred to as *job files* and come in one of two formats:

- **.xit**. This is a standard 32-bit .xit file. It can be opened by both 32- and 64-bit versions of SA, but cannot have extremely large data sets. It is limited to storing data that will fit in a 32-bit software environment.
- **.xit64**. This is a 64-bit SA file. It can only be opened in the 64-bit version of SA, but is able to store very large data sets (such as extremely dense scan sets).

SpatialAnalyzer job files are backward compatible—you can open an .xit or .xit64 file saved in an older version of SA in a newer version. However, the reverse is not true—attempts to open a job file saved in a newer version of SA into an older version will fail.

To Open an Existing File:

 **Note:** If you double-click a file in Windows Explorer, it will be opened in the most recent version of SA that has been opened.

- From the menu, select **File>Open**, or
- Click the  icon on the toolbar, or
- Press the default keyboard shortcut of **Ctrl+O**, or
- Double-click an SA file in Windows Explorer.

To Open a Recent File:

- From the menu, select **File>[Filename]** toward the bottom of the file menu.

Saving Files


You can save a SpatialAnalyzer job file in one of two formats, as described in “Opening Files” on page 166. Any current job file can be saved in either format, however it’s important to keep a few things in mind:

- If you want to be able to open the file in either the 32-bit or 64-bit version of SA, you must save the file in the 32-bit (**.xit**) format.
- If the file contains a very large dataset, and you want to preserve the integrity of the data, it must be saved in the 64-bit (**.xit64**) format and cannot be opened in the 32-bit version of SA.
- A very large file can be saved in the 32-bit (**.xit**) format if you’d like, however there will likely be loss of data. You will be warned if this occurs.


File Checksums

To protect against data integrity issues, SA can perform a checksum of your SA file immediately after a save operation. This essentially verifies that the file on disk matches the currently opened file, assuring that the save was successful.

To Verify File Checksums After Saving:

1. In the User Options  dialog, navigate to the *Machine Configuration* tab.
2. Enable the *Verify File Checksums After Saving* option.

To Save an SA file:

1. From the menu, select **File>Save**, click the  icon on the toolbar, or use the default keyboard shortcut (**Ctrl+S**).
2. In the *Save As* dialog, select the desired format from the *Save as type:* dropdown, then click **Save**.

The first time you save a file, you will be prompted to choose a name and format. Subsequent saves will simply save the file and not prompt you for a file name or format.

Save As

If you choose to save the current file under a different filename or format, you can select the **Save As** command at any time. This permits you to specify a new filename or format, even if you've already saved the current file.

To Save the File With a New Name or Format:

1. From the menu, select **File>Save As**.
2. In the *Save As* dialog, specify a new filename and format, then click **Save**.

Backing Up Files

SA can be configured to capture periodic, automatic backups of your working file which requires no user intervention. However, you can also automatically save a backup copy of your current file. This backup will be saved to the current backup folder and the filename will be appended with a timestamp.

To Force A Backup File To Be Saved:

- From the menu, select **File>Backup Now**. After a short delay, the file will be saved to the backup folder.

Automatic Backups

Losing measurement data due to crash or computer malfunction is

always a risk. For that reason SA defaults to using automatic backups. These backup settings are controlled in the *User Options* on the *Machine Configuration* tab. There are 3 types of backups that can be performed at a selectable interval:

- **Job File Backups.** These backup files save the same information as a standard job file. (In fact, job file backups are simply regular SA files that are saved automatically for you).
- **Measurement Backups.** These backups only store instruments and measurements.
- **MPs.** If MP scripts are changed they can also be saved automatically as you work.


Backups are saved to the designated SA directory (which by default is *C:\Analyzer Data\Backup*).

So why are there two types of job backups? The main reason is to save space. You might be working in a file with a very large complex nominal CAD file. There is simply no reason why all of this information should be saved in a backup, because the CAD surfaces are typically not modified. As a result, they mostly just waste hard disk space. Measurement backups, on the other hand, only store instrument and measurement data which is much more critical and significantly smaller in file size. As a result, frequent measurement backups can be made without wasting hard drive space or causing performance delays, whereas frequent job file backups would cause a much bigger hit to performance and available drive space. Measurement backups are also regular SA files, and can easily be re-merged into the SA file containing the nominal CAD data using the **File>Import** command.

Sometimes, you might open a file, make a few changes, and accidentally overwrite your original file, which could be disastrous without a backup. To help prevent this type of mistake, you can optionally ask SA to create a file backup automatically upon opening the file, so you have a pristine, untouched backup in place.

You can also ask SA to automatically create Measurement Plan backup files while making changes in the MP Editor. If the specified time has elapsed and changes have occurred to your MP, SA will store a backup of your script to the Backups folder.

To Enable Automatic Backups:

1. Click the User Options  icon in the Main toolbar.
2. In the *Automatic Backups* section of the *Machine Configuration* tab, click the *Enabled* checkbox for the appropriate backup type.
3. Select a time interval at which enabled backups should occur.
4. If desired, enable the *Automatically Create on File Open* option

to store a backup upon opening a file.

or

- To quickly toggle automatic backups without changing other backup settings, toggle the **Edit>Automatic File Backup** setting in the menus (this does not apply to MP Backups).

To Restore a Job File Backup

- Select **Edit>Undo to Restore Point** from the menus. Select a desired restore point from the list, or
- Select **File>Open** and select the desired file in your backup folder.

To Restore a Measurement Backup

- Select **File>Open** and select the desired file in your backup folder.

To Restore a Measurement Plan Backup

- In the MP Editor, select **File>Open** and select the desired .MP file in the backup folder.

Restore Points

Restore points can be thought of as “quicksaves”. At any time, you can create a quick backup of your current files by creating a restore point. You can then revert back to the restore point to continue from that point. They are not a complete alternative to an undo system (which is not available in SA) as they require you to consciously create a restore point. However, they are very useful as quicksaves of your working job file.

To Create a Restore Point:

Do one of the following:

- From the menu, select **Edit>Create Restore Point**, or
- Press **Alt+Space** (the default keyboard shortcut).

To Undo to a Restore Point:

Do one of the following:

- From the menu, select **Edit>Undo to Restore Point**, or
- Press **Ctrl+Z** (the default keyboard shortcut).

Password-Protecting SA Files

For safety purposes, you may choose to password-protect your SA file to prevent accidental or malicious modification. This involves assigning a password to the current job file, then saving it. When the file is opened in the future, it requires a password to access the data.

To Password-Protect a Job File:

Warning: The password protection scheme is not intended to be a secure method of data storage. Password protection can be circumvented by a hacker relatively easily. If you need high security, you should use a tool designed for secure data storage.

1. From the menu, select **File>Password Protection>Set Password**.
2. In the *Set Password* dialog, enter your password into both fields.
3. When prompted, confirm that you wish to apply a password to the file.
4. Save the file. It will now require a password to open.

To Remove a Password From a Job File:

1. From the menu, select **File>Password Protection>Remove Password**.
2. At the prompt, confirm that you would like to remove the password.
3. Save the file. A password will no longer be required to open it.

File Import


SA supports a wide variety of common file formats, making data sharing easy. You can work with various CAD formats, text-based point formats, custom formats and embedded formats. The act of importing a file will always merge it into the existing job file.

When a file is imported, the data is transformed by aligning the working frame or active coordinate system of the imported file to the working frame of the current file.

Auto Import

You can let SA choose the proper import method for a file based on its file extension. You do this using the **Auto Import (by file extension)** command. Be aware that this depends on the file extension to match the format of the file. If there is a mismatch, the import will fail.


To Auto-Import a File by Type:

1. Do one of the following:
 - From the menu, select **File>Auto Import (by file extension)**, or
 - In the toolbar, click the Auto Import  icon, or
 - Drag the file from Windows Explorer into the graphical view, or
 - In the tree, switch to the *Explorer* tab, find the file to import, then double-click it.
2. The appropriate file import command will be triggered.

Importing SpatialAnalyzer Job Files

It is often desirable to merge another SA job (.xit file) into the current SA file. This is particularly useful when portions of measurement have been accomplished on various computers and files. You can choose which collections should be imported.

To Import Another SA File Into the Current Job Using the Menus:

 **Note:** If any imported collections have the same name as existing collections, they will be automatically renamed.

1. Choose **File>Import>SpatialAnalyzer Job File (.xit)**. Choose the file to import.
2. From the **SA File Import Filter** dialog, check the collections you'd like to import.
3. Click **Import Selected Collections**.

Direct CAD Access

The **Direct CAD Access** import option allows a user to import native CAD files directly into SpatialAnalyzer. The following is a list of supported formats:

Format	Extensions	Version	Polygonized Surfaces	Surfaces	Annotations
3MF	.3MF	1.2.3	✓	N/A	N/A
ACIS	.SAT, .SAB	Up to 2023	✓	✓	N/A
AutoCAD - 2D	.dwg, .dxf	DWG 2018, RealDWG up to 2021	✓	N/A	✓
AutoCAD - 3D	.dwg, .dxf	Up to 2019	✓	✓	N/A
Autodesk 3DS	.3DS	Any	✓	N/A	N/A
Autodesk DWF	.DWF, .DWFx	Any	✓	N/A	N/A
Autodesk Inventor	.IPT, .IAM	Up to 2024	✓	✓	✗
Autodesk Navisworks	.NWD	From 2012 to 2024	✓	✓	✗
CATIA V4	.MODEL, .SESSION, .DLV, .EXP	Up to 4.2.5	✓	✓	✓
CATIA V5	.CATPRODUCT, .CATPART, .CATSHAPE, .CGR	Up to V5_6 R2023	✓	✓	✓
CATIA V6 / 3DEXperience	.3DXML	Up to V5-6 R2019 (R29)	✓	N/A	✓
COLLADA	.DAE	Any	✓	N/A	N/A
Creo - Pro/E	.ASM, .NEU, .PRT, .XAS, .XPR	Pro/Engineer 19.0 to Creo 10.0	✓	✓	✓
DGN ⁵	.DGN	7	✓	N/A	N/A
FBX	.FBX	Ascii: from 7100 to 7500, Binary: All	✓	N/A	N/A
GL Transmission Format	.GLTF, .GLB	Version 2.0 only	✓	N/A	N/A

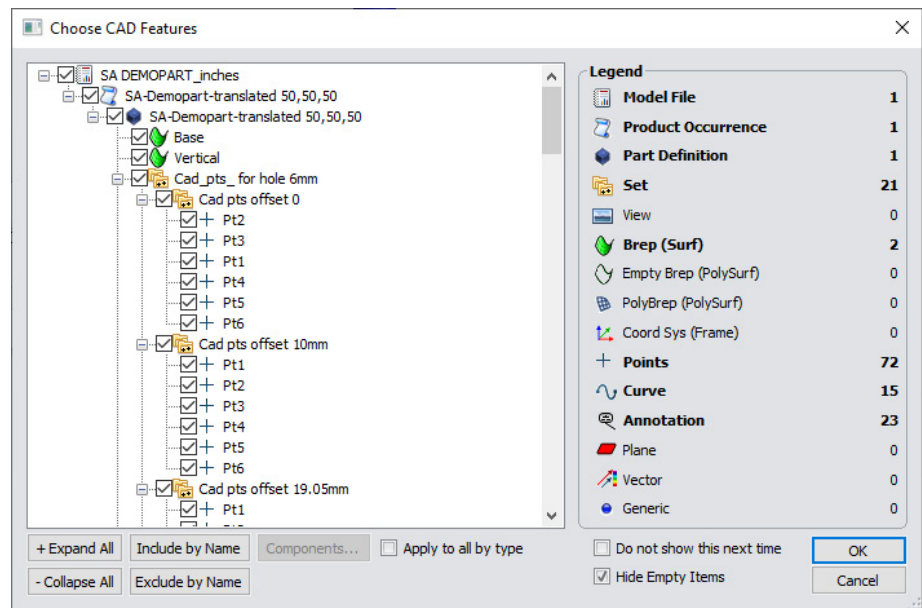
Format	Extensions	Version	Polygonized Surfaces	Surfaces	Annotations
I-deas	.MF1, .ARC, .UNV, .PKG	Up to 13.x (NX 5), NX I-deas 6	✓	✓	✓
IFC	.IFC, .IFCZIP	IFC2x Editions 2, 3 & 4	✓	✓**	N/A
IGES	.IGS, .IGES	5.1, 5.2, 5.3	✓	✓	N/A
FBX 7	FBX	ASCII: from 7100 to 7500. Binary: All.	✓	N/A	N/A
JT	.JT	Up to v10.9	✓	✓	✓
NX - Unigraphics	.PRT	UG11 to UG18, UG NX, NX5 to NX12, NX1847 to NX2306	✓	✓	✓
Parasolid	.X_T, .X_B, .XMT, .XMT_TXT	Up to v36.0	✓	✓	N/A
PDF	.PDF	All Versions	✓	✓	✓
PRC	.PRC	All Versions	✓	✓	✓
Revit	.RVT, .RFA	2015 to 2024	✓	✗	✗
Rhino3D	.3DM	v4 to v7	✓	✓	N/A
Solid Edge	.ASM, .PAR, .PWD, .PSM	1 to 20, ST1 to ST10, 2019 to 2024	✓	✓	✗
SolidWorks	.SLDASM, .SLDPRT	From 97 up to 2024	✓	✓	✓*
STEP	.STP, .STEP	AP 203 E1/E2, AP 214, AP 242	✓	✓	✓
STEP/XML	STPX, STPXZ	Any	✓	✓	✓
Stereo Lithography (STL)	.STL	All Versions	✓	N/A	N/A
U3D	.U3D	ECMA-363	✓	N/A	N/A
VDA-FS	.VDA	Version 1.0 and v2.0	✓	✓	N/A
VRML	.WRL, .VRML	V1.0 and v2.0	✓	N/A	N/A
Wavefront OBJ	.OBJ	Any	✓	N/A	N/A
				Supported:	✓
				Unavailable in this CAD Format:	N/A
				Not currently offered in SA:	✗
* Our SolidWorks PMI reader is limited.					
**IFC files can be imported as surfaces when saved as IFC Advanced B-reps only					

To Natively Import a CAD File:

1. From the menu, select **File>Import>Direct CAD Access**.
2. Select the file you'd like to import.

3. In the **Direct CAD Access** dialog (if you've chosen to show this dialog), choose the desired import options.
4. If the **Selective Import** option is checked in the **Direct CAD Access** dialog, the **Choose CAD Features** dialog will be shown (Figure 7-1). Choose the desired features to import, then click **OK**. For more on this dialog, see “**Selective CAD Import**” on page 173.

Figure 7-1. The selective CAD import dialog.



Selective CAD Import

The **Choose CAD Features** dialog (Figure 7-1) is only displayed when the **Selective Import (Choose which Components to Import)** option in the **Direct CAD Access Settings** dialog is enabled (see “**Direct CAD Access Settings**” on page 173). It lists all components of the CAD model or assembly in a tree-like view on the left, and a legend at right indicates the type and quantity of all of the entities to be imported. In this dialog you can choose to deselect entities from the source CAD file that you do not wish to import. This provides several benefits:

- It saves you the effort of having to clean up the CAD file of unwanted items after import.
- It makes the import process shorter.
- It allows you to circumvent portions of a CAD file that may be causing import errors.

Direct CAD Access Settings

The Direct CAD Access Settings control a number of options related to import via Direct CAD Access.

To Open the Direct CAD Access Settings:

- From the menu, select **File>Import>Direct CAD Access Settings**, or

- From the User Options  *Machine Configuration* tab, click the **Direct CAD Access Settings** button.

The options for this dialog are explained below.

Entity Types to Import:

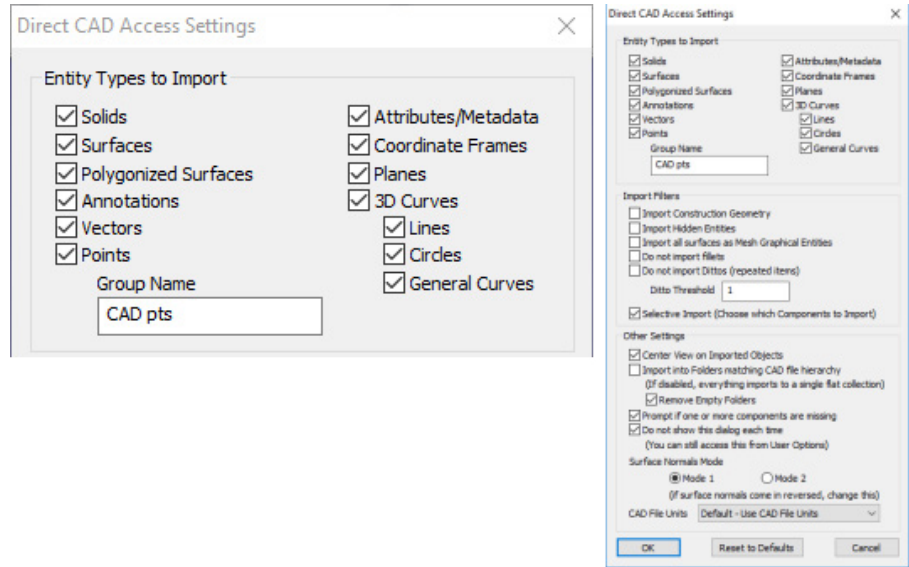


Figure 7-2. The *Direct CAD Access Settings* dialog, section 1.

- Entity Types to Import.** These checkboxes give you control over which types of entities to import. Any unchecked entities will not be imported (Figure 7-2).
- Group Name.** All points from the CAD file will be imported into a group with this name.

Import Filters:

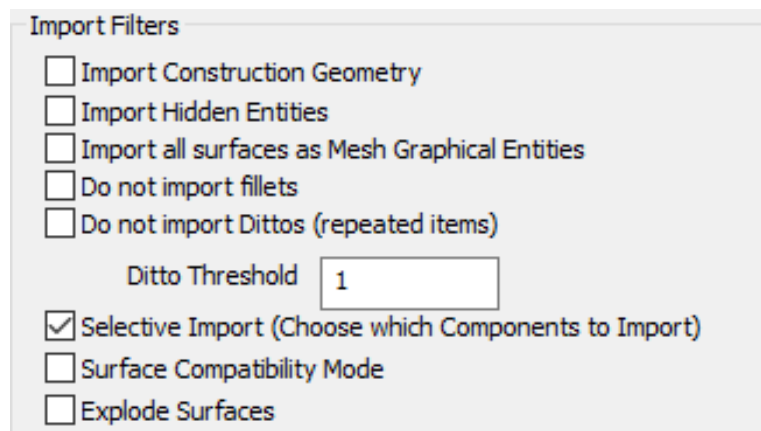


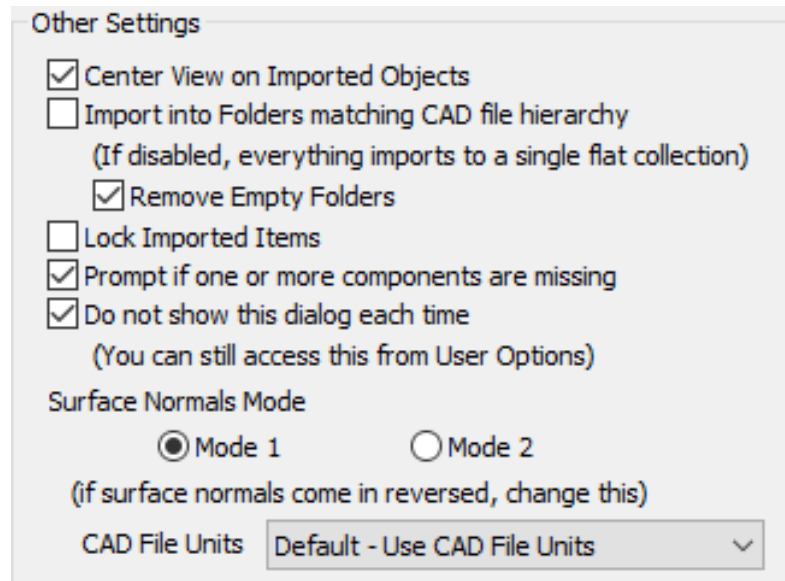
Figure 7-3. The *Direct CAD Access Settings* dialog, section 2.

- Import Construction Geometry.** For CAD formats that support it, unchecking this box will exclude construction geometry from the import (Figure 7-3).
- Import Hidden Entities.** For CAD formats that support it, unchecking this box will exclude hidden entities from the import.

- **Import all surfaces as Mesh Graphical Entities.** When checked, surfaces from the CAD file are imported as Mesh Graphical Entities instead of true surfaces. Mesh Graphical Entities are useful for visualizing surfaces and have a significantly lower requirement for computing resources than surfaces, but you cannot analyze against them.
- **Do not import fillets.** Usually, CAD surfaces contain minor features that may not be important to a metrology inspection. Fillets or modeled weld beads are one example of this. To keep your imported model as efficient as possible, check this option to exclude fillets from the import (if the file type supports marking the features as such).
- **Do not import Dittos (repeated items).** Some file formats support “dittos”, which are efficient memory copies of surfaces that are repeated over and over in a CAD file. For instance, hardware such as bolts, nuts, and screws are often modeled as dittos to save memory in the CAD application. These are features that are rarely inspected by metrology—check this option to exclude them from the import.
- **Ditto Threshold.** Some formats support multiple levels of dittos (dittos of dittos, for instance). If necessary, you can modify this number to indicate the threshold below which dittos are not imported. Higher numbers import less dittos.
- **Selective Import (Choose which Components to Import).** When enabled, the *Choose CAD Features* dialog is displayed to allow you to more finely control which entities are imported from the CAD file (see “[Selective CAD Import](#)” on page 173).
- **Surface Compatibility Mode.** When enabled, this function attempts to adapt surfaces (brep) from the imported CAD to improve compatibility with SA rendering. Try enabling this when running into import problems with specific CAD files where faces appear broken or are missing.
- **Explode Surfaces.** When enabled, individual surfaces will be created for each face in the imported model. Individual face colors will be set as the object colors, preserving more of the color information from the model during import. Use this function with caution when importing large models because abundant surfaces can slow down tree operations.

Other Settings:

Figure 7-4. The *Direct CAD Access Settings* dialog, section 3.



Tip: Importing while matching CAD hierarchy can result in a somewhat cumbersome, cluttered SA job file. Use this option sparingly, and only when needed.

- **Center View on Imported Objects.** When enabled, the graphical view will be centered on the imported objects after import (Figure 7-4).
- **Import Into Folders Matching CAD File Hierarchy.** When enabled, if the source CAD format supports a hierarchy, that hierarchy will be preserved in the SA tree using a hierarchy of folders and collections.
- **Remove Empty Folders.** Importing a CAD file into folders matching the CAD file hierarchy (see previous option) can sometimes result in empty folders in the tree. Enabling this option removes these empty folders after the import operation.
- **Lock Imported Items.** When enabled imported items that can be locked will have this status set automatically.
- **Prompt if One or More Components Are Missing.** Several CAD formats support the idea of a product or assembly file, which is a master file that has several supporting component CAD files. When importing these master files, this option will warn you if any subcomponent CAD files that are part of the master assembly are not found.
- **Do Not Show This Dialog Each Time.** When enabled, the *Direct CAD Access Settings* dialog will not display on import, but will use the previous settings. If you want to access the settings again, follow the steps in “[To Create A New File:](#)” on page 165.
- **Surface Normals Mode.** This setting has two possible options: Mode 1 or Mode 2. Some CAD formats or CAD packages do not consistently define or store the direction of surface normals. Incorrectly oriented CAD surface normals will cause incorrect

analysis in SA. While surface normals can always be corrected manually, if you find that one mode is importing surfaces in the wrong direction, trying Mode 2 might solve the problem.

Importing Standard CAD Formats

SA can import several standard CAD formats using an additional second import utility called the “classic” importer. These formats include:

- **STEP.** Import via **File>Import>Standard CAD Formats>STEP.**
- **IGES.** Import via **File>Import>Standard CAD Formats>IGES.**
- **VDA.** Import via **File>Import>Standard CAD Formats>VDA.**
- **SAT.** Import via **File>Import>Standard CAD Formats>SAT.**

In addition to providing additional import formats, you can also use this classic import method if you encounter problems attempting to import using Direct CAD Access. During the CAD import you can select from a variety of import options (Figure 7-5):

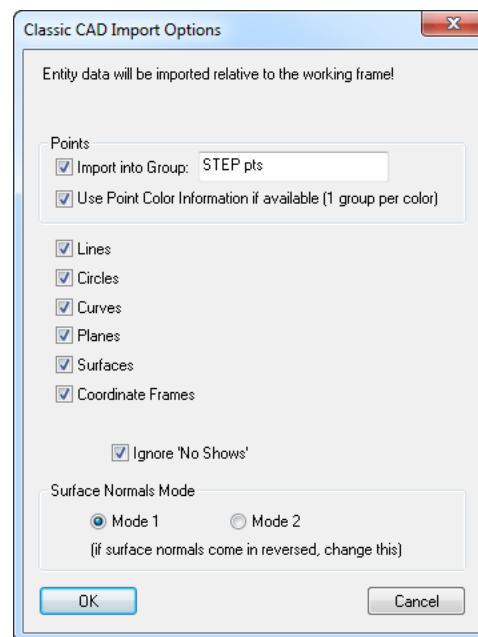


Figure 7-5. The *Classic CAD Import Options* dialog.

- **Import into Group.** Points from the CAD file will be imported into a group with the specified name.
- **Use Point Color Information if available (1 group per color).** Upon importing points from the CAD file, this option will set the group color to match the current point color from the source file, if that color exists. Each point will be organized into a group by its color.
- **Lines/Circles/Curves/Planes/Surfaces/Coordinate Frames.** Choose which types of entities to import.
- **Ignore ‘No Shows’.** When selected, items hidden or on a “no

show" layer in the file (if supported) will not be imported.

- **Surface Normals Mode.** Equivalent to the same option in the *Direct CAD Access Settings* dialog (see "[Direct CAD Access Settings](#)" on page 173).

To Import a Standard CAD Format Using the Classic Importer:

1. Choose **File>Import>Classic CAD Importers>(Format)**.
2. Select the file to import from the dialog, and click **OK**.
3. Choose the desired CAD import options, then click **OK**.
4. A summary of the file information will be displayed. To save it to file, click **Save to File**. Otherwise, click **Accept**.
5. If the current file's units do not match the source units, you will be prompted whether you'd like to convert the units. Click **Yes** or **No**.

Note: If units between files differ, the units are converted (if you choose). The CAD data is not physically scaled at all.

Importing Predefined ASCII Formatted Files

ASCII text files are common in coordinate metrology. They provide a standard, human-readable format that many software packages can easily import. SpatialAnalyzer supports many formats of ASCII data. [Figure 7-1](#) shows an example of a file in "Point Name, X, Y, Z" format.

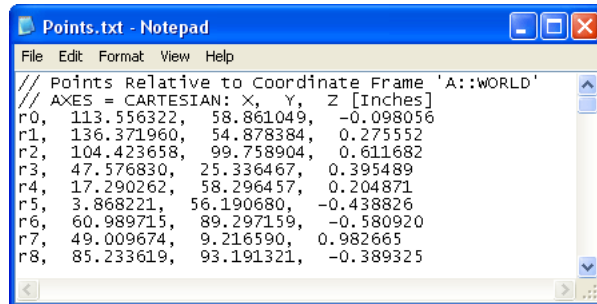


Figure 7-6. A text file containing point data in Pointname, X, Y, Z format.

When a file is selected, you will be presented with several options to define how SA will import the file ([Figure 7-7](#)):

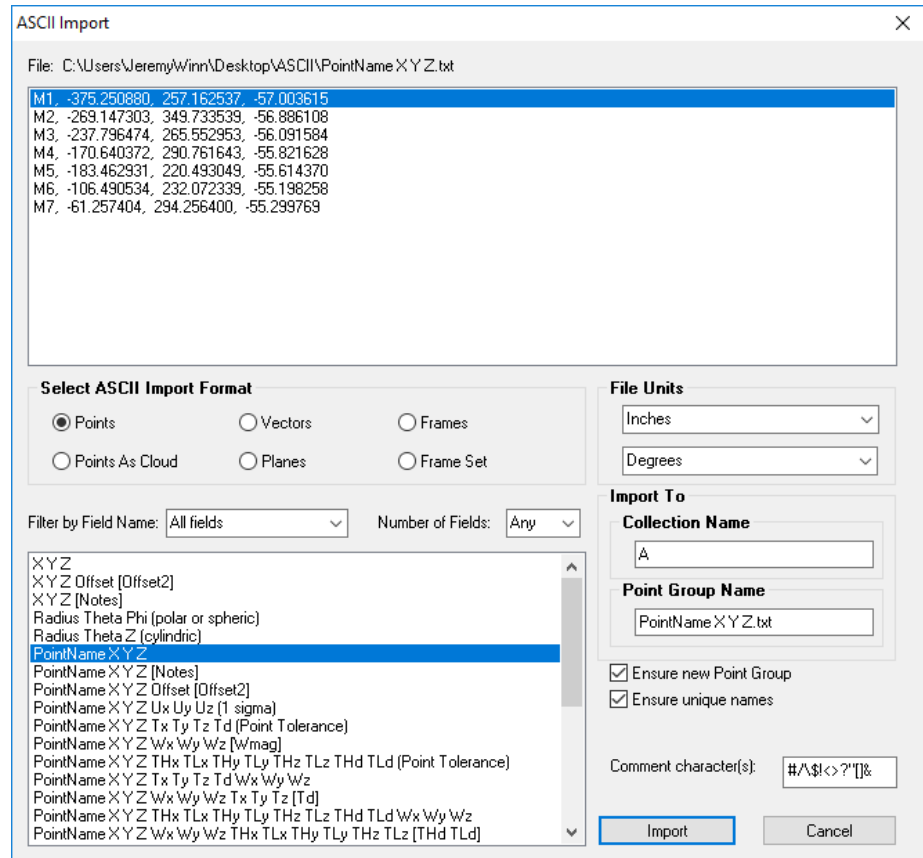


Figure 7-7. The ASCII Import dialog.

- **Select ASCII Import Format.** Choose the format of the ASCII file. SA will automatically delimit using commas, tabs, or spaces (depending on how the ASCII file is set up). The format types can be filtered in the selection view based upon type, name, and number of fields represented by the ASCII line.
- **File Units.** Define the units used in the ASCII file.
- **Collection Name.** Specifies the collection to which the data will be imported. This applies to all formats that do not contain a collection designation as part of the format.
- **Point Group Name.** Specify the name to import the points into (only applicable if not specified already by the format).
- **Ensure new group.** Select this option to ensure that points are placed into a new point group if the specified group already exists.
- **Ensure unique names.** Verifies that imported point names are unique. If there is a naming conflict, the offending point will have an asterisk appended.
- **Make Point Cloud.** Select this option to import points as a point cloud instead of a point group.
- **Comment character(s).** Any lines that start with the specified

characters are treated as comment lines and are ignored.

To Import an ASCII-Formatted File:

1. Select **File>Import>ASCII: Predefined Formats**.
2. Choose the file to import.
3. In the *ASCII Import* dialog, specify the desired options, then click **Import**.
4. The file will be imported, and you will receive a summary of lines imported and ignored.

Importing ASCII Formatted Points With Offsets

There are two additional ASCII formats for importing points with offsets, depending on the format of your source data.

To Import Points With Offsets:

- From the menu, select **File>Import>ASCII: Points with Offsets><Format>**.

<Name><X><Y><Z><Offset>

This option is used when the source file is in the format **<Name><X><Y><Z><Offset>**.

2 Files: <Name><X><Y><Z> + <Name><Offset>

This option is used when there are two source files: One in the format **<Name><X><Y><Z>** and the other in the format **<Name><Offset>**.

Importing XML Files

XML files containing nominal points and coordinates can be imported into an SA file. At a later time, measured point data can be merged back into the XML file, which will associate the actual coordinates with the nominal coordinates. The nominal points will be imported into a group matching the name of the XML file.

XML Format: Nominal Points

The format of the XML file should have the standard XML format, such as the introductory tag:

```
<?xml version="1.0" encoding="UTF-8" ?>
```

Following this, an enclosing group may contain one or more **<POINT>** elements, containing a **NAME** attribute and a child **<NOMINAL>** element which itself contains **X**, **Y**, and **Z** attributes. For example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<left>
  <POINT NAME="Center1">
    <NOMINAL X="1.000" Y="4.000" Z="7.000" />
```

```

</POINT>
<POINT NAME="Center2">
  <NOMINAL X="3.000" Y="6.000" Z="9.000" />
</POINT>
</left>

```

XML Format: Nominal and Measured Points

Once measured data has been obtained, the measured point names are matched up with the nominal names in the XML file, and an **<ACTUAL>** element is added as a child of each matching **<POINT>** element. This **<ACTUAL>** element contains **X**, **Y**, and **Z** attributes holding the measured coordinates. The above XML file might look like the following after merging measured data:

```

<?xml version="1.0" encoding="UTF-8" ?>
<left>
  <POINT NAME="Center1">
    <NOMINAL X="1.000" Y="4.000" Z="7.000" />
    <ACTUAL X="1.002" Y="3.996" Z="7.003" />
  </POINT>
  <POINT NAME="Center2">
    <NOMINAL X="3.000" Y="6.000" Z="9.000" />
    <ACTUAL X="3.007" Y="6.005" Z="8.998" />
  </POINT>
</left>

```

To Import an XML File:

1. From the menu, select **File>Import>XML**, then select the desired XML file.
2. In the XML import dialog, the structure of the points to import will be listed (Figure 7-8). Click the **Import Points** button.

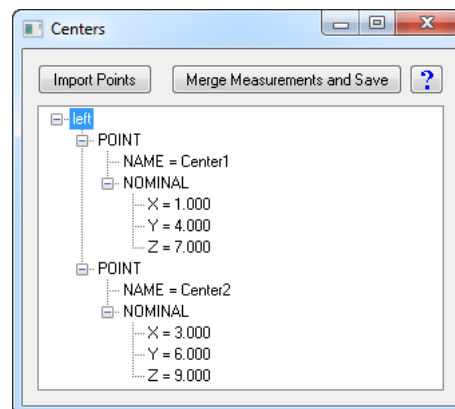


Figure 7-8. The XML import dialog for the above example file.

To Merge Measurements to an XML File:

1. From the menu, select **File>Import>XML**, then select the desired XML file.
2. In the XML import dialog (Figure 7-8), click the **Merge Measure-**

ments and **Save** button.

3. From the **Object Selection** dialog, select the point group you'd like to merge.

Cloud Files: Standard Formats

Point Cloud data including color and intensity information can be imported in the following standard formats:

- **E57**. Which is a ASTM standard format in a compressed form.
- **PTX**. Is an ASCII point cloud format used by several instrument manufacturers. This format can be used to import point cloud data, intensity and RGB information.

Using PTX for Point Cloud Import

PTX is a handy format because it provides a means to import cloud data with Intensity and RGB color for each cloud point. Any ascii file can be modified and saved as a *.ptx file for this purpose as long as you understand the format.

The header includes the transform for an instrument within a job file with respect to the working frame.

- number of columns (for unstructured data enter 1)
- number of rows (number of points to import)
- st1 st2 st3 ; scanner registered position
- sx1 sx2 sx3 ; scanner registered axis 'X'
- sy1 sy2 sy3 ; scanner registered axis 'Y'
- sz1 sz2 sz3 ; scanner registered axis 'Z'
- r11 r12 r13 0 ; transformation matrix
- r21 r22 r23 0 ; this is a simple rotation and translation 4x4 matrix
- r31 r32 r33 0 ; just apply to each point to get the transformed coordinate
- tr1 tr2 tr3 1 ; use double-precision variables

Therefore you header for un-transformed data would look like this:

```
0,0,0; 1,0,0; 0,1,0; 0,0,1; 1,0,0,0; 0,1,0,0; 0,0,1,0; 0,0,0,1
```

Each row below the header would then include the XYZ location of the cloud point, intensity in a range from 0 -1 and 3 color integer values (RGB in a range from 0 - 255). An example cloud point entry would look like this:

```
4.147541 12.075727 1.136227 0.486915 76 74 91
```

Custom Formats

Many custom file formats can be imported, including the following:

- **xyz, ijk Files (IJK).** Files in IJK format (containing both position and probe orientation data).
- **Digital Network Level Measurements.** Measurements from a digital level. This is an ASCII file expected to be of the format `<group>`, `<target>`, `<elevation>`.
- **VSTARS .xyz Files.** .xyz files from VSTARS.
- **VSTARS Cameras.** Camera files from VSTARS in the outstar.txt format.
- **DMIS Files.**
- **Polyworks Files.** .POL, .PIF, .PF, and .DPI formats are all supported.
- **IMETRIC Files.**
- **Leica GSI8/GSI16 Files.** Files from Leica total stations in GSI8 or GSI16 format.
- **Leica SDB File.** Imports a point cloud to the selected instrument. *Note - Any additional transformations performed within the Leica Job file cannot automatically be applied.
- **Steinbichler AC Files.** This imports data from a Steinbichler scanner (such as the T-Scan) originating in .AC format. Note that, depending on the scanning mode selected (Surface or Feature), T-Scan Collect can automatically post-process the data creating a mesh. When exported, the mesh or the point cloud will have the same format *.ac, and SA only supports *.ac format if it's a point cloud. "Surface" will create a mesh and "Features" a point cloud.
- **Wavefront Objects Files.** This imports data in Wavefront Technologies' Objects file (.obj).
- **Leica ADF format.** Files in Leica's .adf format.
- **1-D Data (DataMyte).**

To Import a Custom File Format:

- From the menu, select **File>Import>Custom Formats>...**

Embedding Files

You can import any file from the file system and embed it in the SA job file. The embedded file is placed into the active collection. This is convenient for transporting relevant files with the job file. Of course, when doing so the SA job file grows accordingly to store the file. At

any time, this file can be exported back to the file system.

To Embed a File:

1. Activate the collection into which you'd like to embed the file.
2. From the menu, select **File>Import>File as Embedded File**.
3. At the prompt, select the file to embed. The file will appear under the *Embedded Files* category of the tree.

Importing Images

You can import any common image format into the SA job. In the tree, embedded images appear under the *Pictures* category in the active collection. Importing images is particularly useful because these images can be used in reports. For example, you might have a few JPEG images of a measured part taken with a digital camera that you'd like to include in your report. Importing these images into the SA file gives you the ability to use them in a report.

To Import an Image Into an SA File:

1. Ensure the collection into which you'd like to import the image is active.
2. From the menu, select **File>Import>Image**.
3. At the prompt, select the image file to import. In the tree, the image will appear in the *Pictures* category of the active collection.

File Export

There are many ways to export your data from SA. Points, Point Clouds, Vector groups and many CAD formats are supported under the **File>Export** menu option.

Exporting ASCII Formatted Points

Points are easily exported from SA to an external ASCII text file. Using this method, the points are always exported in the working coordinate frame. There are many export options to give you fine control over the export format and data (Figure 7-9). These options are summarized below:

- **Target Name.** The name of each exported target can be one of four options: *Collection Group Target*, *Group Target*, *Target*, or *None*. These options indicate which parts of the point/target name are included. For example, exporting using *Collection Group Target* exports in the format **CollectionName Group-Name TargetName x y z**.

- **Coordinate System.** Point coordinates can be exported in Cartesian, Cylindric, or Spherical coordinates.
- **Tolerance.** If selected, point tolerances will be exported for each point (if available).
- **Coordinate uncertainties.** If selected, coordinate uncertainties for each point will be exported (if available).
- **Time Stamps.** If selected, the point's time stamp will be exported.
- **Target Offsets.** If selected, the planar and radial offset of each point will be exported.
- **Target Comments.** If selected, any comments stored with the point will be exported.
- **Measurement Details.** If selected, measurement details (such as RMS error, measurement mode, and sampling method) will be exported, if available.
- **Separator.** Data can be delimited by spaces or commas in the exported format.
- **Decimal Precision.** Numerical values can be exported using fixed decimal precision (you specify the number of digits) or using full precision scientific notation.
- **Include SA Version and Frame Comments.** If selected, the SA version and working coordinate frame name will be listed at the beginning of the file, preceded by `//` to denote that the line is a comment.
- **Include Axis Comments.** If selected, information about the exported coordinate system will be listed at the beginning of the file, preceded by `//` to denote that the line is a comment.

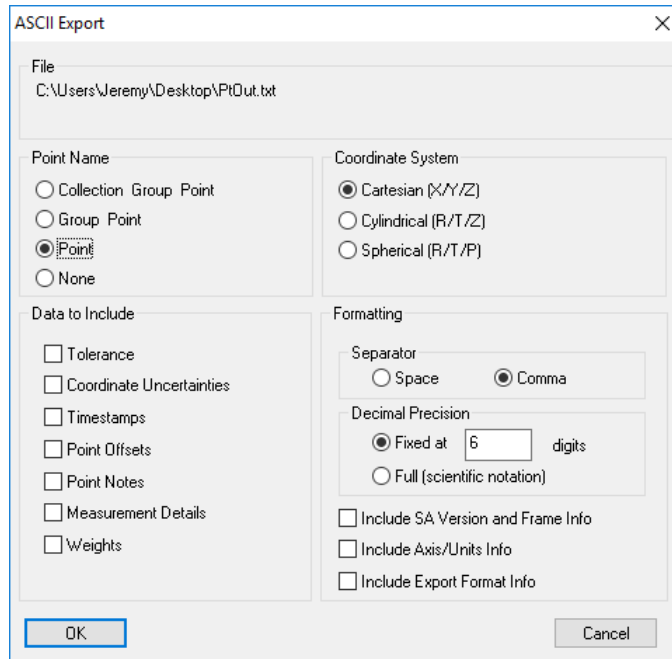


Figure 7-9. The options for exporting ASCII points.

To Export Selected Points:

1. From the menu, select **File>Export>Points (ASCII File)**.
2. When prompted, select the desired points to export, then press **Enter**.
3. In the **Save As** dialog, specify the name for the exported file, then click **Save**.
4. When the **ASCII Export** dialog appears, choose the desired options, then click **OK**. For convenience, the resulting exported file will be opened in Notepad.

To Export an Entire Point Group:

1. In the tree, right-click the desired point group to export, then select **Export to ASCII File** from the context menu.
2. In the **Save As** dialog, specify the name for the exported file, then click **Save**.
3. When the **ASCII Export** dialog appears, choose the desired options, then click **OK**. For convenience, the resulting exported file will be opened in Notepad.

Exporting Point Clouds

You can export point clouds to an external ASCII file. You can specify whether each point is assigned a label in the exported file.

To Export One or More Point Clouds to an ASCII File:

1. From the menu, select **File>Export>Point Clouds**.

2. Select one or more clouds to export.
3. In the *Save As* dialog, specify the filename to export, then click *Save*.
4. When prompted, indicate whether you would like each cloud point to have a label in the exported file.

To Export a Single Point Cloud to an ASCII File:

1. Right-click any point cloud in the tree and select **Export to ASCII File**.
2. In the *Save As* dialog, specify the filename to export, then click *Save*.
3. When prompted, indicate whether you would like each cloud point to have a label in the exported file.

Exporting Vector Groups

Vector groups can be exported to an ASCII file. The file is exported in the following format:

- **Name, X, Y, Z, dX, dY, dZ, Signed Magnitude**

To Export a Vector Group to an ASCII File:

1. From the menu, select **File>Export>Vector Group**.
2. At the prompt, select the vector group to export.
3. In the save dialog, specify the name for the ASCII file to export. For convenience, the exported file will be opened for you in Notepad.

Exporting Geometry

The fundamental characteristics of any geometry type (either constructed or as part of Geometry Relationship) can be export to a *.csv file for easy access in Excel. Each parameter is included in a separate column.

1. From the menu, select **File>Export>Geometry**.
2. Select the objects or geometry relationships you would like to export (objects included in a relationship are listed under the relationship category).
3. In the *Save As* dialog, specify the filename to export, then click *Save*.

Exporting Object Frames

The coordinate frames for any objects that have representable frames (such as coordinate frames, circles, and spheres) can be exported in

any of the following formats:

- **XYZ Fixed Angles.** FrameName, X, Y, Z, Rx, Ry, Rz, [Timestamp]
- **XYZ Euler Angles.** FrameName, X, Y, Z, Rx, Ry, Rz, [Timestamp]
- **ZYX Euler Angles.** FrameName, X, Y, Z, Rx, Ry, Rz, [Timestamp]
- **ZYZ Euler Angles.** FrameName, X, Y, Z, Rx, Ry, Rz, [Timestamp]
- **ZXZ Euler Angles.** FrameName, X, Y, Z, Rx, Ry, Rz, [Timestamp]
- **Frame Name, Transformation Matrix (4x4).** Frame-Name, r0c0, r0c1, r0c2, r0c3, r1c0...[Timestamp]
- **Transformation Matrix (4x4).** r0c0, r0c1, r0c2, r0c3, r1c0...[Timestamp]

The Fixed angle and Euler Angle options exports the name of the objects and their positions and orientations in space. The Transformation Matrix options export the 16 elements of their transformation matrix in addition to the other selections.

To Export Object Frames:

1. From the menu, select **File>Export>Frames (and objects' frames)**.
2. Select all of the objects defining the frames you'd like to export.
3. In the **Frame Export Settings** dialog, select the desired format for the exported file, then click **OK**.
4. In the **Save As** dialog, specify the name for the exported file. Notice that this file is automatically exported as a .CSV file. For convenience, the file is automatically opened after successful export.

Exporting to CAD Formats

SA jobs can be exported to several different formats, with support for points, geometry and surfaces. Only the entity types supported by the selected file format will be exported. The supported formats are:

- **STEP.** You can export the entire SA file or specific objects.
- **IGES.** You can export the entire SA file or specific objects.
- **VDA.** You can export the entire SA file or specific objects.
- **SAT.**

- **DXF.**
- **Metric Vision Instrument Compensation Data Format.**
- **Imageware Cloud File.**
- **Surface Wireframe Curve File(s) (.crv).** Writes a wireframe representation of the selected surfaces to a text file containing comma-separated values.
- **Polyworks POL File.**
- **STL.** Files can be exported in either ASCII or binary format.

To Export an SA File to a CAD Format:

1. From the menu, select **File>Export>[Format]**. If supported, specify whether the entire model or specific objects should be exported.
2. If applicable, select the objects you'd like to export.
3. When prompted, specify the name of the file to export.

Q-DAS SPC Data

Under *Vender Specific File Formats* is an option to export Hexagon's **Q-DAS (Qualitative Data Analysis Software) DFQ Files**. Hexagon's Q-DAS Process Control Solution provides multiple software products and services for quality assurance in industrial production HxGn Q-DAS Product Line (hexagonmi.com).

SA also offers MP export options for more comprehensive export and automation. For more information on these options refer to:

<https://kinematics.my.site.com/SA/s/article/Integrating-Q-DAS-export-with-SA-for-process-control>

Exporting Window Placements

When automating SA using Measurement Plans or the SDK, it is sometimes desirable to preserve the placement and size of SA windows and dialogs. Consider an example in which a number of client computers running at a specified screen resolution will be using a series of automation routines. In order to ensure that these measurement computers have all of the windows and dialogs in the exact same place as the "development computer", you can export window placements on the development computer, then have the automation routine import these placements on the client computers using the **Import SA Windows Placement** MP command. This will ensure that the client computers position and size the windows exactly the same as the development computers.

To Export Window Placements:


1. From the menu, select **File>Export>Window Placements**.
2. Specify a name for the exported file. The file will be exported as a .POS file.

Templates

When you create a new file or start up SpatialAnalyzer, it opens in a “factory default” configuration. However, you might wish to specify your own read-only configurations for one or more files that are used as starting points for your work. Or, you may wish to create a configuration that is used as the default starting point for your job when starting up SA or creating a new default file. For instance, you likely have a preferred set of default units, decimal precision settings, colors, and other settings you wish to use. Setting these manually each time you start a new job would be tedious.

To satisfy this need, SA employs *templates*. A template is not very different from a regular job file. In fact, it *is* a job file. The only difference is that it is marked as read-only and is typically stored in a standard folder on disk (C:\Analyzer Data\Templates), although this is not a requirement.

To Create a New Template File:

 **Note:** Template files are typically stored in C:\Analyzer Data\Templates, although this is not mandatory.


1. Configure a job file as desired.
2. From the menu, select **File>Save As Read-Only Template**.
3. In the dialog, specify the name and location for the template file. It will be set as read-only.

To Open an Existing Template File:

1. From the menu, select **File>Open Template File**.
2. From the dialog, choose the file to open.

The Default Template

Instead of having templates as starting points for a specific job, you may also have a preferred default starting configuration for all files. SpatialAnalyzer provides the capability to define a *default template*. A default template is a template that is automatically loaded in the following situations:

- SpatialAnalyzer is started from the Windows start menu or a desktop shortcut.
- **File>New** is selected from the menu, or the New File icon  is selected in the toolbar.

Default templates help you avoid repeating the same job file preparation steps by allowing you to define a starting point for all of your job files. They also provide a convenient method for standardizing file setup and settings across an organization. If a default template does not exist, SA will use factory default settings.

To Create a Default Template

1. Configure an SA file with the settings for your template. Remember to set all settings that are saved with an SA file (such as the active units and decimal precision settings) as desired.
2. From the menu, choose **File>Save As Read-Only Template**.
3. In the save dialog, ensure you are in the Analyzer Data\Templates folder (SA will take you there by default), then save the template as `default.xit`. When you start SA or a new file, this file will automatically be restored.

To Remove a Default Template

1. Navigate to SA's Analyzer Data\Templates folder in Windows Explorer.
2. Delete the `default.xit` file. Without this file, the factory default settings will be used.

Screen & Graphics Capture

At any time, you can capture either the entire screen or the graphical view for printing or output to a standard image file (BMP, JPG, PNG, GIF, or TIFF).

To Print or Capture the Screen or Graphical View:

1. From the menu, select **File>Capture Screen[Capture Graphics]>To Printer[To File (BMP/JPG/PNG/GIF/TIFF)]** as appropriate.
2. If capturing to printer, choose the printing options and confirm. If capturing to file, select the filename and type, then click **Save**.
3. The **Capture Graphics > To File (BMP/JPG/PNG/GIF/TIFF)** command allows you to specify a render scale. A value of **1.0** uses the actual pixel count of the graphical view for the resulting image. A value of **3.0** will create an image three times the resolution of the graphical view.

This Page Intentionally Left Blank.

Points are the most fundamental aspect of any measurement. They are the basis from which virtually all analysis is performed in portable metrology—therefore having a good understanding of how they work in SpatialAnalyzer is crucial.

About Points

A point is defined as a specific location in three-dimensional space, and may be represented in Cartesian (X/Y/Z), Spherical ($r/\theta/\Phi$), or Cylindrical ($r/\theta/Z$) coordinates (depending on the application). With a few exceptions coordinates are reported to the user in the working coordinate frame.

Points can be created by many methods. For example, they can be created from a point construction command or can be the result of measurement with a portable metrology device. They may be imported from a CAD file, extracted from sphere scans, or the result of a fit to existing data.

Point Groups

In SA, all points must exist inside of a *point group*. A point group is a container for a set of points that are related in some way. The point group itself is considered an object, whereas its contained points are not objects. All points inside of a point group are treated as a rigid body. That is, when you move or rotate a point group, all of the enclosed point groups move or rotate together, as if rigidly connected.

The point group is the organizational element that is used to organize your point data by feature. All measurements of a specific hole should typically be placed into their own point group. All measurements of reference (tie-in) points should typically be placed into their own group. Likewise, measurements of a complex surface should typically be placed into their own group as well. As mentioned earlier in this

manual, organizing your point data in this manner will make analysis and reporting easier later on.

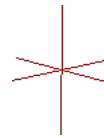
Since point groups are objects and points themselves are not, only point groups have a color. All points inside that group inherit the color of their point group.

There are two types of points that you will see in SA: *Constructed Points* and *Measured Points* (also referred to as *Targets*). Both of these types of points may appear in the same point group.

Constructed Points

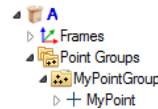
A *constructed point* is a point that is not associated with any instrument observations or measurements. There are many ways to create constructed points. They may have been created manually by entering coordinates, by intersecting a line with a plane, or may be the result of importing data from an ASCII file. A constructed point may also result from deleting a target's individual measurement observations.

Figure 8-1. A constructed point as seen in the graphical view.



A constructed point is represented in the graphical view and the SA tree by a 3-D "plus" symbol, as pictured in [Figure 8-1](#) and [Figure 8-2](#).

Figure 8-2. A constructed point in the tree named **MyPoint** sits inside the **MyPointGroup** group.



Expanding the triangle next to any point will display its coordinates in the working frame.

Measured Points

A *measured point* (also referred to as a *target*) is a point which has been determined from one or more direct instrument observations. For traceability purposes, targets they maintain a record about who and what measured them, when and how they were measured, weather data, sampling statistics, raw measurement data from the instrument, and any other pertinent metadata. Because they are tied to instrument measurements, measured points cannot be created through typical point construction or analysis methods--they must be measured. (There is, however, a way to create targets through measurement simulation).

Measured points are represented in the graphical view by a 3-D plus symbol with a diamond around it ([Figure 8-3](#)).

Figure 8-3. A measured point as it appears in the graphical view.




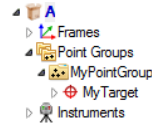
In the tree, targets are represented by a crosshair  icon (Figure 8-4).

Figure 8-4. A measured target named **MyTarget** as it appears in the tree.

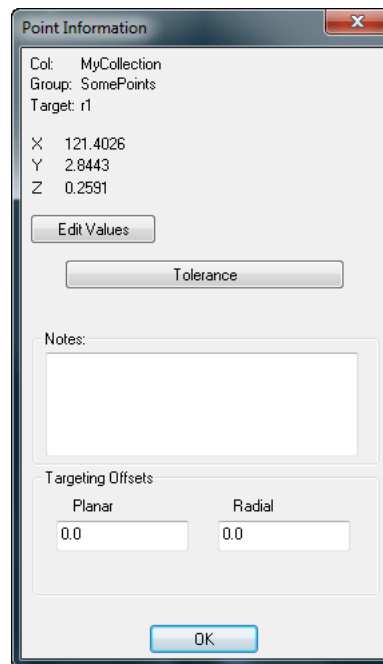


Expanding the triangle next to any point will display its coordinates in the working frame.

Point Characteristics

There are a number of characteristics common to both constructed and measured points. The *Point (Target) Information* dialog, shown in Figure 8-5, contains the properties of an individual point. In this particular figure, a constructed point's properties are shown.

Figure 8-5. The *Point Information* dialog for a constructed point.



To Display the Properties for a Point:

- Double-click a point in the graphical view or tree, or
- Right-click a point in the graphical view or tree and select **Properties**.

At the top of the *Point Information* dialog, the collection, group, and target name are listed. Below that are the coordinates of the point in the working coordinate frame.

To Edit the Coordinates of a Point:

1. Click the **Edit Values** button. The **Edit Point Coordinates** dialog will appear as shown in [Figure 8-6](#).

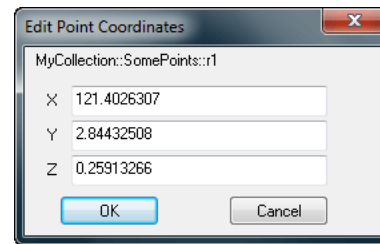


Figure 8-6. The *Edit Point Coordinates* dialog.

2. Type in the new values and click **OK**.

If you attempt to edit the coordinates of a measured point (something you typically should not do), you will receive a warning and the measured point will turn into a constructed point. The original measurement data is still preserved with the point, but the measurements become deactivated and they need to be reactivated (and the target recomputed) to restore the measured point.

Point Tolerance

Each point can be assigned a tolerance ([Figure 8-7](#)). Tolerances that have been assigned to a point will be used for watch windows and other comparisons where applicable. A point tolerance must be associated with a specific coordinate frame (because it may be asymmetrical). Tolerances can be applied on any combination of axes, and in any combination of sides (high or low).

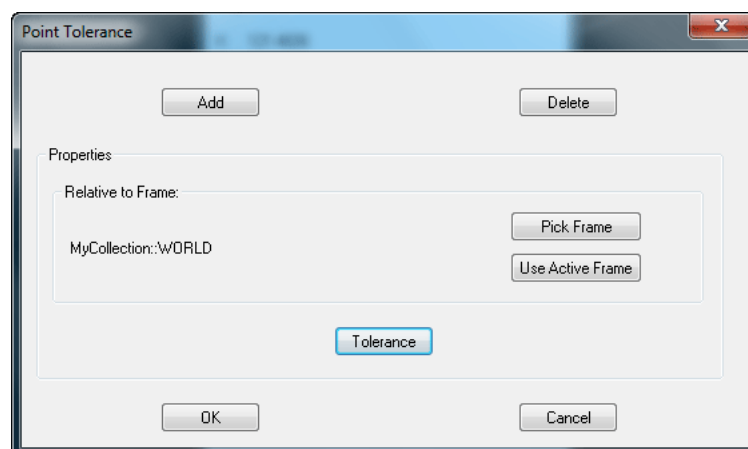


Figure 8-7. The *Point Tolerance* dialog allows you to specify the tolerance for a point.

To Add a Tolerance to a Point:

1. In the **Point (Target) Information** dialog, click the **Tolerance** button, or right-click the point in the tree or view and choose **Tolerance**.
2. In the resulting **Point Tolerance** dialog, click the **Add** button, as shown in [Figure 8-7](#).

3. To pick the applicable coordinate frame from a list, click the **Pick Frame** button and choose the frame from the resulting dialog. To specify that the tolerance will apply to the current working frame, click the **Use Active Frame** button.
4. To assign the numeric tolerance value, click the **Tolerance** button, which will open the **Vector Tolerance** dialog (Figure 8-8).

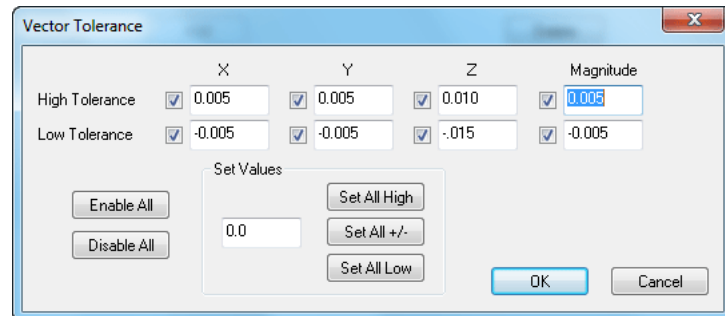


Figure 8-8. The Vector Tolerance dialog.

5. Check the tolerances that you want to apply, and enter the numeric values. The **Enable All** and **Disable All** buttons are shortcuts to check and uncheck all tolerances. If a value is entered into the **Set Values** field, then clicking the **Set All High**, **Set All +/-**, or **Set All Low** buttons apply the value to all of the high tolerances, all of the high and low tolerances, or all of the low tolerances, respectively.
6. Click **OK** to apply the tolerance.

Notes

The **Notes** field in the **Point (Target) Information** dialog can be used to enter descriptive notes about an individual point.

Targeting Offsets

Each point can be assigned a planar and radial target offset. Enter the values in these fields to change them for a single point. These offsets will be automatically compensated for when comparing the point to surfaces and other geometry. By default, constructed points have offset values of zero.

Planar Offset

The planar offset is the most commonly-used offset. It is used in almost all situations. To provide examples, the planar offset is used when compensating in the planar direction for a pin nest (Figure 8-9), when comparing to a cylinder or best-fitting a cylinder to measured points, or when comparing a set of measured points to a surface.

Radial Offset

The radial offset is only used in one situation: when using tooling in-

tended to measure circular and elliptical holes that has a radial offset associated with it—that is, when using a pin nest. In [Figure 8-9](#), the radius of the pin nest’s shank determines how far outward or inward the circle radius or ellipse axis must be offset.

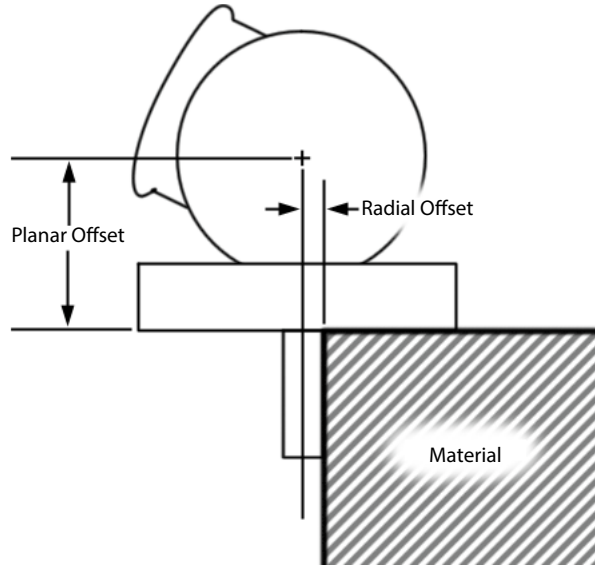


Figure 8-9. Measuring a surface hole with a laser tracker’s SMR and Pin Nest.

Spherical probes (such as SMRs and balls) have the same planar and radial offset, although technically the radial offset is not used.

The following are some sample measurement instrument/tooling combinations, and the proper planar and radial offsets associated with each:

Measurement Tooling	Planar Offset	Radial Offset
PCMM arm with 6 mm probe	3 mm (0.1181")	3 mm (0.1181")
PCMM arm with 3 mm probe	1.5 mm (0.0591")	1.5 mm (0.0591")
PCMM arm with Point Probe	0	0
Tracker with 1.5" (38.1 mm) SMR	0.75" (19.05 mm)	0.75" (19.05 mm)
Tracker with 1.5" SMR and Flat Nest ¹	1.0" (25.4 mm)	N/A (0)
Tracker with 1.5" SMR and Pin Nest	1.0" (25.4 mm)	0.125" (3.175 mm)
Tracker with 1.5" SMR and Edge Nest	1.0" (25.4 mm)	0
Laser Radar (non-contact measurement)	0	0

Accessing A Point’s Group Menu From the Graphical View

Sometimes, you want to select a point and access the properties of its enclosing point group. For instance, to change the color of a point you would need to access its parent point group. The graphical view’s context menu allows you to do just that.

¹ When a flat nest is used, a feature such as a hole cannot be measured because there is no tooling feature to properly locate the SMR radially. As a result, there is no valid radial offset for this tooling combination.

To Access a Point's Enclosing Group Menu:

- In the graphical view, right-click a point and select **Group Menu** from the context menu. The enclosing group's context menu will be displayed.

Measurements

Measured points carry a number of additional properties that are important to understand.

How SA Computes a Measured Point's Location

Determining the location of a measured point at first seems trivial. However, it gets significantly more complex when you realize that some instruments (such as theodolites) are angle-only measurement devices, and do not measure a target's position in space. With theodolites, at least two measurements of a target need to be recorded from different points in space to triangulate the location of the point.

Another complication is that one or more instruments may measure a single target several times. We will discuss these situations, but let's first look at how SA computes a measured point's location in a simple case, when an instrument provides a point position in space.

When measurements are recorded, SA always records the raw measurement data that came from the instrument. For theodolites, this means a horizontal angle (azimuth) and vertical angle (elevation). For other line-of-sight devices, it means azimuth, elevation, and distance, and for non-spherical measurement devices (such as portable arms) the coordinates are provided from the instrument in Cartesian space.

The instrument always provides the location of the measurement relative to its own internal reference system—typically at the optical center or at the base of an instrument. Since each instrument has a three-dimensional position and orientation in the job file, SA takes these raw measurements and calculates their coordinates in job space based on the position of the instrument. When an instrument moves, the position of all of its measurements are recalculated.

At any time, you can move a point group containing measured points. When you do so, you will break the connection between the measured points and their instrument, and the measured points will instantly turn into constructed points to show that you are no longer looking at actual measurements. You haven't lost the original measurement data, however. You can recalculate these measured points and turn them back into measurements.

To Recalculate Measured Points:

- From the menu, select **Analysis>Re-Compute Targets From Shots**. This will snap all measurements back into place relative to the instrument and convert them back into measured points.

About Target Observations

An observation is a single reported measurement from an instrument. A target, by definition, must have one or more instrument observations. While the coordinates for a constructed point are easy to understand and unambiguous, the coordinates for a target take more explanation. It is fairly common, particularly with theodolite measurements, for a given target to have multiple observations that together define the coordinate of the point. Consider, for instance, the case where three theodolites have measured a common point (see [Figure 8-10](#) on page 201). Since each measurement has finite error (due to encoder error, operator differences, alignment, and other factors), the observations will not coincide in space. Put another way, each instrument will “see” a slightly different location when measuring the same point in space. (The error in [Figure 8-10](#) is greatly exaggerated for clarity).

It would be easy to simply take an average of each observation to determine the computed target location, but that is not an optimal technique, because it assumes that each observation has equal error. Instead, SA uses a more accurate technique of computing the coordinate for the target based on an optimization. In [Figure 8-10](#), the computed target location is closer to the leftmost instrument’s observation, due primarily to the fact that it is closest to the measured point and therefore should have the least measurement error.

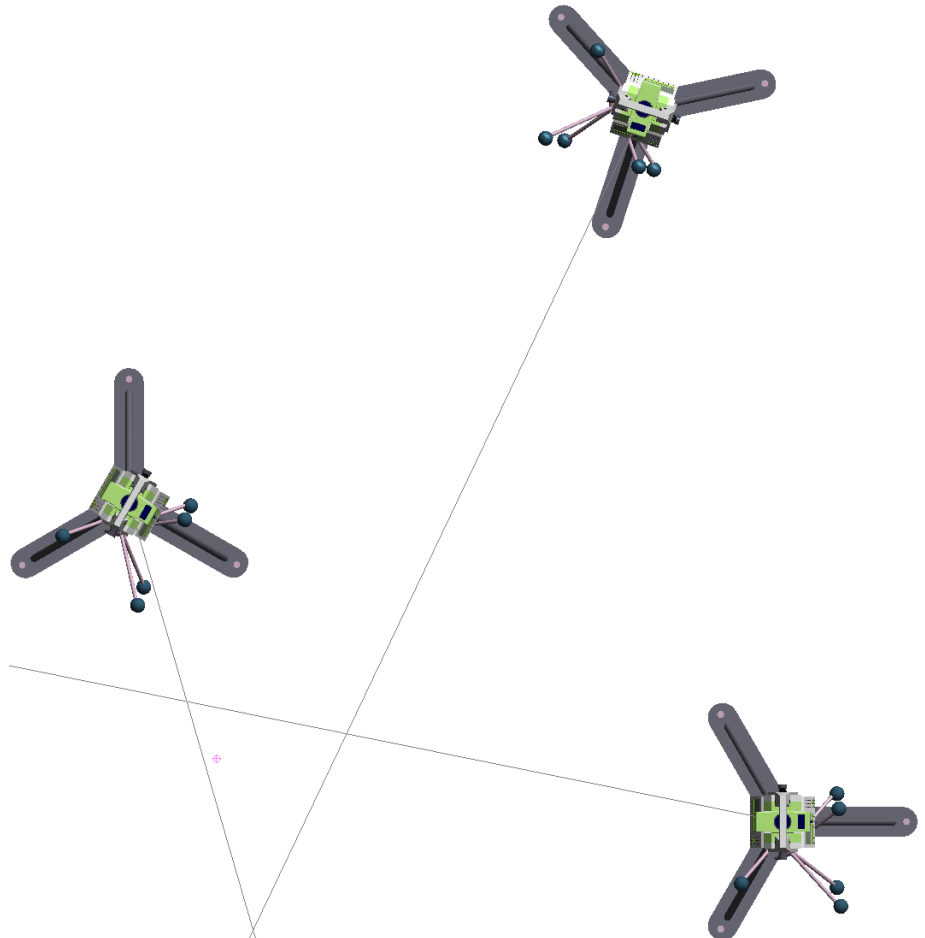
A view of this situation from above (see [Figure 8-11](#) on page 201) makes it even clearer: The closest instrument is most accurate, and gets the most influence in calculating the target location. The net result of this optimization technique is that the computed target location’s coordinates are more accurate than they would be if the average of the observations were used instead.

In the case of a single instrument measuring a point multiple times, the end result is that the target coordinate is almost exactly an average of the individual observations, as the distance to the target is approximately constant for all observations.

Figure 8-10. A measured target as computed by observations from three instruments (exaggerated for clarity).



Figure 8-11. The instrument closest to the target has the greatest accuracy and therefore gets the largest influence in target computation.



Target Computation

Different instrument types require different default approaches to shot computation. Within the User Options on the Analysis Tab is a selection for how measured points are computed based on the contributing observations.

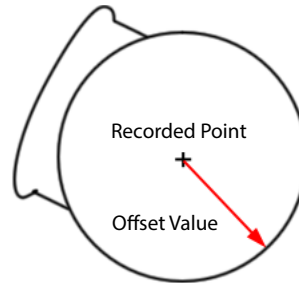
- **User only most recent shot.** This method considers only the most recent observation recorded from each instrument in computing the coordinates of a point. Prior observations will be disabled as new observations are added.
- **Use most recent shot from each face.** This option considers both the most recent shot from each face (front/back) as well as shots from each contributing instrument.
- **Don't change prior measurements at all.** This option will append additional observations to the existing list without altering the prior shot status.
- **Force a new point for each measurement (add * if necessary).** This option ensures that any additional observations will be recorded in separate points. If you measure "P6" for example and try to record the same point a second time it will be recorded in the tree as a separate point in the same group named "P6*".
- **Deactivate All Prior Shots.** This option ensures that only the most recent shot is used to compute the point. All shots from all other instruments will be disabled.
- **Remove All Prior Shots.** This option will delete any prior observations on the current measured point.

About Target Offsets

Regardless of the type of instrument used, when a measurement is performed in SA, the point that is recorded is the center of the probe (or the surface intersection point for non-contact measurements). For example:

- Laser tracker measurements record the center of the SMR.
- PCMM arm measurements record the center of the probe tip.
- Laser radar measurements record the returned point of contact with the object's surface.

Figure 8-12. The center of the probe is the recorded point, and the offset is its radius.



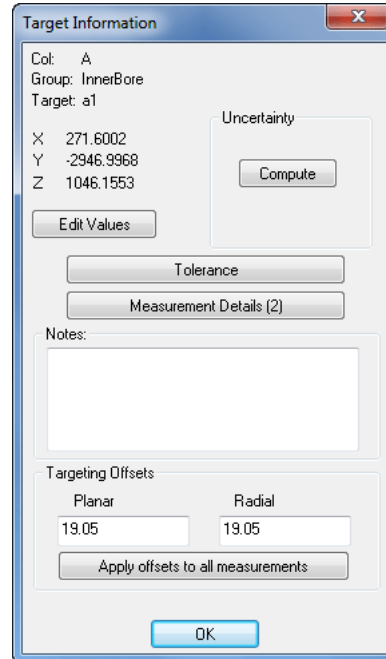
In order to properly compare measured points to geometry or construct geometry from measured points, SA must always know the planar and radial offset for the measurement tooling that is currently in use. In normal use, an instrument interface has an offset assigned based on the current probe and tooling being used, and this offset is automatically applied to points as they are measured. For example, suppose a pin nest and SMR are being used to measure the inside diameter of a hole with a laser tracker. If SA is set up to construct a circle as a result of the measurement, then once the circle is fit to the measured probe centers, it will be offset normal to the circle by the planar offset amount, and its radius will be increased by the radial offset value. These offsets will align the constructed circle with the true location of the edge of the hole, regardless of the measurement tooling or probe used for the measurements.

In addition to the target's offsets, for traceability purposes each individual measurement observation has a radial and planar offset as well.

Characteristics Unique to Measured Points

There are a few characteristics that are unique to measured points that you will not see on constructed points (Figure 8-13).

Figure 8-13. The *Target Information* dialog looks like the *Point Information* dialog for constructed points, but has a few subtle (but important) differences.



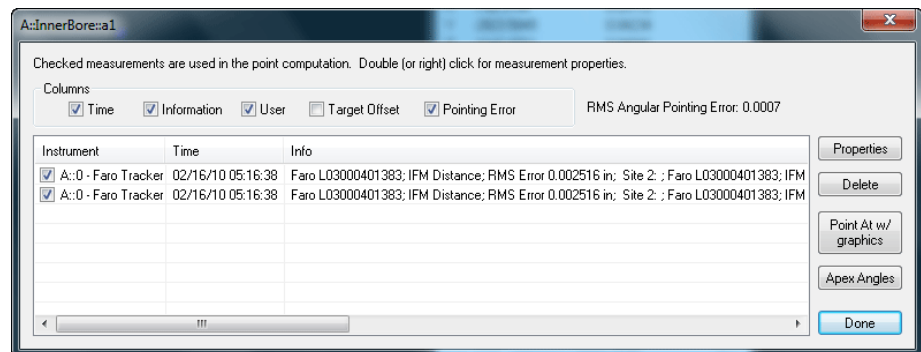
Uncertainty

There is an *Uncertainty* box with a *Compute* button inside. This is used for computing the uncertainty of the measured point and is discussed in more detail in the chapter on Measurement Uncertainty.

Measurement Details

To view the detailed measurement information that comprises a target, click on the *Measurement Details* button. (Note that the button will display the number of active observations that are being used to calculate the target position).

Figure 8-14. Two observations comprise this measured target, and both are active.



Warning: Deactivated observations are not used when calculating the target coordinate.

In this example (Figure 8-14), there are two active instrument observations that are being used to calculate the target coordinates. The checkboxes at the top of this dialog control which items are visible in the table.

- **Time.** Toggles whether or not the timestamp (date and time) for the measurement is displayed.
- **Information.** Toggles whether measurement details (such as

instrument serial number, mode, weather information, etc.) are displayed.

- **User.** Toggles whether the username is displayed.
- **Target Offset.** Toggles whether the stored offsets for each observation are displayed.
- **Pointing Error.** Displays the instrument's angular and linear *pointing errors*. (Pointing error is the angular error between the observation and the computed target location).

Next to these checkboxes, the dialog displays the RMS value of the combined angular pointing errors.

To Activate or Deactivate an Observation:

- Check or uncheck the observation in the *Instrument* column.

To Activate or Deactivate All Observations on a Target:

- Right-click the point in the tree or graphical view and select **Activate All Measurements** or **Disable All Measurements** as applicable.

To Activate All Measurements to All Targets:

- From the menu, select **Analysis>Activate All Measurements**.

To Permanently Delete an Observation:

- Select the row in the table and click the **Delete** button.

An *Apex Angle* is the angle between two measurement rays, or lines of sight, from measurement devices (see [Figure 8-15](#)). Ideal measurements with the least amount of error and uncertainty are obtained with apex angles of 90 degrees. As the apex angles increase or decrease from this value, the measurements have more uncertainty and are less reliable. SA can calculate and display the worst apex angle from a series of observations comprising a target.

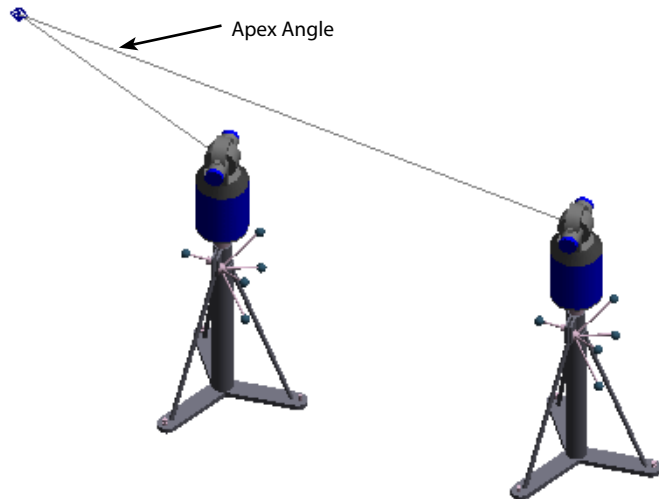


Figure 8-15. The apex angle between two lines of sight to a common target.

To Determine the Worst Apex Angle Comprising a Target:

- Click the **Apex Angles** button. SA will determine the two observations that form the worst apex angle, list the angle, and also list the two observations that form this angle.

To Point an Instrument’s Graphical Model at its Observation:

- Select the observation of interest and click the **Point At w/ Graphics** button. The instrument’s head will point at the measurement in the graphical view.

Observation Details

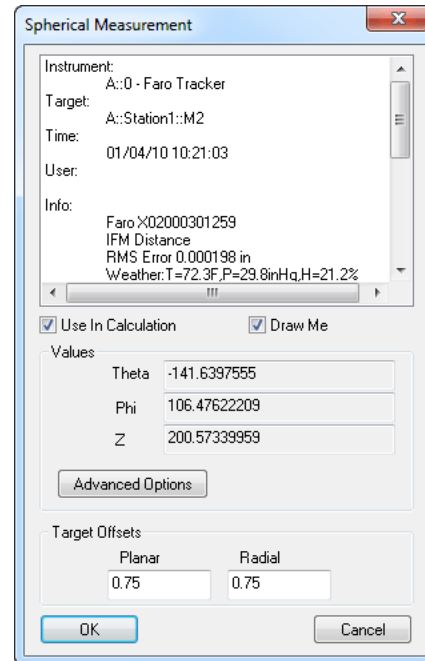
You can view the details for an individual instrument observation.

To View the Details for a Single Observation:

- Double-click or right-click the observation, or
- Select the desired row and click the **Properties** button.

The details for an observation are displayed in the *Polar Measurement* dialog (see [Figure 8-16](#) on page 207).

Figure 8-16. A single observation's details. This shows a laser tracker's observation information.



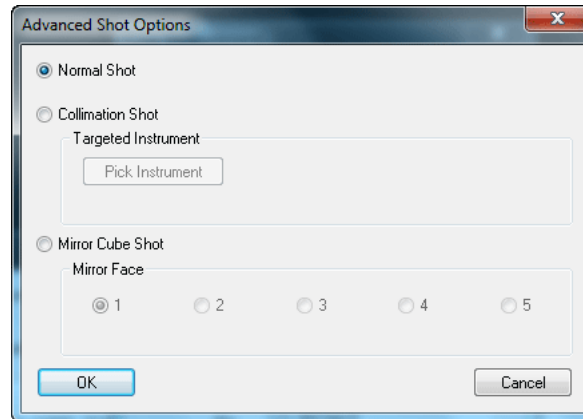
The top section displays all of the metadata associated with the observation.

- **Use In Calculation.** Toggling this option is equivalent to activating/deactivating an observation in the *Measurement Details* dialog.
- **Draw Me.** Turn this option off to prevent SA from drawing the instrument shot (ray) for the observation.
- **Theta/Phi/Z (X/Y/Z).** Read-only values showing the raw data from the instrument, expressed in the instrument's coordinate frame.
- **Advanced Options (theodolites only).** Mark observations as normal, collimation shots, or mirror cube shots (see below).

Each observation can store a planar and radial offset. These offsets are normally recorded when the measurement is taken. However, you can change them by filling in the *Planar* and *Radial* offset values to assign custom offsets to a specific observation.

The *Advanced Options* button allows you to specify whether the observation is a *Normal Shot*, a *Collimation Shot* (you must specify the instrument you're collimating to), or a *Mirror Cube Shot* (you must specify which mirror face was measured). These settings are used to help constrain bundle adjustments (see [Figure 8-17](#) on page 208).

Figure 8-17. Advanced shot options for a spherical measurement.



Targeting Offsets

Individual observations can have their own stored planar and radial offsets. Constructed points do not have to deal with this, as they do not have any observations.

A button labeled **Apply offsets to all measurements** (see Figure 8-13 on page 204) is visible below the planar and radial offset fields for measured points (you will not see it for constructed points). Clicking this button will apply the specified planar and radial offsets to all individual observations for the target.

Hidden Point Measurements

It is often necessary to be able to take measurements and compute point location in places that cannot be directly reached or measured. Hidden-Point Bar definitions provide a means to do that. Hidden points can be computed in two ways:

- Using a *Hidden Pt. To SA* measurement profile with a laser tracker (refer to the Instruments Manual).
- Using the **Hidden-Point Bar Database** saved within the SA Users Options. These options are described in the following sections.

The Hidden-Point Bar Database

Hidden-Point Bar definitions are saved with the **Users Options**. Bars can be defined in either of three ways:

- **Gravity Frame.** A gravity frame is a single point offset point. Added to support survey applications it can be used to compute the location of a point shifted a predefined distance along the Z axis of a reference frame.
- **2 Point Bars.** Define the distance of a third point (C) based upon the vector between the two measured or selected points and the entered distance from one of those points (A or B).

- **N-Point Bars or Fixtures.** With 3 or more points defined, a 3D position can be computed anywhere in space relative to the measured or selected points. N-Point Fixtures are defined by selecting a set of points and also selecting a measurement location to return when those points are measured.

Automatic Hidden Point Computation

The most effective way to use hidden point bars or fixtures is to use a naming convention to designate points as part of a defined hidden point bar or fixture. To do so follow this naming convention:

Point Name \$ (Bar Index) _ (Point Index)

- **Point Name.** The point name for the computed point should also be used for all the measurements of the reference points used to built it.
- **Bar Index.** The Hidden Point Bars defined in the user options are not named but instead use an integer index. By specifying "MyPoint\$0_0" you are specifying a point named "MyPoint" and telling SA to use the "0" index bar.
- **Point Index.** Its important to specify a measured point as A or B for a 2 point bar to be computed correctly. Also the specific index of the points within an N Point Bar need to be defined. The final piece of the naming convention is the point index. So "MyPoint\$0_1" tells SA to use the 0 index bar and that this point represents the "1" index point.

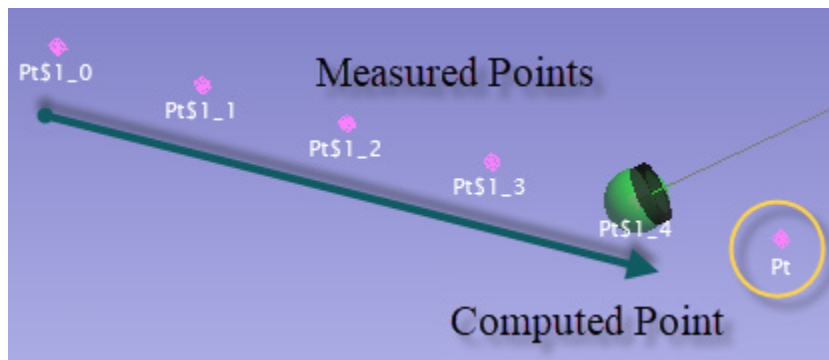


Figure 8-18. Hidden Point Naming Convention in use.

If this naming convention is followed hidden point locations can be updated/recomputed at any point by using the menu command **Analysis>Re-Compute Hidden Points**. This can be quite helpful if hidden points are computed from multiple instrument plants and their relative positions are changed do to an alignment.

Gravity Frame

A **Gravity Frame** hidden-point definition was added to facilitate survey applications and pin nest measurements. A single point can be used to compute a point shifted along an axis with respect to a specified

frame. This allows a single point to be all that is needed for this Hidden-Point Bar measurement.

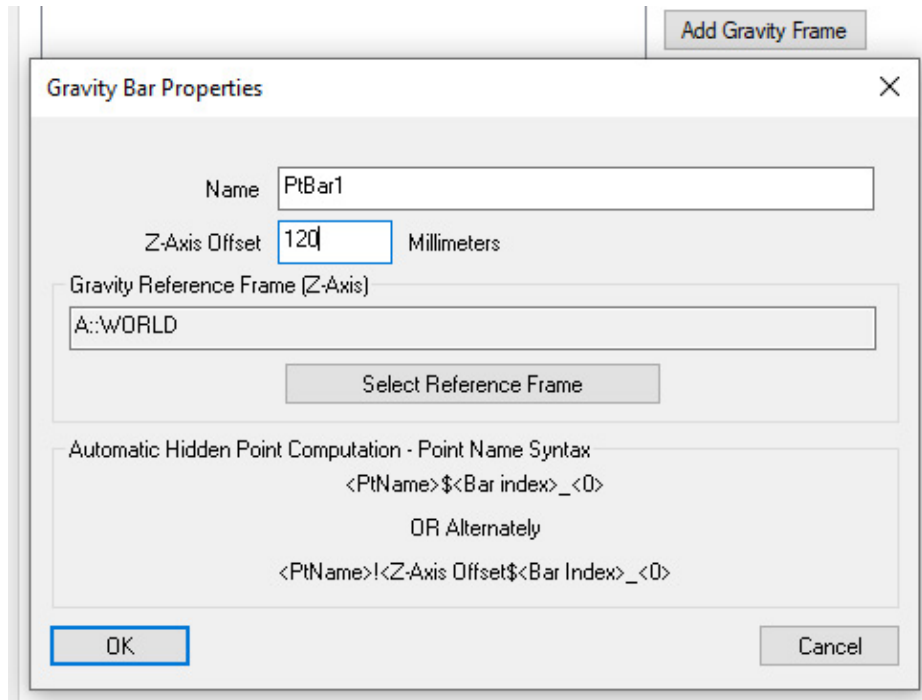


Figure 8-19. Gravity Bar Properties Dialog.

Gravity Frame measurements are compensated along the Z-axis of the specified Reference Frame by the specified Offset or by the offset value entered in the measurement string.

Measurement examples are as follows:

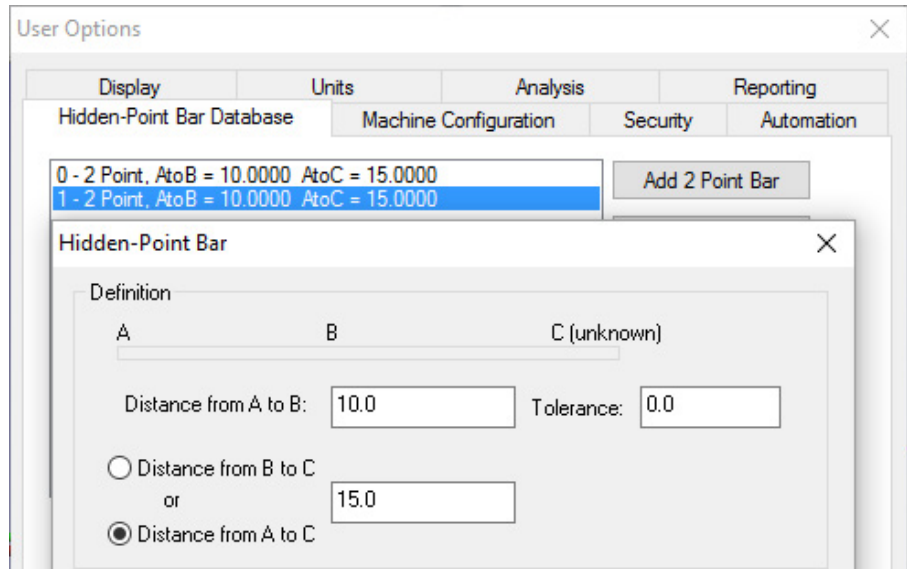
- If you measure a point "<PtName>\$<Bar index>" the hidden point will be computed using the default bar offset
- If you measure a point "<PtName>!<Z-Axis Offset>\$<Bar index>" the hidden point will be compensated by the entered value "!<Z-Axis Offset>".

Note that you can also use the Gravity Bar Name directly instead of the Bar index.

2 Point Hidden-Point Bar

With a 2 point bar (Figure 8-20), an **A** and **B** point are used to define the vector of the hidden point bar, and the hidden point (**C**) is constructed along this vector at a prescribed distances from either the **A** or **B** points.

Figure 8-20. 2 Point Hidden Point Bar Definition



To Construct a Hidden Point:

1. From the menu, select **Construct>Points>Hidden Point Bar**.
2. When prompted, select the **A** point.
3. Select the **B** point.
4. The *Pick Hidden Point Bar* dialog will appear. Select the appropriate hidden point bar, then click **OK**.
5. The *Construct New Point* dialog will appear (Figure 8-29). Specify the point name, leaving the coordinates alone—then click **OK**.

To Measure a Hidden Point:

1. In the instrument name control type the <point name> \$<Bar Index_0 for the point A (eg. "Point1\$1_0"), and measure a point
2. Check that the name incremented correctly: <point name> \$<Bar Index_1 for point B (eg. "Point1\$1_1"), and measure a point.

That's it...Measure 2 points and it will compute the hidden point for you (eg. "Point1")

N Point Bars or Fixtures

Hidden points can also be computed from an array of point locations. Having 3 or more input points makes it possible to compute a hidden point location anywhere in 3D space with respect to the reference points (Figure 8-21).

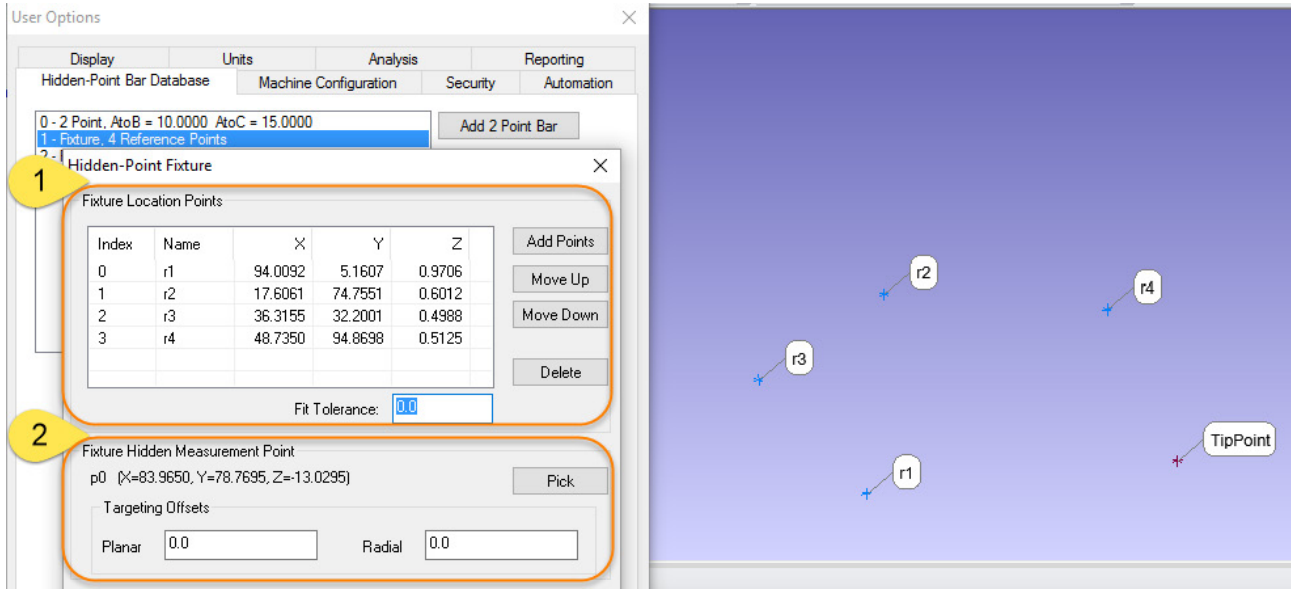


Figure 8-21. N Point Hidden Point Fixture

To build a Hidden- Point Fixture do the following:

1. First build a nominal bar. Use **Construct>Points>Enter** and build the measurable points defining the fixture in their exact nominal XYZ locations relative to the working frame, and also build the nominal design target point relative to these fixture points.
2. Open the Use Options and press the **Add N Point Frame** button to build a new Fixture definition.
3. In the Hidden-Point Fixture dialog press the **Add Points** button and select the constructed (measurable) points.
4. In the *Fixture Hidden Measurement Point* section use the **Pick** button and choose the constructed Target point. In this case where a point probe is to be used, simply press **OK** and be done. However, if you are using fixture with a spherical probe tip be sure to also add the correct offsets so that they are automatically applied to the computed point.

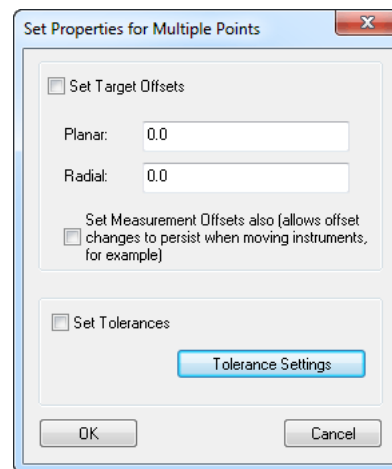
To use new fixture definition, measured the fixture points using the hidden point name convention described in the next section. When measuring an N Point Hidden Point Bar the points are fit to the nominal definition. Therefore, as soon as 3 points are measured a computed point is built. The accuracy of the computed point increases with each additional measurement as error is distributed equally through a best fit process.

Setting Properties for Multiple Points

Some situations arise in which you'd like to change the properties of multiple points at once. For instance, if you accidentally measure a large number of points with the wrong target offsets set in the instrument interface, you will need to correct these offsets in order for SA to properly apply probe compensation.

You can apply new tolerances or offsets to multiple points using the *Set Properties for Multiple Points* dialog (Figure 8-22).

Figure 8-22. The *Set Properties for Multiple Points* dialog.



To Change Offsets or Tolerances For Multiple Points:

- From the menu, choose **Edit>Set Properties for multiple points** and select the points to modify. This will call up the *Set Properties for Multiple Points* dialog.

To Change Offsets or Tolerances For An Entire Group:

- Right-click a point group and select **Set Point Properties** from the context menu. This will call up the *Set Properties for Multiple Points* dialog.

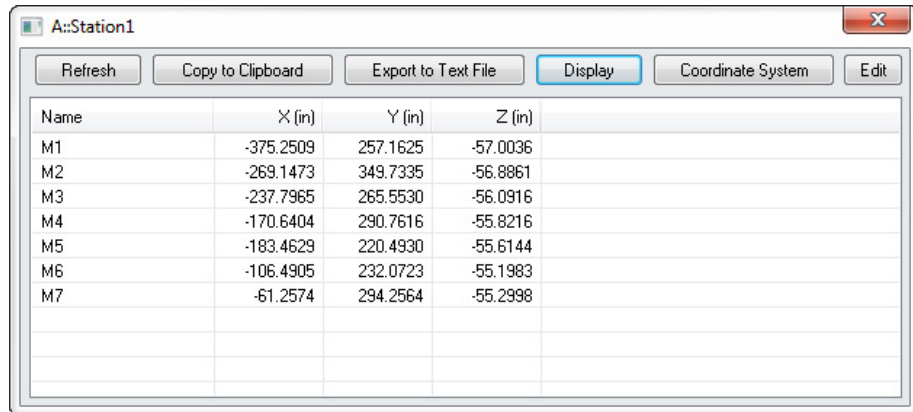
This dialog has two regions. The top region is for changing the stored target offsets, and the bottom region is for changing the point tolerances.

- Set Target Offsets.** Check this option to change the planar and radial offsets of the measured points to the values entered into the *Planar* and *Radial* fields.
- Set Measurement Offsets Also.** This applies the entered offsets to the individual observations within a target as well. Very rarely would you not want to select this option.
- Set Tolerances.** When checked, the tolerances specified via the *Tolerance Settings* button will be applied to the points.

Note: The tolerance settings dialog is the same as described in “Point Tolerance” on page 196.

Working With Point Lists

It is often useful to view a list of points and point coordinates in tabular format, so that you can easily compare against nominals or quickly modify a set of points at once (similar to working in Excel). SA's point lists provide this functionality (Figure 8-23).



Name	X (in)	Y (in)	Z (in)
M1	-375.2509	257.1625	-57.0036
M2	-269.1473	349.7335	-56.8861
M3	-237.7965	265.5530	-56.0916
M4	-170.6404	290.7616	-55.8216
M5	-183.4629	220.4930	-55.6144
M6	-106.4905	232.0723	-55.1983
M7	-61.2574	294.2564	-55.2998

Figure 8-23. A point list.

To View a Point List:

Do one of the following:

- From the menu, select **View>Point Lists for Groups**, then select the group(s) to view, or
- Right-click a group and select **View Point List** from the context menu (to view a single group).

The list has several buttons at the top:

- **Refresh.** Refreshes the list to ensure it's showing the latest information.
- **Copy to Clipboard.** Copies the selected rows (or all rows if nothing is selected) to the clipboard.
- **Export to Text File.** Exports the selected rows (or all rows if nothing is selected) to a text file.
- **Display.** Display provides a list of additional point properties that can be displayed including tolerances, weights, offsets, etc.
- **Coordinate System.** Click this button to switch among Cartesian, Cylindrical, and Spherical coordinates.
- **Edit.** Opens up a new window allowing you to modify the point information (see below).

Instead of just viewing the information, a point list can be used for modifying point names, coordinates or properties. This is particularly useful because you can change multiple values at once. For instance, you can quickly change the x-coordinate of multiple points in one

action.

To Edit a Point List:

Do one of the following:

- With the point list already displayed (see “[To View a Point List:](#)” on page 214), click the **Edit** button, or
- Right-click a point group and select **Edit Point List** from the context menu to edit that group of points.

The **Edit Point List** dialog is displayed in [Figure 8-24](#).

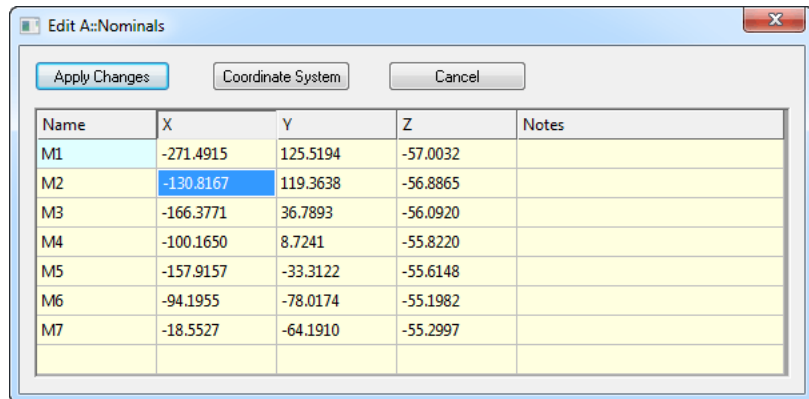


Figure 8-24. The **Edit Point List** dialog.

To Change a Point’s Name, Coordinate, or Notes:

- Double-click the field you’d like to modify, enter a new value, then press **Enter**.

To Modify Multiple Values At Once:

1. Select multiple fields using **Shift+Click** (standard Windows range selection) or **Ctrl+Click** (standard Windows non-contiguous selection).
2. Double-click one of the selected fields, type a new value, and press **Enter**. All fields will be updated with the new value.

The dialog provides a few additional buttons at the top.

- **Apply Changes.** Modifications are not actually applied to the points until you click this button.
- **Coordinate System.** Choose to switch the display between Cartesian, Cylindrical, or Spherical coordinates.
- **Cancel.** If you cancel, no changes will be applied the points.

Point Sets

Point Sets

Point Sets provide a merged point format that allows many individual points to be saved as part of a single object. Using Point Sets provides the advantage of keeping the tree manageable and allowing many more points to be measured in a single job file at faster rates (Figure 8-25) than are available with traditional points. Unlike a point cloud, each point within a point cloud retains critical information including: Point name and (when available) instrument measurement time.

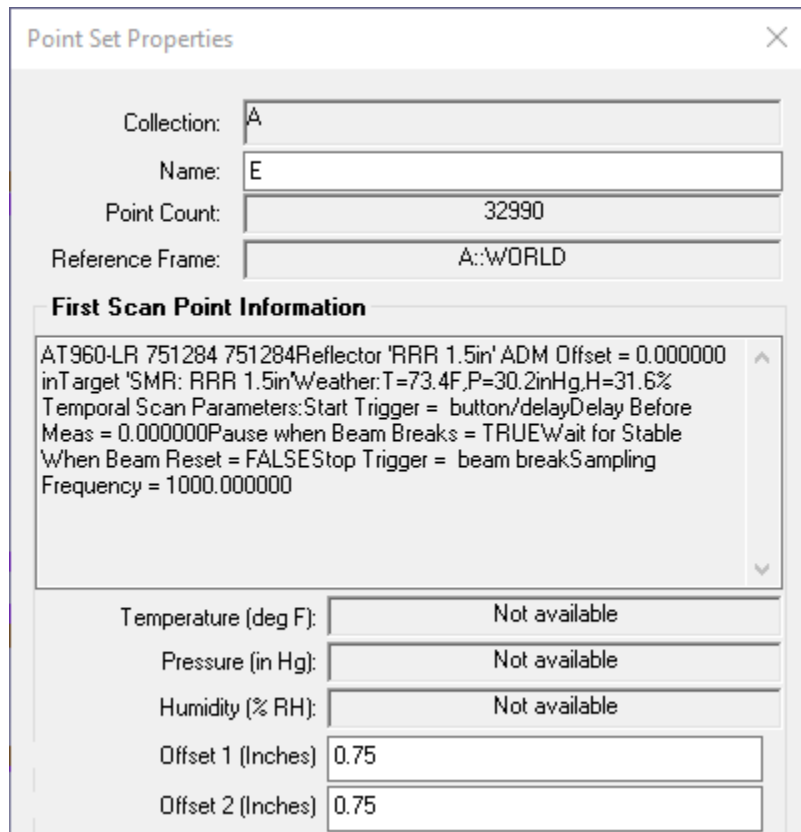


Figure 8-25. Point Set Properties dialog include a full set of measurement details

Measuring with Point Sets

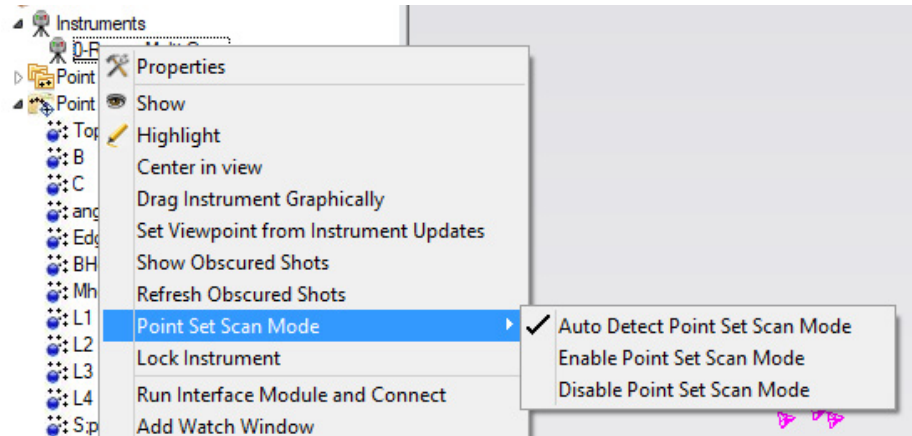
Point Sets can only be recorded as part of a Scan but are available from any measurement device that can scan points. Single Discrete measurements will still always be recorded as individual points.

This method of data capture is an option that must also be turned on through the right-click instrument menu by selecting **Enable Point Set Scan Mode** (Figure 8-26):

- **Auto Detect Point Set Scan Mode.** When set individual point measurements will be recorded as individual points but scans will be recorded in Point Sets.

- **Enable Point Set Mode.** This option enables data recording within a point set for newly recorded points.
- **Disable Point Set Scan Mode.** This option turns off any Point Set data acquisition.

Figure 8-26. Enable Point Set Mode must be active to record measurements in Point Sets

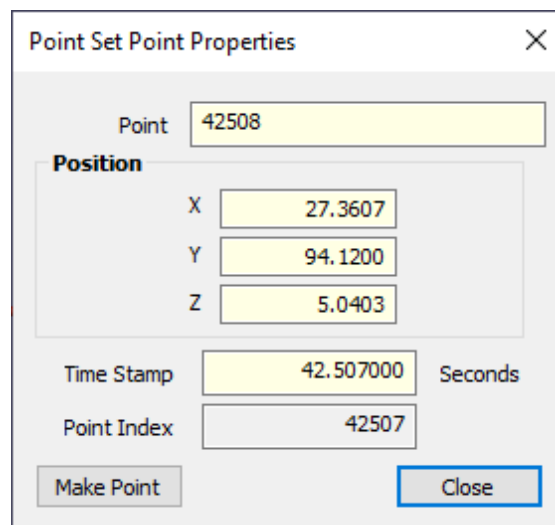


Individual Point Selection from a Set

Double left clicking on a single point in the graphics provides the ability to select a single point from within the point set. Doing so provides direct access to the position and time stamp recorded with that particular measurement.

Time stamps are recorded as a fundamental property of a point set, when available from the instrument, and reflect the instrument's own controller time. This controller time is quite accurate and reflects the time from the beginning of the scan.

Figure 8-27. Individual Point Selection from within a Point Set



Constructing Points

There are many different ways to create points in SA. You can explicitly create a point at a specified coordinate or create points through an array of geometric analysis functions.

Entering Point Coordinates

The most basic way to create a point is to explicitly enter a point by specifying its collection, group, target, and coordinate system. This will create a single constructed point in your job.

To Enter a Point:

1. From the menu, select **Construct>Points>Enter**, or press **Ctrl+P** (default keyboard shortcut). The **Add Points to Model** dialog will appear (Figure 8-28).

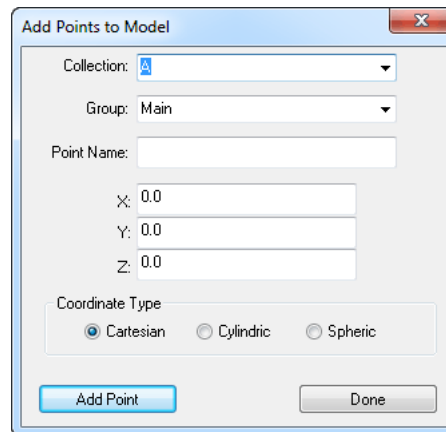


Figure 8-28. Constructing a point by entering its coordinates.

2. Enter a collection, group, and point name (or select one from the dropdown of existing collections or groups).
3. Specify the desired coordinate type (*Cartesian*, *Cylindric*, or *Spheric*).
4. Enter the desired coordinate values, then click the **Add Point** button to create the point and continue adding more, or click **Done** to create the point and close the dialog.

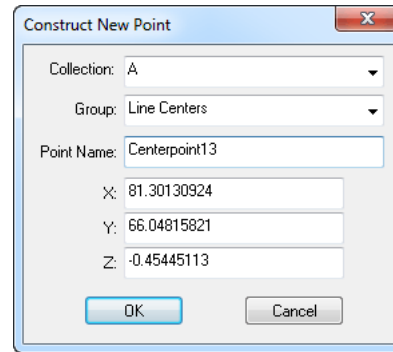
Constructing a Point at the Center of a Line

Creating a point at the center of a line constructs a point halfway between its endpoints.

To Create a Point at the Center of a Line:

1. From the menu, select **Construct>Points>Center>Line**.
2. Select the desired line.
3. The **Construct New Point** dialog will appear (Figure 8-29).

Figure 8-29. The *Construct New Point* dialog.



4. Leave the coordinates as-is (they are calculated for you), and specify the desired collection, group, and point name (or select an existing collection or group from the dropdown).
5. Click the **OK** button.

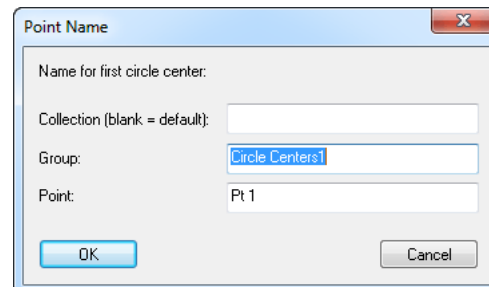
Constructing Points at the Center of Spheres and Circles

You can construct points at the center of multiple circles or multiple spheres with a single command.

To Construct Points at the Center of Circles or Spheres:

1. From the menu, select **Construct>Points>Center>Circles** or **>Spheres**. The *Point Name* dialog will appear (Figure 8-30).

Figure 8-30. The *Point Name* dialog.



2. Enter a collection, group, and point name for the first point to construct. (Leave the *Collection* field blank to put the points into the active collection). Click **OK**.
3. Select one or more circles or spheres. Each center point will be created, and successive point names will be incremented.
4. When finished creating centers, press **Esc** or **Enter**.

Constructing Points at the Center of a Cylinder

When creating points at the center of cylinders, SA will actually create three points: one point at each end on the cylinder's axis (called **EndA** and **EndB**), and a third point halfway between the endpoints (called **Center**).

To Create Points at the Center of Cylinders:

1. From the menu, select **Construct>Points>Center>Cylinder**.
2. Select the cylinder. The points will be placed into the active collection, into a group with a name matching that of the cylinder.

Projecting to the Closest Point on Objects

Projecting points to the closest point on an object is equivalent to creating points on the object where that object's surface normals intersect with the original points.

With this command, you can select multiple objects and SA will find the closest point among all selected candidate objects. You can also select multiple points, and SA will find the closest point for each of the candidate points.

You can also offset points away from the surface. This is particularly useful when you have nominal points on a surface that you'd like to back away from the surface by the probe radius (so that those points can serve as proximity triggers for measurement).

To Project Points to the Closest Points on an Object:

1. From the menu, select **Construct>Points>Project Points to>Objects>Closest Point**.
2. Select one or more objects to project to.
3. Select one or more points to project. The *Point Naming* dialog will appear (Figure 8-31).

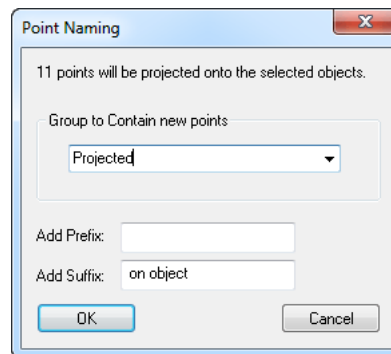


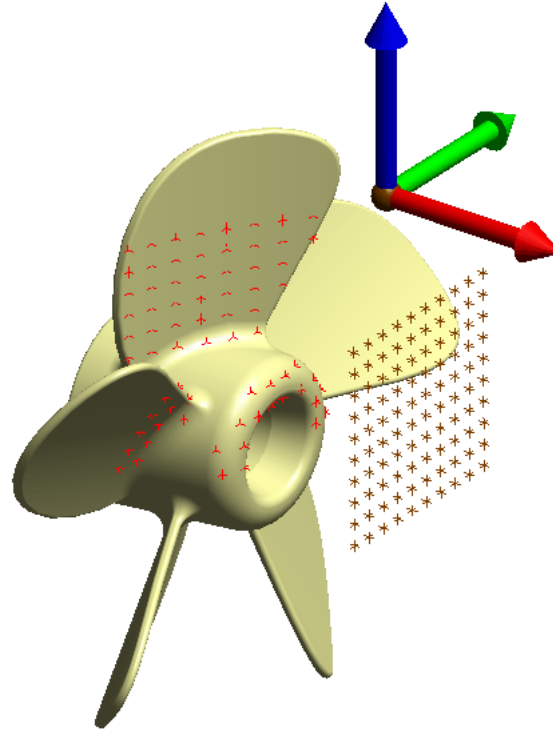
Figure 8-31. The *Point Naming* dialog.

4. Enter or select the desired group to contain the projected points. In the *Add Prefix* or *Add Suffix* field, enter a prefix or suffix that you'd like to attach to the point names, then click **OK**.
5. The *Probe Offset* dialog will appear. Enter a distance to offset the projected points back away from the object's surface. A positive value shifts along the positive direction of the surface normal.

Projecting Points Parallel to a Working Coordinate Frame Axis

You can project points along a working coordinate frame axis until it intersects with one or more surfaces (Figure 8-32). (If there is no intersection, no projected point will be created).

Figure 8-32. Projecting a grid of points along the working X axis to a surface.



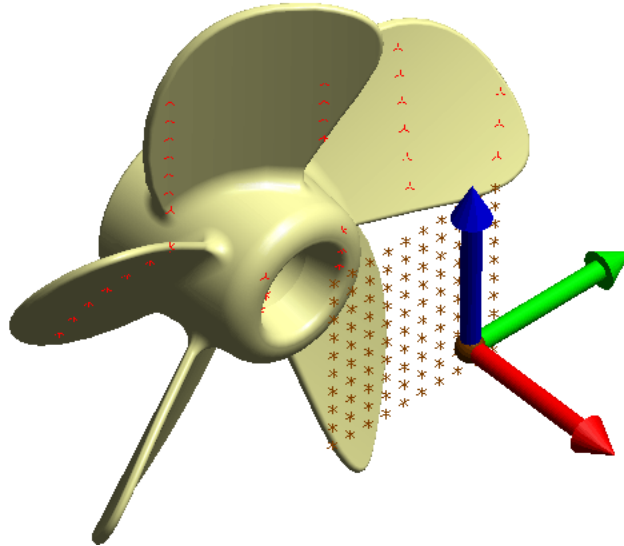
To Project Points Along the Working Coordinate Frame Axis:

1. From the menu, select **Construct>Points>Project Points to>Objects>Relative to Coordinate Frame Axis>Parallel to WCF Axis**.
2. When prompted, pick the surfaces to project to.
3. Select the points you'd like to project.
4. When the **Select Axis** dialog appears, choose the axis to project along.
5. The **Point Naming** dialog will appear (Figure 8-31). Enter or select the desired group to contain the projected points. In the **Add Prefix** or **Add Suffix** field, enter a prefix or suffix that you'd like to attach to the point names, then click **OK**.

Projecting Points Radially from a Working Coordinate Frame Axis

You can also project points radially from a coordinate frame axis. Here, one of the axes of the working coordinate frame serves as the axis of a cylindrical coordinate system, and the points are projected out radially from that axis until they intersect with one or more surfaces (Figure 8-33).

Figure 8-33. Projecting radially from a coordinate frame axis.



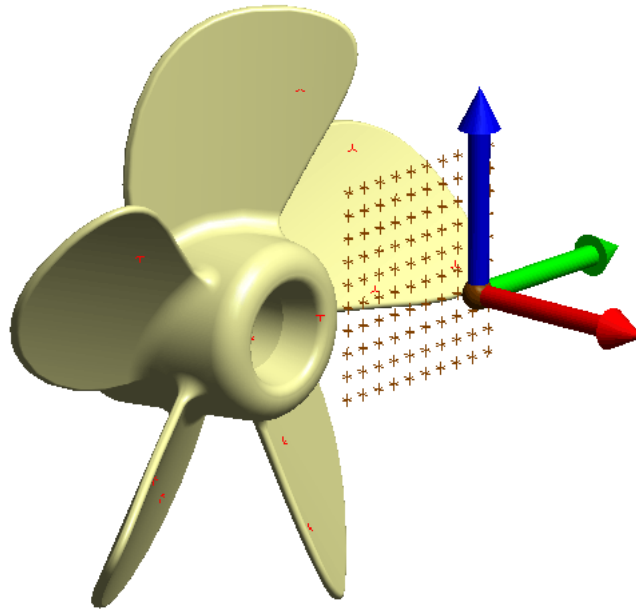
To Project Radially from a Working Coordinate Frame Axis:

1. From the menu, select **Construct>Points>Project Points to>Objects>Relative to Coordinate Axes>Radial from WCF Axis**.
2. When prompted, pick the surfaces to project to.
3. Select the points you'd like to project.
4. When the **Select Axis** dialog appears, choose the axis from which to project radially.
5. The **Point Naming** dialog will appear (Figure 8-31). Enter or select the desired group to contain the projected points. In the **Add Prefix** or **Add Suffix** field, enter a prefix or suffix that you'd like to attach to the point names, then click **OK**.

Projecting Points Spherically from a Working Coordinate Frame Origin

Points can be projecting spherically outward from the origin of the working coordinate frame (Figure 8-34).

Figure 8-34. Projecting points spherically from the working coordinate frame origin.



To Project Points Spherically From the Working Coordinate Frame Origin:

1. From the menu, select **Construct>Points>Project Points to>Objects>Relative to Coordinate Frame Axis>Spherical From WCF Origin**.
2. When prompted, pick the surfaces to project to.
3. Select the points you'd like to project.
4. The **Point Naming** dialog will appear (Figure 8-31). Enter or select the desired group to contain the projected points. In the *Add Prefix* or *Add Suffix* field, enter a prefix or suffix that you'd like to attach to the point names, then click **OK**.

Projecting Points to a Curve

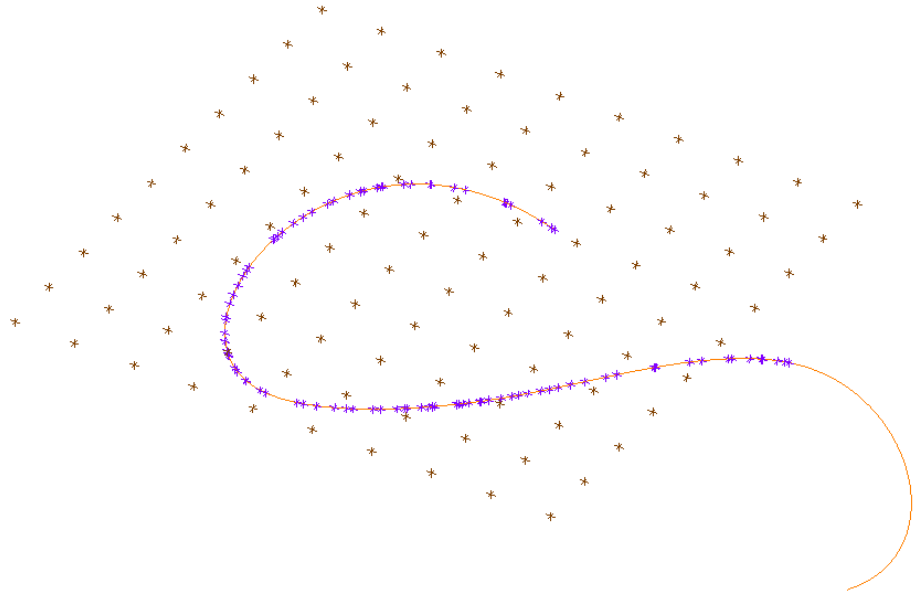
You can project points directly to any arbitrary curve or curves (Figure 8-35). The projected points can lie directly on the curves or can be backed off the curve by a specified amount. Each point will be projected to the closest curve among all candidates.

To Project Points to a Curve:

1. From the menu, select **Construct>Points>Project Points to>Curves**.
2. At the prompt, select the curves to project to.
3. Select the points to project.
4. The **Point Naming** dialog will appear (Figure 8-31). Enter or select the desired group to contain the projected points. In the *Add Prefix* or *Add Suffix* field, enter a prefix or suffix that you'd like to attach to the point names, then click **OK**.
5. At the **Probe Offset** dialog, enter a distance to offset the pro-

jected points back away from the curve. This value must be positive, and always shifts away from the curve.

Figure 8-35. Projecting points to a curve.



Creating Points at a 3-Plane Intersection

A point can be created at the intersection of 3 planes. This of course requires that no two planes may be parallel.

To Create a Point at the Intersection of 3 Planes:

1. From the menu, select **Construct>Points>Intersection>3 Planes**.
2. Select the three planes successively.
3. The *Construct New Point* dialog (see [Figure 8-29](#) on page 219) will appear. Leave the coordinates as-is (they are calculated for you), and specify the desired collection, group, and point name (or select an existing collection or group from the dropdown).
4. Click **OK** to create the point.

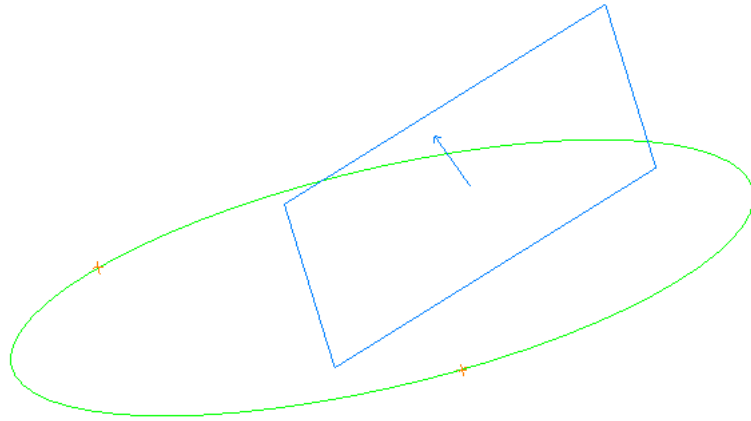
Creating Points at the Intersection of a Plane and Circle

Intersecting a plane and a circle will actually create two points as seen in [Figure 8-36](#). It should be noted that the plane and circle cannot be parallel or coplanar.

To Create Points at the Intersection of a Plane and Circle:

1. From the menu, select **Construct>Points>Intersection>Plane and Circle**.
2. Select the plane, then the circle. Two new points will be created in the active collection's **Intersections** group named **Plane-Circle1** and **Plane-Circle2**.

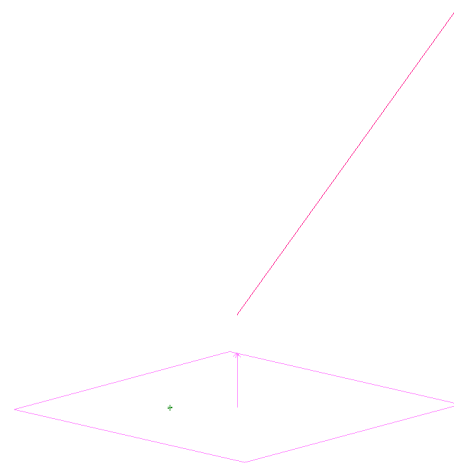
Figure 8-36. Intersecting a plane with a circle.



Constructing Points at the Intersection of a Line and Plane

You can intersect a line with a plane to create a point at the intersection (Figure 8-37).

Figure 8-37. A line and plane intersection creating a point, even though their depictions do not intersect.



There are a few things to keep in mind:

- The line and plane need not physically intersect, since planes and lines are treated as infinite in size.
- If the line lies in the plane, or the line is parallel to and not in the plane, no points will be created.

To Create a Point at the Intersection of a Line and Plane:

1. From the menu, select **Construct>Points>Intersection>Line and Plane**.
2. Select the line, then the plane.
3. The **Construct New Point** dialog (Figure 8-29) appears.
4. Leave the coordinates as-is (they are calculated for you), and specify the desired collection, group, and point name (or select an existing collection or group from the dropdown).
5. Click the **OK** button.

Constructing a Point at the Intersection of 2 Lines

You can create a point at the intersection of two lines (Figure 8-38). Or, if the lines do not intersect, a point will be created at their mutual perpendicular midpoint. (The mutual perpendicular midpoint is half-way between the closest points between the two lines). If the lines are parallel, no point will be created.

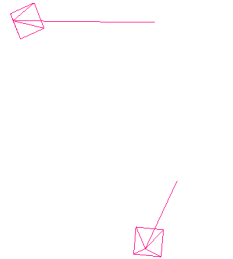


Figure 8-38. A point constructed at the mutual perpendicular midpoint between two lines. In this case, these two lines do not intersect.

To Create a Point at the Intersection of Two Lines:

1. From the menu, select **Construct>Points>Intersection>2 Lines (mutual perpendicular midpoint)**.
2. Select the first line, then the second.
3. The **Construct New Point** dialog (Figure 8-29) appears.
4. Leave the coordinates as-is (they are calculated for you), and specify the desired collection, group, and point name (or select an existing collection or group from the dropdown).
5. Click the **OK** button.

Constructing Points at the Intersection of a Line and Circle

Points can be created at the intersection of a line and a circle (Figure 8-39). The line is first projected to the plane on the circle and then the intersection points are created.

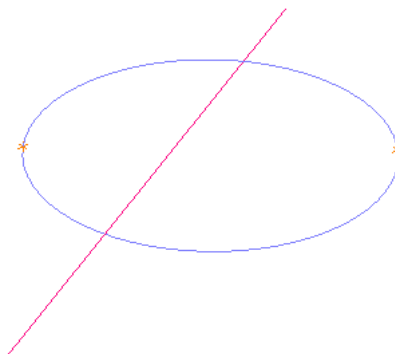


Figure 8-39. A line and circle intersection creating two points.

To Create Points at the Intersection of a Line and Circle:

1. From the menu, select **Construct>Points>Intersection>Line and Circle**.
2. Select the line, then the circle. Two new points will be created

in the active collection's **Intersections** group.

Constructing Points at the Intersection of a Line and Cylinder

You can create a point at each intersection of a line and cylinder. The cylinder and line need not physically intersect, since cylinders and lines are treated as infinite in length (Figure 8-40). However, the line must pass within the cylinder's radius.

+

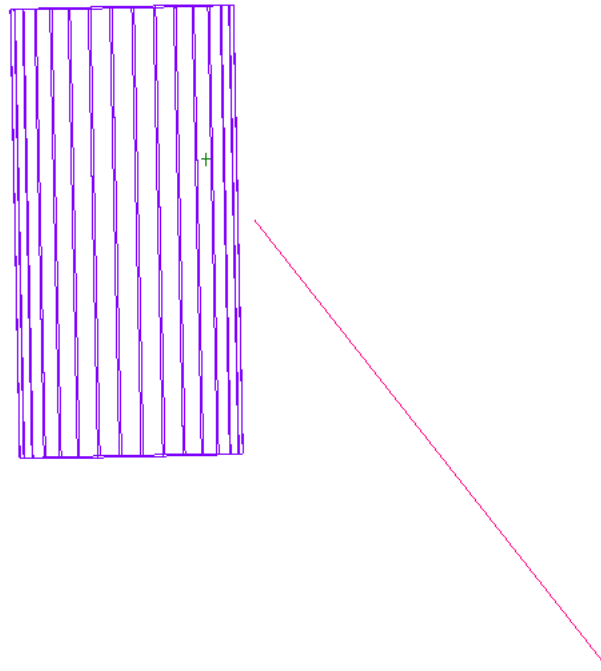


Figure 8-40. A point at each intersection of the line and cylinder are created, even though their depictions do not intersect. Note the length of the cylinder does not need to be increased, since cylinders are infinite in size.

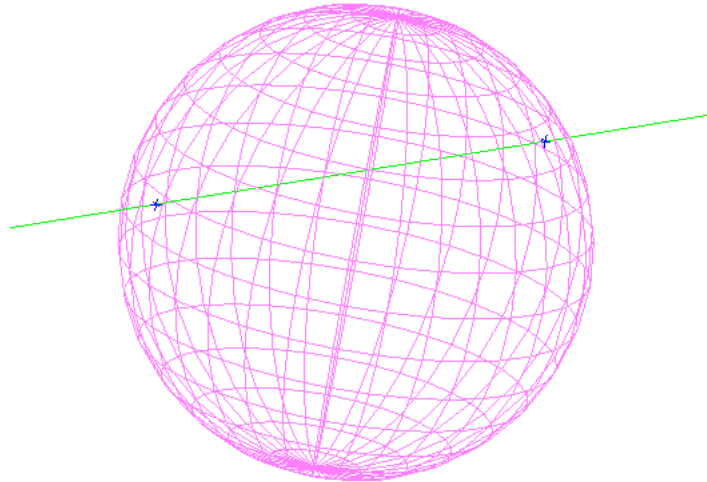
To Create a Point at the Intersection of a Line and Cylinder:

1. From the menu, select **Construct>Points>Intersection>Line and Cylinder**.
2. Select the line, then the cylinder. A naming dialog appears (Figure 8-42).
3. Specify the desired collection, group, and point names (or select an existing collection and group from the dropdown).
4. Click **OK** to create the point.

Constructing Points at the Intersection of a Line and Sphere

You can create a point at each intersection between a line and a sphere.

Figure 8-41. A point is constructed at each intersection of the line and sphere.



To Create a Point at the Intersection of a Line and Sphere:

1. From the menu, select **Construct>Points>Intersection>Line and Sphere**.
2. Select the line, then the sphere.
3. A naming dialog appears (Figure 8-42).
4. Specify the desired collection, group, and point names (or select an existing collection and group from the dropdown).
5. Click **OK** to create the point.

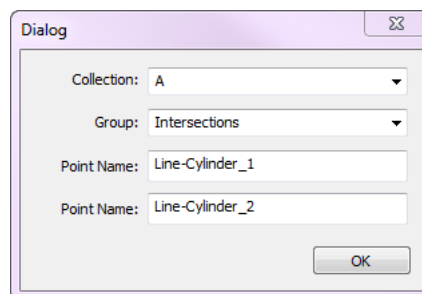


Figure 8-42. The naming dialog for line intersections with cylinders and spheres.

Constructing Points at the Intersection of 2 B-Splines

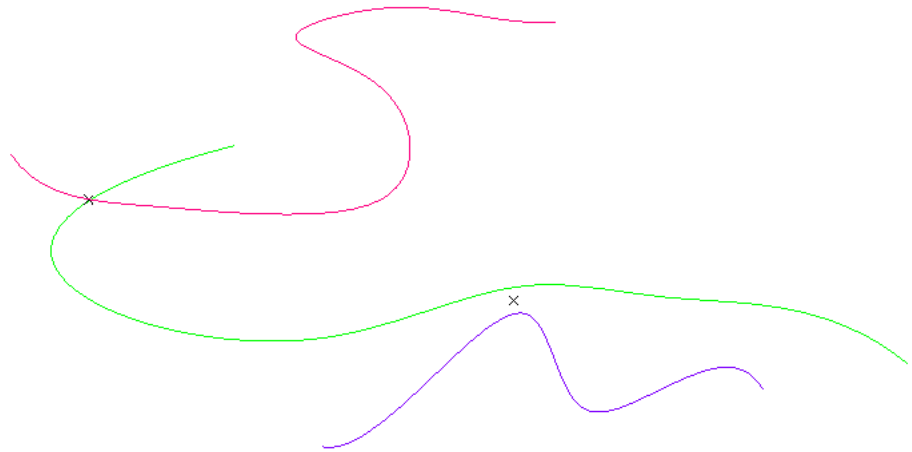
Points can be constructed from the intersections of two B-Splines. A point will be created at the midpoint of the closest location between the two B-splines if they do not intersect (Figure 8-43).

To Create a Point at the Intersection of 2 B-Splines:

1. From the menu, select **Construct>Points>Intersection>2 B-Splines**.
2. Select the first B-spline, then the second.
3. The **Construct New Point** dialog (Figure 8-29) will appear. Leave the coordinates as-is (they are calculated for you), and specify the desired collection, group, and point name (or select an existing collection or group from the dropdown).

- Click **OK** to create the point.

Figure 8-43. The pink and green B-splines create a point where they physically intersect, whereas the green and purple B-splines create a midpoint between their closest locations since they do not intersect.



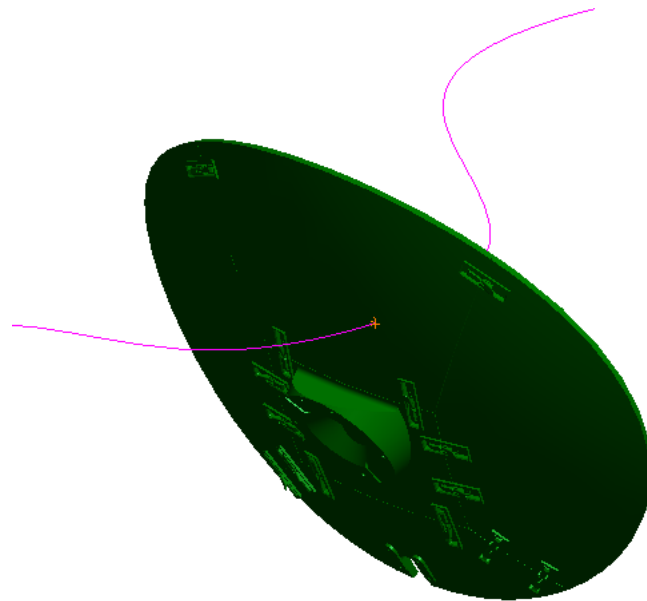
Constructing Points at the Intersection of a B-Spline and Surfaces

Note: If the B-Spline intersects multiple times or intersects multiple selected surfaces, only one point is created. Which point is created is not generally clear.

You can create a point at the intersection of a B-Spline and a surface (Figure 8-44). The command follows the B-Spline from its tail toward its head and creates a point when the spline first pierces the surface.

You can select multiple surfaces with this command, but still only one intersection point will be created.

Figure 8-44. Intersecting a B-Spline with a surface to create a point.



To Create a Point at the Intersection of a B-Spline and Surfaces:

- From the menu, select **Construct>Points>Intersection>B-Spline and Surfaces**.
- Select the B-Spline.
- Select one or more surfaces to test.

4. When prompted, enter an approximation tolerance. Generally, the default value is fine.
5. The **Construct New Point** dialog (Figure 8-29) will appear. Leave the coordinates as-is (they are calculated for you), and specify the desired collection, group, and point name (or select an existing collection or group from the dropdown).
6. Click **OK** to create the point.

Constructing Points at the Intersection of Principle Object Axes and Surfaces

A point can be constructed at the intersection of an object's axis and a surface (Figure 8-45). The definition of the axis depends on the type of object, but the axis must intersect at a point on the surface and not through a hole in the surface.

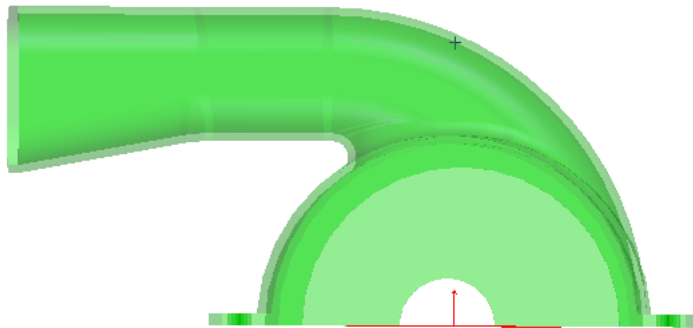


Figure 8-45. A point created at the intersection of a plane's axis and a surface.

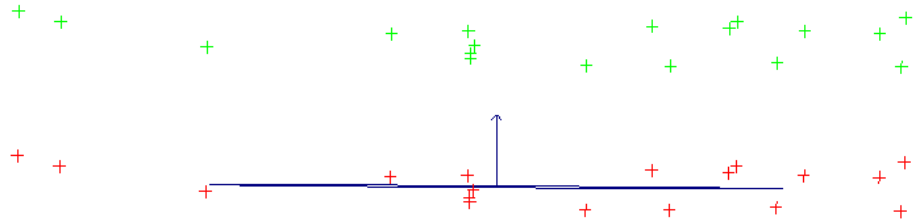
To Construct a Point at the Intersection of an Object Axis and Surface:

1. From the menu, select **Construct>Points>Intersection>Principle Object Axes and Surfaces**.
2. Choose the axis object, followed by the surface.
3. Enter the destination point group name. You also have the option of entering a suffix for the points (which will have the same name as your object).
4. Click **OK**.

Shifting Points In a Direction

You can use a selected object to define a direction in which to shift a point or selection of points by a specified distance (Figure 8-46). If a frame is selected as the object, you will be prompted to choose along which axis to shift the points.

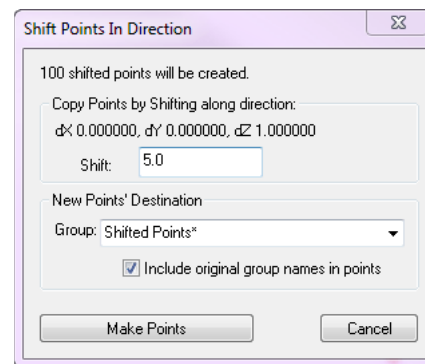
Figure 8-46. Green points were copied from the original points (red) and shifted along the circle's surface normal.



To Shift Points in a Direction:

1. From the menu, select **Construct>Points>Shift Points>In a Direction**.
2. Select the object defining the shift direction, then select the points you would like to shift.
3. The *Shift Points in Direction* dialog will appear (Figure 8-47). Enter the shift distance and the point group destination.

Figure 8-47. Shifting points along a direction.

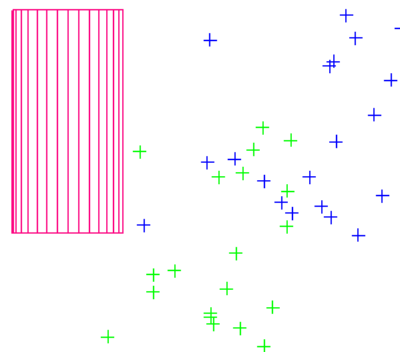


4. Click **Make Points** to create the shifted points.

Shifting Points In Cylindric Coordinates

Points may be shifted cylindrically. Any object that contains directional information may be selected to define the axis of the cylinder about which the points are shifted (Figure 8-48).

Figure 8-48. The cylinder is the object defining the cylindrical coordinate system and the blue points were copied from the original points (green) and shifted 5.0" along the planar and radial directions.



To Shift Points in Cylindric Coordinates:

1. From the menu, select **Construct>Points>Shift Points>In Cylindric Coordinates**.
2. Select the object defining the cylindric coordinate system,

followed by the points to be shifted.

3. The *Shift Points in Cylindric Coordinates* dialog will appear (Figure 8-49). Enter the radial and planar shifts and the point group destination.

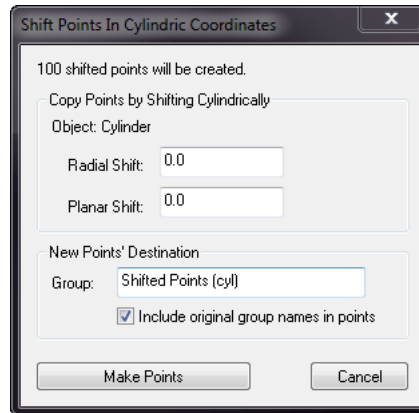


Figure 8-49. Specifying the radial and planar shift.

4. Click **Make Points** to create the shifted points.

Laying Out Points on a Grid By Count

A grid of points can be laid out with respect to the XY plane of the working coordinate frame. Using this command, you specify the range of the grid and the number of points desired along each axis, and SA determines the spacing for you automatically.

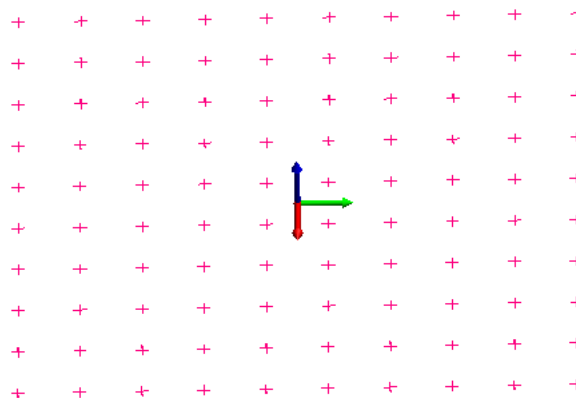


Figure 8-50. 10x10x1 Grid layout of 100 points.

To Lay Out Points on a Grid by Count:

1. Activate the frame you'd like to use to define the grid orientation.
2. From the menu, select **Construct>Points>Layout>Grid by Count**.
3. The *Make a Grid of Points* dialog will appear (Figure 8-51). In the *Min* and *Max* fields, specify the limits along the coordinate directions for the three-dimensional grid. In the *Count* fields, specify the number of rows of points along each direction. You can also enter a destination collection, group, and target

prefix for the resulting points.

4. Click **Create Points** to construct the grid.

Figure 8-51. Entering the X, Y, and Z range of the grid, and the number of rows of points along each axis direction.

The dialog box 'Make a Grid of Points' contains the following fields and values:

- X Axis:** Min: 0.0, Max: 100.0, Count: 10
- Y Axis:** Min: 0.0, Max: 50.0, Count: 10
- Z Axis:** Min: 0.0, Max: 10.0, Count: 1
- Destination:** Collection Name: (empty), Group Name: PointGrid, Point prefix: P

Laying Out Points on a Grid By Distance

A grid of points can be laid out with respect to the XY plane of the working frame. Using this command, you specify the range of the grid and the distance between points, and SA determines the number of points for you automatically.

To Lay Out Points on a Grid by Distance:

1. Activate the grid you'd like to use to define the grid orientation.
2. From the menu, select **Construct>Points>Layout>Grid by Distance**. The *Make a Grid of Points* dialog will be displayed (Figure 8-52).

Figure 8-52. Defining parameters to create a grid by spacing.

The dialog box 'Make a Grid of Points' contains the following fields and values:

- X Axis:** Min: 0.0, Max: 100.0, Distance: 10.0000
- Y Axis:** Min: 0.0, Max: 50.0, Distance: 5.0000
- Z Axis:** Min: 0.0, Max: 10.0, Distance: 0.0000
- Destination:** Collection Name: (empty), Group Name: PointGrid, Point prefix: P

3. Specify the range of the grid in all three component directions, as well as the desired spacing between points along each direction. You can also enter a destination collection,

group, and target prefix for the resulting points.

4. Click **Create Points** to construct the grid.

Laying Out Points On a Circle

It is often useful—particularly when laying out nominal bolt hole patterns—to lay out points on a circle. A circular point layout can be constructed in the XY plane of the working frame (Figure 8-53).

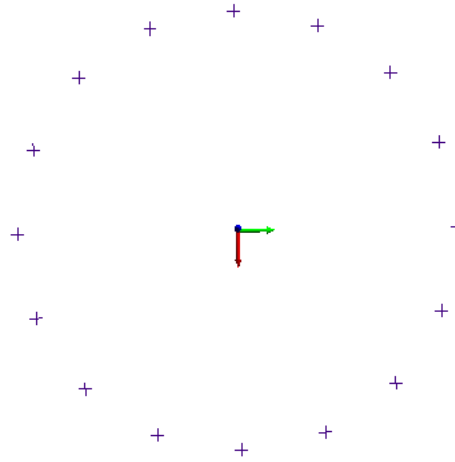


Figure 8-53. Laying out 16 evenly-spaced points on a circle.

The *Layout Points on Circle* dialog is shown in Figure 8-54. A number of options are available when laying out points, which are discussed below:

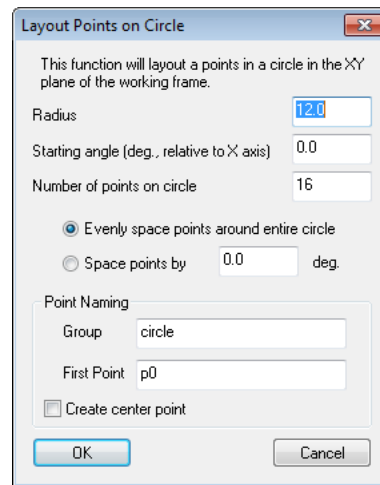


Figure 8-54. The *Layout Points on Circle* dialog.

- **Radius.** The radius of the circle defining the points.
- **Starting angle.** The angle in the working frame’s XY plane, from the X axis, at which to lay out the first point.
- **Number of points on circle.** The number of points to create on the circle.

There are two methods for laying out points on the circle:

- **Evenly space points around entire circle.** The points will be

constructed evenly around the entire circle.

- **Space points by.** You can specify the angle between points. If (based on the specified spacing and the number of points) more than 360° of the circle is specified, the points will continue to be laid out around the circle.

You can also control the naming of the resulting points using the *Group* and *First Point* fields, and whether a center point is created with the *Create center point* option.

To Construct a Layout of Points on a Circle:

1. From the menu, select **Construct>Points>Layout>Circle**.
2. The *Layout Points on Circle* dialog will appear (Figure 8-54).
3. Fill in the desired parameters, then click **OK** to create the points.

Generating Random Points

Occasionally, you may come across the need to generate random points. (This tends to be more common in training and testing than actual production environments).

When generating random points (Figure 8-55), you can specify the number of points and the range of the points along each of the three axis directions. The points will be randomly created inside this defined box.

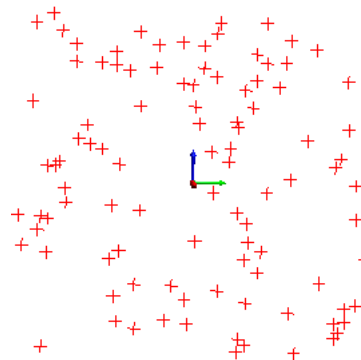
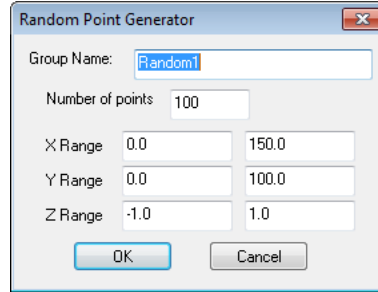


Figure 8-55. 100 randomly-generated points.

To Construct a Random Layout of Points:

1. From the menu, select **Construct>Points>Layout>Random Points** or use the default keyboard shortcut **Ctrl+Alt+Z**.
2. The Random Point Generator dialog will appear (Figure 8-56). Specify the number and range of point coordinates as well as the desired group name.

Figure 8-56. The *Random Point Generator* dialog.

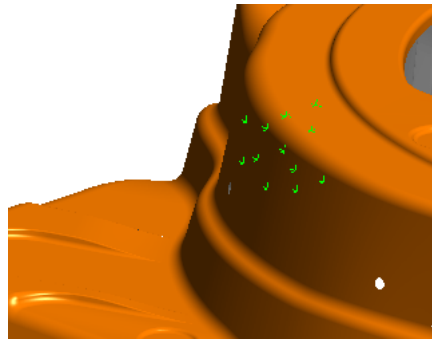


3. Click **OK** to create the points.

Laying Out Points On A Surface By Clicking

You can click anywhere on a surface to create a point on that surface (Figure 8-57).

Figure 8-57. Laying out points on a surface.



To Lay Out Points on a Surface:

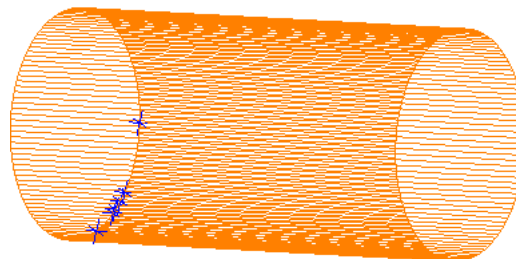
Note: The first click on a surface will incur a small delay as the software performs some calculations. Remaining clicks on the surface will be much faster.

1. From the menu, select **Construct>Points>Layout>On Surface By Clicking**.
2. Place the cursor anywhere on the surface and click to create a point at that location. If the cursor is not over a surface, a point will not be created.

Laying Out Points On An Object Vertex By Clicking

Points can be constructed on an object’s vertices simply by clicking near the vertex (Figure 8-58).

Figure 8-58. Several points have been laid off on this cylinder’s vertices.



To Lay Out Points on Object Vertices by Clicking:

Note: The points will always be created on the surface, regardless of render settings.

1. From the menu, select **Construct>Points>Layout>On Object Vertex by Clicking**.

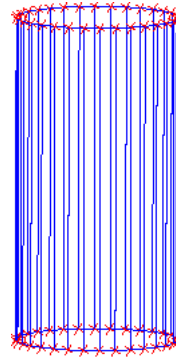
Tip: Increasing surface rendering detail will create more vertices for a given surface.

2. At the prompt, enter a desired group and starting point name.
3. Click the object near the desired vertices.
4. Press **Enter** when finished.

Laying Out Points On An Object's Vertices

You can construct points on all vertices of one or more objects at once (Figure 8-59).

Figure 8-59. Points laid out on a cylinder's vertices.



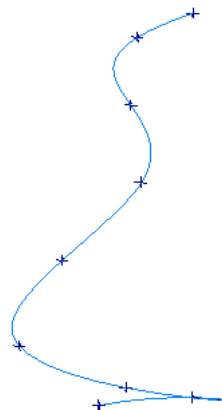
To Lay Out Points on an Object's Vertices:

1. From the menu, select **Construct>Points>Layout>On Object's Vertices**.
2. Select the objects on which you'd like to create vertex points.
3. Enter a name for the destination group, then click **OK**.

Laying Out Points Evenly On Curves

You can construct any number of evenly-spaced points on one or more B-Spline curves (Figure 8-60).

Figure 8-60. 10 points spaced evenly on a B-Spline.



To Lay Out Evenly-Spaced Points on a Curve:

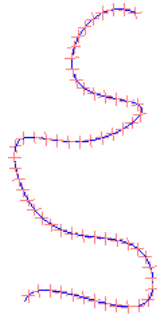
1. From the menu, select **Construct>Points>Layout>On Curves>n Spaced Points**.
2. Select one or more B-splines.

3. Type a name for the destination group, then click **OK**.
4. At the prompt, enter the number of points to create on each curve, then click **OK**.

Laying Out Points On Curves With Prescribed Spacing

SA can construct points on one or more curves spaced by a prescribed distance (Figure 8-61). This is not a straight-line distance, but a distance along the curve (as if you'd laid a measuring tape of the prescribed length along the curve). When creating the points, SA places the first point exactly on the starting end of the curve.

Figure 8-61. Points on a B-spline spaced exactly 2" from each other along the curve.



To Create Points Spaced on a Curve by a Prescribed Distance:

1. From the menu, select **Construct>Points>Layout>On Curves>Spaced at Distance**.
2. Select one or more B-splines.
3. Type a name for the destination group, then click **OK**.
4. Enter the specified distance between points, then click **OK**.

Laying Out Points On Curves by Clicking

This command allows you to click anywhere on one or more curves and create a point at that position on the curve (Figure 8-62).

Figure 8-62. Several points created by clicking on a B-Spline.



To Create Points on a Curve by Clicking:

1. From the menu, select **Construct>Points>Layout>On Curves>By Clicking**.
2. Enter a destination group name for the constructed points, then click **OK**.
3. Type a starting point name, then click **OK**.

4. Select one or more curves to consider.
5. Click at the desired locations. SA will create the point on the closest point of the closest curve.

Laying Out Points on Curves Max Chordal Deviation

Builds points on a curve at higher densities where the curve is at a tighter radius. This can be controlled in a couple of ways. It will ask you to select:

- **Maximum Chordal Deviation.** Which defines the greatest distance the mid point of a line between any two points on a curve can deviate from that curve without adding an additional point.
- **Maximum Trim Edge Angle.** Which sets an additional threshold such that if the angle between any two chordal lines is greater than the threshold a new points is built.
- **Maximum Chord Length.** Allows you to set a maximum spacing between points along the curve. This can be helpful in ensuring a minimum point spacing along straight sections.

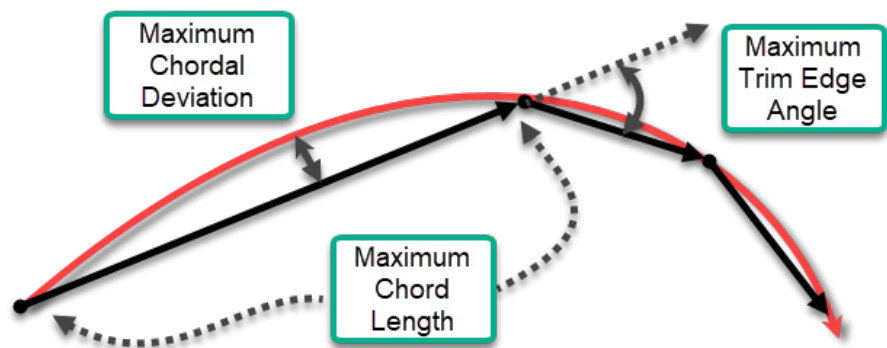


Figure 8-63. Depiction of the available controls provided by the Max Chordal Deviation tool

All the points built with this command will be constructed perfectly on the mathematical definition of the curve. * Note* that the graphical depiction of a curve also uses a similar approximation and if the graphical display is not equal to or higher than the construction tolerance the points may not appear to be on the curve even when they are (to control the graphical display go to the **User Options>Display tab, Surface Rendering Options**).

Laying Out Points On Lines

As with a curve, you can lay points out on a line by specifying the number of points (in which they're evenly spread along the line) or by

specifying the spacing (Figure 8-64).

Figure 8-64. A line with 10 evenly-spaced points.



To Lay Out Points on a Line:

1. From the menu, select **Construct>Points>Layout>On Lines**.
2. Select one or more lines.
3. The *Layout Points on Lines* dialog will appear (Figure 8-65).

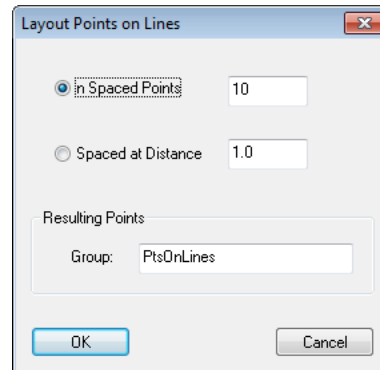


Figure 8-65. The *Layout Points on Lines* dialog.

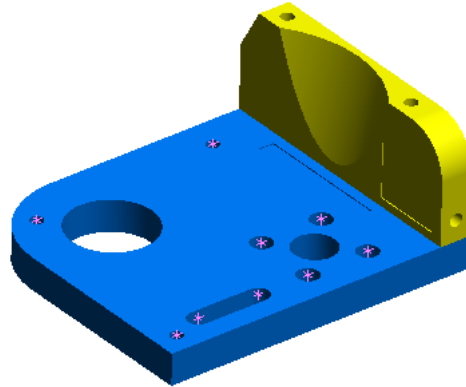
- **n Spaced Points.** SA will create the specified number of points spaced evenly along each of the selected lines.
 - **Spaced at Distance.** SA will create points on each line starting from the tail of the line, using the prescribed spacing.
 - **Group.** The name for the resulting point group.
4. Choose the desired options, then click **OK**.

Laying Out Points At The Center of Surface Holes

It is often useful to create points at the center of holes on a surface. You can click on a single surface containing surface holes, and SA will create points at the center of the holes based on prescribed minimum and maximum diameter thresholds (Figure 8-66).

If a surface is dissected and you therefore select just a single surface face with holes, the points will be created along that face. If a surface is merged and you select a closed shell, points will be created on each end of the hole (if it is a thru hole).

Figure 8-66. These points have been constructed at the center of surface holes.



SA considers any surface opening that has a circular arc for an edge as a surface hole.

To Construct Points at the Center of Surface Holes:

1. From the menu, select **Construct>Points>Layout>At Center of Surface Holes**.
2. Select one or more surfaces with holes.
3. The *Points from Surface Holes* dialog will appear (Figure 8-67).

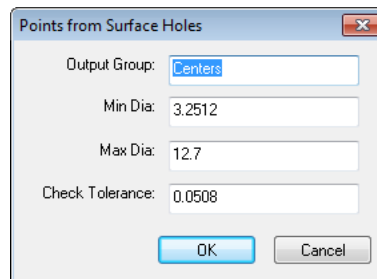


Figure 8-67. The *Points from Surface Holes* dialog.

- **Output Group.** The name for the destination group.
 - **Min Dia.** The minimum diameter hole for point construction. Holes smaller than this diameter will not have center points constructed.
 - **Max Dia.** The maximum diameter hole for point construction. Holes larger than this diameter will not have center points constructed.
 - **Check Tolerance.** This describes the accuracy of the diameter check. Smaller values may take incrementally longer, but will more accurately check the diameter of the hole. Typically, you need not adjust this value unless you encounter unexpected behavior.
4. Select the desired options, then click **OK**.

Laying Out Points In The View By Clicking

Primarily intended for training and testing purposes, SA enables you to construct points by clicking anywhere in the graphical view. These points will be created on a plane perpendicular to the viewing direction (Figure 8-68).

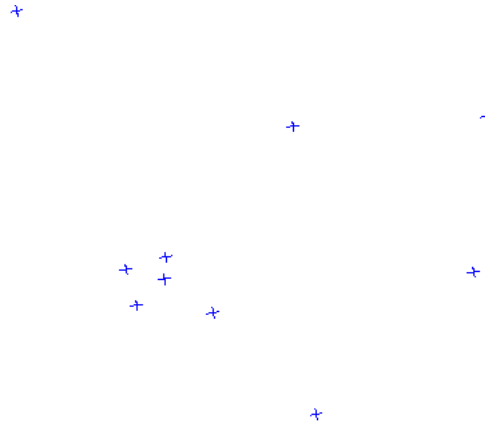


Figure 8-68. Random points laid out in the view.

To Construct Points in the Graphical View by Randomly Clicking:

1. From the menu, select **Construct>Points>Layout>In View By Clicking**.
2. Enter a name for the destination point group.
3. Specify a name for the starting point (subsequent point names will be incremented).
4. Click anywhere in the graphical view to create each point. Press **Enter** when finished.

Copying Existing Points



Note: Any copied measurements become constructed points.

Any set or subset of existing points can be easily copied. The copies are placed into a new point group.

To Copy Points from Existing Points:

1. From the menu, select **Construct>Points>From Existing Points>Copy**.
2. Select the points that you want to copy.
3. When prompted, type a destination group name, then click **OK**.

Moving Existing Points Into A New Group

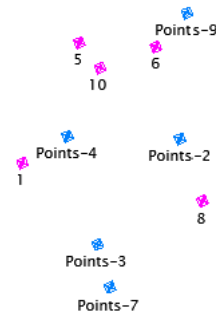


Note: This command does *not* physically move points, as the name might imply.

You can quickly and easily relocate a set of points into a new group (Figure 8-69). The points may start in the same group or separate groups, but they will always end up in the same group. The points are relocated (not copied), which means that measurements remain as measured points after the command is executed. After the command,

the point names will carry a prefix indicating their source group.

Figure 8-69. Relocating points into a new group.



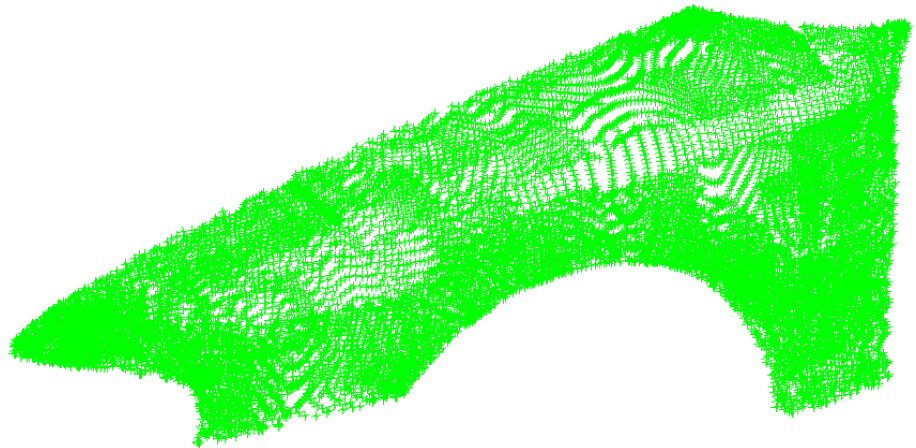
To Move Existing Points to a New Point Group:

1. From the menu, select **Construct>Points>From Existing Points>Move**.
2. Select the points you want to move.
3. At the prompt, enter a name for the destination group, then click **OK**.

Creating a Subset of Points With Greatest Spacing

It is not uncommon to be confronted with point data that is more dense than is necessary for a certain operation (Figure 8-70).

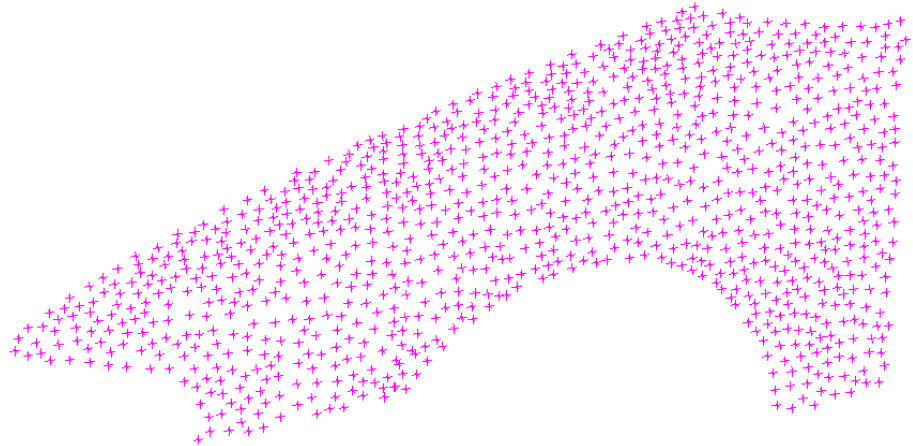
Figure 8-70. An overly-dense point group.



SA can sample an existing point group and create a new group of points at a lower spatial density than the source group (Figure 8-71). You can specify the desired number of points, and SA will choose a subset of points spread out to match that number.

The original points are untouched. Any subsampled points are a copy of the originals.

Figure 8-71. A subset of the original point group.



To Create a Subset of Points with Greatest Spacing:

1. From the menu, select **Construct>Points>From Existing Points>Subset with Greatest Spacing**.
2. Specify the number of points to create.
3. Enter a name for the destination point group.

Auto-Corresponding Two Groups By Proximity

Given a group of reference points and a group of roughly-aligned corresponding points, this command will copy the corresponding group and automatically rename the copied points to match the names of the reference points.

To Auto-Correspond Two Groups by Proximity:

1. From the menu, select **Construct>Points>Auto-Correspond 2 Groups>Proximity**.
2. When prompted, select the reference group of known point names.
3. Select the group to copy and rename. The **Autocorrespond Points** dialog will appear (Figure 8-72):

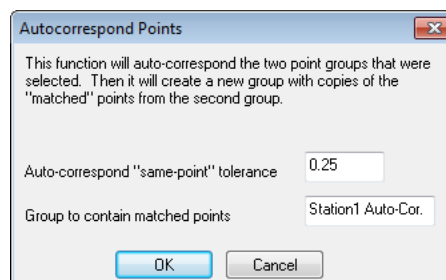


Figure 8-72. Setting parameters for an auto-correspond operation.

- **Auto-correspond “same point” tolerance.** Each point must be within this proximity to a reference point, or it will not be renamed.

- **Group to contain matched points.** The name for the copied group with the corrected point names.
- 4. SA will propose new names for your review. Click **Accept** to create a copy of the source point group and assign the appropriate point names.

Auto-Corresponding 2 Groups by Inter-Point Distance

Given a set of reference points and a group of source points, this command copies the source group and renames the points based on the reference point names. This command does not require the source and reference groups to be roughly aligned.

To Auto-Correspond Two Groups by Inter-Point Distance:

1. From the menu, select **Construct>Points>Auto-Correspond 2 Groups>Inter-Point Distance**.
2. Select the reference group.
3. Select the group to be copied. The *Autocorrespond Points* dialog will appear (Figure 8-72).
 - **Auto-correspond “same point” tolerance.** Once “aligned”, each point must be within this proximity to its reference point to be considered a match.
 - **Group to contain matched points.** The name for the copied group with the corrected point names.
4. SA will propose new names for your review. Click **Accept** to create a copy of the source point group and assign the appropriate point names.

Averaging Points

You can create a single averaged point from a set of points.

Figure 8-73. The average of 5 selected points.



To Average a Set of Points:

1. From the menu, select **Construct>Points>Fit to Points**.
2. Select the points to average, then press **Enter**.
3. The *Construct New Point* dialog appears (Figure 8-29).
4. Specify the name for the resulting point—leave the calculated coordinates alone—then click **OK**.

Creating Points at the End of a Line

You can construct a point at either or both ends of a line by clicking near the desired end of the line.

To Construct a Point at the End of a Line:

1. From the menu, select **Construct>Points>End of a Line**.
2. The *Point Name* dialog will appear (Figure 8-74).

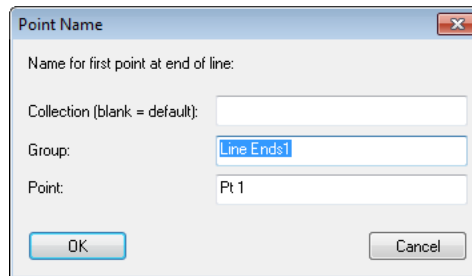


Figure 8-74. The *Point Name* dialog.

3. Specify the name for the first created point, then click **OK**.
4. Click near the end of a line to create the point. When finished creating points, press **Enter**.

Creating Points at a Plane Cross Section:

This command utilizes the database order of a given set of points to create a new set of points at a cross section of a plane.

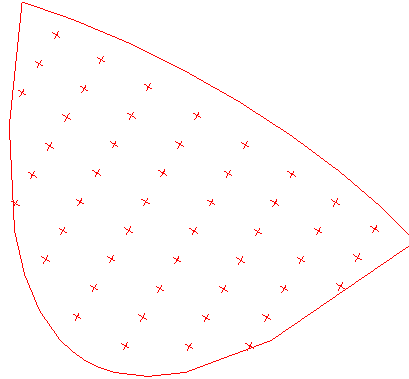
To Construct Points at a Plane Cross Section:

1. From the menu, select **Construct>Points>Plane Section Crossing**.
2. When prompted, select the *Group of Points*.
3. Select the *Section Plane*.

Constructing Points Evenly Spaced on a Surface:

Given a surface, this command allows you to lay out a grid of evenly spaced points along both the U and V directions (Figure 8-75).

Figure 8-75. Points laid out in ten divisions along U and V.



To Construct a Grid of Points Evenly Spaced on a Surface:

1. From the menu, select **Construct>Points>Mesh Surfaces in UV**.
2. When prompted, select the appropriate surface(s), then press **Enter**.
3. The **Make Points From Surface** dialog will appear (Figure 8-76).

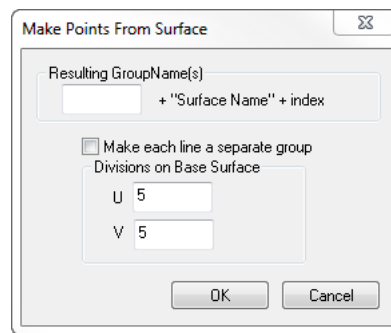


Figure 8-76. The **Make Points From Surface** dialog.

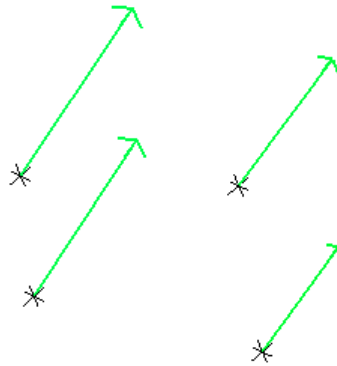
Note: The resulting point group name will be the prefix specified + "Surface Name" + index.

4. Specify a prefix for the resulting point group.
5. If you would like each row to be in a separate group, check the **Make each line a separate group** option.
6. Specify the number of divisions along the U and V directions, then click **OK**.

Constructing Points From Vectors

This command will create a point at the origin of each selected vector (Figure 8-77).

Figure 8-77. A point is constructed at the origin of each vector.



To Construct Points from Vectors:

1. From the menu, select **Construct>Points>From Vectors**.
2. When prompted, select the appropriate vector(s), then press **Enter**.
3. The **Input** dialog will appear (Figure 8-78).

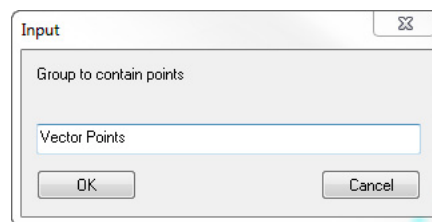


Figure 8-78. The **Input** dialog.

4. Specify a name for the point group containing the resulting points, then click **OK**.

Constructing a Single Point through Patch Normal Shift

This command allows you to project a single point to a defined set of patch points. A theoretical plane is created from the patch points and the end result is a constructed point at a specified offset along its probing direction (Figure 8-79).

Figure 8-79. A point constructed from a selected point and shifted 0.75" along the probing direction of the theoretical patch point plane.



To Construct a Point by Shifting along Normal:

1. From the menu, select **Construct>Points>Patch Normal Shift>Single Point** or press **Ctrl + M**.
2. When prompted, select the point, then select the patch points you wish to use. Press **Enter**.
3. The **Add Shifted Point** dialog will appear (Figure 8-80).

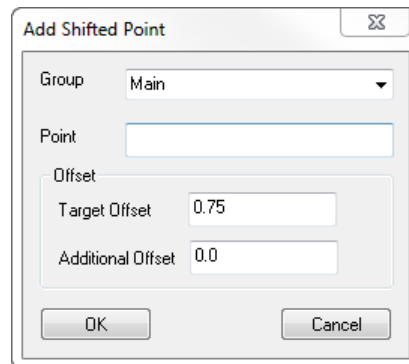


Figure 8-80. The **Add Shifted Point** dialog.

Note: The target offset of the selected initial point will populate in the dialog by default.

4. Specify the point group and point name of the resulting point, as well as the target and additional offsets (if applicable).
5. Click **OK**.

Scale Points

This command build a copy of the selected points scaled relative to the working frame.

Copy Groups Excluding Obscured Points

This command builds a new point group from the selected group including only points directly visible from the instruments base frame. It excludes points on the far side of surfaces.

Auto Filter to Faces

Creates a new set of points, groups, or clouds based on the proximity of points to a selection of surfaces.

A selection of useful controls are available which include:

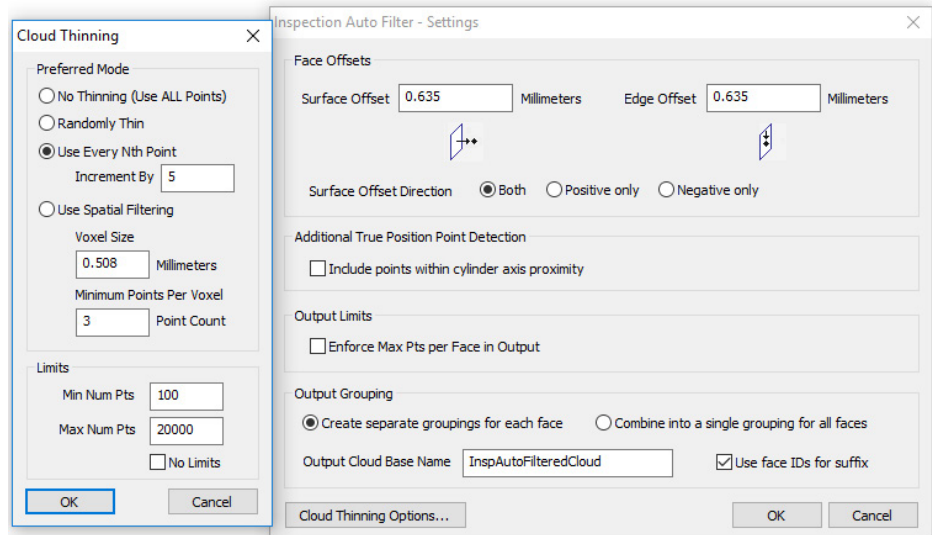


Figure 8-81. Auto Filter to Surface Faces Dialog.

- **Surface Offset.** The necessary proximity to the surface
- **Edge Offset.** The required distance from a surface edge, which allows removal of points too close to an edge.
- **Surface Offset Direction.** Which can be set to both above and below the surface or in only a single direction.
- **Output Limits.** This control ensures that no more than the specified number of points are included per face.
- **Output Grouping.** This selection determines if a single point cloud is built or if a cloud per surface face is returned.
- **Cloud Thinning Options.** A list of cloud thinning controls are also available including randomized thinning, Nth point thinning, and min/max limits.

When combined with the user of cloud based geometry relationships this command makes it very easy to extract features from a cloud using the available CAD to filter the cloud. For more information on geometry relationships refer to “[Minimizing Relationships](#)” on page 444.

A variation on this same command can be found under **GD&T>Feature Inspection Auto Filter**. This command displays the same dialog and selection of controls but does the following:

1. Detects which surface faces have **both** annotations attached and valid features checks built.
2. It filters the data using the specified criteria to those faces
3. Associates the newly built points with the corresponding feature checks.

From Surface Faces

This command currently builds a single point at the proximal extent of the cylinder it defines. Much like **Construct>Circles>From Surface Faces**.

Closet points between 2 Surfaces

This command compares the provided surfaces and builds 2 points on the surfaces at the closest point locations on those surfaces, displaying the minimum distance between them.

This Page Intentionally Left Blank.

Spatial Analyzer provides a lot of flexibility when it comes to defining a base coordinate for a job file. Frames in SA refer to both coordinates systems and any defined 6D location in space or measurement. “Frames” may also be called: Reference Systems, Coordinate Reference Systems, Coordinate Reference Frames, Frame of References, or 6D locations.

6D Locations and Measurements

As noted in the SA Fundamentals section, frames are the foundation for any measurement job. In order to define both the position and orientation of an object you need both a base reference point and orientation. A coordinate system or coordinate frame is a common way to report this information and is a fundamental piece of any computer aided graphical design or inspection software.

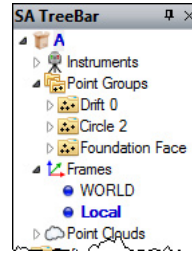
Coordinate reference frames and measurement frames are interchangeable within SA.

The Working Frame In SA

SpatialAnalyzer allows you to create a virtually unlimited number of coordinate frames (referred to simply as frames in SA), each representing a unique position and orientation within the 3D graphical environment.

Frames are contained within their own category in the SA TreeBar (Figure 9-1).

Figure 9-1. This collection has two coordinate frames, **World** and **Local**. The **Local** frame is active (working).



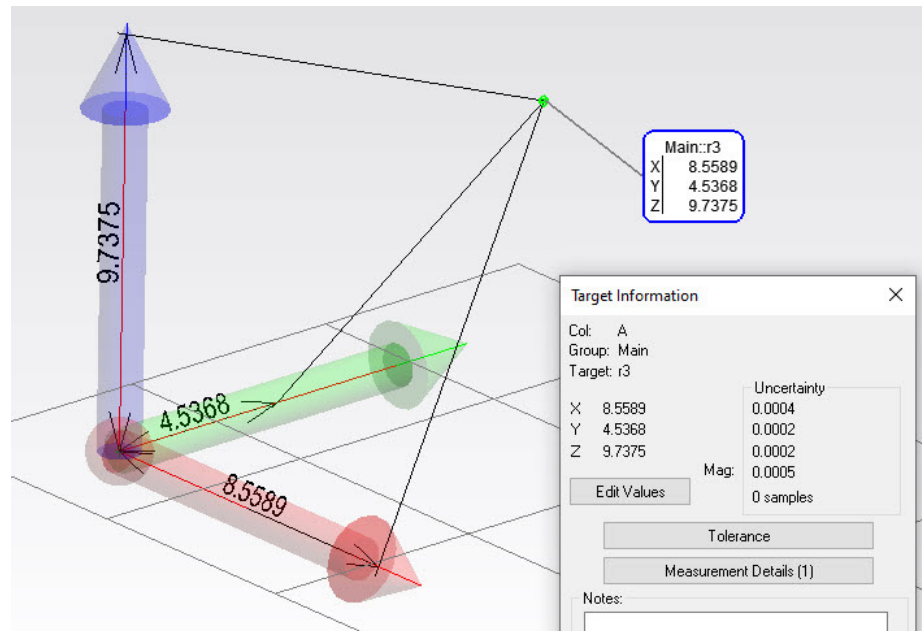
While there can be many coordinate frames in a given SA job, only one can be designated as the *working* frame. When a frame is designated as the working frame, it is shown in bold blue letters in the tree and listed at the bottom of the SA window just after the SA version and active color selection (Figure 9-2).



Figure 9-2. The Working frame selection control at the bottom of the SA graphics window.

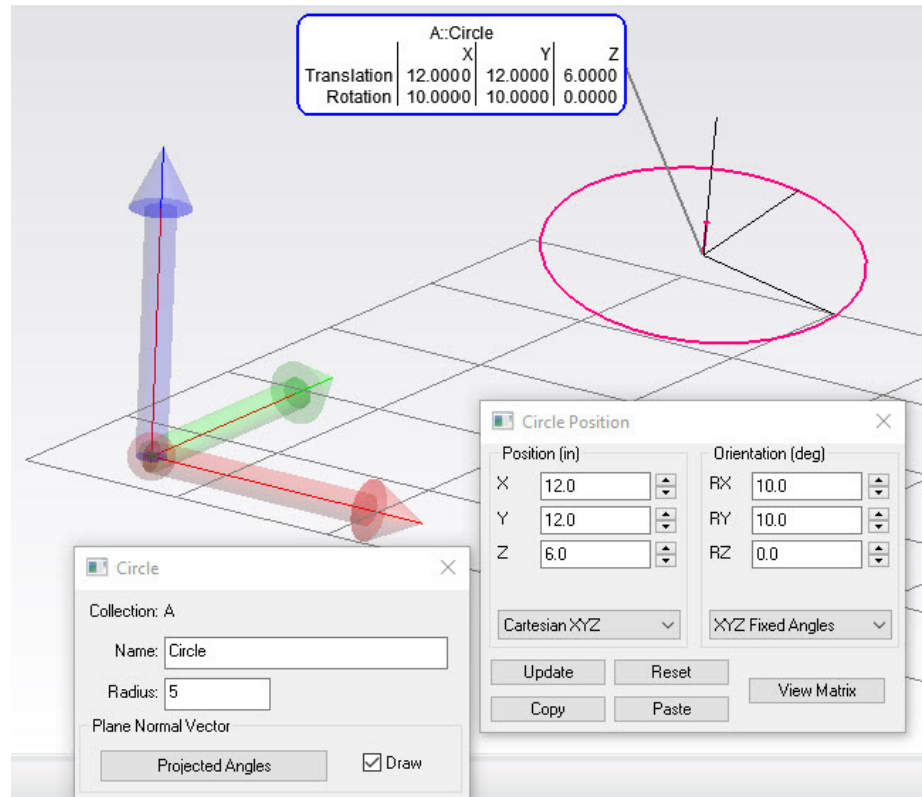
All coordinates, query results, and other values are always displayed relative to the active coordinate frame (Figure 9-3), unless explicitly specified otherwise. For example, when you view the coordinates of a point, you are seeing the coordinates relative to the active coordinate frame.

Figure 9-3. Working Frame is used to define the position of any point in 3D space and is the foundation for XYZ cartesian reporting.



The position and orientation of any object within SA is also defined by the working frame in the job. Each object, even a plane or cylinder for example has a base frame that defines its origin and orientation. Figure 9-4 shows the base frame of a circle. its position and orientation can be seen by opening the circles properties and looking at its Transform.

Figure 9-4. The transform of an object such as a circle is reported based on the objects base frame.



To Activate a Coordinate Frame:

Any frame in the job file can be set as the working frame at any time. This will change the base reference system for the job and update the XYZ positions reported for all other items. To change the working frame do one of the following:

- Right-click a frame in the graphical view or SA tree and select **Make Working Frame**.
- Click the active frame button at the bottom of the graphics window and select the frame you wish to set as working.
- From the menus, choose **Edit>Pick Working Frame** and select the frame to activate.
- Press **Alt+W** (default hotkey) and choose the desired frame from the **Object Selection** dialog.

Cylindrical Coordinate Systems and Spherical Coordinates

SA does not have a concept of a cylindrical or spherical coordinate system on a job level. Instead, what SA offers is the ability to edit the **Reporting Options** within the properties of the item you wish to edit. For example if you want to report the cylindrical position of a set of points measured around a circle you can build a local frame centered on the circle and go to the properties of the point group and select

that frame as the reporting frame for the group. This allows you to have complete control over the way in which a point group is reported. For more details refer to the Reporting Chapter.

Frame Properties

Frames, like virtually all entities in SA, have properties associated with them.

To View a Coordinate Frame's Properties:

Do one of the following:

- Right-click the frame in the SA tree or graphical view and choose **Properties**.
- Double-click a frame in the SA tree.
- Click the **Object Properties** icon in the **Main Toolbar**, then double-click a frame in the graphical view (Figure 9-5).

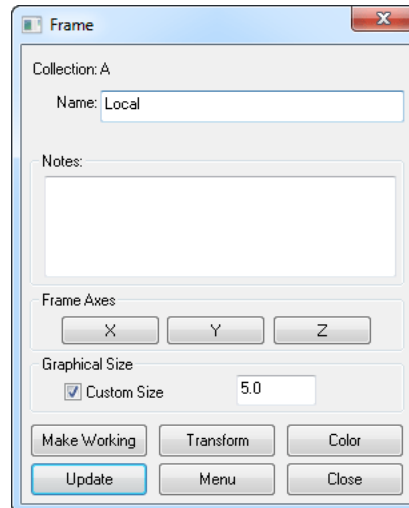
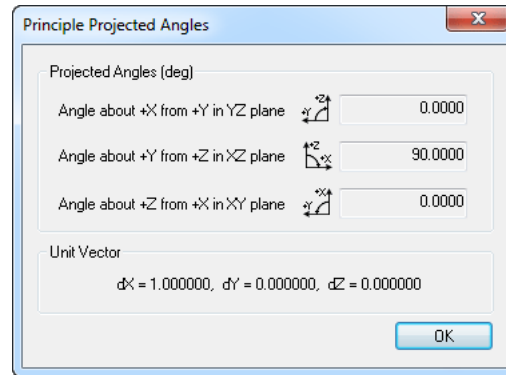


Figure 9-5. The Frame properties dialog.

- **Collection.** Indicates the collection in which the frame resides.
- **Name.** The name for the coordinate frame.
- **Notes.** The notes field allows you to add text-based notes to a coordinate frame, which you can reference at a later time.
- **Frame Axes.** By clicking on the X, Y, or Z buttons, you can view the projected angles for the selected coordinate axis, presented relative to the working coordinate frame (Figure 9-6).

Figure 9-6. A frame's principle projected angles.



Projected Angles can be reported for most any object, but unlike frames, these default Projected Angles are reported for the Z axis of that object only. Projected angles represent the 2D component of the vector or line as projected onto the respective planes of the reporting frame (Figure 9-7).

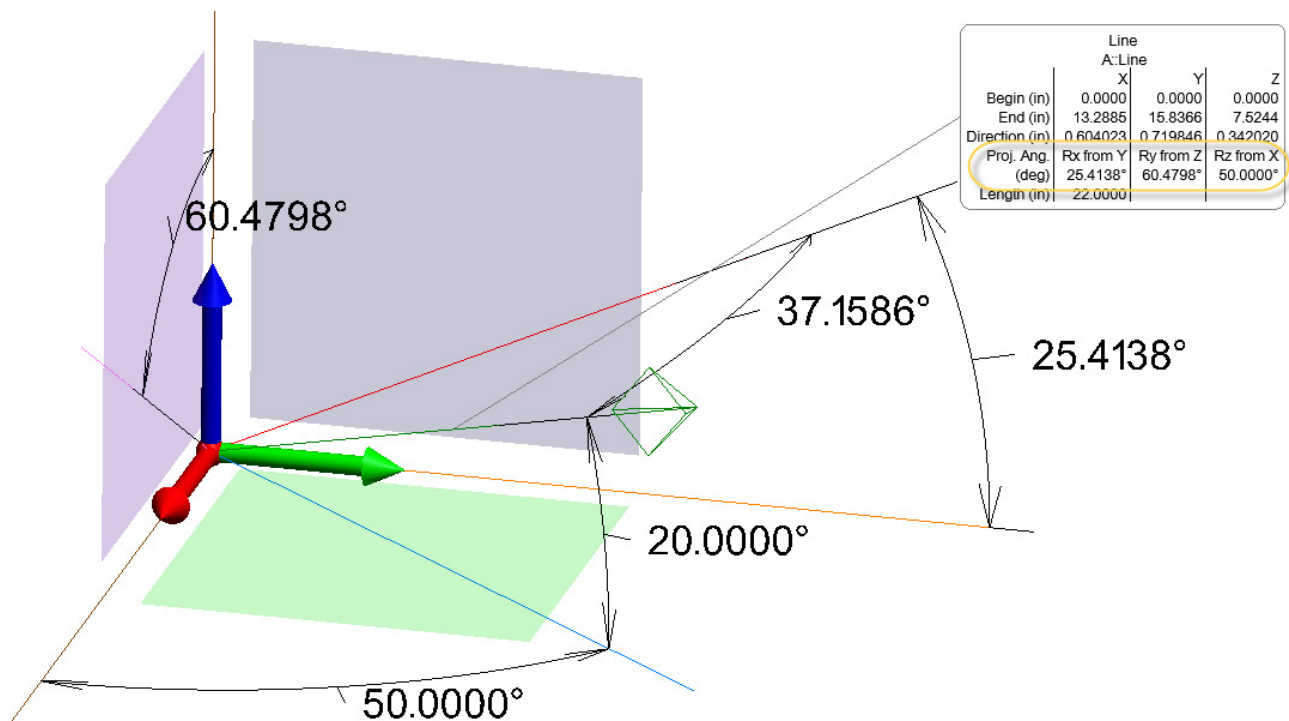


Figure 9-7. Visualizing Projected Angles for a line or axis

- **Custom Size.** Enable this option to assign a custom size to the coordinate frame, and enter the size in the resulting text field.
- **Make Working.** Sets the coordinate frame active so that it becomes the working coordinate frame.
- **Transform.** Opens the transformation dialog to adjust the position and orientation of the frame.

Note: The assigned color of the frame is only shown when it is not the working frame.

- **Color.** Opens the color dialog so that you can assign a new color to the coordinate frame.
- **Update.** Updates SA with any changes in the object properties without actually closing the **Frame** dialog. Note that closing the dialog also applies changes to SA.
- **Menu.** Exposes object-specific menu options. This is equivalent to right-clicking the object in the tree.
- **Close.** Closes the properties dialog.

Frame Transform Representation

The position of a frame (or object) in SA is saved using the full four-by-four matrix which can be seen from the properties dialog.

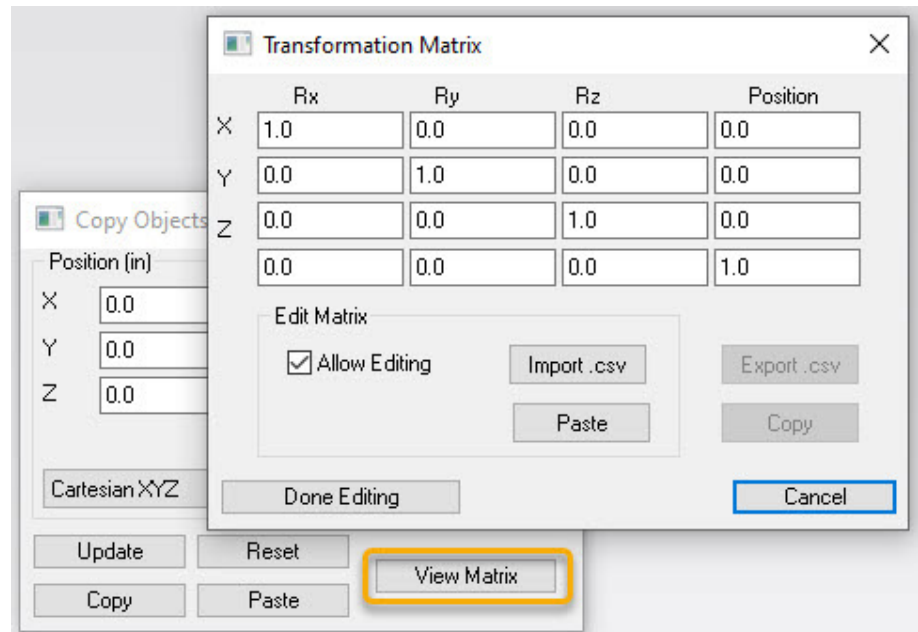


Figure 9-8. The View Matrix button allows you to see the full 4x4 matrix used to the position of a frame or object.

Angular transform components can also be reported in several different way:

Fixed Angles:

- **XYZ Fixed Angles.** SA's default reporting method. Fixed angles provide a sequential rotation reference where rotations are performed in order using a fixed reference.
- **ZYX Fixed Angles.**

Euler Angles:

- **XYZ Euler Angles.** Euler angle references use a sequential reference, in this case X, then Y from the new position, and then Z, much like an airplane would make sequential turns.
- **ZYX Euler Angles.**

- **ZYZ Euler Angles.**
- **ZXZ Euler Angles.**

Additional Formats:

- **Equivalent Angle Axis.** Where KX, KY, KZ are unit vectors that define an axis and Θ is the rotation about that axis.
- **Euler Parameters.** The four parameters $e_0, e_1, e_2,$ and e_3 describing a finite rotation about an arbitrary axis.

Frame Wizard

The Frame Wizard (**Construct>Frame>Frame Wizard**) is an all in one construction tool used to create a coordinate frame based on different combinations of entities, which can then be optionally offset in the new frame's X, Y, and Z directions. It is also the foundation for GR-Frame construction.

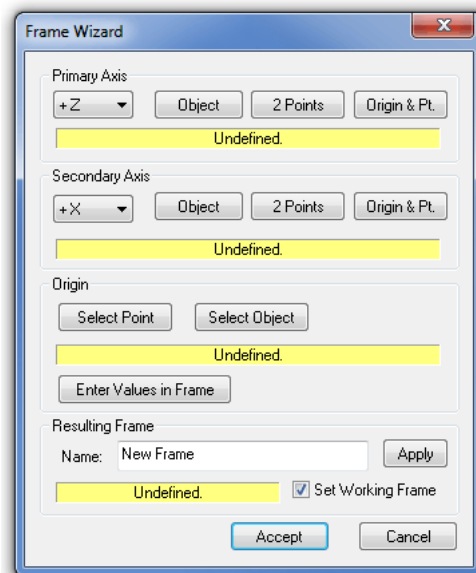


Figure 9-9. The Frame Wizard dialog.

The *Frame Wizard* dialog (Figure 9-9) allows you to interact with the graphical view while changing the settings for the frame. Once enough parameters have been defined to construct a coordinate frame, you see the frame in the view and it changes as you modify the settings.

The following describes the specific settings for this command:

- **Primary Axis.** The axis to use as the primary axis for the coordinate frame. Since this axis is primary, it is perfectly parallel to the vector constructed by its defining entities. The primary axis and secondary axis must make sense—for example, you cannot

define the +Z axis as primary and the -Z axis as secondary.

- **Secondary Axis.** The axis to use as the secondary axis for the coordinate frame. The plane defined by the primary and secondary axes will be parallel to the vector constructed by the secondary axis's defining entities.
- **Origin.** A single point in space that (assuming no offset) defines the origin of the coordinate frame.
- **Enter Values in Frame.** Click this button to define a new XYZ coordinate. The coordinate frame will be shifted such that the originally defined origin point ends up at the supplied coordinates.
- **Name.** The name for the resulting coordinate frame (in the active collection). To change a name, type it in, then click the **Apply** button.
- **Set Working Frame.** If checked, the new coordinate frame will be set as the active coordinate frame.

Figure 9-10. A frame defined by an origin at point Origin, a primary axis (+Z) by a cylinder, and a secondary axis (+X) by the origin and a point named Clocking. No offsets have been applied.

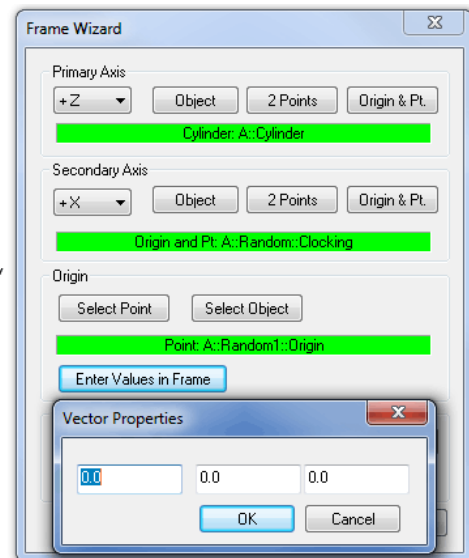
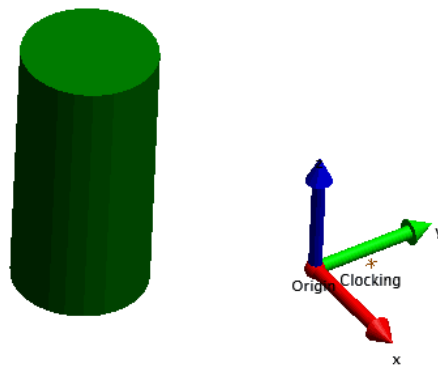
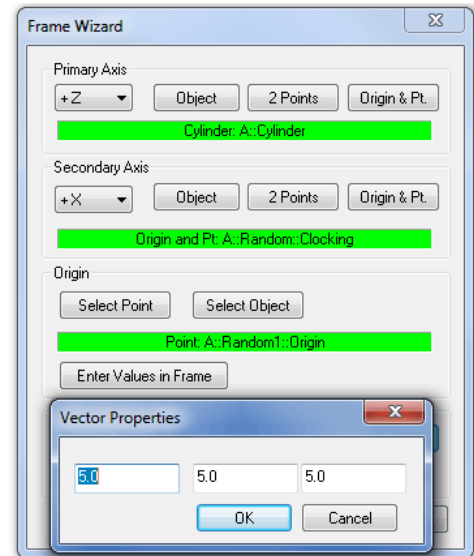
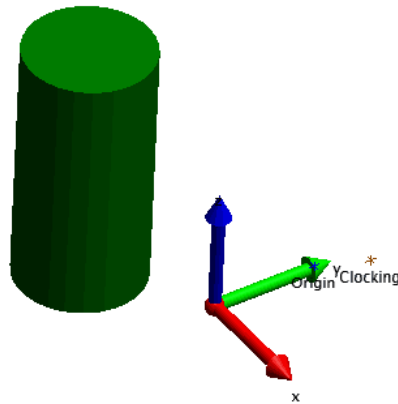


Figure 9-11. The same frame which has been shifted using Enter Values in Frame so that the origin ends up at (5, 5, 5).



The primary and secondary axes can be defined by an object, 2 points, or an origin and point:

- **Object.** The object's internal normal direction defines the direction of interest. For example, if a plane is selected, the normal to the plane is used. Note that some objects (such as spheres) do not define a direction and therefore cannot be used with this option.
- **2 Points.** The vector from the first point to the second defines the direction of interest.
- **Origin & Pt.** The vector from the defined origin to the selected point defines the direction of interest. An option is also available to select an object. When this is done the selected Object's origin point is treated as the point definition.

The origin can be defined by either an object or a point:

- **Select Point.** A single point will define the position of the origin. Point offsets are not applied.
- **Select Object.** An object's internal origin defines the origin of the frame. For example, if a sphere is selected, the center of the sphere is used.

Working with CAD Models Directly

In addition to the object and point select options a right-click on an **Object**, **2 Points** or **Origin & Pt.** button opens up a list of options to directly extract and build those objects or points by selecting surface faces on the CAD model directly.

To Create a Frame with the Frame Wizard:

1. Select the appropriate primary and secondary axes to define.
2. Click the appropriate buttons to define entities for the primary axis, secondary axis, and origin. These entities can be defined in any order.
3. If desired, enter coordinates to shift the frame using the **Enter Values in Frame** button.
4. Type a desired name for the frame and click **Apply**.
5. When finished, click the **Accept** button.

Additional Frame Construction

Spatial Analyzer offers many methods for constructing Frames. To fully understand frame construction it is important to understand the components needed to construct a frame. A frame has four basic components: an origin point, primary axis, secondary axis and tertiary axis. A frame can usually be constructed by defining an origin point and two of the axes, and the third axis becomes fully defined as a result. The simplest and more useful tool for this application is **Frame Wizard** (see section below).

During frame construction many commands will prompt the user to define the primary, secondary and tertiary axis. An axis can be described by a point or a vector (object direction). When using a point to describe an axis, a vector is formed from the origin point and the selected point. If an object is selected its normal will be used to define the axis. The following methods provide several combinations of ways in which points and vectors can be defined and use to build a frame.

Enter

Creates a frame at the origin and orientation of the active frame. The transform dialog will appear allowing the user to enter the position and orientation values.

Plane, Line, Point (3-2-1)

A 3-2-1 frame consists of a locator point for the origin, line/vector for one axis and plane for the second axis ([Figure 9-12](#)). The Plane, Line, Point Transformation dialog allows control over the axis for the plane and line/vector. The user may also specify an alternate coordinate for the locator point.

Figure 9-12. The Plane, Line, Point Transformation dialog.

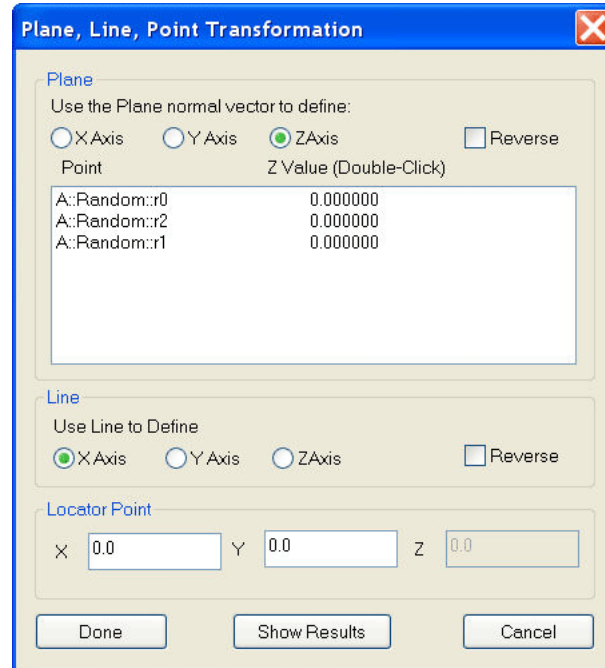
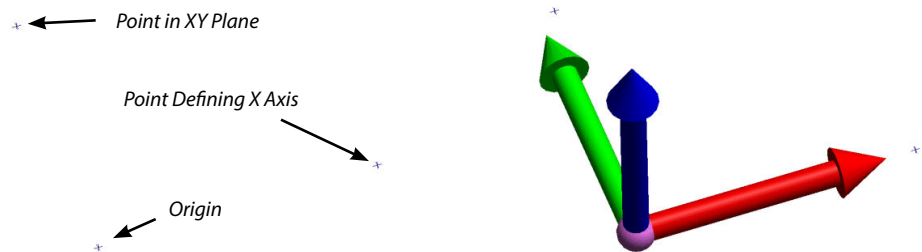


Figure 9-13. Points used to define a 3-2-1 frame (left) and the resulting constructed frame (right).



Plane, Line, Line (Sheet Metal Corner)

Uses a plane and two lines to define a frame. A user can select the axis for each object selected.

Using Points and Vectors

Example using only three points. **Construct>Frame>3 points>Origin, X-axis>Point on XY plane**

3 Points

Uses three point to describe the origin, primary axis and secondary axis.

Z-Axis

Constructs a frame with the Z-axis normal to a plane, circle or a line and uses the selected points for origin and X-axis.

On Instrument

This will construct a frame on the instrument base with the Z-axis being the standing axis. A frame can also be made on the “scope or end effector”. This will generate a frame using the probing direction as the vector for the Z-axis.

Using Working Frame Orientation

Uses the current active frame orientation to define a new frame. A copy can be made by defining a new origin point or maintain the Z-axis and clock the remaining axis.

3 Planes

Uses three planes to create a coordinate system. User selects X, Y and Z planes. The user interface allows control over priority and direction.

Mirror Cube

Often, users wish to construct a coordinate frame based on the faces of a mirror cube (see Figure 9-14). The Mirror Cube frame assignment method provides a convenient method for repeatedly and accurately locating a coordinate frame with two instruments (typically theodolites). SpatialAnalyzer contains a powerful mirror cube frame assignment tool.

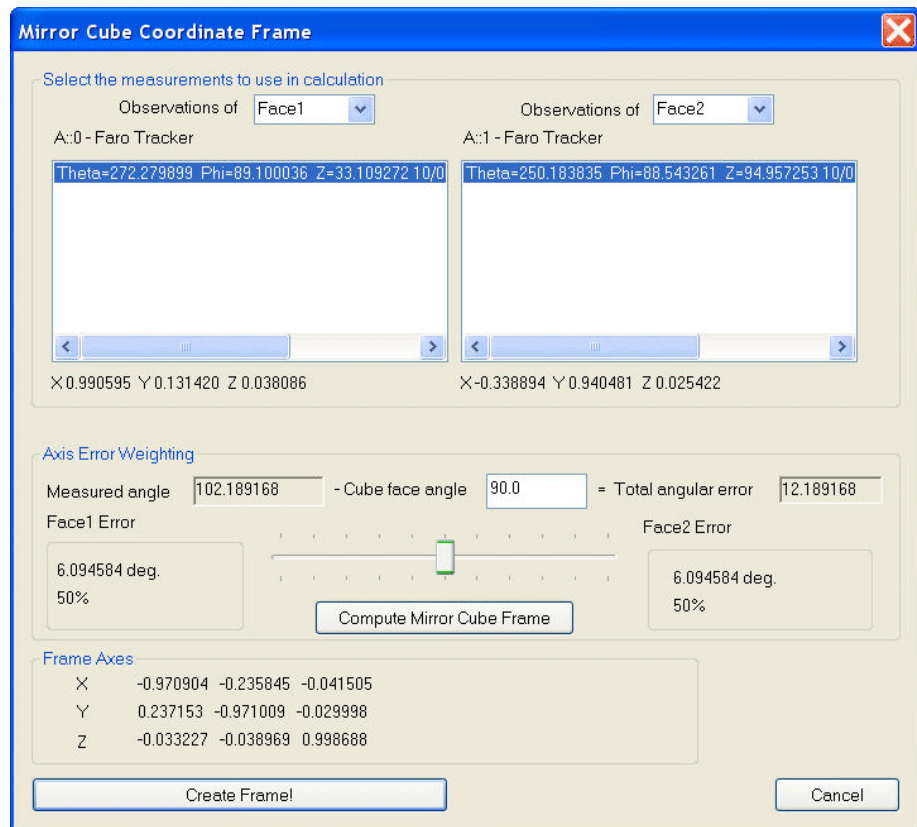


Figure 9-14. The Mirror Cube Coordinate Frame dialog.

To accomplish a mirror cube frame assignment, point two theodolites at the mirror cube and collimate the instruments with the mirror faces. Record a shot from each theodolite at the cube using the same target name, just as if you were shooting the mirror as a target. If you desire reversed shots, reverse the scope and take another shot, or take multiple shots in each face. In SpatialAnalyzer, select Construct, Frame, and Mirror Cube. You will be prompted to select the target for the mirror. Pick the common target name you used for the measure-

ments. The mirror cube frame assignment dialog will appear. At this point, you may press the **Create Frame** button to make the coordinate frame, or you may change the options.

Possible Options:

- You may select which instrument represents the X-axis of the coordinate frame.
- You may select or deselect specific shots from each instrument. When there are multiple shots, their vector directions are averaged to calculate the vector displayed below the measurement box.
- The check box for coordinate axes determines whether the axes of the frame are outward along the theodolite vectors or inward toward the theodolites.
- The Axis weighting box allows you to control the distribution of the errors between the two axes.
- Set the Cube face angle (normally 90 degrees). Press **Compute** after doing this to update!
- The difference between the cube “ideal” angle and the theodolite vector angle will be displayed.
- Use the slider bar to control how far the frame axes are away from the theodolite axes. Sliding the bar to the left places more of the error on the “right-side” instrument.
- Choose carefully when distributing the error. In some cases, it may be better to have no error on one axis since it is difficult to tell if the “error” is coming from the theodolite measurements or the mirror cube itself.

On Object

Constructs a frame with the Z-axis normal to the selected object.

Fit to Coordinate Components

In this method of frame assignment, we assume that we have several points which share a known X coordinate, several points which share a Y coordinate, and several points which share a known Z coordinate. This is often used in a factory setting where a designer has established a reference block with one face representing a certain XY, one plane representing a certain XZ plane, and one face representing a certain YZ plane. Initially the user will be prompted for the points of known X value, then Y and Z in turn. After selection of the points is completed, the user will see the dialog shown below. Enter the known values and hit **OK**. The frame assignment will be made.

Average of Other Frames

Creates a frame based on the average of selected frames. Usually used

to create an average between gravity frames.

Copy and Make Left-Handed

Makes a copy of the selected frame and prompts for the axis to reverse.

Frame Sets

Frame Sets provide a merged frame format that allows many individual frames to be saved as part of a single object, much in the same way point clouds record the point information. Using Frame Sets provides the advantage of keeping the tree manageable and allowing many more frames to be measured in a single job file at faster rates (Figure 9-15).

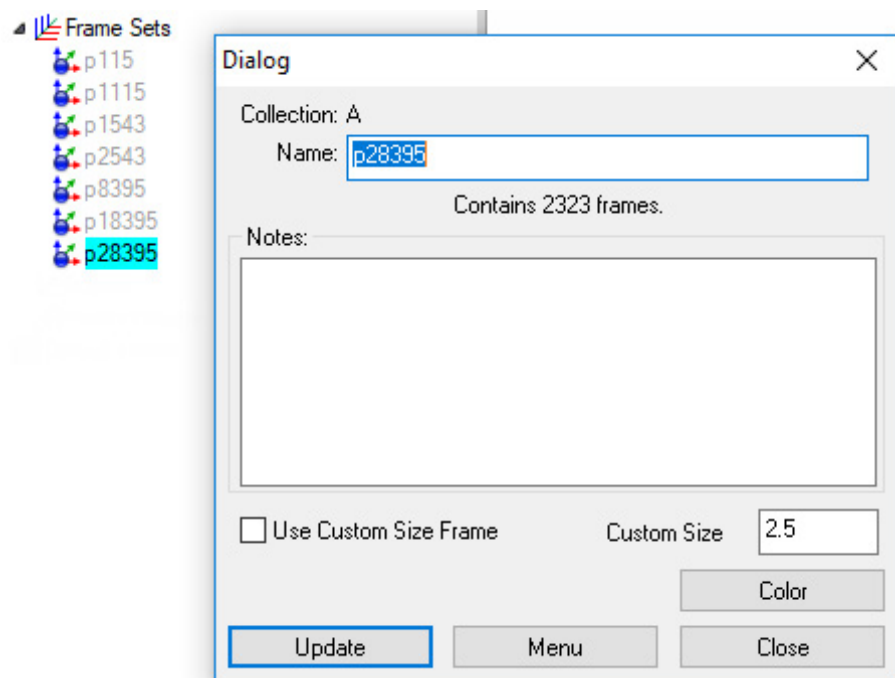


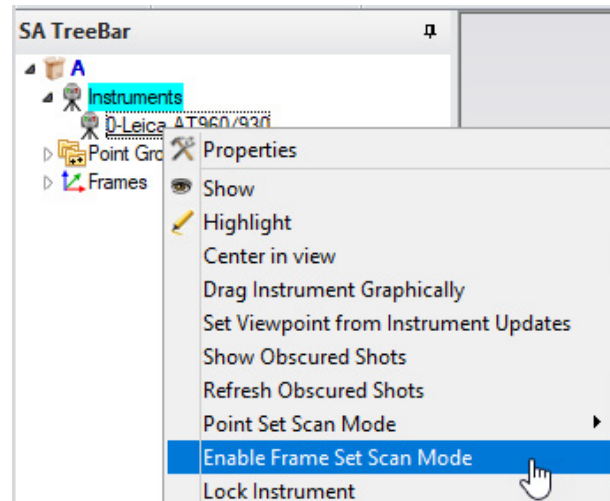
Figure 9-15. Frame Set Properties dialog

Measuring with Frame Sets

Frame Sets currently can only be recorded as part of a 6D Scan. Single Discrete measurements will still create individual frames.

This method of data capture is an option that must be enabled. In 2023.1 the control for this mode has been added to the instrument's right-click menu, which allows different instruments to be handled independently (Figure 9-16):

Figure 9-16. Enable Frame Set Mode must be enabled to record Frames in Frame Sets



The global default for newly added instruments will take on the value set for the last instrument configured. Instruments for legacy job files will be set as per the global setting stored to the job file.

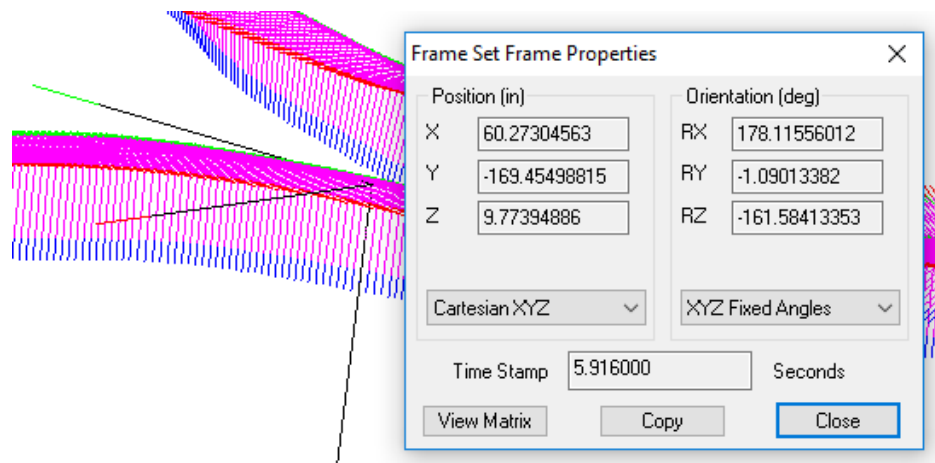
In prior versions this option was only available through the right-click Frame Category menu.

Individual Frame Selection

Double left clicking on a single frame in the graphics provides the ability to select a single frame from within the frame set. Doing so provides direct access to the transform and time stamp recorded with that particular frame measurement.

Time stamps are also recorded as a fundamental property of a frame set, when available from the instrument, and reflect the instrument's own controller time. This controller time is quite accurate and reflects the time from the beginning of the scan.

Figure 9-17. Individual Frame Selection from within a Frame Set



By pressing **Copy** the transform is saved and using **Construct>Frames>Enter** then provides the ability to use **Paste** in the frame transform to easily construct an exact duplicate of any individual frame in the frame set.

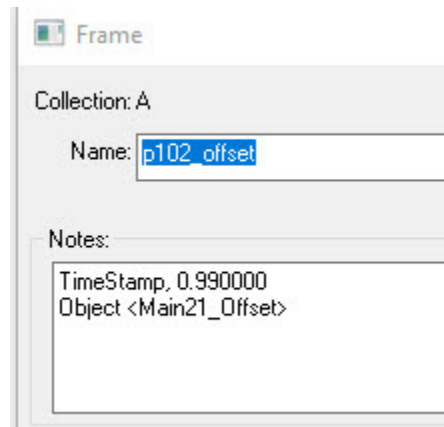
Full Ascii Predefined format Frame support is also available in both import and export for frame sets.

Recording Multiple Frame Sets At Once

In 2023 the ability to record multiple frame sets at once was added. This is particularly helpful for photogrammetry systems that are recording the 6D transform of several objects at one time. In order to ensure the correct frames are added to the correct set in the tree, a naming convention must be used.

SA expects the use of the object notes to drive the placement of incoming frames in the correct frame set. The current standard is to use the first line in the frame set's notes for a timestamp and the second line for the name of the destination frame set in the SA tree (Figure 9-18).

Figure 9-18. Expected Structure in the properties of a frame to ensure it is entered into the correct frame set when measuring multiple sets at one time.



In the above frame the structure includes:

- **Name.** The name of the frame saved within the Frame Set (in this case the name is "p102_offset")
- **Notes (Time).** The first line in the notes category is used to define the timestamp from when the scan began (TimeStamp, 0.990000).

Note that if an entry with a timestamp of zero is encountered, this will trigger the renaming of the current frame set container and the creation of a new one taking on the name of the object as specified in the second note entry.

- **Notes (Object).** The second line defines the name of the frame set within which the frame will be recorded. Use the following format (Object<Main21_Offset>) to record a new frame using the frame name in the "Main21_Offset" frame set.

Anything that has both a position and orientation in space—with the exception of instruments—is considered an object in SA. In this chapter, we'll discuss how to work with objects, and what you can do with them.

Locations and Measurement

For more detail on points and point measurements refer to the Chapter on “Points” on page 193.

Point Group

When points are created (constructed or measured), they are placed into point groups. This grouping added an additional layer to the tree structure. Unlike other geometry such as planes and cylinders, point groups have no specific purpose in themselves but act as “containers” to hold points, and provide an additional layer of organization.

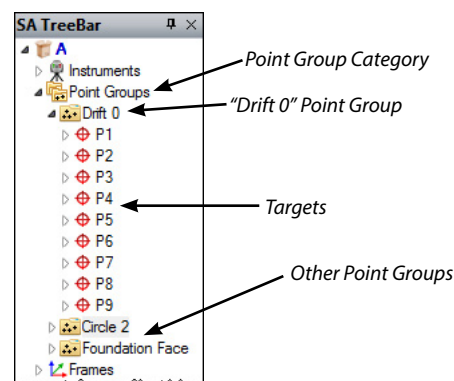


Figure 10-1. Point groups as they appear in the tree.

Point groups are listed under the *Point Groups* category in the treebar (see Figure 10-1). Each individual point group can be expanded to reveal the contained points by clicking the disclosure triangle next to the point group name. Click the triangle again to collapse the list of points.

A point group is an object, whereas an individual point is not. As such, the point group can be moved, scaled, rotated, mirrored, etc. When a point group is manipulated in this manner, all of the points inside the group are treated as a rigid body.

Points within the same point group are all the same color.

A point group also offers a very convenient means to access and select a set of points.

Point Group Reporting Options

Points can be reported in a number of different formats. These include:

Coordinate System

You can choose the way in which the coordinates of points are reported using the following 3 options:

- **Cartesian (X/Y/Z).** SA's default reporting method is to report the coordinates of points in cartesian XYZ coordinates relative to the working frame.
- **Cylindrical (R/T/Z).** Points can be reported using a cylindrical coordinate system in which the position are reported using a radial distance from the cylinder axis, a theta or clocking angle and a z value defining the height of the point along the axis.

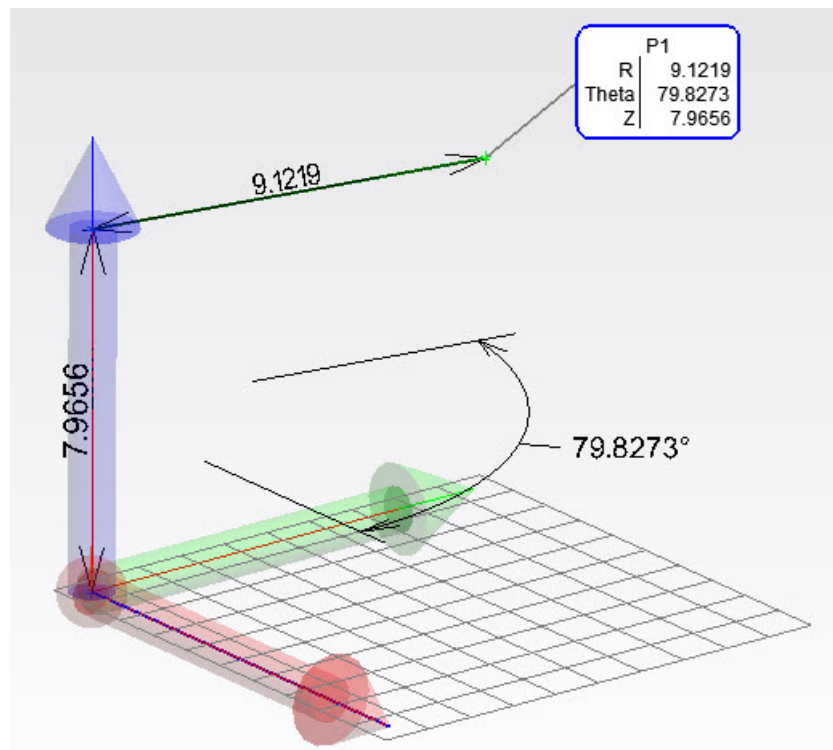


Figure 10-2. Example of cylindrical reporting for a point coordinate

- **Spherical (R/T/P).** Points can be reported using a full spherical representation as well. With this selection point locations are

a distance R from the origin of the frame, an angle Θ from the X -axis and an angle Φ from the XY plane.

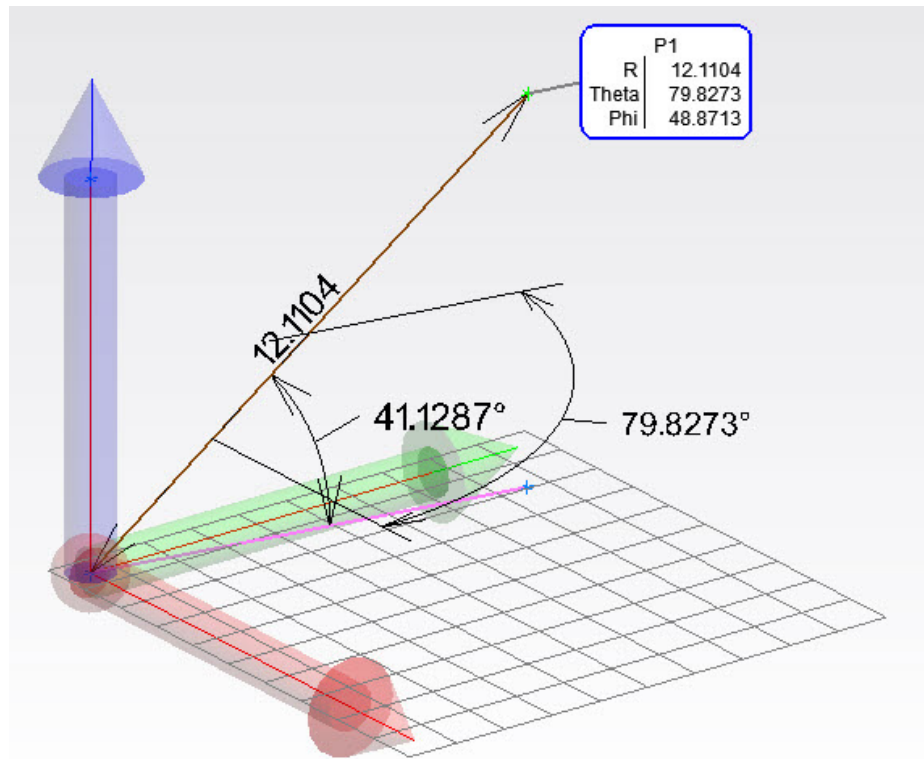


Figure 10-3. Example of spherical reporting for a point coordinate

Components to Report

The individual components can be turned on or off as needed.

- **X.** The X component which also corresponds to R in cylindrical or spherical display.
- **Y.** The Y component which also corresponds to Θ in cylindrical or spherical display.
- **Z.** The Z component which also corresponds to Φ in spherical display.
- **Offsets.** When enabled the planar and radial offsets saved with the points are included in the report.
- **Uncertainty.** When computed the uncertainty for the point is reported in each coordinate and as a magnitude value. This corresponds to the XYZ, Cylindrical or Spherical reporting system selected and reports the point uncertainty with respect to the instruments base.
- **Notes.** Lists any notes added by a user.

Measurements (if available)

- **Show Measurements.** When selected, the contributing measurements are reported. This can be quite helpful for theodo-

lites or total stations where multiple observations were taken of the same point. Observation 1 (Theta), 2 (Phi), and 3 (Radius), are displayed as well as the average point error and observation status.

- **Show Meas Details.** Enabling this option allows you to include the full measurement details string from each observation.

Theodolite Information

- **Show RMS Angle, Pointing Error and Worst Apex Angle.**

Format

- **Sort by Point Names.**

These settings can be set as default for the job file or applied to other point groups using the Default Settings buttons at the bottom of the Report Options dialog.

Objects vs. Geometry Relationships

Spatial Analyzer provides the capability to build both static and dynamic points and geometry. For example, a cylinder can be constructed from a point at a defined location with a preset length and diameter, or dynamically update as measurements are taken. In order for geometry to dynamically update a relationship is used to connect the list of points measured with the object that is fit to those points. However, the object that is built is the same in each case and will be described here. For more information on Geometry Relationships and dynamic features refer to "[Geometry Relationships](#)" on page 263.

Lines

Properties

- **Collection.** Specifies collection name where object resides.
- **Name.** Object name.
- **End-Point Method.** Enter two point coordinates to determine length, position and orientation of the line.
- **Ray Method.** Control the length or the transformation of the line.
- **Projected angles.** Displays the Principle Projected Angles dialog. These angles represent the projection of the object normal vector onto the coordinate planes of the active frame.
- **Draw.** Displays the normal vector in the graphics.

- **Notes.** Contains information entered by user or information regarding construction method, fit, etc.
- **Reverse.** Reverses the normal direction of the object.
- **Shift Ends.** Enter values to shift beginning and end point along the vector of the line.
- **Color.** Opens color dialog and allows the object color to be changed.
- **Update.** Updates SA with any changes in the object properties.
- **Menu.** Exposes object specific menu options. Emulates right-clicking the object in the tree.
- **Close.** Closes the object properties dialog.

Construction

Fit to Points

Constructs a best-fit line for the selected points. See the Geometry Best-Fit section for more information.

Enter

This will allow you to enter the end-points of the line or designate a direction and a length. These values are entered using the Line Property dialog.

2 Points

Constructs a line between two selected points. The lines direction is from the first to second point.

Mutual Perpendicular

This function will ask for selection of two lines. It will then find the closest point to the intersection between those lines and build a line that passes from one line, through this point, to the other.

Projection, Line to Plane / Circle

After selecting either a plane or a circle as the destination, you will be prompted to select the line you wish to project. A new line will be created on the projection XY plane of the selected object.

Between a Point and a Line

This function creates a line between a user-selected point and a line.

2 Plane Intersection

A line is created at the intersection of two planes. If the planes are parallel, the function reports that the line can not be created.

Center of Cylinder

Constructs a line from the cylinder end points and inherits the cylinder's direction.

Center of Slot(s)

Builds a line defining the axis of the slot. Its extents will define the center points of each radii defining the ends of a rounded slot or connect to the center of each straight end section of a square slot.

Along Frame Axes

Constructs a line down each axis of the working frame.

From Instrument Shot

Creates a line from a measured point and its associated instrument. The point selected must be a measured point.

Normal to Object(s)

Creates a line normal to an object. All objects (plane, line, cylinder, etc) have a normal direction. The line created will represent that normal direction.

Normal to Objects, through Point

Creates a line parallel to the selected object's normal and intersects to selected point.

Parallel to Line, through Point

Creates a line from the XY plane of the reference object, normal to it's Z axis and extending to the selected point.

Cross Product

The cross product is a function that takes two lines (vectors) and produces a line that is perpendicular to them. Imagine selecting two lines that define the Z and X axes of a frame, this function will built the Y axis line.

Bisection of 2 lines

Builds a line defining the average or mid line between the two selected lines.

From a Vector Group

Creates lines from each vector in a vector group.

From 2 Groups (name matched)

Much like a group to group query, this function will build a set of lines (rather than vectors) linking the points in each group with common names.

From CAD Surfaces

This function, like construct circle from CAD surfaces, finds a set of holes in a selected surface within a specified size range. It then builds the hole axes as individual lines.

From CAD Faces

Allows user selection of individual CAD faces. When a cylindrical CAD face is selected, it will build the center axis of the cylinder defined by that surface face.

Circles

Properties

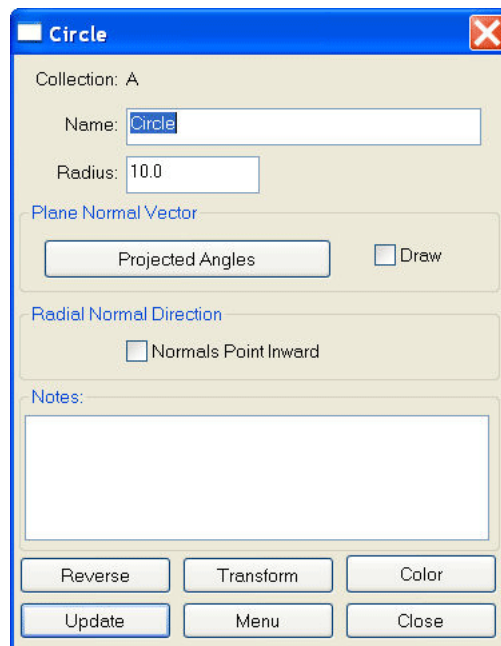


Figure 10-4. The properties dialog for a circle.

- **Collection.** Specifies collection name where object resides.
- **Name.** Object name
- **Radius.** Radius of circle
- **Projected angles.** Displays the Principle Projected Angles dialog. These angles represent the projection of the object normal vector onto the coordinates planes of the active frame.
- **Draw.** Displays the normal vector in the graphics.
- **Radial Normal Direction.** Reverses the normal radial vector direction of the circle.
- **Notes.** Contains information entered by user or information regarding construction method, fit, etc.

- **Reverse.** Reverses the normal direction of the object.
- **Transform.** Opens the transformation dialog.
- **Color.** Opens color dialog and allows the object color to be changed.
- **Update.** Updates SA with any changes in the object properties.
- **Menu.** Exposes object specific menu options. Emulates right-clicking the object in the tree.
- **Close.** Closes the object properties dialog.

Construction

Fit to Points

Constructs a best-fit circle for the selected points. See the Geometry Best-Fit section for more information.

Enter

Constructs a circle where the normal is in the Z-axis of the working frame. Use the *Circle Properties* dialog to edit parameters.

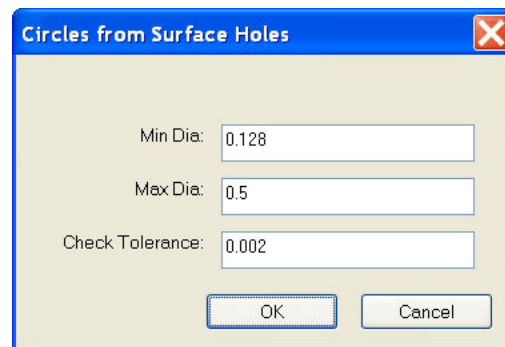
Maximum Material Condition (from Scan Patch)

Creates a best-fit circle with a diameter that bounds all of the selected points. All of the points will either be touching the circle or inside of the circle.

From CAD Surfaces

Uses selected CAD surfaces to look for circular segments and creates circles based on the search criteria.

Figure 10-5. Creating circles from surface holes.



Normal to Object, Centered at Point

Creates a circle using the selected object's normal and is centered at the selected point.

Planes

The position of a plane is reported differently than other objects. We report the position and orientation of a plane using ABCD notation. Because the origin and rotation of a plane in space is not clearly defined, ABCD is used as a simplified expression of a full transform. ABC is the unit direction vector for the plane and D is the distance from the reporting frame origin to the closest point on the reported plane (Figure 10-6).

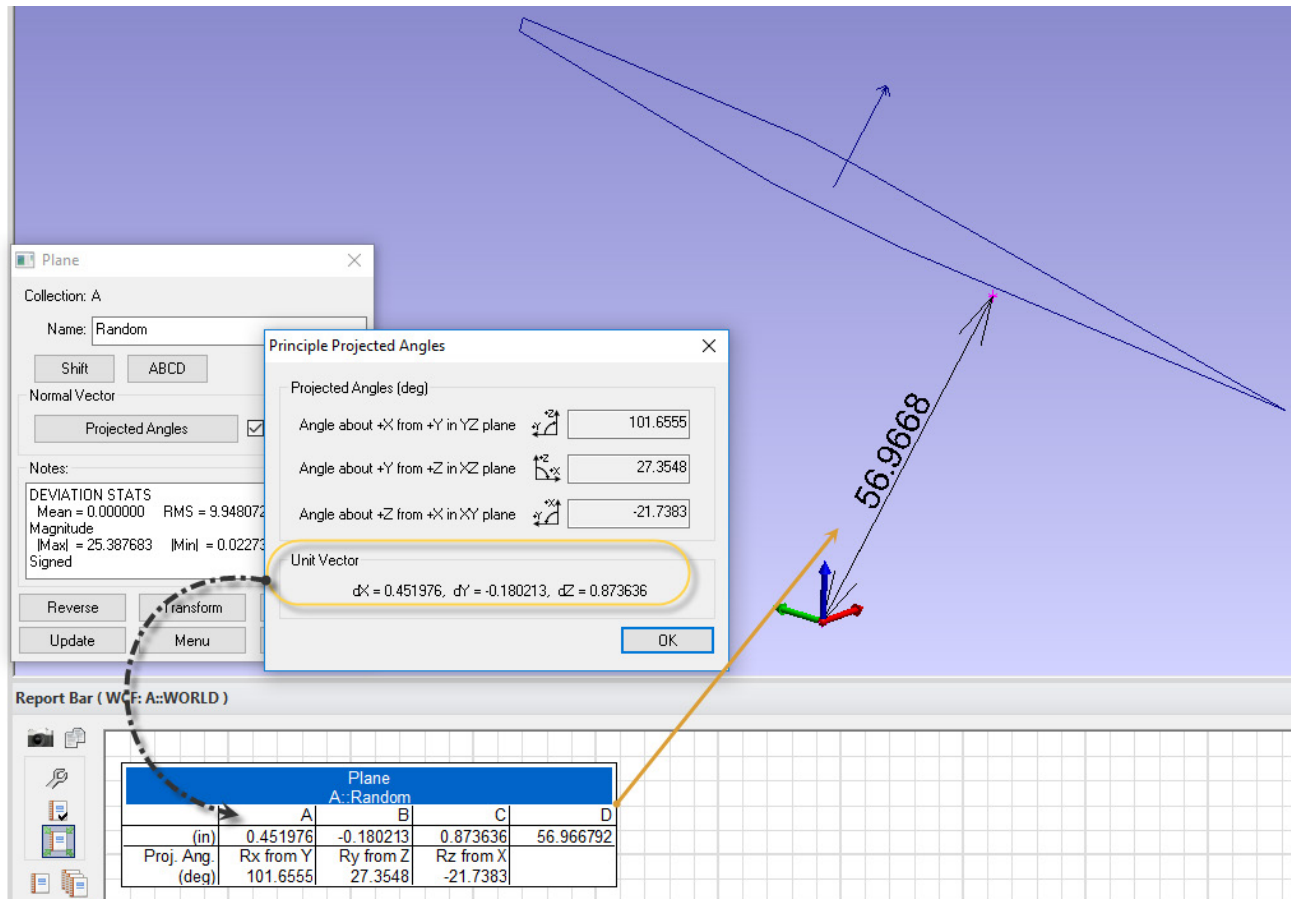


Figure 10-6. ABCD Definition

Construction

Fit to Points

Constructs a best-fit plane for the selected points. See the Geometry Best-Fit section for more information.

Enter

Constructs a plane where the normal is in the Z-axis of the working frame. Use the *Plane Properties* dialog to edit parameters.

Normal to Object, Through Point

Plane created normal to a selected object and passes through a selected point.

Normal to Line, Through End

A plane is created that is normal to the line and passes through a selected end of line.

Mirror from Two Measured Points

The mirror plane function is used to determine the location of a mirror by measuring a point both directly and through the mirror.

1. Pick the instrument.
2. Pick the point measured directly (not through mirror).
3. Pick the virtual point that is really the same point, but measured through the mirror.

SA creates a plane representing the mirror location (see Figure 10-7 on page 278). You can make a frame on this plane if needed.

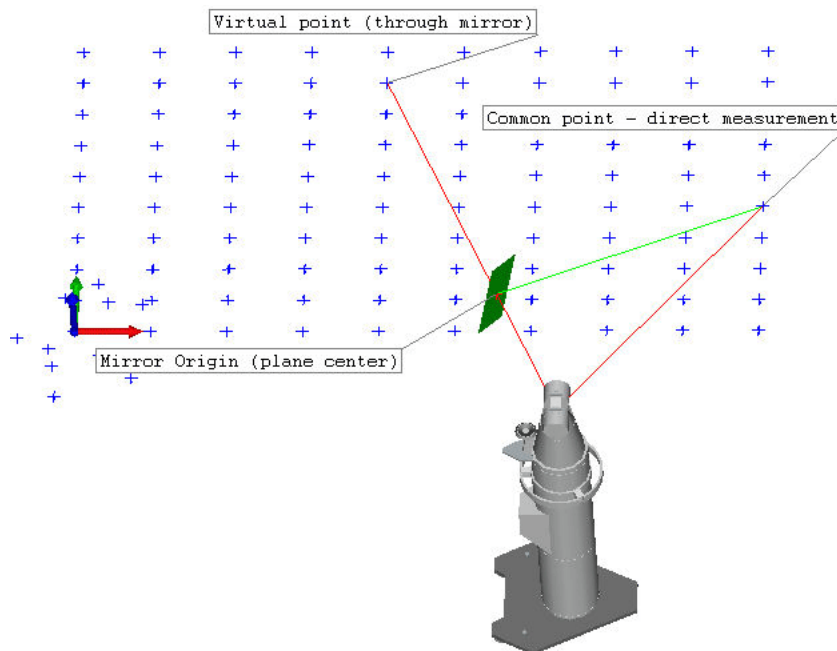


Figure 10-7. Creating a plane from two measured points using a mirror.

If the points used in the construction have uncertainty fields (**Analysis>Coordinate Uncertainty>Create Point Uncertainty Fields**), SA will automatically compute the uncertainty of the mirror plane and add the results to the notes field for the constructed plane.

Establish Bounding Planes for Points

Creates two planes parallel to the selected plane and are spaced so that they bound the selected points.

From Surface Faces

Creates planes from the selected surface faces.

Slots

Properties

- **Collection.** Specifies collection name where object resides.
- **Name.** Object name.
- **Length.** Controls the full length or the slot from end to end, in both square and rounded slots (not circle center points).
- **Width.** Defines the distance across perpendicular to the length.
- **Round or Square.** Round slots are defined as two half circles connected with two tangent lines, where as a square slot is rectangular. A slot can be defined in either form and draw in either from.
- **Projected Angles.** Displays the Principle Projected Angles dialog. These angle represent the projection of the object normal vector onto the coordinates planes of the active frame.
- **Draw.** Displays the normal vector in the graphics.
- **Notes.** Contains information entered by user or information regarding construction method, fit, etc...
- **Reverse.** Reverses the normal direction of the object.
- **Transform.** Opens the transformation dialog.
- **Color.** Opens color dialog and allows the object color to be changed.
- **Update.** Updates SA with any changes in the object properties.
- **Menu.** Exposes object-specific menu options. Emulates right-clicking the object in the SATreebar.
- **Close.** Closes the object properties dialog.

Construction

Fit to Points

Constructs a best-fit cylinder for the selected points. See the Geometry Best-Fit section for more information.

Enter

Constructs a cylinder where the normal is in the Z-axis of the working frame. Use the *Cylinder Properties* dialog to edit parameters.

From Surface Faces

Constructs a slot by searching the surrounding faces of a click point on a CAD model. The inner cylindrical or planer faces should be selected for this process.

Ellipses

Properties

- **Collection.** Specifies collection name where object resides.
- **Name.** Object name.
- **Transform.** Opens the transformation dialog.
- **Major Axis Radius/Minor Axis Radius.** The major (long) and minor (short) axis lengths (Figure 10-8).

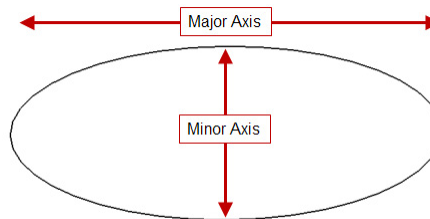


Figure 10-8. The major/minor axes of an ellipse.

- **Notes.** Contains information entered by user or information regarding construction method, fit, etc...
- **Update.** Updates SA with any changes in the object properties.
- **Color.** Opens color dialog and allows the object color to be changed.

Construction

Fit to Points

Constructs a best-fit ellipse for the selected points. See the Geometry Best-Fit section for more information.

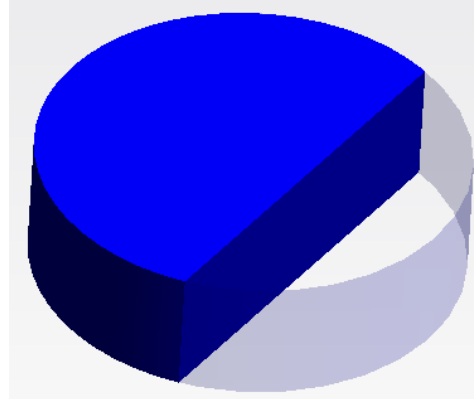
Enter

Constructs an ellipse about the working frame. Use the *Ellipse properties* dialog to edit parameters.

Cylinders

Cylinders can be displayed either using a full cylinder representation or using a cylinder slice, displaying measurement coverage. This coverage display is an option controlled from the *User Options, Display Tab, Geometry* control. When enabled the Theta Extents are set when the cylinder is fit to points. The length of a cylinder is always displayed based upon the extent of the measurements.

Figure 10-9. Cylinder coverage display is optional for measured cylinders.



Properties

- **Collection.** Specifies collection name where object resides.
- **Name.** Object name.
- **Transformation Method.** Controls the length or the transformation of the cylinder.
- **End-Point Method.** Enter two point coordinates to determine length, position and orientation of cylinder.
- **Diameter.** Diameter of the cylinder.
- **Surface Normals.** Reverses the surface normal direction.
- **Theta Start in degrees.** The starting rotation of the cylinder display when displaying measurement coverage.
- **Theta Span in degrees.** The span of display of the cylinder slice when displaying coverage.
- **Enable Theta Extent Mode.** This check box option enables coverage display to be enabled or disabled for a particular cylinder.
- **Top End Cap.** When enabled the upper end cap of the cylinder will be displayed.
- **Bottom End Cap.** When enabled the lower end cap of the cylinder will be displayed, the origin end.

- **Projected angles.** Displays the Principle Projected Angles dialog. These angle represent the projection of the object normal vector onto the coordinates planes of the active frame.
- **Draw.** Displays the normal vector in the graphics.
- **Notes.** Contains information entered by user or information regarding construction method, fit, etc...
- **Reverse.** Reverses the normal direction of the object.
- **Color.** Opens color dialog and allows the object color to be changed.
- **Update.** Updates SA with any changes in the object properties.
- **Menu.** Exposes object-specific menu options. Emulates right-clicking the object in the SATreebar.
- **Close.** Closes the object properties dialog.

Construction

Fit to Points

Constructs a best-fit cylinder for the selected points. See the Geometry Best-Fit section for more information.

Enter

Constructs a cylinder where the normal is in the Z-axis of the working frame. Use the *Cylinder Properties* dialog to edit parameters.

From Surface Faces

Constructs a cylinder from a click point on a CAD model. The inner cylindrical should be selected for this process.

Along Line(s)

Constructs a cylinder from a selection of lines. The orientation and length of the of the newly constructed cylinders will match that of the selected line and a prompt will allow manual entry of the cylinder diameter(s).

Spheres

Properties

- **Collection.** Specifies collection name where object resides.
- **Name.** Object name.
- **Graphical Settings.** Control the length, span and facets of the

graphical representation of the Sphere.

- **Radius.** This value controls the radius (diameter) of a sphere.
- **Surface Normals.** Reverses the surface normal direction.
- **Notes.** Contains information entered by user or information regarding construction method, fit, etc.
- **Transform.** Opens the transformation dialog.
- **Color.** Opens color dialog and allows the object color to be changed.
- **Update.** Updates SA with any changes in the object properties.
- **Menu.** Exposes object specific menu options. Emulates right-clicking the object in the tree.
- **Close.** Closes the object properties dialog.

Construction

Fit to Points

Constructs a best-fit sphere for the selected points. See the Geometry Fit section for more information.

Enter

Constructs a sphere at a position and orientation relative to the working frame. Use the *Sphere Properties* dialog to edit parameters.

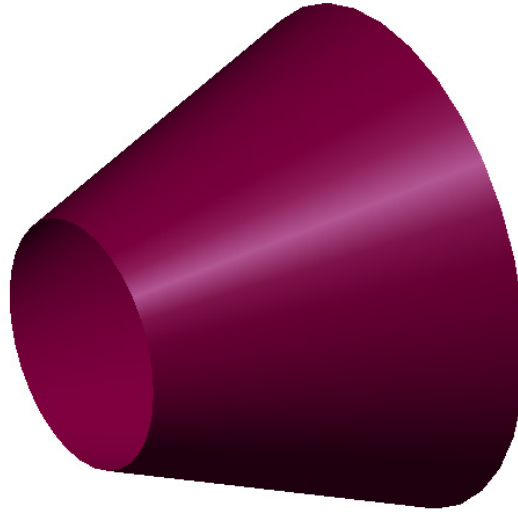
From Surface Faces

Constructs a sphere from a click point on a CAD model. The spherical face should be selected for this process.

Cones

Cones can be displayed either using a full cone representation or using a cone slice, displaying measurement coverage. The length of a measured cone is displayed based upon the extent of the measurements.

Figure 10-10. Cone coverage display which uses the cut length to limit the length display to the measured extents of the cone.



Properties

- **Collection.** Specifies collection name where object resides.
- **Name.** Object name.
- **Length.** Length of the cone measured from the apex.
- **Cut Length from Apex.** Hidden segment of the cone used to reduce graphical display to the measured segment of the cone.
- **Number of Height.** Number of graphical segments drawn along the length of the cone.
- **Theta Start in degrees.** Angle at which the graphical display begins
- **Theta Span in degrees.** The span of the graphical display.
- **Number of Theta.** Number of segments drawn around a cone.
- **Included Angle.** The entire angle that contains the taper.
- **Surface Normals.** Reverses the surface normal direction.
- **Notes.** Contains information entered by user or information regarding construction method, fit, etc.
- **Transform.** Opens the transformation dialog.
- **Color.** Opens color dialog and allows the object color to be changed.
- **Update.** Updates SA with any changes in the object properties.
- **Menu.** Exposes object specific menu options. Emulates right-clicking the object in the tree.
- **Close.** Closes the object properties dialog.

Construction

Fit to Points

Constructs a best-fit cone for the selected points. See the Geometry Best-Fit section for more information.

Enter

Constructs a cone about the working frame, use the *Cone properties* dialog to edit parameters.

From Surface Faces

Constructs a Cone from a click point on a CAD model. The conical face should be selected for this process.

Paraboloids

Properties

- **Collection.** Specifies collection name where object resides.
- **Name.** Object name.
- **Graphical Settings.** Control the length, span and facets of the graphical representation of the cone.
- **Geometry Parameters.** Control length and alternative parameters.

Construction

Fit to Points

Constructs a best-fit paraboloid for the selected points. See the Geometry Best-Fit section for more information.

Enter

Constructs a Paraboloid about the working frame, use the *Plane properties* dialog to edit parameters.

Toruses

Toruses are defined as a circular section revolved about an axis to form a donut shape. These objects can come in hand in modeling pipe runs and other circular sections.

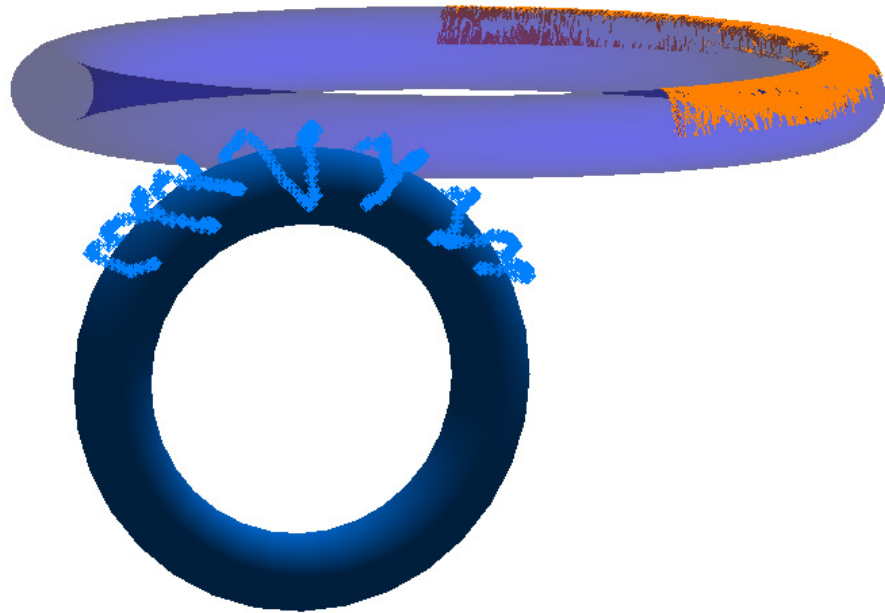


Figure 10-11. Example Toruses fit to points and cloud points

Properties

- **Collection.** Specifies collection name where object resides.
- **Name.** Object name.
- **Major Radius.** The radius of the rotation of the torus
- **Minor Radius.** The radius of each circular cross section of the torus.
- **Number of Major Segments.** Number of graphical segments drawn perpendicular to the major radius of rotation.
- **Number of Minor Segments.** Number of graphical segments drawn about the axis of rotation
- **Theta Start.** Graphical starting point about the object's base frame.
- **Theta Span.** The degree of rotation of the graphical representation.
- **Notes.** Contains information entered by user or information regarding construction method, fit, etc...
- **Transform.** Opens the transformation dialog.
- **Color.** Opens color dialog and allows the object color to be changed.
- **Menu.** Exposes object specific menu options. Emulates right-clicking the object in the tree.

Construction

Fit to Points

Constructs a best-fit torus for the selected points. See the Geometry Best-Fit section for more information.

Enter

Constructs an torus about the working frame. Use the *torus properties* dialog to edit parameters.

Perimeters

Perimeters appear graphically much like a series of connected lines that define a 3D volume. Perimeters are used with scanning devices to define a region of interest for measurement purposes.

Properties

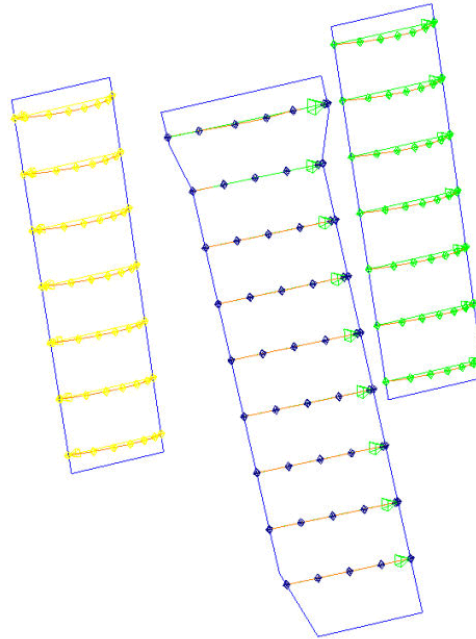
- **Collection.** Specifies collection name where object resides.
- **Name.** Object name.
- **Closed Perimeter.** When unchecked the perimeter is opened.
- **Notes.** Contains information entered by user or information regarding construction method, fit, etc.
- **Color.** Opens color dialog and allows the object color to be changed.
- **Apply.** Updates SA with any changes in the object properties.
- **Cancel.** Closes dialog without saving changes.

Construction

Manual Point Selection

This command will create an open or closed perimeter object from selected points. Its purpose is to provide a boundary for scanning instruments. The below example shows perimeters to capture data from a targetless scanner.

Figure 10-12. Perimeters defining the boundary of measurements from a total station.



From CAD Faces

This option will build a bounding perimeter for each and every surface face selected. This makes it quite easy to identify specific CAD faces to be measured.

Bounding Surface from Instrument Viewpoint

When a scan of an entire model is needed, this option delivers. It uses the base frame of the selected instrument as the viewpoint, allows a selection of a selection of surfaces and builds a bounding perimeters that is grown 10% larger than the model to ensure complete surface coverage from the instruments perspective.

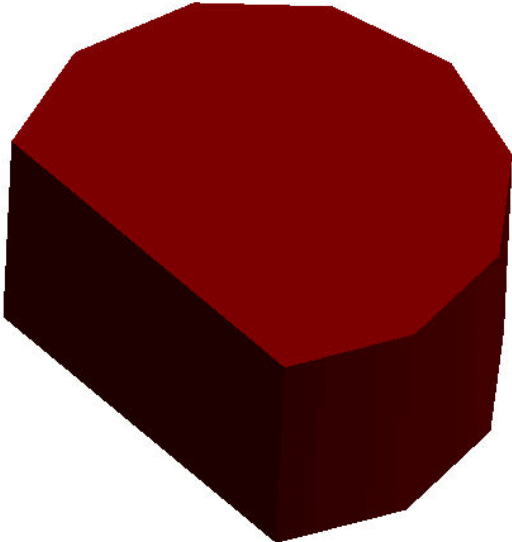
Graphical Entities

SA offers the ability to build a selection of graphical objects that are primarily used for display purposes. Once built these are listed in the tree under the Graphical Entities category. These include:

Cylindrical objects such as:

Cylinder Slice

Figure 10-13. Creating a cylinder slice.



Pie Wedge

Figure 10-14. Creating a pie wedge.



Pipe

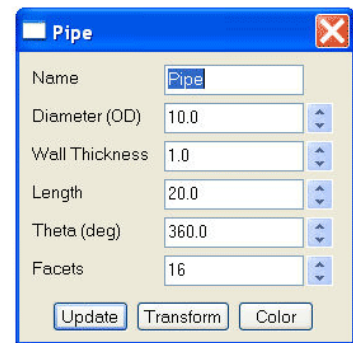
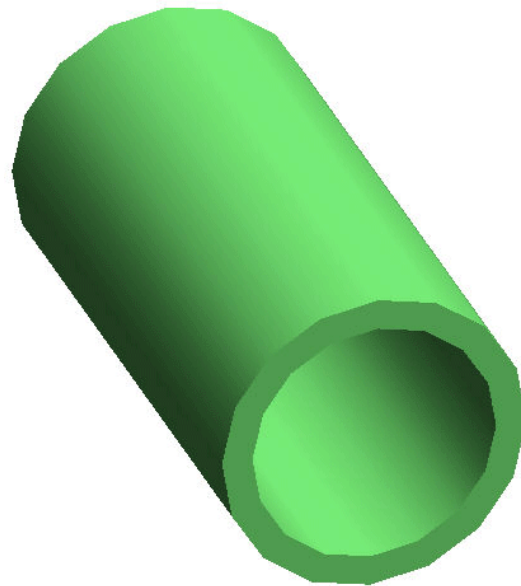


Figure 10-15. Creating a pipe.

Boxes

Tapered Pipe

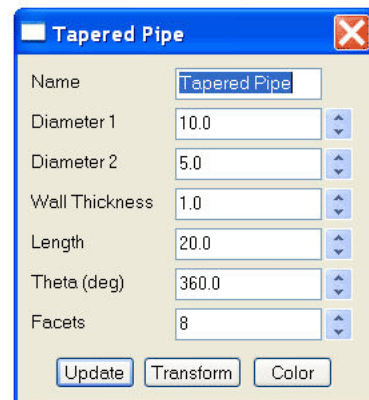
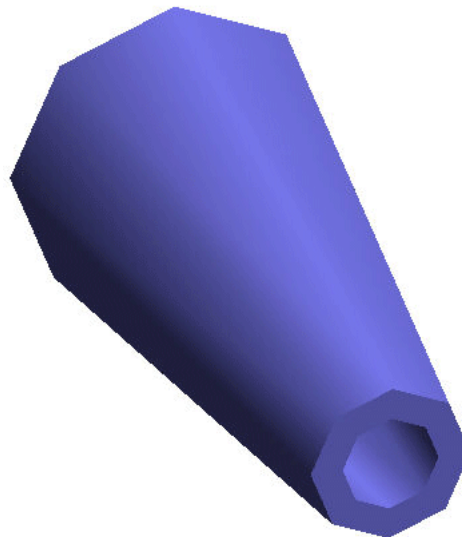
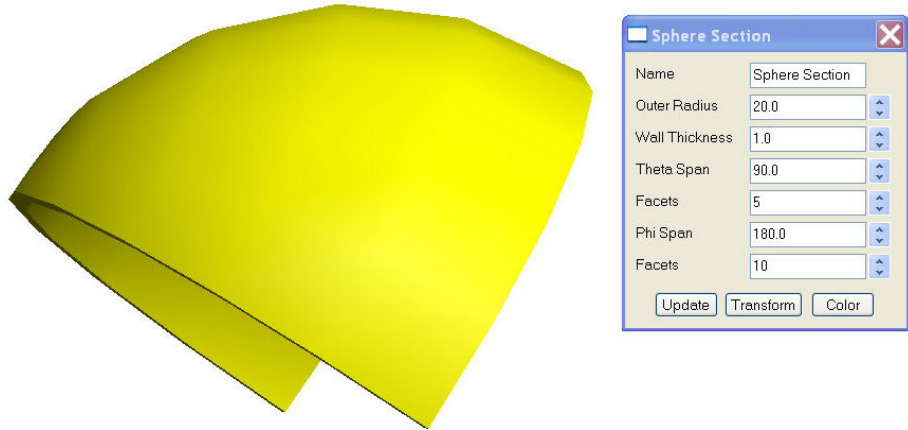


Figure 10-16. Creating a tapered pipe.

Sphere Section

Creates a spherical section. Use the *Sphere Section* dialog to edit parameters.

Figure 10-17. Creating a spherical section.



Ellipsoid

Ellipsoids can be constructed and displayed with a set length in X, Y, and Z and the transform edited.

Boxes

Box display options include:

- **Box.** A box has a length width and height as well as a position transform.
- **Tapered Box.** A tapered box offers a front and back face length and width control as well as a height and transform.
- **Rounded Box.** A rounded box offers an Length, and two width parameters that control the radius of the ends. A depth and transform is also available.
- **Arched Box.** An arched box has 1 rounded end and offers length, width and depth controls along with a transform.
- **Rounded Plate.** Defines a box with 4 rounded corners of a given radius.

Pyramid

A pyramid offers Length, Width, Height and Transform controls.

This Page Intentionally Left Blank.

Instruments

Instruments are the devices that gather accurate measurement data and provide a foundation for this data's position in a digital world. This chapter discusses how to work with instruments in general.

Refer to the Instruments Manual for details specific to particular instrument types.

Spatial Analyzer (SA) supports a wide range of measurement devices, from PCMM arms and laser trackers to scanners, laser radars, projectors, digital levels, and theodolites. An instrument model in SA contains the basic properties of the instrument and defines its location in the workspace. It is the foundation for measurements from a live instrument in a single plant, and a new model is added each time the live instrument is moved in the real world.

To add an instrument model to SA select **Instrument>Add**. You will then connect your live instrument to this model when taking measurements.

Instruments appear in the treebar under the Instruments category, and each instrument is assigned a number (starting from zero), referred to as the *Instrument Index Number*. This number appears next to the name of the instrument in the treebar (Figure 11-1).

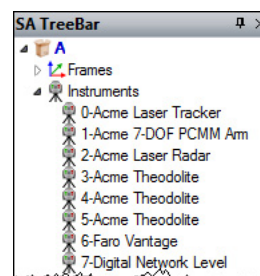
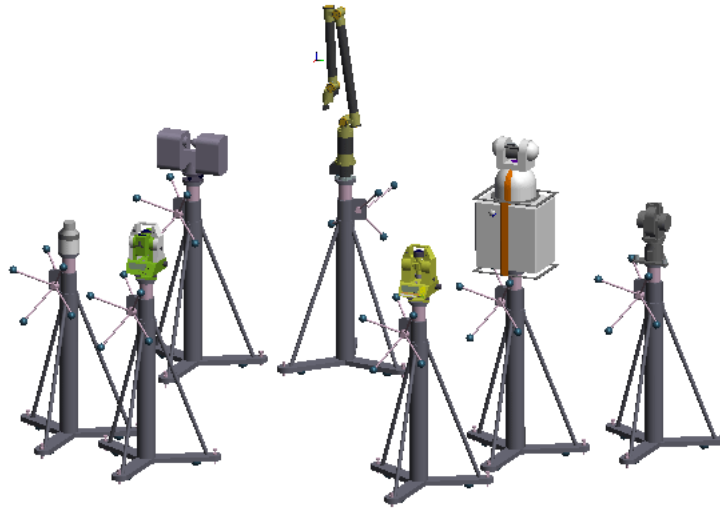


Figure 11-1. Instruments in the tree.

It is very important to keep in mind that *instruments are not objects*. In SA, objects are treated differently than instruments. Commands and functions that explicitly ask for objects will not accept an instrument.

They are completely independent entities.

Figure 11-2. A number of instruments in the graphical view.



For spherical measurement devices (total stations, laser trackers, and the laser radar, for example), the position of the instrument defines the origin of the instrument's measurement volume. For a PCMM arm, the instrument position defines the base of the kinematic chain of linkages leading toward the probe.

When a point is measured by an instrument, the instrument reports back a point in its own internal coordinate system. This point is recorded in the measured target's observation details. For spherical measurement devices, the coordinate is represented in a spherical coordinate system. For a Cartesian measurement device (such as an arm), the coordinates are represented in XYZ coordinates.

Instrument Properties

To Access an Instrument's Properties:

- Right-click the instrument in the tree or graphical view and choose **Properties** from the context menu,
- Double-click the instrument in the tree, or
- Select **Instrument>Properties**, and double-click the instrument of interest.

The properties for an instrument ([Figure 11-3](#)) will vary depending on the class of instrument (laser tracker vs. arm vs. scanner, etc.).

Figure 11-3. The Instrument Properties dialog.

- **Instrument Number.** The index number for the instrument (referred to as the *Instrument Index Number*).
- **Model.** The name assigned to the instrument. (By default, this will be the manufacturer's name and instrument type).
- **TCP/IP Socket Port.** The TCP/IP port that the instrument has been assigned. This is a random port number that is assigned by SA.

Sensor Control

The sensor control section allows you to adjust the graphic model of the instrument. For spherical devices, you can adjust the vertical, horizontal, and distance degrees of freedom. For an arm, you can adjust the rotation value for each link. Other instrument types (such as laser scanners) may not have any adjustment at all. After selecting the degree of freedom and associated value, click the **Apply** button to see the changes. Or, click the up/down arrows to increment the rotation value.

Scale Factor

Warning: Scaling an instrument's measurements requires caution. Once an instrument's measurements have been scaled, querying one measured point to another will not represent the true measured real-world distance between the two points.

It may be desirable to apply a scale to an instrument's measurements to account for thermal expansion and contraction of a measured part. For example, suppose you have a nominal model that has been designed for 68°F (20°C) conditions. If you are laying off nominal features from that model on a hot day--say 95°F (35°C)--then you will want to scale the instrument's measurements down from 95°F to 68°F so that the instrument and the nominal model are adjusted to the same temperature reference. One way to achieve this is to scale the instrument's measurements.

When an instrument scale factor is applied, all measurements from the instrument will be scaled appropriately. For example, an instrument observation that is normally at a radius of 100" from the instrument will be 101" from the instrument at a scale factor of 1.01. Instruments that are currently using a scale factor other than 1.0 will appear with a special yellow icon in the tree and will be listed as a scaled instrument, as depicted in [Figure 11-4](#).

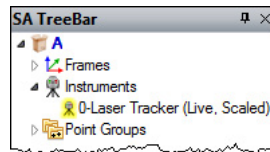


Figure 11-4. This laser tracker has a scale applied.

The scale for an instrument can be set in several different ways. It can be set manually (by directly entering a scale factor), set by measuring scale bars, or set in a command (such as **Instrument>Locate (Transform to Part)>Measure Nominal Points**), which allows you to vary the instrument's scale as part of the best-fit procedure.

To Set an Instrument's Scale Factor Manually:

1. In the *Instrument Properties* dialog, click the **Set Manually** button.

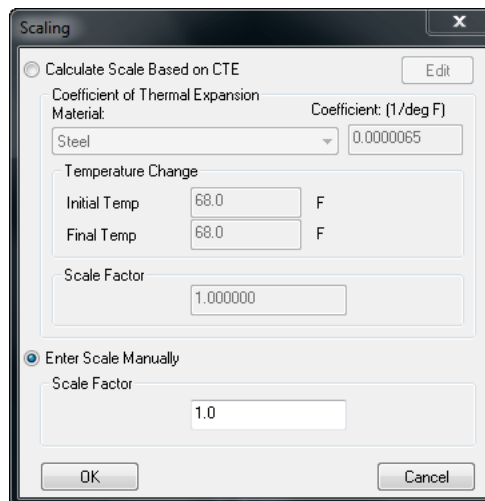


Figure 11-5. The scaling dialog.

2. In the *Scaling* dialog ([Figure 11-5](#)), select the *Enter Scale Man-*

ually radio button and enter a scale factor, then click **OK**.

To Calculate a Scale Based on a Coefficient of Thermal Expansion:


Consider an example of measuring a steel part sitting in the sun, which has expanded in the 95°F midday heat. If we want to compare measurements against the nominal model for this part, which is at 68°F, then it is important to scale the instrument's measurements from 95°F (initial temperature) to 68°F (final temperature). This would be a situation where entering a scale for the instrument based on a CTE could be appropriate. The resulting scale factor will shrink the instrument's measurements so that the measurements can be compared to the 68°F CAD model, effectively removing the differences caused by thermal expansion.

1. In the *Instrument Properties* dialog, click the **Set Manually** button.
2. Select the *Calculate Scale Based on CTE* radio button.
3. In the dropdown, choose the desired material for the part that will be measured. If the material of interest is not available, click the **Edit** button, which allows you to add, delete, and edit materials and their associated coefficients of thermal expansion.
4. Enter an initial and final temperature. A higher final temperature will result in a scale increase, and a lower final temperature will result in a scale decrease.

To Calculate a Scale Based on Measured Scale Bars:

Scaling based on measured scale bars should only be performed when the material of the scale bar matches the material of the part to be measured, and when the nominal scale bar temperature matches the nominal (modeled) CAD temperature of the part. Consider an example of measuring a steel part sitting in the sun, which has expanded in the 95°F midday heat. If a steel scale bar with a nominal length of 36" at 68°F is then measured in the same conditions as the part, it will also have expanded, and the measured scale bar points will be more than 36" apart. Scaling based on this measured scale bar will scale the instrument's measurements down to match the nominal (68°F) length of 36". Subsequent measurements from the instrument will now be properly compensated to the nominal 68°F temperature.

1. In the *Instrument Properties* dialog, click the **Set from Scale Bars** button.
2. If the instrument has measured points that belong to a defined scale bar, the scale factor will be instantly calculated.
3. Review the results of the scale calculation, and click **Yes** to apply the scale factor to the instrument.

 **Note:** If the scale bar points have not been measured by the instrument in question, then no scale will be applied.

- **Measurements Active.** Unchecking this box will deactivate all of this instrument's measurements from target calculations. This might be useful, for example, if you've measured the same point with several instruments, and have found an instrument to be out of calibration to the point where its measurements are corrupting the overall coordinates of the targets. Re-check this box to activate the measurements again. (Note that this will require you to recompute targets from shots using **Analysis>Re-Compute Targets From Shots**).

Uncertainty Variables

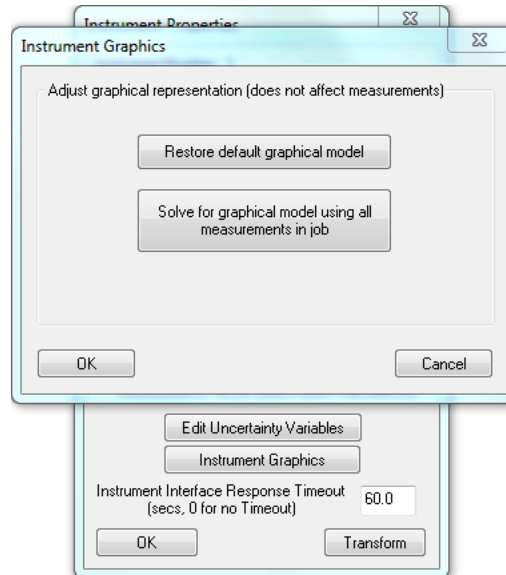
All instruments have error in their measurements, and as a result every measurement from an instrument has some uncertainty. Each class of instrument (laser tracker, PCMM arm, total station, etc.) has different set of uncertainty characteristics. For example, laser trackers have uncertainties for the horizontal and vertical encoders, as well as an uncertainty for the distance measurement. PCMM arms, on the other hand, have an angular uncertainty for each link of the arm. Additionally, each instrument within a specific instrument class has different values for the uncertainties. The uncertainty values for a newly calibrated laser tracker in pristine condition from a manufacturer will be different from those of an old, dirty, poorly calibrated tracker from the same manufacturer.

For more information on uncertainty and uncertainty parameters refer to the chapter: "[Measurement Uncertainty](#)" on page 761.

Instrument Graphics (PCMM arms only)

PCMM arms have an **Instrument Graphics** button that appears directly under the **Edit Uncertainty Variables** button. This button calls up the **Instrument Graphics** dialog ([Figure 11-6](#)).

Figure 11-6. The Instrument Graphics dialog for portable arms.



Note: The Instrument Graphics options only affect how the PCMM arm is depicted in the graphical view. They have no effect on measured data.

- **Restore Default Graphical Model.** The kinematics for a PCMM arm are refined as points are measured. These kinematics only affect how the arm is depicted in the graphical view—not the accuracy of the measurement data itself. Clicking this button will restore the default graphical model that has been stored with the instrument definition.

SA offers the ability to manually calibrate an arm graphical model based on measurements recorded in the job file. Once calibrated this new calibration is saved in C:\Analyzer Data\Persistence. If a calibration is found for you arm SA will load it automatically when it connects.

- **Solve for Graphical Model Using All Measurements in Job.** This option will consider the measurements that the arm has observed, and recalculate the kinematics for the arm.
- **Instrument Interface Response Timeout.** This value determines (in seconds) how long an instrument interface will attempt to connect to the instrument until it times out with an error, stating that the instrument cannot be found. Enter zero to continue checking indefinitely.
- **Transform.** The Transform button is used to transform (move and rotate) the instrument. More information about transformation can be found in the Move section.

Instrument Communication Controls

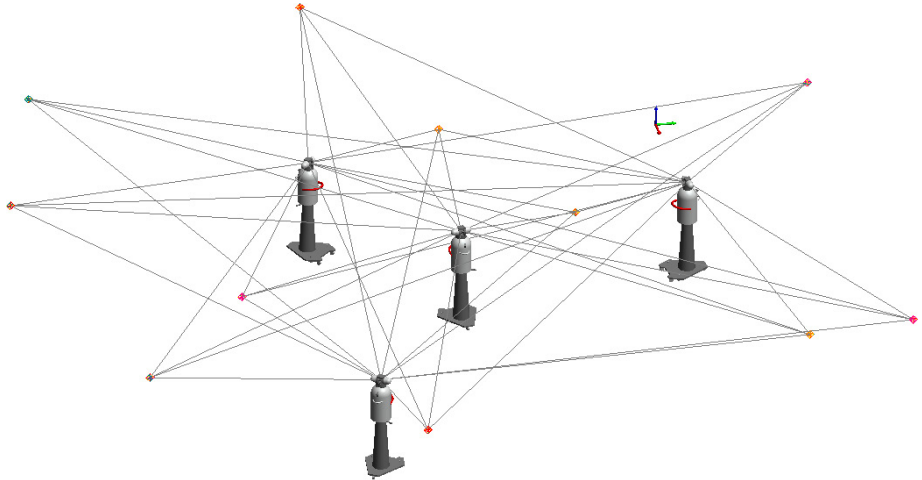
Several instrument communication controls have been added to allow users to tailor performance based upon their computer performance and instrument requirements. These include:

- **Instrument Interface Response Timeout.** This value is used to define how long SA waits for a response from an instrument in seconds. If no response is received after the specified duration a break in communication will be assumed to have occurred such as might be caused by a firewall turning on or a cable getting unplugged. Also, some data transmitted to an instrument may be such that the function sending it will wait for a response before exiting the function. This value establishes the amount of time the function will wait before it exits on a “timeout” condition.
- **Packet Warning Threshold.** Measurements from an instrument are stored in packets and processed by SA as fast as possible. However, when recording large amounts of data at a fast rate a backlog can occur and a user notification can be useful. This warning threshold defines a minimum threshold below which no warning dialogs will be shown. This a dynamic threshold which determines, based upon the number of packets processed and the time required to process them, how many packets could be expected to be processed over an interval of 5 seconds.
- **Packet Bundle Interval.** This interval is used to control the minimum update rate of the graphics and user interface while measuring. The instrument socket thread will present data for the incoming “PendingPacket” buffer as fast as the instrument will send them. The data is only processed upon receiving a notification from the instrument socket thread that there is data available. The time interval between these notifications will not be less than this time interval.
- **Filter Instrument Update Packets.** This check box controls whether graphical updates from an instrument are stored and processed or filtered out and ignored to accelerate data processing speeds. If the filter is asserted, only the last instance of an update packet will be processed even if there are several packets of this sort in the packet bundle being processed.

Instrument Shots

It is often useful to draw instrument shots—that is, the line of sight from instruments to their corresponding measurements, as depicted in [Figure 11-7](#). Instrument shots (also referred to as shot lines) help you visualize which instruments have measurements to which points, and also helps with visualizing the angles and distances to your shots, as well as the entire instrument network as a whole.

Figure 11-7. Shot lines for an instrument network.



To Control Whether Instrument Shots are Drawn in the Graphical View:


- Toggle the **View>Show Instrument Shots** setting.

Instrument Shot Modes

There are four modes for viewing instrument shots:

- **Draw only last shot of active instruments.** Draws only the most recent shot from any active instruments.
- **Draw all shots.** Draws every shot line from every instrument to every point (used in [Figure 11-7](#)).
- **Draw only active (enabled) shots.** Any shots which have been disabled will not have shot lines drawn—but all others will.
- **Draw only inactive (disabled) shots.** Any shots which are enabled will not have shot lines drawn—but all others will.


To Change How Instrument Shots Are Drawn:

1. Click the User Options icon  in the toolbar.
2. In the *Display* tab, click the **Shot Graphics Options** button.
3. Select the desired mode from the *Mode* box.

Shot Colors

In addition to changing the instrument shot mode, you can also set the color used for active shots, inactive shots, or the latest (Current) update.

To Change Instrument Shot Colors:


1. Click the User Options icon  in the toolbar.
2. In the *Display* tab, click the **Shot Graphics Options** button.

3. Click the **Active Shots**, **Inactive Shots**, or **Current Update** button to indicate which color you'd like to modify.
4. Select a color from the dialog, then click **OK**.

Angle Only Shots

Instruments that do not measure distance (such as theodolites) cannot measure a point in 3D space, therefore the shot line length cannot be determined. A default length of 200 units is used for drawing these shots, although you can lengthen or shorten this value as appropriate to the scale of the measurement job.

To Change the Length of Angle Only Shots:

1. Click the User Options icon  in the toolbar.
2. In the *Display* tab, click the **Shot Graphics Options** button.
3. Enter a length into the *Default Length* field.

Instruments and Measurements

A unique characteristic to SpatialAnalyzer is the association of points to their respective instrument. SA treats an instrument and its points as a rigid body. If an instrument is transformed (moved), the instrument and its points will move with it. This is a key concept that is used extensively throughout SA. When a point is measured the observation is built and the link is created. For more details refer to “[Measured Points](#)” on page 194.

Objects can also be associated to an instrument. In some cases, such as with point clouds and frames, objects are directly handled like measured points and the link between them and the instrument can be rebuilt if things are moved. This is done using **Analysis>Reset Associated Object Transforms**.

Geometry fit to measure points, all measured from a single instrument station, is also automatically associated with that instrument so that when an instrument moves the associated points and objects will move as well.

Finally additional objects can be linked to an instrument manually using **Instrument>Object Associations**.

XYZ Reference Frame

A special instrument called a **XYZ Reference Frame** instrument exists for the specific purpose of linking things together within a job file. It's tool to link other objects together for monitoring applications. It's position will update automatically to match a specified frame in a job file and any associated objects will also move with it. This provides a means to take a series of 6D measurements, updating a frame's po-


sition and see the resulting associated objects move dynamically to match.

Using Instruments

Before you can begin measuring with an instrument, you need to add one to your current job.

To Add an Instrument:

Do one of the following:

- From the menus, choose **Instrument>Add**, or
- Press **Alt+I**, the default keyboard shortcut, or
- In the Main Toolbar, click the Add Instrument  button.

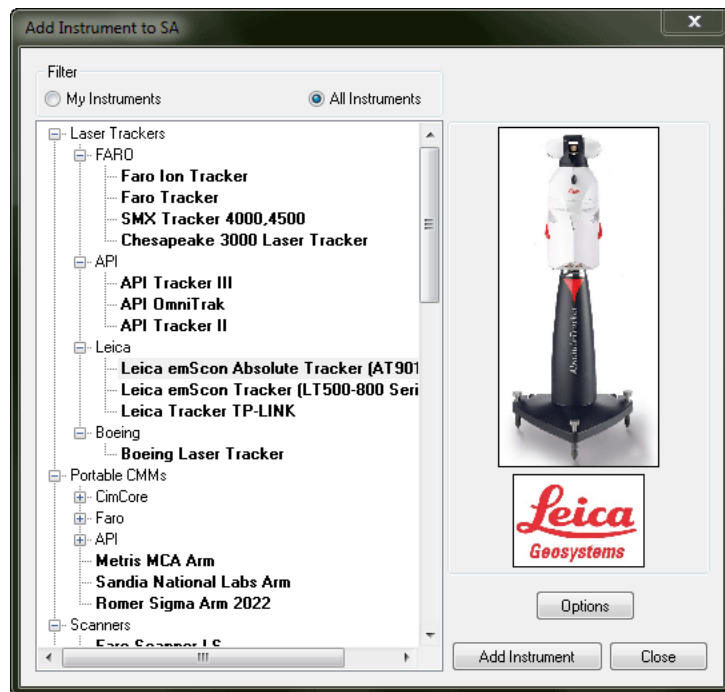


Figure 11-8. The Add Instrument to SA dialog.

The *Add Instrument to SA* dialog (Figure 11-2) will list all available instruments that can be used in SA. The instrument list can be filtered to only show the instruments that you have added in the past; each time an instrument is added to a job, it's added to this filtered list.

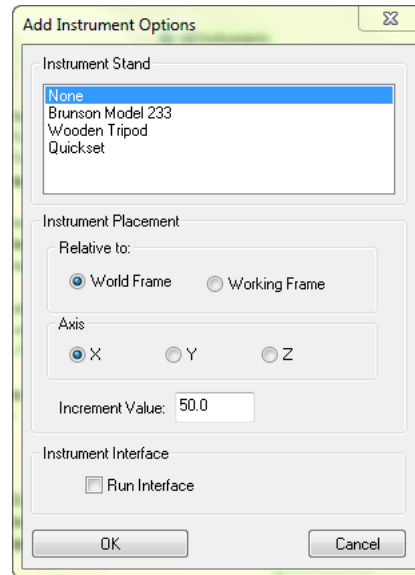
To Switch Between the Filtered Instrument List and the Full List:

- Select the **My Instruments** or **All Instruments** radio button.

To Reset Which Instruments Appear in the Filtered List:

- Navigate to the Analyzer Data\Persistence folder and delete the My Instruments.bin file.

Figure 11-9. The options available for adding an instrument.



Clicking the **Options** button opens the **Add Instrument Options** dialog (Figure 11-3). In this dialog, you can specify an instrument stand (for visual representation purposes only), specify where to locate the newly added instrument, and specify whether to automatically run the instrument's interface when the instrument is added. These settings are optional conveniences and are not critical to the procedure of adding instruments. The settings persist and will apply to the addition of future instruments until the settings are changed.

To Specify an Instrument Stand:

- Choose **None**, **Brunson Model 233**, **Wooden Tripod**, or **Quickset** from the Instrument Stand list.

To Specify Where a New Instrument Should Be Located:

- Choose **World Frame** (to add the instrument relative to the World frame) or **Working Frame** (to add it relative to the current working frame).
- Specify whether to add the instrument along the chosen frame's X, Y, or Z axis.
- Enter an Increment Value. This value specifies (in job units) how much further down the selected frame's axis each subsequent instrument should be added.

To Specify Whether a Newly Added Instrument's Interface Should Automatically Be Started:

- Check or uncheck the **Run Interface** option.

Instrument Options and Properties

When an instrument is added to SA, it is added to the graphical view as well as the tree. Each instrument is identified by an Instrument Index Number and the name of the instrument. In a given collection,

the Instrument Index Number begins counting from zero.

Right-clicking an instrument in the tree will expose a number of commands that apply to instruments.

Instrument Properties

The instrument properties give access to several important properties. Here the user can set the scale of the instrument either using Scale Bars, Material CTE or a Manual entry. Instrument Properties are covered in SA Fundamentals | Entities | Instruments.

Instrument History

The instrument history (Figure 11-4) is a log of events that have occurred to that particular instrument. Events such as when an instrument was connected, instrument specific information, how the instrument was transformed, etc.

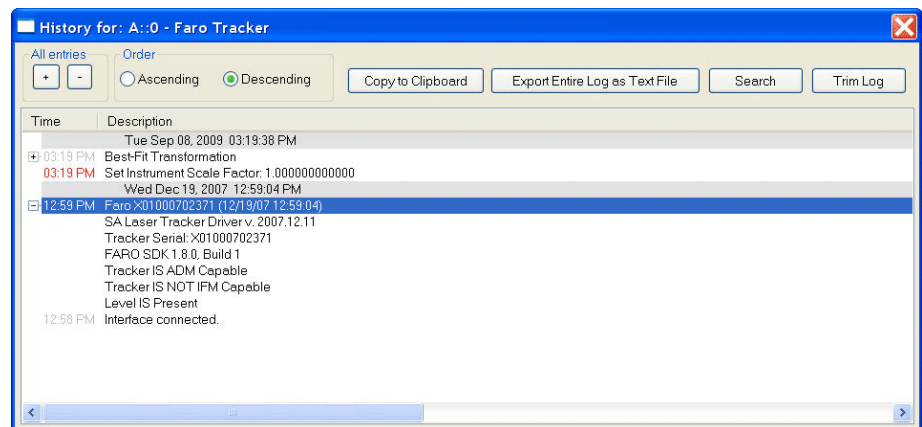


Figure 11-10. A tracker's history.

The Instrument Interface

One of the most powerful aspects of SpatialAnalyzer is the modularity of the instrument interfaces and how they interact with the core analysis platform. These interfaces are able to work independently or together, on a single computer or across networked computers.

Instrument interfaces communicate with SpatialAnalyzer using the robust TCP/IP network protocol. This affords several distinct advantages. First, these connections are extremely fast and reliable. Second, since TCP/IP is a “network” protocol, the instrument interfaces may be run on either the same computer as the instance of SA, or on another computer accessible on the same network. Third, and perhaps most impressive: since it is the “Internet Protocol” (IP), the computers running the instrument interfaces could be in the same room, across the country, or around the world. Likewise, if activated, it is possible to remotely monitor a measurement job in progress using this same TCP/IP connection over the Internet or on a local network.

Modular instrument interfaces are the basis of the data acquisition aspect of SpatialAnalyzer. These provide an easy-to-use and friendly interface for the control of instruments during a measurement job.

Establishing the instrument interfaces as separate entities in the architecture affords several key advantages. First and foremost, this provides flexibility to adapt to tomorrow's instrumentation. The number of instrument interfaces available for SpatialAnalyzer is ever-expanding, much like printer drivers and other peripheral devices for the Windows operating system. Second, this approach enables SA to provide a common interface for each type or class of instrument. A portable CMM manufactured by Company A would have the same top-level interface as a portable CMM manufactured by Company B. This allows users to easily select the instrumentation best suited for a particular task. It also allows users to easily transfer the expertise they have gained with one instrument to another. A direct benefit of this capability is to improve a user's flexibility when moving from task-to-task and it reduces the training time for a user to become productive on a different instrument. Third, this allows for a more objective view when it comes time to replace or purchase new hardware, since re-training is minimized by the common instrument interface in SpatialAnalyzer.

In all cases, the instrument interfaces perform four key functions:

- They provide a common and simple user interface that can be used under SpatialAnalyzer control to guide a user through a particular measurement task.
- They communicate over a TCP/IP network to convey essential measurement data to SpatialAnalyzer.
- They communicate via a native format to the instrument. For example: serial RS-232 connections for many theodolites, Network protocols for many laser trackers.
- They provide an efficient means to collect the raw measurements, the conditions and techniques under which they were collected, and to perform operational checks on the instrument to establish that it is meeting its performance specifications. It also supports simulating measurements and measurement tasks before the job is performed. By simulating a measurement process before the job is performed, accuracy and time estimates can be made to help you make the best choices regarding optimizing the placement of the instrument(s) relative to the subject of the survey.

Starting an Instrument Interface


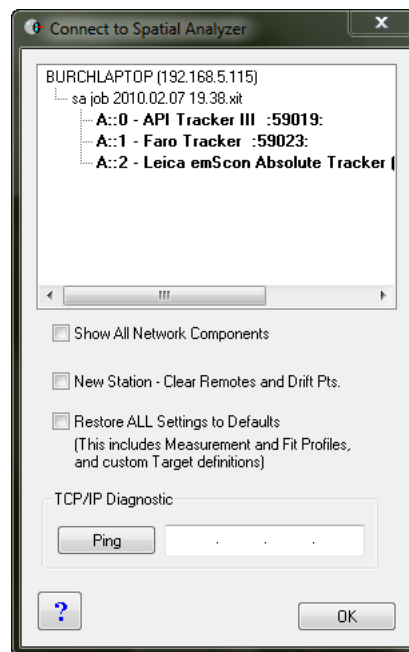
To start an instrument interface, choose **Instrument > Run Interface Module** (or click the dropdown arrow next to the Run Instrument Interface  icon). Then select the instrument interface of choice. Once selected, the **Connect to Spatial Analyzer** window will display (Figure 11-5). This window displays all network-accessible and running instances of SA containing instruments. The network location (computer name and IP address) appears in the list, in addition to the open job filename. Any applicable instruments not already connected to an interface will appear in boldface.

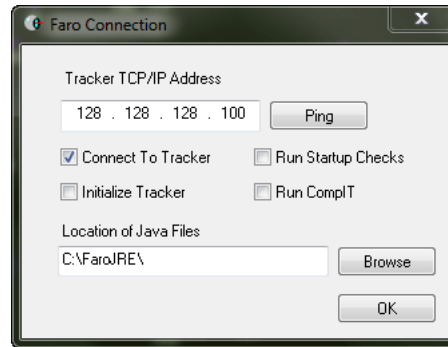
Figure 11-11. The list of available instruments to connect to.



In this window, you're choosing the instrument in an SA job file to which you'd like to send the measured data. While it is usually the file open on the same computer, this is not required. Choose the instrument to which you wish to connect and click **OK**.

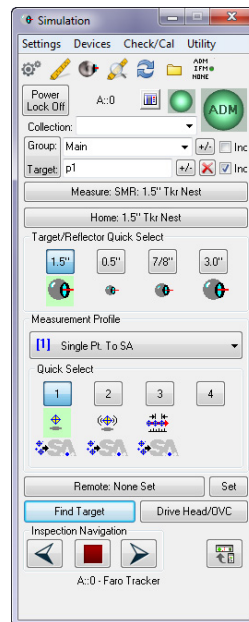
The next dialog will display connection options that are specific to the instrument being added. In Figure 11-12, a Faro tracker is being added.

Figure 11-12. Connection options for this instrument (a Faro tracker).




Once accepted, the instrument interface will appear (Figure 11-13).

Figure 11-13. A simulated tracker interface.



See the Interface section for information on instrument-specific interfaces.

If a single instrument is in the job file, you can click directly on the Run Instrument Interface  icon, and the instrument’s interface will be connected and started up. This is equivalent to the **Instrument>Run Interface Module and Connect** command.

Running Multiple Instrument Interfaces

One of SpatialAnalyzer’s strengths is its ability to both connect and gather data simultaneously from multiple instruments in the same job file at the same time. This can be done with any combination of instruments, and users may connect and disconnect instrumentation at will. The process for adding additional instruments is exactly the same as adding the first. Just follow the procedure outlined in earlier sections. Users for this are described in several other sections:

- (see “Synchronized Measurement” on page 328)

-
- (see “Transformation Tracking” on page 400)

Connection Configurations

Instruments can be categorized by the connection type used to interact with the computer. These fall into several basic categories:

- **Serial Devices.** Serial devices (such as RS-232) require a functional serial port on the user’s computer and the use of a serial cable. USB to serial adapters are available should your computer not have a serial port (We have had success with DIGI Edgeport adapters, which although are more expensive, have been quite reliable). Refer to the individual following sections for more specific configurations.
- **Bluetooth devices.** Bluetooth devices require a Bluetooth receiver on the user’s computer. If your computer does not have a built-in Bluetooth connection, a relatively inexpensive module, such as a Rocketfish™ Micro Bluetooth Adapter, can be purchased from most office supply stores.
- **Universal Serial Bus (USB).** USB connected devices typically require the installation of a USB driver for the instrument. The latest drivers for supported instruments can be found on NRK’s FTP site at: <ftp://ftp.kinematics.com/pub/SA/Install/DriverDownloads/>.
- **Network-connected devices (TCP/IP).** The majority of instruments used in SpatialAnalyzer are network-connected devices. This connection type provides extremely fast and reliable connections either directly to the host computer or over a network.

IP-Based Instruments

IP Address Basics

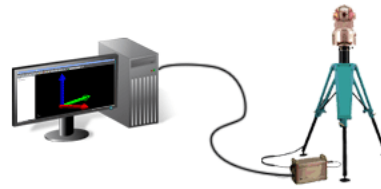
An IP address is an identifier for a computer or device on a TCP/IP network. Networks using the TCP/IP protocol route messages based on the IP address of the destination. The format of an IP address (at least for the original, IPv4 version) is a 32-bit numeric address written as four numbers separated by periods. Each number can be zero to 255. For example, 1.160.10.240 could be an IP address. Within an isolated network, you can assign IP addresses at random as long as each one is unique.

The four sets of numbers can be thought of as a mailing ad-

dress. State, City, Street and House number. Anyone can live in the same State and City and reside on the same street, however there can't be duplicate house numbers. So if an instrument has an IP address of 128.128.128.100, the computer must have a different "house number", such as 128.128.128.99. The subnet mask determines which numbers are used as part of the address and which are part of the house number. SA expects a subnet a mask of 255.255.255.0. This blocks the first 3 numbers out as part of the address and only leaves the last number to identify the house (or device).

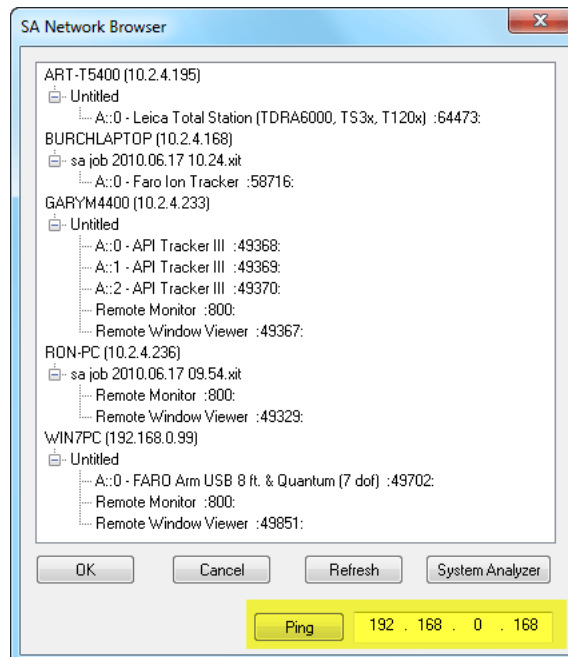
Computer and Instrument Setup

Figure 11-14. A tracker connected directly to a computer via an ethernet cable.



- Host computer and instrument have unique IP addresses.
- It is typically easier to place the host computer on the same IP family as the instrument.
- As an example the instrument IP address is 192.168.0.168
- The host computer would need a unique IP address on the same IP family.
- Change the host computer to 192.168.0.100 for example.
- To test the connection use the **Ping** function in the SA network browser (Figure 11-15).

Figure 11-15. Pinging an instrument to check a network connection.




If multiple instruments exist on the network and they belong to the

same family, no changes are needed. If they do not belong to the same family the host computer must be configured to have multiple IP addresses. See the section below regarding Multiple IP Setup.

Changing a Static IP Address

Windows 7

1. Navigate to the control panel via the Start Menu
2. Select **View network status and tasks**  to view the current connection.
3. Select the connection of interest.

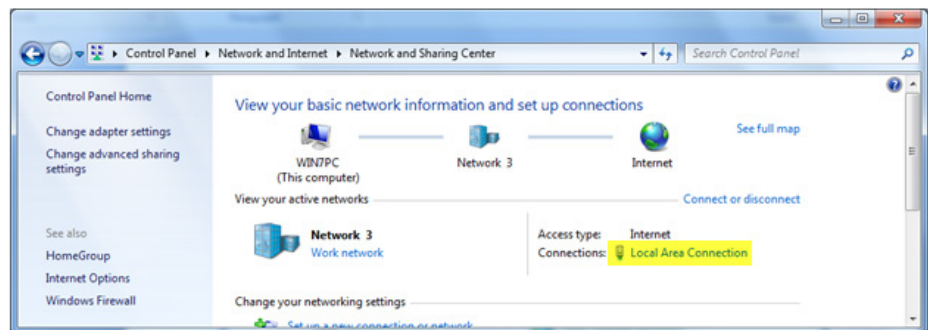


Figure 11-16. The Windows 7 Network Connections dialog.

4. Select **Properties**. In the *Properties* dialog, choose **Internet Protocol Version 4 (TCP/IPv4)** and click the **Properties** button.
5. Enter the desired static IP address and subnet mask.

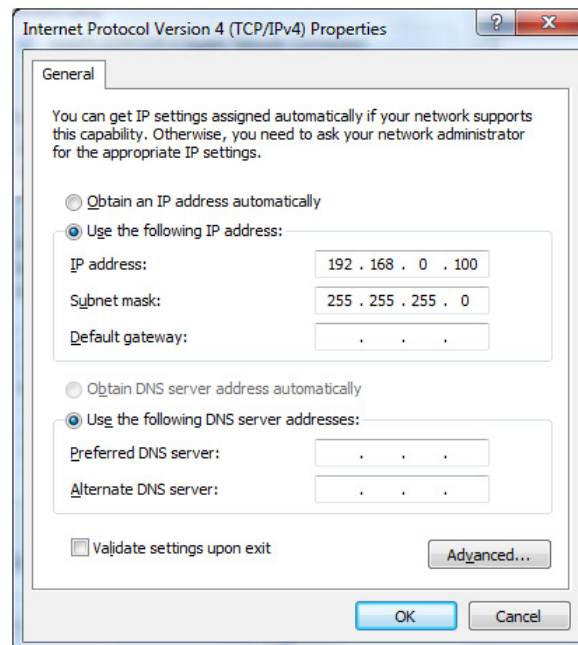


Figure 11-17. The Windows 7 IP settings dialog.

Multiple IP Setup (Dual IP)

Cases exist where the host computer and multiple instruments are unable to reside in the same IP family (subnet). In this case, the host computer can be configured to handle multiple IP addresses.

1. Follow the above directions to set a single static IP address.
2. Once the first static IP address is set, select the **Advanced** button at the bottom of the dialog.
3. Click the **Add** button and enter the additional IP addresses (Figure 11-18).

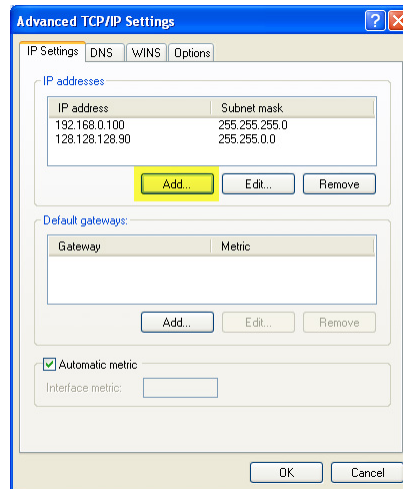


Figure 11-18. Assigning dual (or multiple) IP addresses to a single port.

Measurement Simulation

SpatialAnalyzer allows for particular instruments to be started and run in a simulation mode. This can be quite helpful in planning a job, analyzing setup uncertainty, or simply for training purposes.

To start an instrument in simulation do the following:

Laser Tracker Simulation

All Laser Trackers models (except the Leica AT960, “[Leica AT960 Simulator](#)” on page 315) can be run in a generic simulation mode. To start a laser tracker interface in simulation do the following:


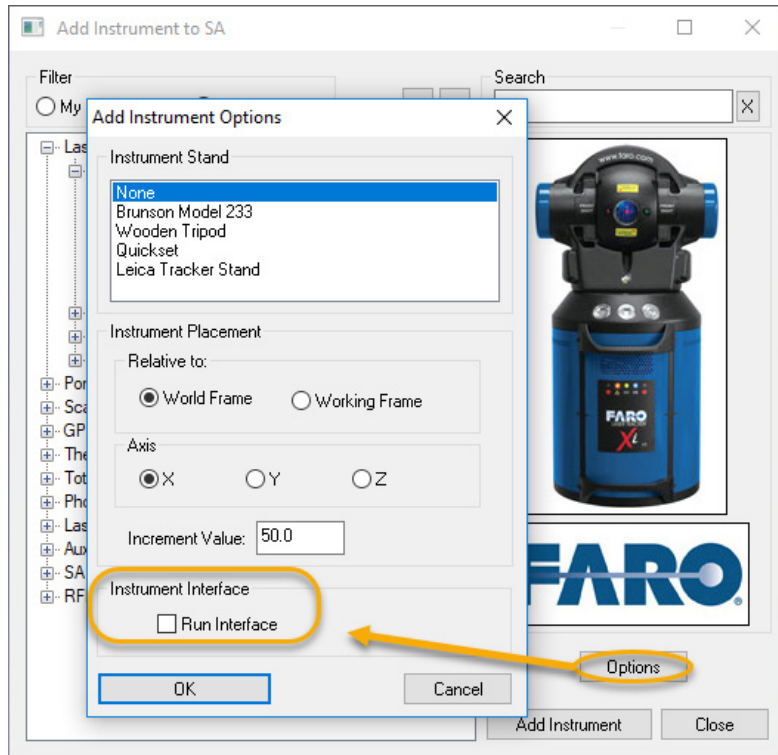
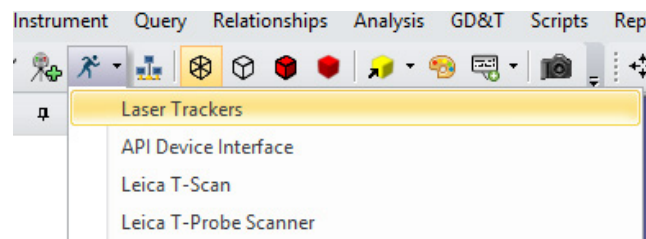
1. Add an instrument model using the instrument add button  or by going to **Instrument > Add**, if an instrument station does not yet exist in your job file. If you have your tracker set to connect automatically this will have to be turned off through the options button in the add instrument dialog (Figure 11-19).

Figure 11-19. Automatic Connection Option in the Add instrument menu must be turned off to run in simulation



2. Next, start the instrument interface by going to **Instrument > Run Interface Module > Laser Trackers** (or **API Device Interface**). Or by using the drop down icon next to the running man (Figure 11-20).

Figure 11-20. Run Interface Module



3. Select your instrument model from the network list by clicking on the name in the list and press **OK**.

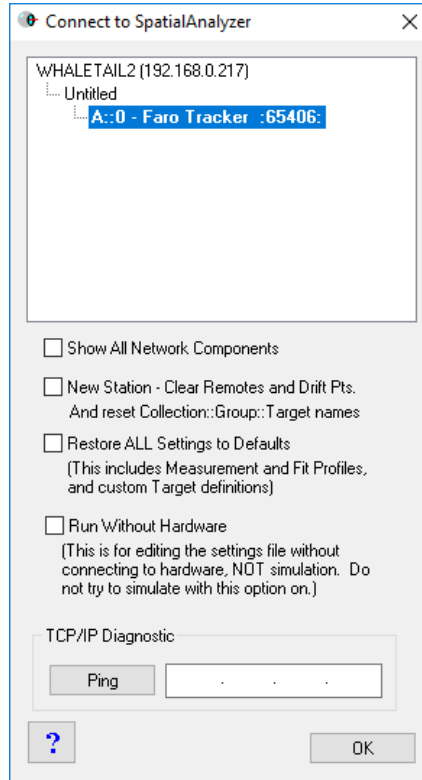


Figure 11-21. Select the instrument model to connect to and press OK to continue

Do not select **“Run Without Hardware”** for simulation. **Run Without Hardware** is a special mode that allows you connect and edit the tracker’s persistence file directly, as if you are connected to your actual instrument. This allows you to edit tooling and measurement profiles for later use, but will not allow measurement simulation.

4. In the Connection dialog uncheck the **“Connect to Tracker”** check box (Figure 11-22).

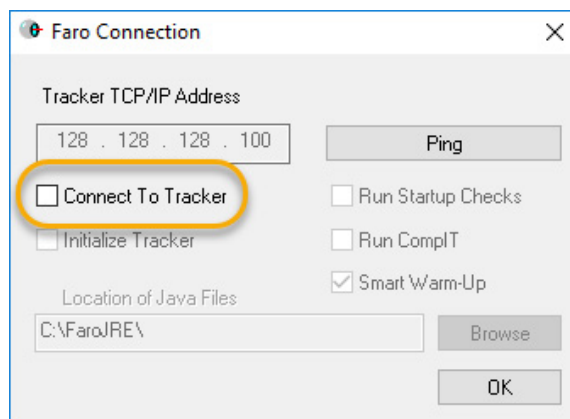


Figure 11-22. Connect to Tracker Check Box in the connection dialog

Once the simulated interface opens you can take measurements, point at targets and interact with the measurement profiles and tooling definitions. However, only a default set of each will be available. Specific Instrument settings are also not available through Simulation.

Leica AT960 Simulator

Leica offers a laser tracker simulator utility for their LMF trackers (AT960, AT600 and AT500). This tool is accessed by adding a AT960/AT600 to a job file and connecting to the instrument in Simulation. When you do so an additional tracker control dialog will open along with the full interface (Figure 11-23).

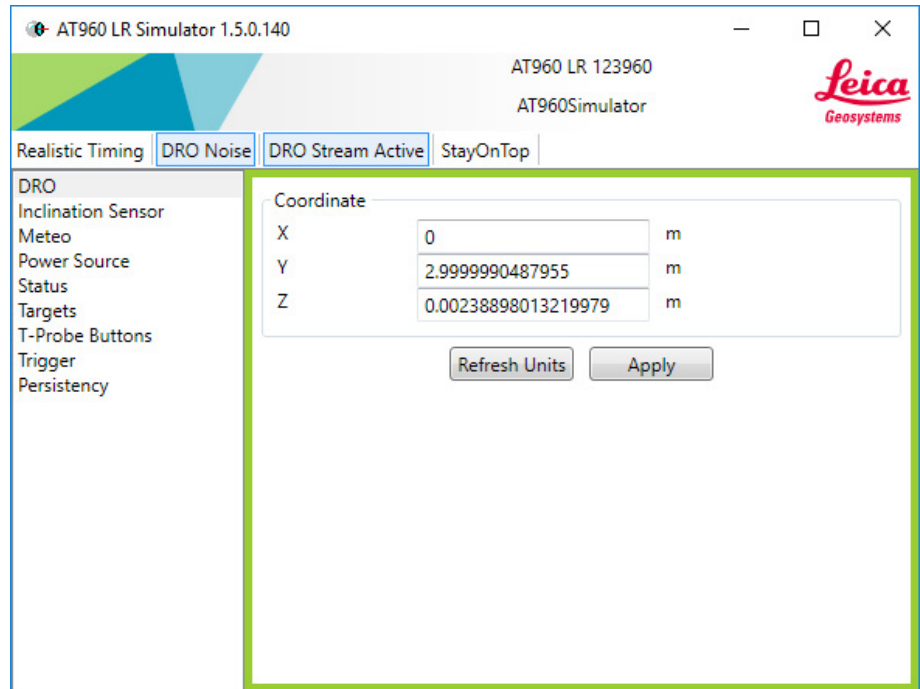


Figure 11-23. AT960 Simulator Control

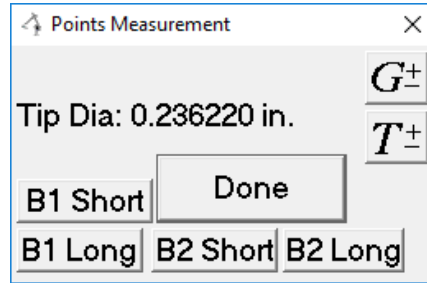
This simulator provides the ability to simulate the tracker directly. You can change beam status, tooling, point the instrument, and simulate error messages in order to observe the resulting response of our SA Laser Tracker interface.

Portable CMM Arm Simulation

Start a generic Portable CMM Arm interface in simulation mode using the same method as described for the SA Laser Tracker interface (**Add instrument, Run Instrument Model>Portable CMM Arms**, Select your instrument model, and uncheck the “**Connect to Arm**” Check box).

When simulation starts an arm button simulation dialog will also open allowing measurements to be triggered by a simulated arm button press (Figure 11-24).

Figure 11-24. Arm Button Simulation Dialog



Arm simulation is more limited than that available for trackers because there are so many links in the kinematic model of an arm. For example the option to Point At a point is not available because there are so many ways to move the arm to get to that point. However, there is a **Simulate Motion** option in the instruments right-click menu that will allow you to manually place an instruments probe or enter a transform.

Scanning is also possible in simulation for arms. A simulated scan will build a series of parallel scan path on a plane using a user defined width, line spacing and rate.

Simulation with Other Instrument Types

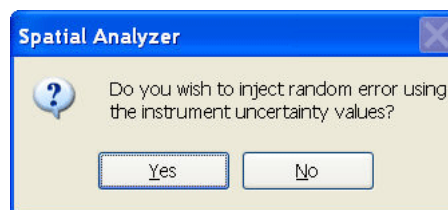
Total Stations, Laser Radar and some photogrammetry systems also offer simulation tools, while still others offer a direct data import function. Typically an option is provided to start in simulation if an attempt to connect is not successful. Refer to the instrument specific quick starts for additional information.

Measurement Fabrication

Imported points can be linked to an instrument station as measurements by using the “Fabricate Measurement” command (**Analysis>Measurement Simulation>Fabricate Measurements**). This will create a simulated observations of the points from the selected instrument, and can be done without starting an instrument in Simulation mode.

Measurement error can be injected to simulate real world conditions. The error is based on the uncertainty model of the selected instrument (Figure 11-25).

Figure 11-25. Measurement simulation.



Taking Measurements from SA

In addition to the standard measurement capabilities of each instrument interface (refer to the SA Instruments Manual) SA has a number of measurement operations that are initiated from within the SA application that can be very helpful in automation applications. These come in three categories:

- **Auto Measure and Batch Measurement.** Auto Measure provides advanced tools for measuring a list of items including individual item selection and iterative measurement. Also in this category are **Batch** measurements which offer full automation by simply measuring a list of items in a single complete pass.
- **Guided Points.** Are designed to lead an operator to a specified location or feature for measurement or to provide proximity based name control using a set of reference items.
- **Proximity Trigger.** Provides a means to add automation to a user guided measurement. This operation guides a user to a target for measurement and, when the probe gets within a proximity threshold, the measurements are triggered at those locations automatically.

Auto Measure

Auto Measure Points

Auto Measure Points provides an automated means to tell a motorized instrument to measure a set of targets on its own. This can be done once or repeatedly. To perform the Auto Measure, select **Instrument>Automatic Measurement>Auto Measure**. Select the group of points to measure and the **Auto Measure** dialog in will appear (Figure 11-26).

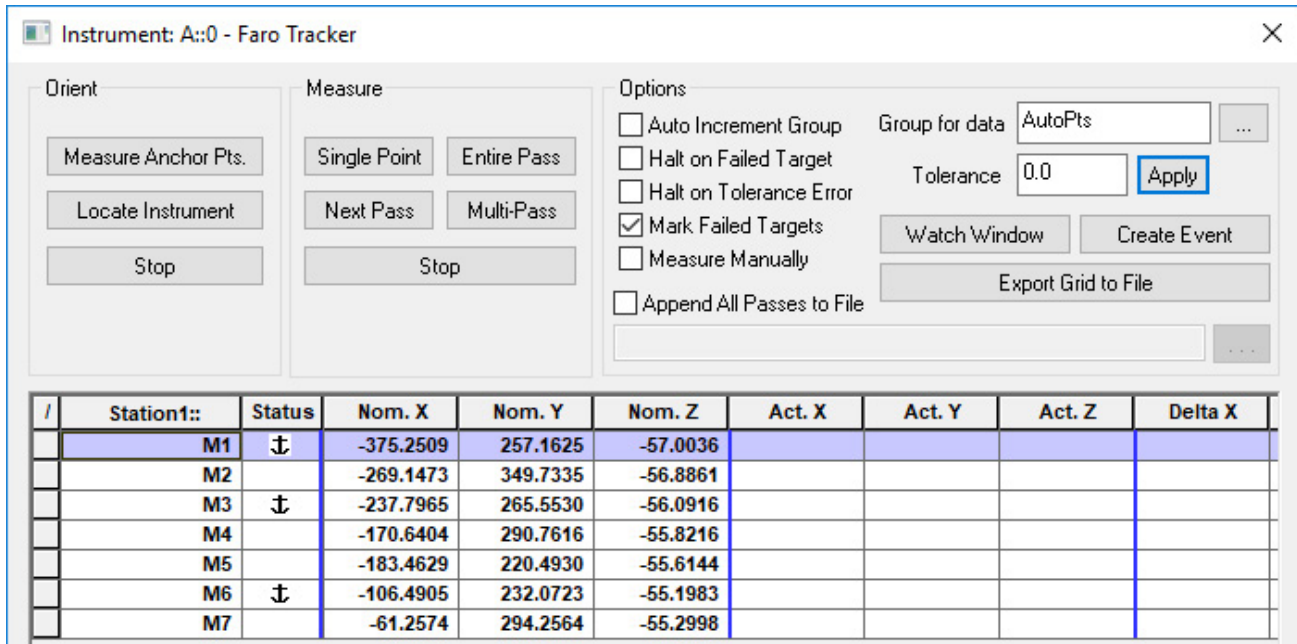


Figure 11-26. Auto Measure Dialog.

The Auto Measure Dialog is organized into 4 sections as follows:

Orient

An instrument must be aligned to the reference points in order to perform an automatic measurement, which is based upon a point at, find, and measure process. It will be told to point at the nominal points in order to find the actual points to measure. Unless already aligned, it can be helpful to mark at least 3 of the points as **Anchor Points**. This is done by right-clicking on the row of the point of interest and changing the **Status** on that point in the Measurement section to define it as an anchor point. The **Measure Anchor Pts.** and **Locate Instrument** buttons can then be used.

Measure

The measure process is initiated and configured from the measurement section of the dialog.

- **Single Point.** By pressing this button a single measurement will be taken at the current reflector/target location and compared to the currently selected target in the Measurement Section. This can be helpful in measuring anchor points as well.
- **Entire Pass.** Pressing the Entire Pass button will trigger a full set of measurements of each of the points included in the Measurement Section.
- **Next Pass.** This function will clear the prior measurement results and reset focus to the first point in the list.

- **Multi-Pass.** This function allows all the points to be measured repeatedly for a specified number of passes and allow for a time delay to be added before each measurement or between passes (Figure 11-27).

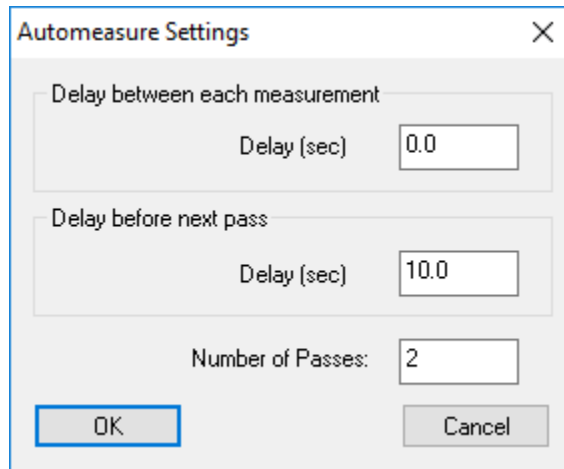


Figure 11-27. Auto Measure Multi-Pass dialog and controls.

- **Stop.** Stop controls are available in both the Orient and Measure sections in order to exit a measurement process when desired.

Options

A selection of options are available for the auto measure process. These options can be defined as default by using the Right-click option on the **Apply** button. These include:

- **Auto Increment Group.** The option increments the point group each time a pass is complete such that passes are saved separately rather than additional observations on the same points.
- **Halt on Failed Target.** When marked the auto measure process stops when a target fails.
- **Halt on Tolerance Error.** When marked the auto measure process stops when a point is out of tolerance.
- **Mark Failed Targets.** This option marks targets as failed and will not attempt to re-measure them in subsequent passes.
- **Append All Passes to File.** Record points to a specified .csv formatted ascii file.
- **Group for data.** Name of the point group used to record the observations.
- **Tolerance.** Tolerance used for each point.
- **Watch Window.** Open a watch window displaying point to point deviations between the nominal and measured points.
- **Create Event.** Saves the currently displayed measurement

window as an event.

The measurement section of the dialog displays the current status of the last pass through the nominal points. This includes both the nominal, actual, and delta statics in a tabular form just as it will appear in an event or ascii export.

The active point (row) is highlighted and the highlighted point will also be the target the instrument measures if a single point measurement is taken. Points can also be marked as anchor points (see “[Orient](#)” on page 318) and **Ignore** such that they are not measured in subsequent passes.

Instrument>Automatic Measurement>Auto Measure (streamlined) provides a much simplified control dialog without a full measurement section display.

Auto Measure (Streamlined)

Auto Measure (Streamlined) provides a simple automated means to tell a motorized instrument to measure a set of targets on its own. To perform the Auto Measure (Streamlined), first align your instrument then select **Instrument>Automatic Measurement>Auto Measure (Streamlined)**. Select the group of points to measure and the **Auto Measure** dialog in will appear ([Figure 11-26](#)).

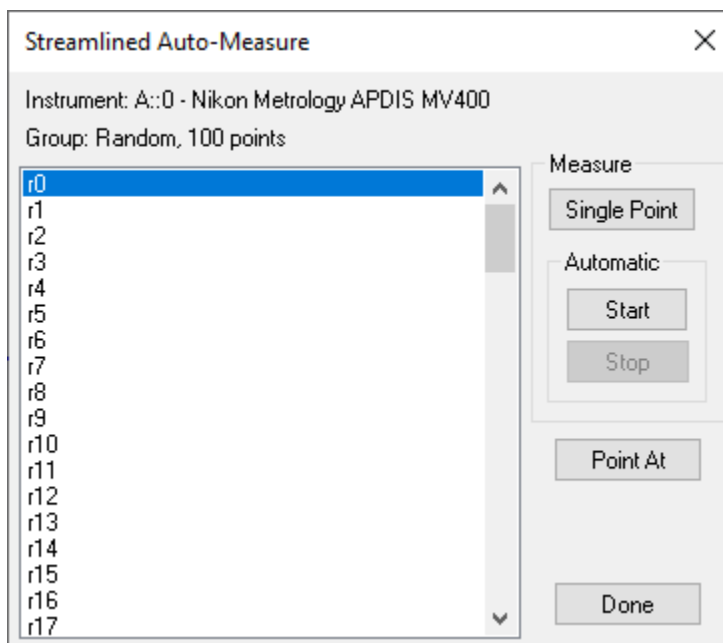


Figure 11-28. Auto Measure (Streamlined)

The Auto Measure (Streamlined) dialog allows you to select a single point from the list and use the **Single Point** button to measure that point. You can also use the **Point At** to drive the laser to the nominal location. The **Automatic Start** and **Stop** controls allow you to begin measuring at the selected point and automatically progress through the list.

Auto Measure Vector Group (Multi-Measure)

Vector groups provide an advantage or points in that they include a direction component. This allows measurements to be taken at a specific location along a vector axis even if there is a vertical offset.

Once an alignment has been performed select **Instrument>Automatic Measurement>Auto Measure Vector Group (Multi-Measure)**. This command provides the means to measure a vector group with many of the controls offered through **Auto Measure** points. This command begins with vector selection and then an option to configure a vector group displaying the point to nominal vector deviations ([Figure 11-29](#)).

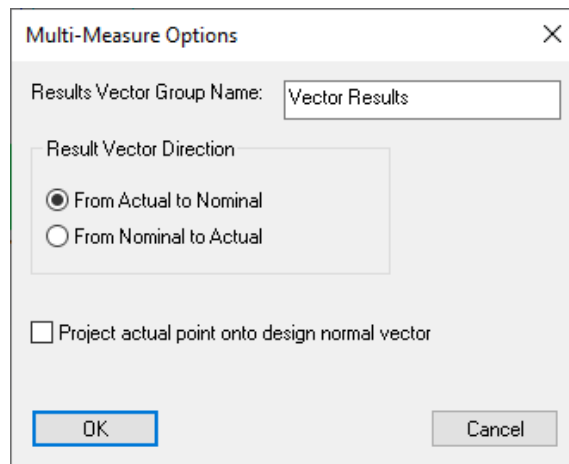


Figure 11-29. Auto Measure Vectors - Deviation vector configuration

The Auto Measure Vector Group dialog will then appear ([Figure 11-30](#)). Settings at the bottom of the dialog control error conditions such as when to skip a vector.

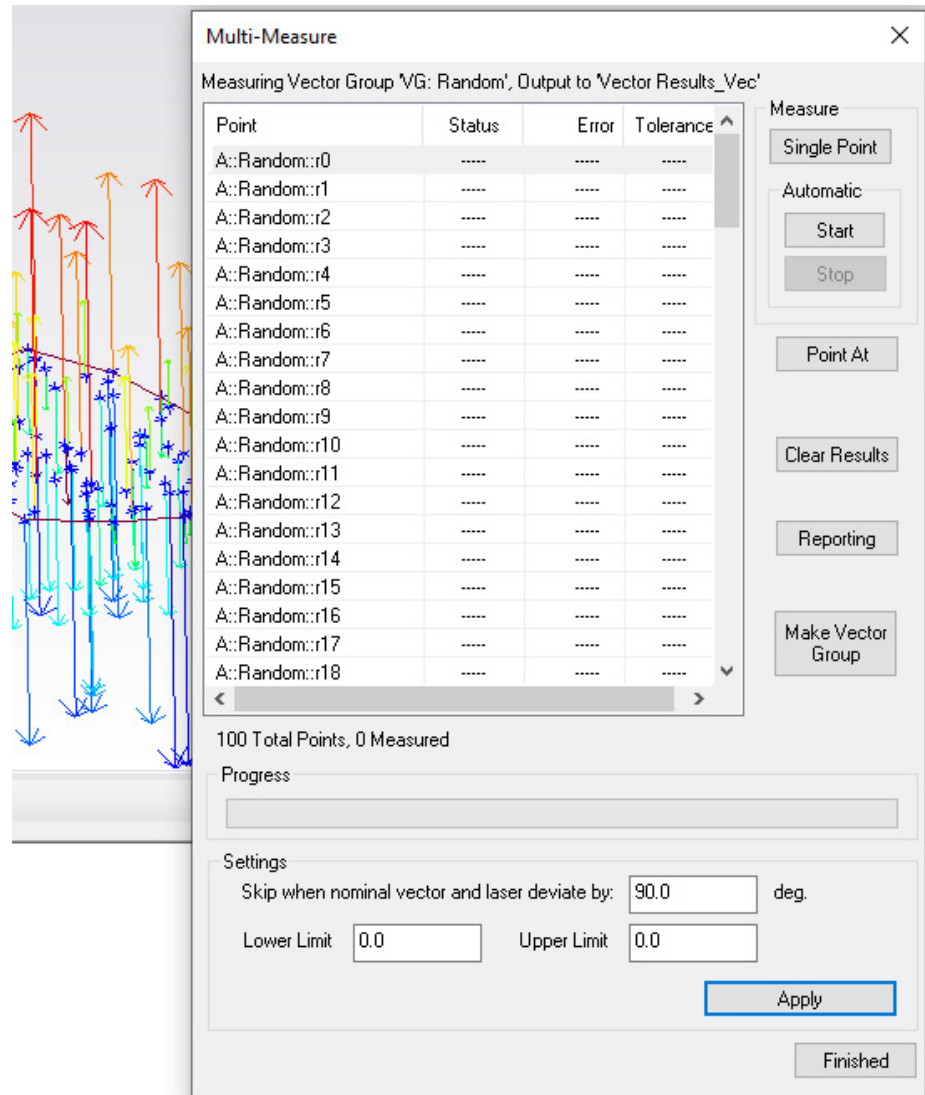


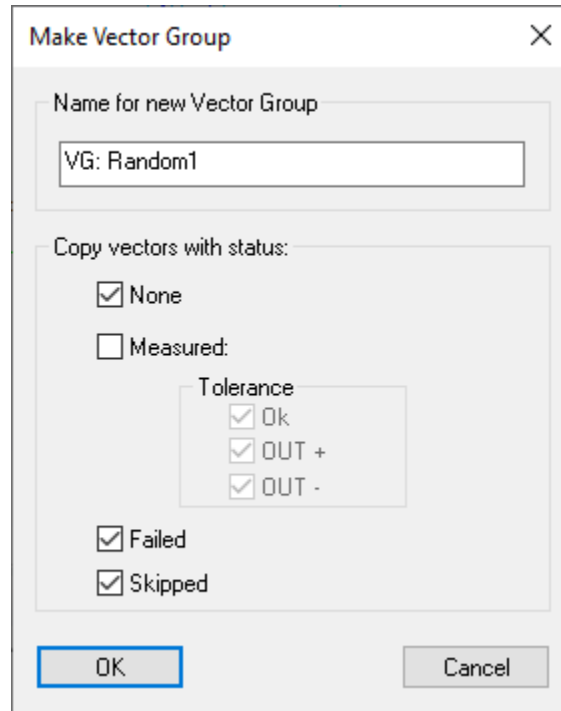
Figure 11-30. Auto Measure Vector Group

Controls for vector measurement directly follow those available for point measurements. The **Single Point** button provides a means to measure the current selected vector. The **Automatic Start** and **Stop** controls allow you to begin measuring at the selected vector and automatically progress through the list.

The **Clear Results** offers an option to clear the current vector measurement status in the display, while **Reporting** offers direct access to the deviation vector group properties dialog.

The **Make Vector Group** button provides several options to generate additional vector groups for re-measurement purposes or reporting (Figure 11-31).

Figure 11-31. Auto Measure Vector Group - Make Vector Group Options.



Vectors are automatically skipped if the error condition settings are exceeded.

- Angle between the vector and measurement beam must be below the specified threshold in degrees.
- The point to vector origin must be within the Lower and Upper limit thresholds (entering 0 for both values skips this validation).

Measure Batch of Vectors

Measure Batch of Vectors was added with a Nikon Laser Radar, Leica ATS600, or MS60 in mind and drives the device to the intersection point between a nominal vector and the measured surface to record a point. It is a fast and efficient measurement of the entire group. The settings for the measurements are controlled from the individual instrument interface.

Measure Batch of Features

Measure Batch of Features was added to support the Nikon APDIS Laser Radar. It allows selection from a list of compare to nominal GR-Features and selection of the active instrument. Currently GR-Circles and GR-Slots are supported. The instrument then processes this list, measuring each feature using its Gauging Engine and the specified measurement definition, which can be selected through the right-click *Inspection > Instrument Configuration* menu for each feature.

Guided Points

Measure Guided Points

Measure Batch of Points provides a tool to guide a user to measure within a specified proximity to a set of nominal points selected in a user defined order. It provides control over an audible and visual proximity indicator and will zoom the view as you get closer the active point (Figure 11-32).

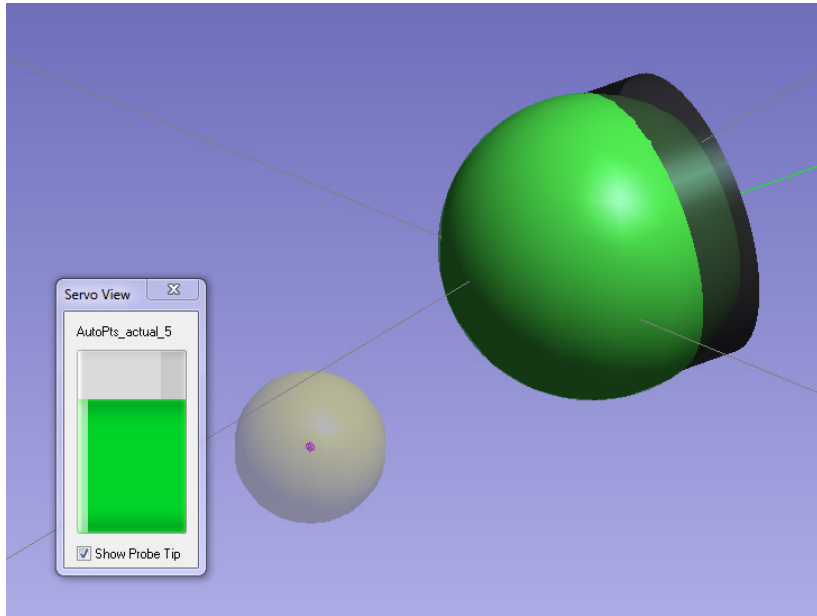


Figure 11-32. Measure Batch of Points graphical guide tools

Auto-Correspond Measured Point to Closest

The command **Instrument>Automatic Measurement>Auto-Correspond Measured Point to Closest Point** offers a means to measure points and automatically name those points based on a proximity threshold to the reference points in a new group (Figure 11-33). It provides a simple alternative to the Measure Nominal Points alignment option.

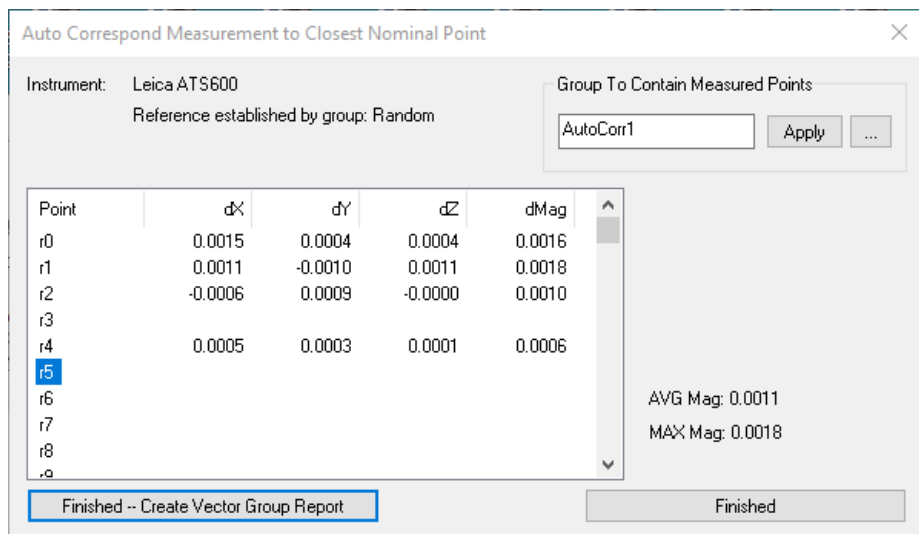


Figure 11-33. Auto Correspond Points to Nominal Points

Proximity Triggered Points

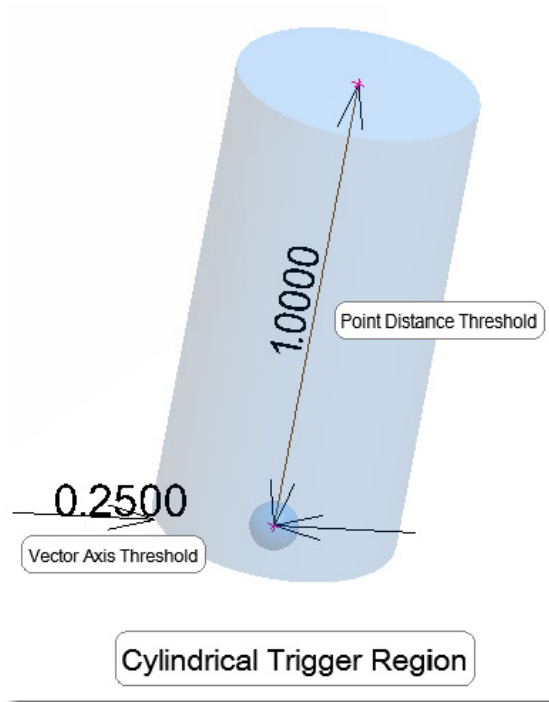
It is often necessary to measure points on a part at precise locations. Proximity Triggers is a special utility for this purpose. It places an instrument in a continuous scan mode and captures a measurement when a probe reaches a specified proximity to a nominal point or vector (**Instrument>Automatic Measurement>Auto-Correspond with proximity trigger**>).

Proximity Triggers asks for a reference point group and lay out a set of bull's-eyes on those points to identify the nominals that have not yet been measured. The points can then be measured in any order and the **Measure each point only once** check box can be used to determine if only a single measurement is capture or if multiple observations are recorded as the probe gets closer and closer to the nominal within the tolerance zone. Additional options include:

- **Group to contain measurements.** This is the name used for the measurements recorded during the proximity trigger operation.
- **Make Vectors while Measuring.** This option allows you to built point to nominal point vectors as part of the measurement process.
- **Project result to nominal vector.** This option ensures that only differences in vertical displacement are displayed by the vectors, not the displacement of the measured point within the tolerance zone.
- **Point Distance Threshold.** Defines the maximum distance from the nominal point at which a point will be triggered. And with vector measurements, defines the length of the cylindrical trigger zone.
- **Vector Axis Threshold.** The maximum distance from the vector axis a point will be triggered (applies to vectors only). This value determines the diameter of a cylindrical trigger zone.

You can use vectors to trigger your measurements rather than the points. Doing so allows you to trigger measurements when you get within a proximity to the vector axis, eliminating any issues with getting close enough to the points you want to measure. The proximity trigger zone for vectors is defined as cylindrical zone with a length equal to the point distance threshold and the width equal to the Vector axis threshold. This fine control of both the width and height of the trigger allows the user to precisely trigger points even in tight regions or slots between adjacent CAD surfaces ([Figure 11-34](#)).

Figure 11-34. Cylindrical Trigger Region



Once the routine starts. You can simply run a probe over the part and it will automatically trigger measurements for specific locations every time you get within the trigger distance threshold, ticking off points locations as you go keeping track for you of what points have been measured and which still need to be taken. Once the majority of the points have been measured you can bring up a Graphical Guide window that will help you navigate precisely to the remaining locations, finishing a precise grid measurement of a part.

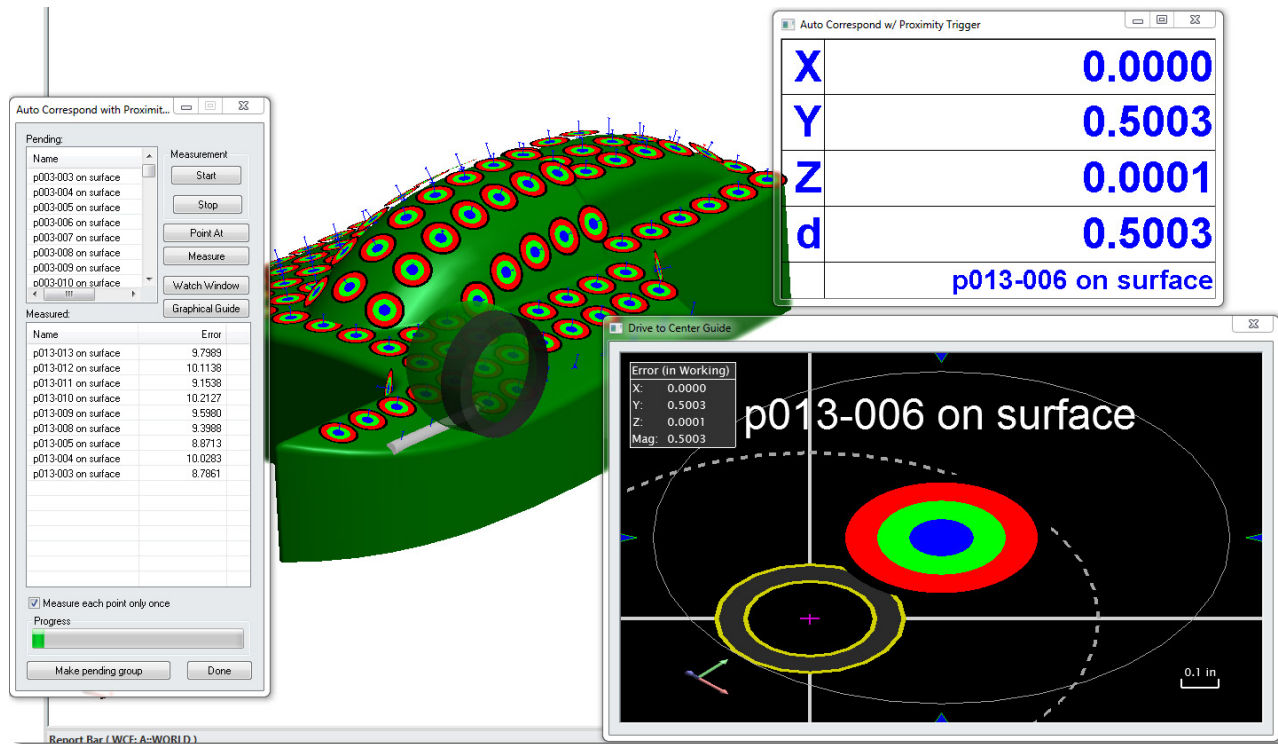


Figure 11-35. Proximity Measured Points in progress

Proximity Measurements with a Scanner

Proximity triggers can also be used with a line scanner. Both Portable CMM arm scanners and tracker-linked scanners (such as the Leica T-scan or LAS) can be used with proximity triggers.

There is a separate option in each instrument's interface properties to enable this function with a scanner. An example of the setting in the CMM arm properties is show here (Figure 11-36):

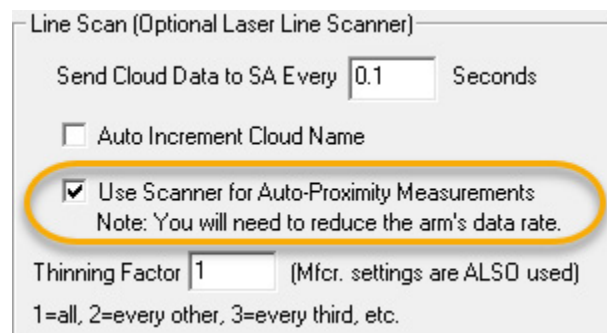


Figure 11-36. The Proximity Settings found in the properties of a CMM arm.

With the **Use Scanner for Auto-Proximity Measurements** option enabled an operator can start the proximity triggers process and begin to scan. The cloud points will only be record when they are within the tolerance zone on a part.

Proximity Measurements with a Shank Probe

Shank Probe measurement can also be acquired using proximity triggers. This provides a very effective method to measure edge points quickly by sliding the shank along the part. This is currently only supported with the T-probe. Refer to the [Instruments Manual](#) in the [Measuring with Laser Trackers](#) chapter.

Synchronized Measurement

It can be helpful to be able to measure from more than one instrument with the press of a single button. There are a number of ways within SA to perform this task.

Master - Slave

This function allows you to select a single instrument as the Master and a second as the Slave. Once started in this mode, a measurement taken from the master instrument will trigger a corresponding measurement from the Slave. The point measurements and the lag time (in seconds) will be displayed with each measurement.

An optional Best-Fit alignment can also be performed as part of this utility by checking the [Locate one of the instruments](#) check box. When you press the Done button, after measurements have been taken, this will close the synchronized measurement utility and automatically open a Best-Fit dialog allowing alignment of the selected instrument.

Polling

Polling provides even more options for synchronized measurement applications. It allows multiple instruments to be selected at one time and opens a master control dialog to trigger measurements.

Figure 11-37. Synchronized Measurement Polling control dialog

Instrument	Delay
A::0 - Faro Vantage	
A::1 - Romer Absolute 7x20SI/SE	
A::2 - API Radian	
A::4 - Leica Nova MS60 Total Station	

Measured Point Data Placement

Place each instrument's measurements in a different group:
Group Suffix: (inst#)_s

All instruments measure the same physical point (bundle solution)
Group Name:

Next Measured Point:
Target Name

Measurement Mode

Single Point

Stream of Points

Delay between measurements (sec):

- **Single Point (Measure Now).** When the **Measure Now** button is pressed it will trigger a point from all of the selected instruments and record those points in the specified point groups.
- **Stream of Points (Begin).** The Stream of Points **Begin** button will start a continuous measurement from all of the instruments. A time delay can be added between measurements to space the points using the **Delay between measurements (sec)** option.

Measurement requests from SA are handled separately from measurements triggered directly in the instrument interface. For example, discrete tracker measurements use a specific measurement profile that can be found under **Settings>Tracker>SA Interaction>Measurement Requests>Single Point Parameters**.

Synchronized Arm/Tracker Scans for Alignment

SA offers the ability to use an arm to start and stop a synchronized scan operation, much like the “Faro ScanArm” alignment. This allows an arm to be quickly located in different positions using a laser tracker.

To do, the following steps should be taken.

Connect and Configure the Instruments

This procedure requires at least 2 instruments be connected at one time and that a common target can be monitored. To locate an arm this typically means a reflector needs to be mounted as a probe tip and calibrated.

Tracker Settings

- The tracker should be placed in a spatial scan mode. An increment of 0.10" has been shown to work well.

Arm Settings

- Open the arm settings and make sure that the "Query Rate During Measurements" is set to a reasonably high rate, 100/sec works well.
- Right-click on the scan button and enable the option to "Control Tracker Meas with Stream On/Off" (Figure 11-38).

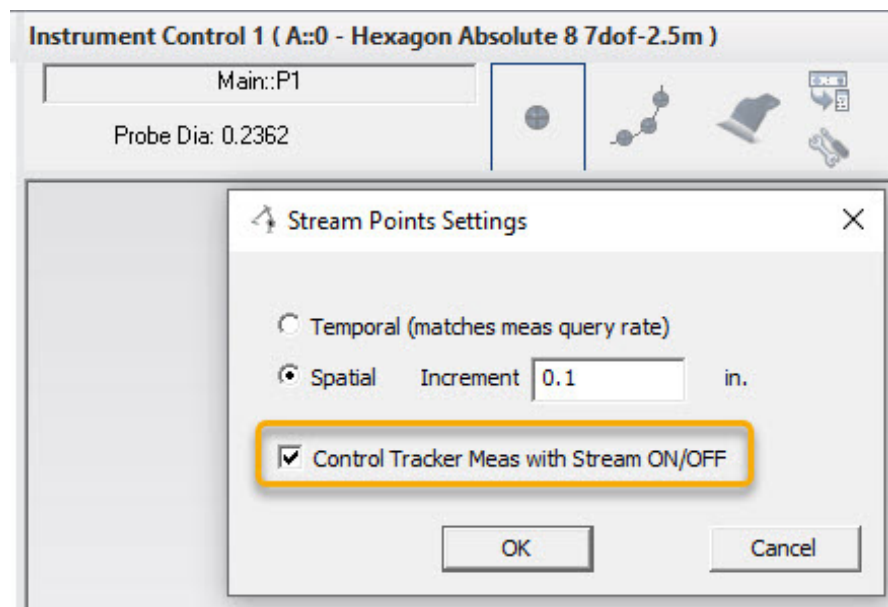
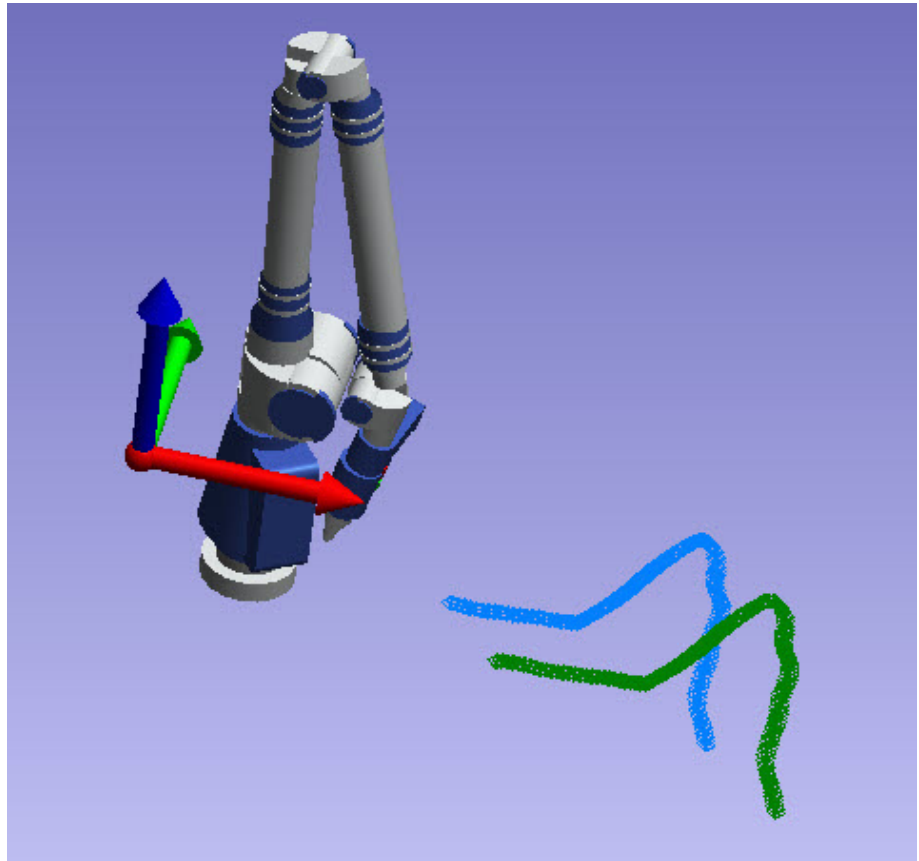


Figure 11-38. Enable Arm Control by right-clicking on the scan profile

Measurement and Alignment Process

1. Lock onto the reflector with the Laser Tracker and make sure it is tracking the arm's movements.
2. Start measuring by pressing the trigger button on the arm. Trace a path with the arm that roughly defines the 3 axes of a frame and check that the scan was duplicated by laser tracker.
3. Stop the scan process by again pressing the button on the arm (Figure 11-39).

Figure 11-39. Scanning a path to locate the arm. A path along 3 axes of a fame works well.



4. Fit a spline through the measurements taken by the arm using **Construct>B-Spline> from Points** and the following settings:

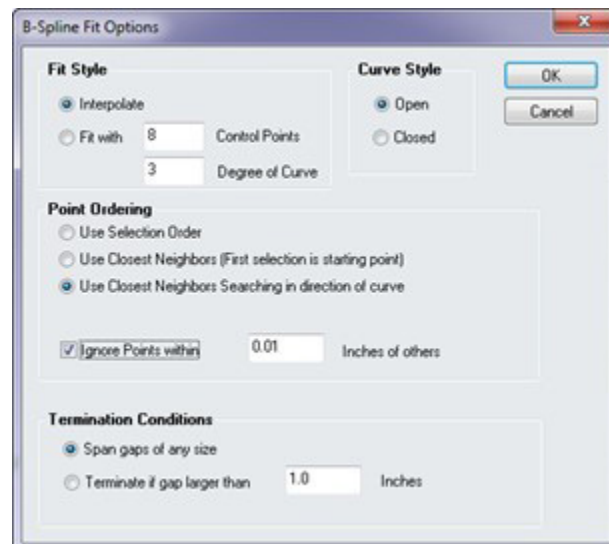
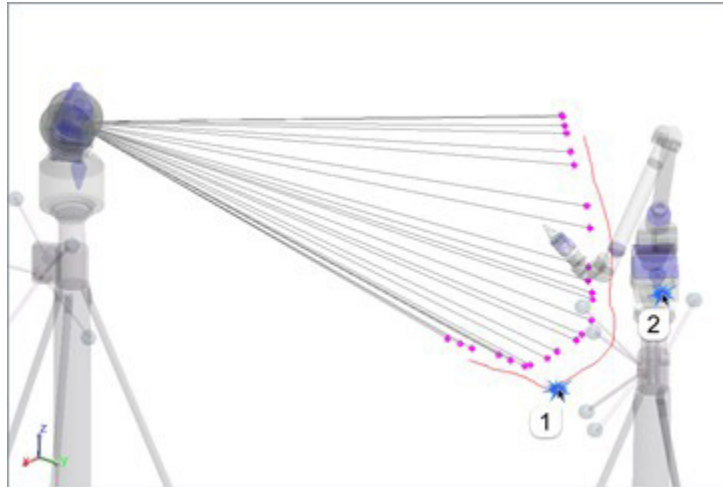


Figure 11-40. B-Spline Settings

5. Build a relationship (**Relationship>Group to Objects**) between the Tracker measurements and the spline built from the arm measurements.
6. Run the optimization by selecting **Relationships>Move Objects by Minimizing Relationships**, and then selecting the b-spline as the

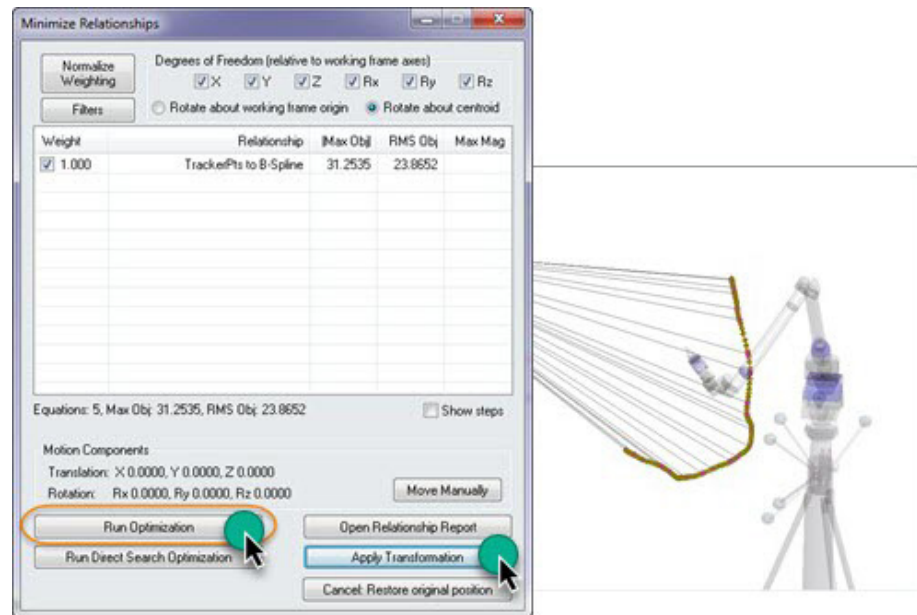
object to move, press enter, and then select the arm to move (Figure 11-41).

Figure 11-41. Selection order when prompted for the Optimization.



7. When the Minimize Relationships window opens you should be able to simply press the Run Optimization button followed by the Apply Transform Button.

Figure 11-42. Running the Optimization and Applying the Fit.



This entire process can easily be automated and an example MP is available on the kinematics.com.

Crib Sheets and Measurement Templates

SA provides a wide spectrum of tools for measurement template construction. The simplest for of the is a Crib Sheet which provides a means of recording target names and the measurement profiles use to measure them along with notes for an operator.

More advanced measurement template construction is available

both in Geometry Relationship Inspection (see “[Feature Inspection](#)” on page 500) and in GD&T Inspect (see “[GD&T Workflow](#)” on page 664). Full Measurement plan integration is also available through SA Ultimate.

Crib Sheets

Crib Sheets provide a simple means to automatically set the Collection, Group, and Target name, as well as the Measurement Mode, for an operator as part of a standard task ([Figure 11-43](#)).

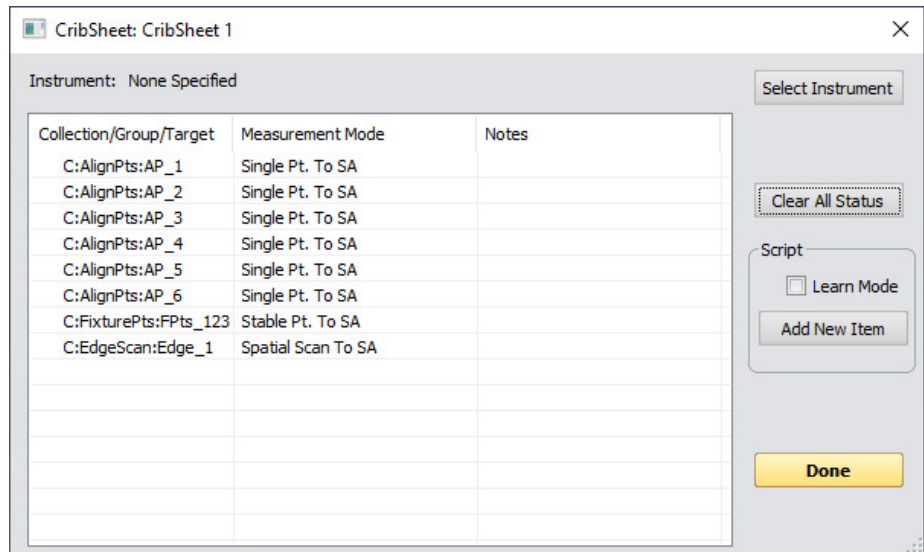


Figure 11-43. Crib Sheet Dialog

Each entry in the Crib Sheet can be considered a command to the instrument and will set both the Collection, Group and Target as well as the measurement mode and any user notes. Therefore, an instrument selection is required.

Crib sheets are set up to record new entries when the *Learn Mode* option is enabled, and the **Add New Item** button provides a means to manually add an entry.

Crib sheets can be run from the properties dialog using the **Measure** button once an instrument has been selected. The Item selection in the list can be used to run steps in any order you wish and completed items will be marked with a check mark.

Monitoring

Watch Windows and Transformation Tracking

Watch windows are discussed in their own chapter “[Watch Windows & Transform Tracking](#)” on page 381.

Tools

Scale Bars

Adding a Scale Bar

After measuring the necessary points, the user should create a scale bar. For instruments that measure angles and distances, it is not necessary to use scale-bars in the bundle process but they are required for theodolite networks. Without them, the instruments will scale freely. To do this you will need to utilize two points that have a known distance from each other. They can be two points designated only for the scale or common points, but either way they need to be recorded from the theodolite before the scale bar is created. Once you have the actual distance select **Construct > Scale Bar > Pick 2 Points**. At this point, select the two points that you just measured the distance between. The window below will appear (Figure 11-44). Select edit to change the length to the actual measured amount. After hitting **OK** the new length should populate. You should also notice that a scale bar is now listed in the tree bar.

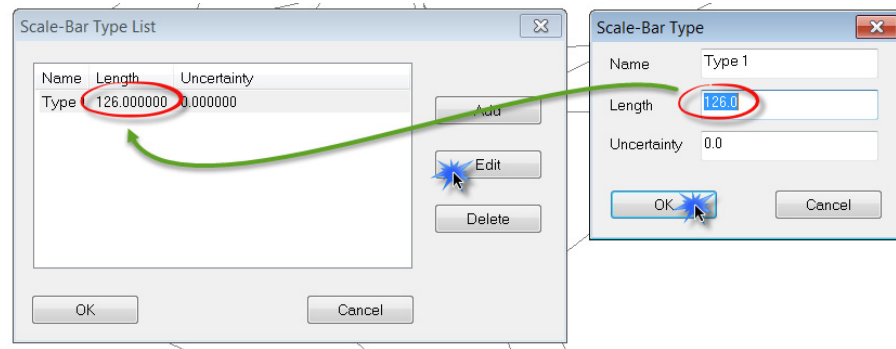


Figure 11-44. Scale-Bar Type List

SpatialAnalyzer: Instrument Scaling Using Scale Bar and CTE - <https://www.youtube.com/watch?v=XOG5du8oOR0>

Perimeter

Perimeters appear graphically much like a series of connected lines that define a 3D volume. Perimeters are used with scanning devices to define a region of interest for measurement purposes. For more information refer to “Perimeters” on page 287.

Target Detection

Survey targets locations can be extracted from clouds using the embedded intensity information. To use these commands, graphically select cloud points containing the target you wish to extract.

- **Survey Points from Triangles.** This command can be used to

extract the center point from two intersecting triangles.

- **Survey Points from Circles.** This command can be used to extract the center of scan circle targets.

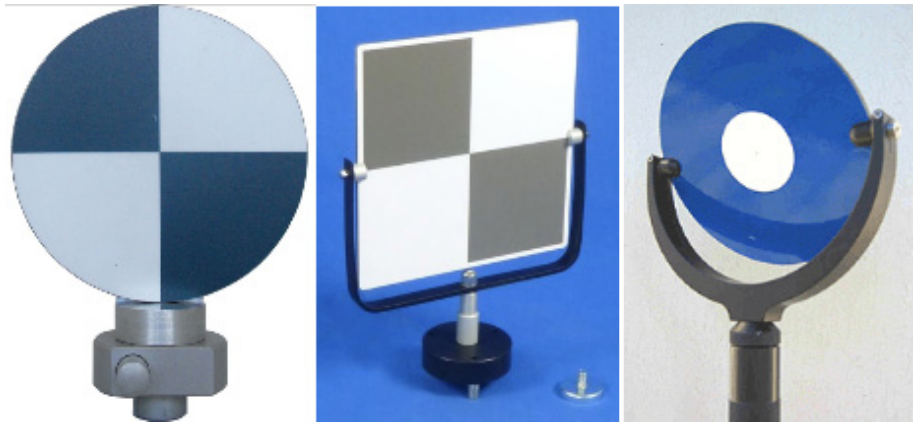


Figure 11-45. Example Triangle and Circular Intensity Targets

Scan Target Detection in SpatialAnalyzer -

<https://youtu.be/BSBPYRHf0>

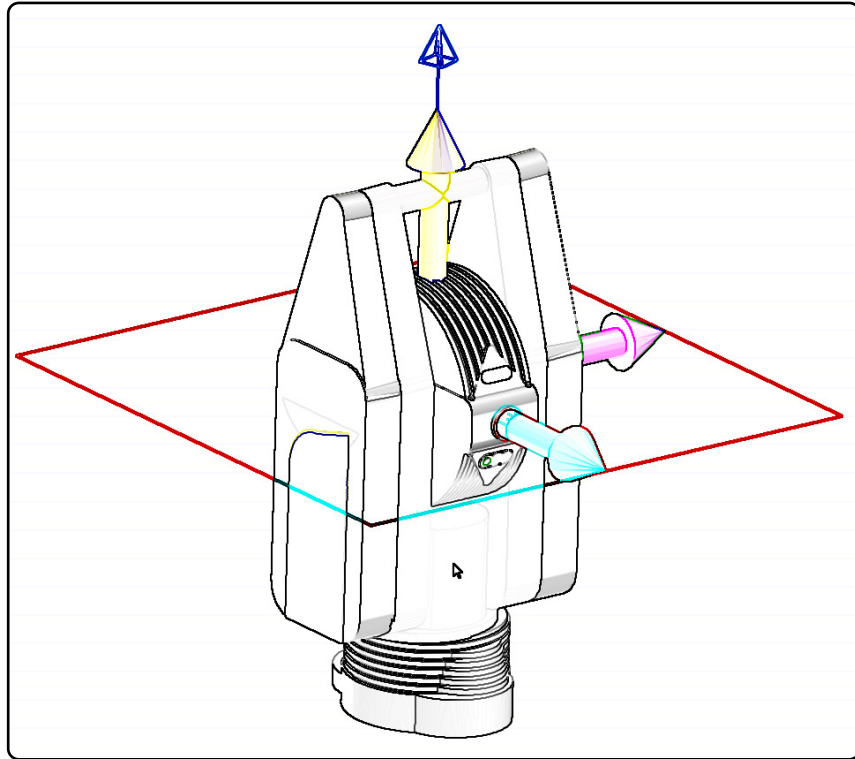
Gravity Measurements in SA

Measuring Level

When a gravity measurement is take within SA, this measurement is made relative to the instrument's standing axis and is recorded as a Frame.

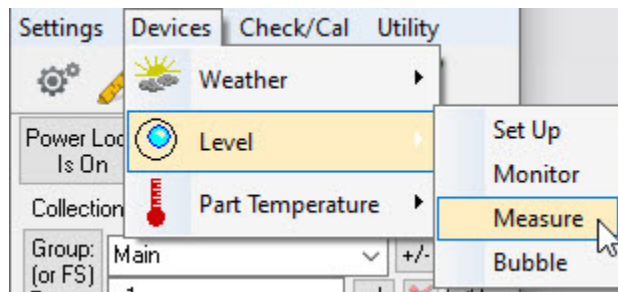
The Z-axis of this frame points upward along the gravity vector, and the XY plane of this frame defines the gravity plane. This frame measurement is saved directly on the instrument's head such that the origin of the level frame is at the center of the instruments *base frame* and at the center of rotation for all angular point measurements (Figure 11-46).

Figure 11-46. Gravity Measurement showing the level plane and gravity vector concept.



To measure level you will need to trigger a level measurement through the instrument interface. For laser trackers this is found under *Devices > Level > Measure* (Figure 11-47).

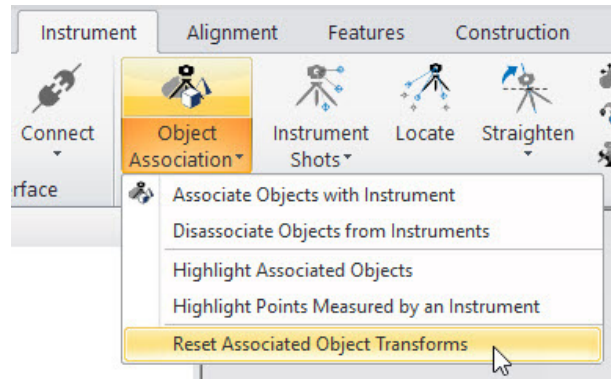
Figure 11-47. Measure Level Option within the Tracker Interface



You can measure level at any time and you can also measure level multiple times from the same station. Like any measurement there will be some minor variation between these level measurements. Level measurement frames are automatically associated to the instrument station. That association means that if you were to move an instrument or locate it, the measured level frame automatically moves with the instrument, as an additional measurement, and will maintain its orientation with respect to the instrument's base frame.

If the level frame is moved for some reason or gets disconnected from the instrument this connection can be re-established using the option *Reset Associated Object Transforms*. This works with any cloud or 6D measurement as well (Figure 11-48).

Figure 11-48. Option to reconnect a level measurement if the association is broken in some way.



If you wish to average multiple level measurements you can use the frame construction command (*Construction > New Frames > From Objects > Frame as Average of Object Frames*). However, this average frame will not be automatically associated with the instrument. To link it to the instrument station use: *Instrument > Object Association > Associate Objects with Instrument*. This will link the new frame to the instrument station as a measurement.

Level Compensation Verses Measurement

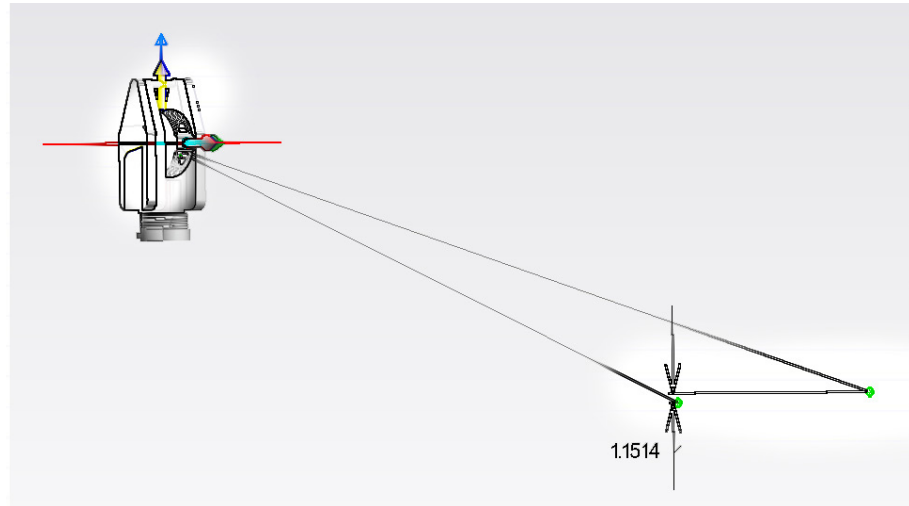
It is important to understand that taking a level *measurement* does not change your measurements or compensate the instrument, it simply records the current orientation of gravity with respect to the instrument's position. An additional step is required in order to perform a level *compensation*.

Basic Level Comparisons

An additional advantage of measuring level as a frame is that performing level comparisons from a single instrument station can be quite easy to perform.

To compare the relative elevation of measurements with respect to gravity you can measure level and then set the level measurement frame as the working frame in the job file. When you do so, the coordinates of all measurements are reported with respect to this gravity frame. The Z component values will then correspond to the level plane displacement and can be directly compared (Figure 11-49).

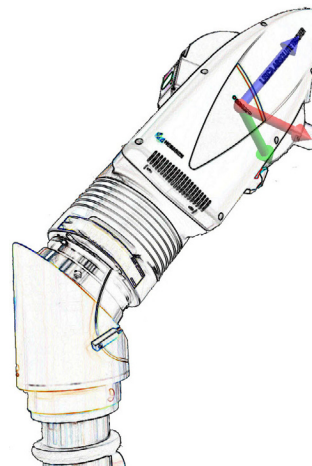
Figure 11-49. Comparing the relative height of two measurements using a level frame as the working frame.



Instrument Level Frame vs Base Frame

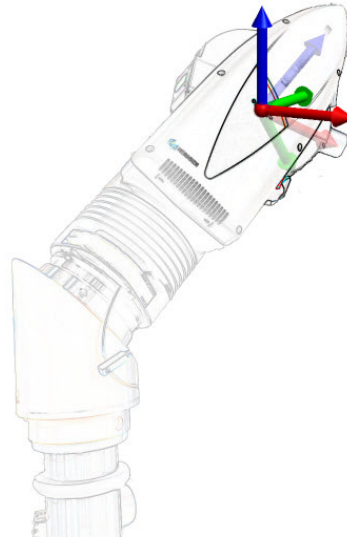
Measurements are recorded from an instrument with respect to the "Base Coordinate" of the instrument. This "Base" does not refer to the attachment to the tripod, but instead refers to the centroid or origin of the measurement process. This location is typically at the center of rotation of the encoders as shown here (Figure 11-50):

Figure 11-50. Instruments Base Frame remains the same regardless of the mount.



A level measurement is recorded relative to the instruments base frame just like all point measurements. A measurement is taken and then a frame is created at the same origin point but tilted such that the Z-axis aligns with gravity. The image below (Figure 11-51) depicts the resulting level measurement frame had a measurement been taken in the same orientation (level measurements are typically only possible when an instrument is nearly level, so this is for illustration only).

Figure 11-51. Level Measurement frame recorded at the instrument's base and tilted to align with the gravity vector.



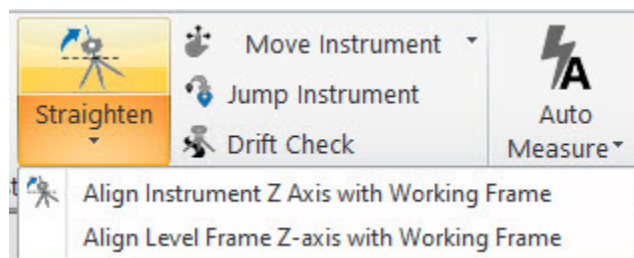
Alignments with Level

It's often necessary to align multiple instruments as part of a job that includes level measurements. There are two ways to handle this sort of scenario.

1. *You can perform a network alignment as usual.* Level measurements are tied to the instrument that measured them and will move with the instrument during an alignment. This means that they will correctly indicate the gravity direction after an alignment is complete. A comparison of level measurements between stations can then be an indicator if there is any issue with the alignment and can be averaged to give you a best indication of gravity for the job.
2. *You can first straighten your instruments with respect to the working frame and then hold level during an alignment.* This option works great in situations where you don't have enough measurements to constrain the alignment such as measuring in a tight hallway.

In order to *hold level* during an alignment an instrument must first be straightened so that its level measurement frame is aligned with the working gravity frame. This is accomplished by using the **Instrument > Instrument Station Controls > Straighten** commands (Figure 11-52).

Figure 11-52. Straighten commands are used to pre-align level measurements with the working frame in order to hold level during an alignment.



- **Align Instrument Z Axis with Working Frame.** This command

is only used with Total Stations or other instruments with an active compensator or stations that have had a level *compensation* performed.

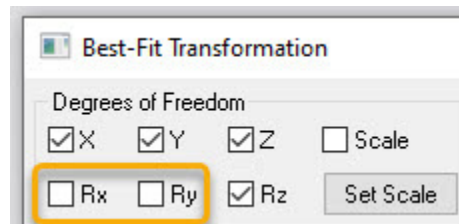
- **Align Level Frame Z-axis with Working Frame.** This second option is used for instruments that record a level measurement frame. It gives the opportunity to select the level measurement to use during alignment. This is the typical option for laser trackers.

Once straightened most alignments offer an option to hold level. There are either of two different methods to do so depending on the alignment.

Holding Level - Degrees of Freedom

In order to hold level during an alignment process you need to restrict the degrees of freedom using in the alignment process. Some alignments such as Measure Nominal Points includes a “Hole Level (Rx Ry off)” check box. What this does is turn off the *Rx* and *Ry* portions of the full degree of freedom selection offered in alignments such as Best-Fit (Figure 11-53).

Figure 11-53. By turning off *Rx* and *Ry* in the Degrees of Freedom section of any alignment you can prevent rotations that would cause an instrument to rotate out of level



It can be helpful to double check that your instrument is still correctly oriented. To do so, select the level measurement frame measured by the instrument in the tree and look at its transform in the report bar. Its important to look at the level measurement not the instrument's transform because your measurement frame is the gravity reference. When you look at the transform check the Rotation (deg) row and verify that the *Rx* and *Ry* values are exactly zero. In Figure 11-54, the yellow *Rx* and *Ry* values indicate that the tracker has been tilted off axis relative to gravity. You will need to use the command *Align Level Frame Z-axis with Working Frame*. To correct for this.

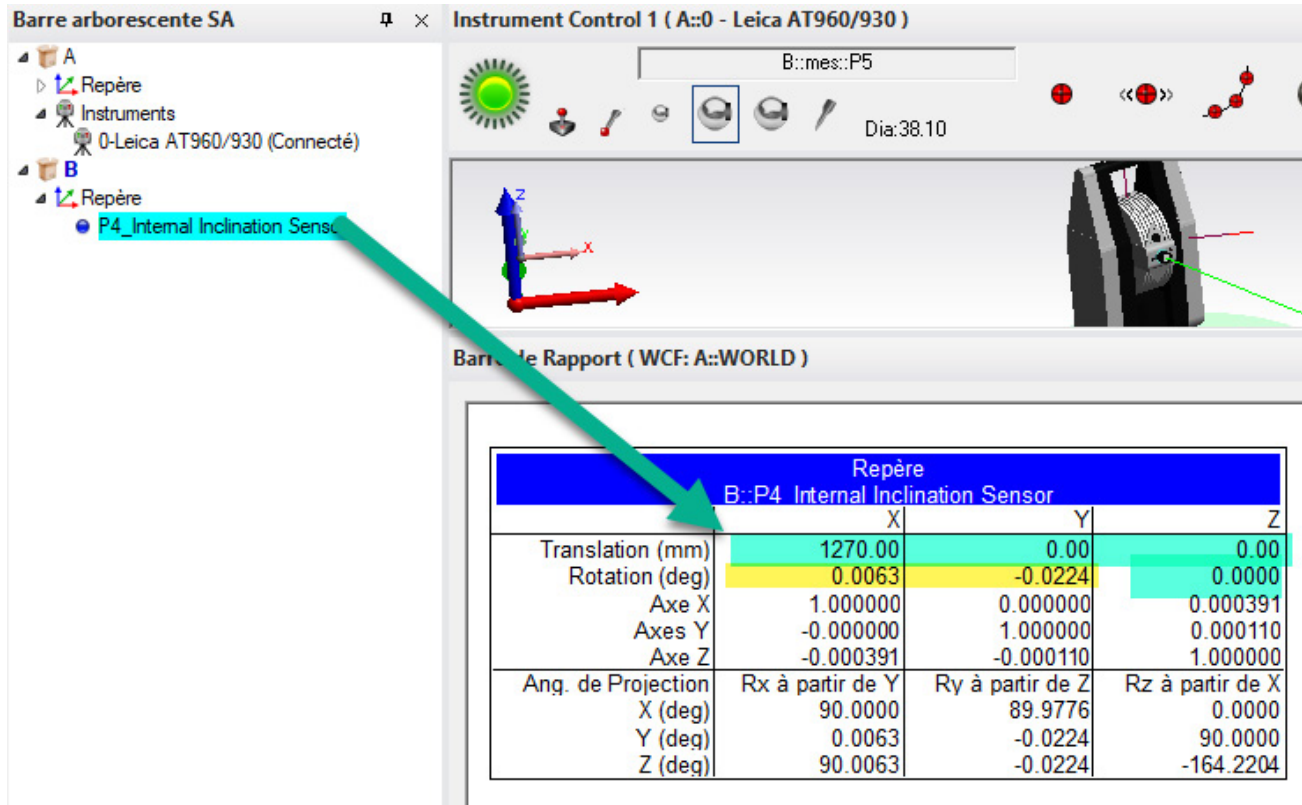


Figure 11-54. To check if your instrument is still level, select the level measurement frame from the tree and check the Rx and Ry rotation values.

However, if your instrument has an active compensator or if you have performed a level compensation the base frame of the instrument can be used as a gravity reference.

Level Compensation

Level *compensation* is different from level *measurement*. Level compensation changes the base reference for points recorded from an instrument. Much like an ADM compensation or axis compensation, this operation changes the base reference from which measurements are recorded.

Operations such as Leica's *Orient to Gravity (OTG)* or API's *Virtual Level* operations perform level *compensations*.

There is an important distinction between level compensation and other instrument compensations however, in that a level compensation is only valid for a single instrument plant. Any time an instrument is moved that level compensation is no longer valid.

Its important to note - Comparing measurements before and after a level compensation would be similar to comparing different instruments to each other. This means two things:

1. A level compensation should be performed after each instrument station move - prior to any point measurements.

2. If measurements have already been taken at an instrument plant, a *jump instrument* operation must be performed prior to performing a level compensation and taking additional measurements.

Notes on Leica’s OTG

Leica’s AT9x0, AT600 and AT500 trackers offer the ability to perform an OTG operation. This is performed within tracker pilot not SA. If you open tracker pilot there are two settings of note:

- **Inclination Monitor.** Which allows you to monitor level and works much like a drift check
- **Inclination to Gravity.** This is where an Orient To Gravity option is offered.

To perform an Orient to Gravity (OTG) operation do the following:

1. Select the **Inclination to Gravity** option from the Settings.
2. Ensure the bubble level is within range and then perform a level Measure (Figure 11-55).

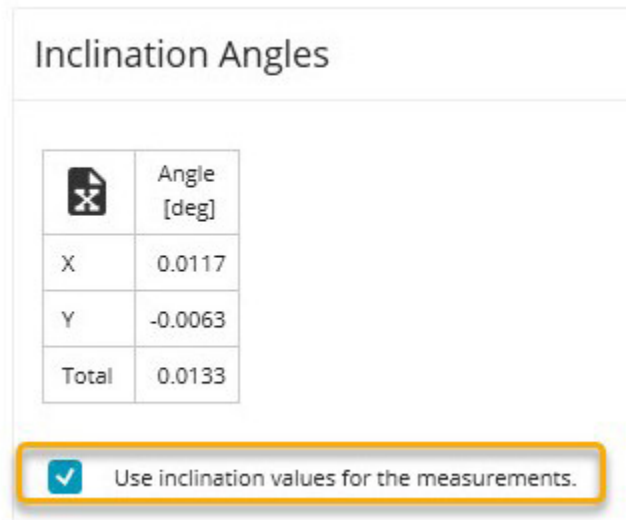


Figure 11-55. Successful level measurement and OTG enabled.

3. Enable the option to **Use Inclination values for the measurements.** option transforms the measurements sent to SA based on the level measurement taken in tracker pilot. This means OTG is enabled. When unchecked its not. So this icon is the key (Figure 11-56):



Figure 11-56. OTG Enabled Icon shown on the front of the tracker controller.

Active Compensation

Some instruments, including most total stations and the Leica AT40x

trackers, have *active* compensators. These devices are actively adjusting the measurements using the gravity sensor which ensures that the measurements are *always* gravity compensated as long as the compensator is enabled.

Orient to gravity operations are not necessary for these instruments because compensation is part of the measurement process.

However, if the compensator is turned off for any reason the same rules apply and a jump instrument and re-alignment operation should be performed if any prior measurements were taken.

When to Perform a Level Compensation

Level *compensation* is not necessary for most applications. Even when you intend to hold level with respect to a local gravity plane, all that is really needed is a gravity measurement and to straighten the instrument model with respect to the measurement. This takes a record of the current gravity orientation at that location and, by straightening the station, aligns the instrument to the gravity plane. A network alignment can then be performed while holding level.

The one time that gravity *compensation* is necessary within SA is when a large area network is needed and you wish to hold level relative to the curvature of the earth. Depending on the scale, the direction of the gravitational pull can be quite complex (Figure 11-57). For more information see https://en.wikipedia.org/wiki/Gravity_of_Earth.

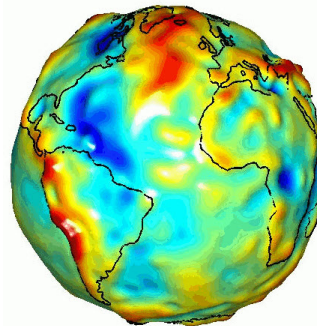


Figure 11-57. Magnetic pull of gravity shown as a geo map

SA's large network alignment solutions such as USMN do not offer an option to hold a non-planar reference during the network solution. For this to happen we would need to both keep track of the instruments to use and the corresponding gravity measurements to using constraining the solution. That's a lot keep track of. In that scenario a level compensation such as OTG or VL is necessary. It aligns the instruments base frame with the gravity measurement. Once a level compensation has been made USMN does offer the ability to perform an alignment while holding the instruments base frames level. This sequence then allows you to hold a local level for each instrument during the alignment process.

Therefore the steps to perform a USMN with a large non-planar network would be as follows:

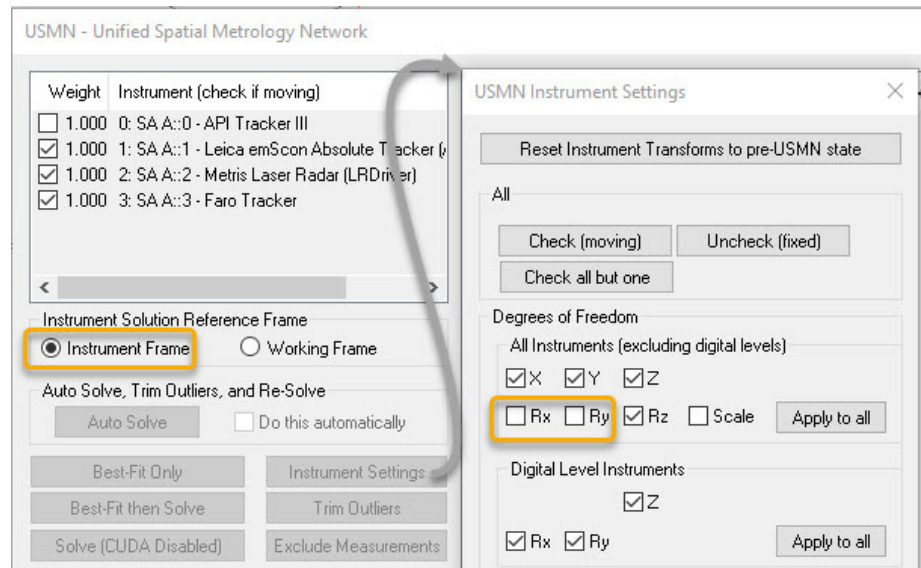
For each instrument station:

1. Perform a level compensation (OTG or VL). This must be performed first in order to align gravity with the instruments base frame.
2. Measure all reference points and features of interest from this station.

Once this is complete you can align the network using USMN and these settings:

1. Perform an initial network alignment to get the instruments correctly located with respect to each other in the job file.
2. Straighten the instrument with respect to a common working frame. This ensure that the instrument base frame is aligned with the working frame (**Instrument > Instrument Station Controls>Straighten> Align Instrument Z Axis with Working Frame**).
3. Select **Instrument Frame** for the Instrument Solution Reference
4. Open the Instrument Settings, disable **Rx** and **Ry** in the Degrees of Freedom section and press **Apply to all**.

Figure 11-58. Select Working Frame as the solution reference and lock the RxRy degrees of freedom for all instruments to hold them level during the solution. NOTE... This only works if you have straightened the instruments relative to each other first.



5. Solve the network and consider any outliers as normal.

Alignment & Transformations

CHAPTER

12

One of the key areas of portable metrology is alignment: transforming an instrument or set of points into the proper position and orientation such that it is aligned to the desired coordinate system. This chapter explores methods for transforming into a desired coordinate system.

Typically, metrology data is of limited use until it has been aligned to some set of reference datums, nominal data, or other measurements. Aligning to a coordinate system is sometimes referred to as *bucking in*, *tying in*, *locating*, or *aligning*. Some of the reasons this is done include the following:

- Governing dimensions and measurements (acceptance criteria) are typically expressed on drawings with respect to a specific datum or frame of reference. Often, evaluating features with respect to other datums is not permitted when evaluating design conformance.
- When comparing as-built measurements to a nominal design, the data must be aligned to the nominal design—otherwise, the comparison cannot be performed.
- Multi-station measurements (measurements taken from more than one position in space) must be “tied” together so that the data is continuous.
- Digital alignment and assembly (virtual fit-up) requires that measured components be “virtually” assembled using some sort of alignment in order to check the assembly for fit and function.

In some situations, this involves creating a coordinate frame from the measured points. In other (less common) situations, it involves transforming CAD data to the measurements. But most often, it involves aligning measurements to a coordinate system of interest. Since measurements are rigidly fixed to the instrument that measured them, this typically involves transforming an instrument in space until its

measurements end up in the desired position.

In this chapter, we will explore the tools available for performing all different types of alignment.

Review of Available Alignment Options

Alignments should be made on an instrument basis whenever possible. Because instruments, their measurements and associated objects move as a unit this is the most effective method.

An instrument can be located several ways. The basic instrument location methods can be found under **Instrument>Locate** or right-click an instrument and select **Locate**. Advanced methods are discussed in later sections and links are provided below.

Measured Point to Reference Alignments

Best-Fit

Locating an instrument via a Best-Fit is one of the most common location methods. It is used heavily during instrument location to nominal control points, instrument relocation and relocation due to drift. The user specifies the instrument to locate, the nominal point group and measured point group. The below fit dialog will appear allowing a view of the best fit. This Best-Fit Transformation dialog is covered in the Best Fit Points to Points section. For more details see “Point to Point Best-Fit” on page 351.

Analysis Best Fit Points to Points

Analysis>Best Fit Transformation>Points to Points is very similar to a basic best fit, the main difference being that the user has the ability to move multiple instruments and additional objects with the computed transform (see “Point to Point Best-Fit” on page 351)

Measure Nominal Points

Measure Nominal Points is a great way to locate an instrument quickly. It functions like a guided best fit. Simply select the nominal points and the instrument to locate.

The **Locate** dialog will listing the points to measure and the user has the option for manual or automatic measurement. Once three points are measured the instrument is transformed via a Best-Fit. With the instrument roughly located the user can easily drive or point to each target. As measurements are recorded the Best-Fit is updated. For more details see “Locate By Measuring Nominal Points” on page 360.

Feature and Reference Point Alignments

Nominal Geometry Relationships

Many geometry relationships are point reducible such as the center of a circle or a sphere. This option allows you to choose a selection of geometry relationships and best fit their center points (see “[Nominal Geometry Relationships](#)” on page 365).

Reference Point System (RPS) Alignment

Added and popularized in the automotive industry this alignment option provides a means to use a set of point reducible features as well as surface points to achieve an Alignment. It provides individual point control over weighting and component influences (see “[Reference Point System \(RPS\) Alignment](#)” on page 366).

Surface Alignments

Quick Align to CAD

A quick and easy method to align to a CAD model. Simply click the model where the points will roughly be measured. A minimum of six points are needed to align to a CAD model. Red dots are used to guide the user where to measure. During the selection of the surface points the current view is saved. As measurements progress the views update (see “[Quick Align to CAD](#)” on page 369).

N-Point Full Fit

Much like a relationship optimization this function finds the optimal transformation that minimizes the errors between the measured points and the CAD surface. This method contains many options for tolerance fits and other advanced fitting methods (see “[N-Point Full Fit](#)” on page 347).

Point Cloud/Mesh to Reference Alignments

A selection of Cloud to Cloud, Cloud to Mesh, Cloud to CAD, Mesh to Mesh and Mesh to CAD alignment options are available. These alignments are discussed in detail in the chapter on point clouds (see “[Cloud Based Alignment/Registration Tools](#)” on page 582).

Frame and Feature Based Alignments

Frame to Frame Transformation

Transform an instrument from a one frame to another. User will be prompted to select a source frame, a destination frame and an instrument to move.

Plane, Line, Point (3-2-1)

A 3-2-1 fit consist of a locator point for the origin, line/vector for one axis and plane for the second axis. The Plane, Line, Point Transformation dialog allows control over the axis for the plane and line/vector. The user may also specify an alternate coordinate for the locator point. Once the transformation is accepted the instrument will be transformed with respect to the active frame.

3 Planes

The 3 Plane instrument locate uses three planes to define a coordinate system that is then transformed to the active frame. The user has control over the axis fit priority and position.

Frame Wizard

Frame Wizard provides a flexible way to select points or objects to define the origin and axis directions of a new frame. Once that frame is built and you accept it, the transformation from the current frame location to the active frame is performed.

Advanced Optimizations

Locating By Minimizing Relationships

Instruments can be located by Minimizing Relationships. This is an advanced option to help equalize any misalignment by using multiple relationships at once all with custom weighting and constraint options. Its only fully available in SA Ultimate and more details can found in the Relationships Chapter (see "[Minimizing Relationships](#)" on page 444).

Sequential Alignment

A Sequential Alignment allows you to align with different relationships in a predefined sequence, applying specific degrees of freedom to each relationship and then running the alignment process much like a script (see "[Sequential Alignment](#)" on page 453).

Network Alignments

Bundle

Designed with theodolites in mind. This network alignment performs a weighted bundle based upon common measurements targets and the resulting position (see "[Bundle Adjust](#)" on page 375).

Unified Spatial Metrology Network (USMN)

USMN is a strong method for locating multiple instruments. It is usually used in a case where three or more instruments are present. This is available in SA Ultimate and can found in the USMN Chapter (see

“What is USMN” on page 785).

Saved Alignments

A selection of alignments provide a means to be saved and recalled at any point during the analysis of a job file.

Repeat Last Alignment

The following alignments can be recalled using the **Repeat Last** button in the Inspection Controls.

- **Measure Nominal Points.**
- **Quick Align to CAD.**
- **Frame Wizard.**
- **Nominal Geometry Relationships.**

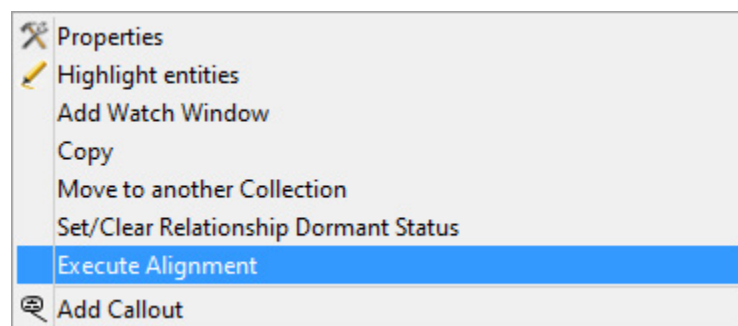
Relationship Construction

Several alignments offer automatic relationship construction. This provides a means to link the measurements to the nominal features such that an alignment can be recalled. These include:

- **Best Fit.** Builds a Group to Nominal Group relationship.
- **Quick Align to CAD.** Builds a Points to CAD Faces relationship.
- **Frame Wizard.** Build a Frame Wizard relationship.
- **Cloud to CAD.** Builds a cloud to CAD relationship.

In order to apply an alignment saved with a relationship, simply right-click on that relationship and select *Execute Alignment*.

Figure 12-1. Right-click option which allows a user assert a saved alignment.



If a relationship is not automatically built as part of the alignment double check that the option to do so is still enabled. Navigate to the Automation Tab of the User Options to find this setting “*Enable Relationship Construction on Alignment*”.

Alignments Saved Separately

More complicated alignments that reference a set of relationships are saved separately in the tree. These include:

- **RPS.** Reference Point System (RPS) Alignment.
- **Sequential Alignment.** Used for sequential relationship fits.

GD&T Datum Alignments

Each GD&T Datum Reference Frame also defines an alignment that can be used directly for alignment. For more information on datum alignments refer to “[GD&T Alignment](#)” on page 634.

Measured Points to Reference Alignments

Best-Fit Transformations

Best-fit transformations are widely used in coordinate metrology to consistently and confidently align measurements into a coordinate frame that is meaningful to a user. Measurements made with an instrument should be presented to users in the part coordinate frame—that is, aligned to the part. The best-fit transformation routine in SpatialAnalyzer is one way to do that.

A best-fit transformation solves for the difference between a group of points and the expected (or reference) coordinates for those points. The difference between the current position and orientation of the points and the ending position/orientation is referred to as the *transformation*. After the solution is reached, SpatialAnalyzer shows you the difference and allows you to apply the transformation to your measurements of the points and to other objects as well. Once applied, SpatialAnalyzer is able to report those measured points and any new measurements in a part coordinate frame that is meaningful to you.

The best-fit alignment technique has many applications; for example, to combine sets of coordinate data or locate an instrument in the part coordinate system using surface features imported from a CAD system. Simply stated, this alignment technique helps you determine the transformation that when applied to one set of coordinates causes the match-up between the two sets to be ideal.

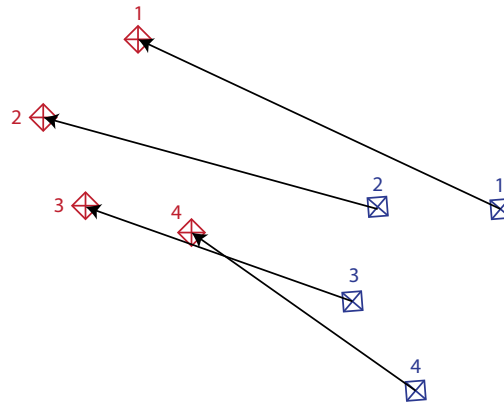
The term “best fit” is the colloquial term given to a mathematical procedure known as a *Linear Least Squares* fit. The transformation solved for and applied is determined using least-squares optimization methods. A least squares optimization seeks to minimize the square of the distance between corresponding points, or between points and cor-

responding surfaces.

Point to Point Best-Fit

A common need for performing a point to point best-fit is when trying to locate to a known coordinate system in which discrete points already exist. For instance, you have a network of control points (fiducials) from a previous station measurement and you'd like to relocate the instrument and remeasure these points in order to align back into the coordinate system. When performing this type of alignment, the group of points being aligned *to* are referred to as the *reference group*, and the points being moved are referred to as the *corresponding group*. They may also be referred to as the *nominal group* and *measured group*, respectively, although if the term *nominal group* is being used, it does not necessarily mean that the points are true nominal design points (Figure 12-2).

Figure 12-2. A group of four points (blue), when best-fit to a set of reference points (red), will move as illustrated. The four blue points move together as a rigid body to try to align with the reference points. In this case, a rotation is involved.



In order to perform a point-based best-fit transformation, the points involved must meet two prerequisites:

- There must be at least 3 points in both the reference and corresponding groups that have corresponding names.
- The reference and corresponding points must be in separate point groups.

Note that the two groups need not have exactly the same number of points. Any points that don't have corresponding pairs are simply ignored in the command, although if they are in the moving point group they will be transformed along with the rest of the points in the group. There are three ways to perform a point-to-point best fit transformation:

- **Instrument-based best-fit.** In this method, you pick an instrument to align, and then select the reference and measured group. The transformation will be applied to the instrument, bringing all of the measured points along with it.
- **General best-fit.** This is nearly identical to the first method,

however it's not instrument-centric, and provides a little more flexibility when needed. You can choose to transform anything, including instruments, point groups, and other objects. There are also a few additional options (discussed below).

- **Locate an instrument by measuring nominal points.** This allows you to measure a set of nominal points, and let SA handle the point naming and alignment for you.

These three methods are described below, but to use them (well, at least two of them) we first need to understand the *Best-Fit Transformation* dialog.

The Best-Fit Transformation Dialog

The *Best-Fit Transformation* dialog (see Figure 12-3) is a simple interface allowing a high degree of control over the fit. Here you can review the fit, apply tolerances, restrict degrees of freedom, edit point constraints, and more. Any time one or more changes is made, the current fit is no longer valid (indicated by a dark blue background on the fit button and the fit results) and needs to be re-solved. The dialog is divided into three main parts: the controls (top left corner), the fit results (top right corner), and the point list (bottom).

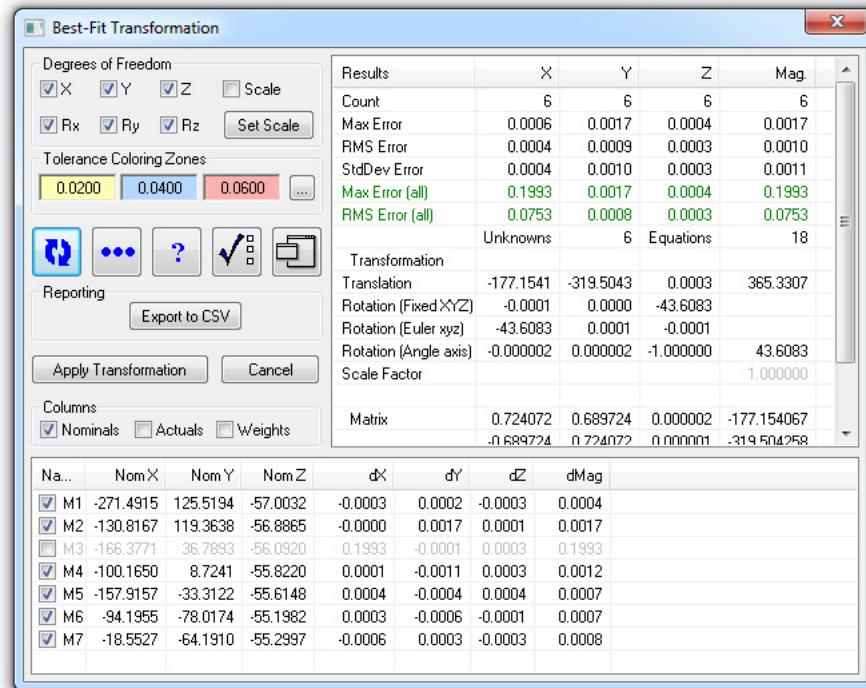


Figure 12-3. The *Best Fit Transformation* dialog.

Fit Results

The fit results are listed in the table in the top right corner of the dialog. These results will be invalid (and therefore will have a dark blue background) whenever the best-fit solution has either not been solved or a change has been made since it was last solved. The fit re-

sults table contains, on a component-by-component basis, a number of statistical values related to the fit that help you determine how well the fit performs. These values include:

- **Count.** The number of points included in the fit for the specified component.
- **Max Error.** The maximum error between a pair of corresponding points. This is an indicator of the “worst fit” for any pair of corresponding points.
- **RMS Error.** The root-mean-square error of the fit as a whole. This is a good indicator of the overall error of the fit. This number is always less than or equal to the max error (although usually it’s noticeably smaller than the max error).
- **StdDev Error.** The standard deviation of the errors. A small value indicates that each pair of points had similar errors. A large value indicates that one or more error values differed significantly from the group as a whole.
- **Max Error (all).** This is the same as **Max Error**, except all points are included (even those which are deselected from the fit).
- **RMS Error (all).** This is the same as **RMS Error**, except all points are included (even those which are deselected from the fit).
- **Translation.** The x, y, and z instrument translation needed for the solution.
- **Rotation (Fixed YZ).** The instrument rotations required for the solution, in Fixed XYZ form.
- **Rotation (Euler XYZ).** The instrument rotations required for the solution, in Euler XYZ form.
- **Rotation (Angle axis).** The instrument rotation required for the solution, in Angle axis form (first three values describe the rotation axis, and the last value describes the rotation angle).
- **Scale Factor.** The scale factor applied for the fit. This will be 1.0 unless the **Scale** degree of freedom is checked (allowed to float).
- **Matrix.** This is the 4x4 transformation matrix that will be applied to the instrument.

Point List

The point list is a table that contains all points that were found to have matching pairs between the reference and corresponding groups. Immediately above this list, in the *Columns* section, are three checkboxes: **Nominals**, **Actuals**, and **Weights**. Checking any of these boxes will display the corresponding values in the table. Any column can be sorted in ascending or descending order by clicking the column header. The

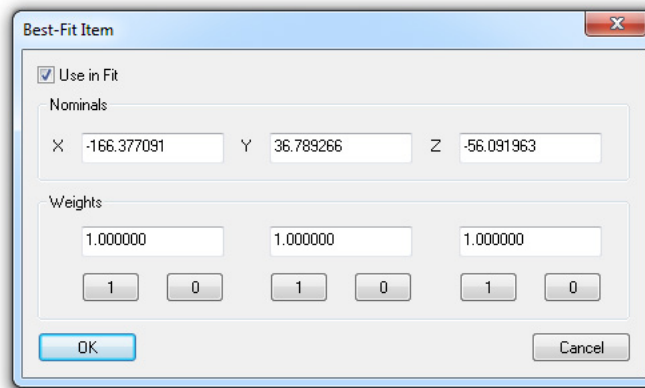
columns in the point list are described here:

- **Name.** The name of the point.
- **Nom X.** The nominal x-coordinate of the point. This is the x-coordinate of the reference (non-moving) point.
- **Nom Y.** The nominal y-coordinate of the point. This is the y-coordinate of the reference (non-moving) point.
- **Nom Z.** The nominal z-coordinate of the point. This is the z-coordinate of the reference (non-moving) point.
- **Act X.** The actual x-coordinate of the point. This is the x-coordinate of the measured (moving) point.
- **Act Y.** The actual y-coordinate of the point. This is the y-coordinate of the measured (moving) point.
- **Act Z.** The actual z-coordinate of the point. This is the z-coordinate of the measured (moving) point.
- **Wt X.** The fit weight for the x-coordinate of the point.
- **Wt Y.** The fit weight for the y-coordinate of the point.
- **Wt Z.** The fit weight for the z-coordinate of the point.
- **dX.** The current x-deviation between the reference and corresponding point.
- **dY.** The current y-deviation between the reference and corresponding point.
- **dZ.** The current z-deviation between the reference and corresponding point.
- **dMag.** The magnitude of the deviation between the reference and corresponding point.

Each point in the fit has a checkbox next to its name in the point list. Deselecting this checkbox will remove the point from the fit, which means that it will have no influence in the final solution. It is standard procedure to remove erroneous points from the fit so as not to corrupt the overall solution. If a point is removed from the fit, it will be grayed out in the point list.

Best-Fit Item Dialog

Figure 12-4. The *Best-Fit Item* dialog.



Double-clicking a point in the list will bring up the *Best-Fit Item* dialog (Figure 12-4). This dialog provides another way to remove the point from the fit (**Use in Fit**), and conveniently allows you to change the reference point's coordinates, as well as the per-component fit weight for the point. The **1** and **0** buttons are simply shortcuts for setting the corresponding fit weight value.

Fit Weights

Fit weights give you control over defining whether a certain component of a point has any influence in the fit process. A value of zero indicates that the component has no influence in the fit, whereas a value of 1 indicates the component has full influence. Any intermediate value may be chosen. If a component is weighted to zero, it is displayed in gray in the point list. If an intermediate value is chosen, it is displayed in blue, and if the weight is set to 1, it is displayed in black.

One reason for reducing the weight of a specific component of a point is to reduce directional errors. Suppose you have a measured point that was performed on a floor that was flexing in the z-direction during measurement. It is reasonable to assume that the measured point has significant error in the z direction when compared to the x or y direction, therefore reducing the weight of the z component may be a wise choice. Another reason is when using discrete points to control specific degrees of freedom, an advanced technique that, while doable using best-fit, is much easier to perform using relationship fits.

Controls

You'll notice that you have a significant amount of control in the fit. Access to changing the fit parameters is provided via the controls in the upper-left hand corner of the dialog.

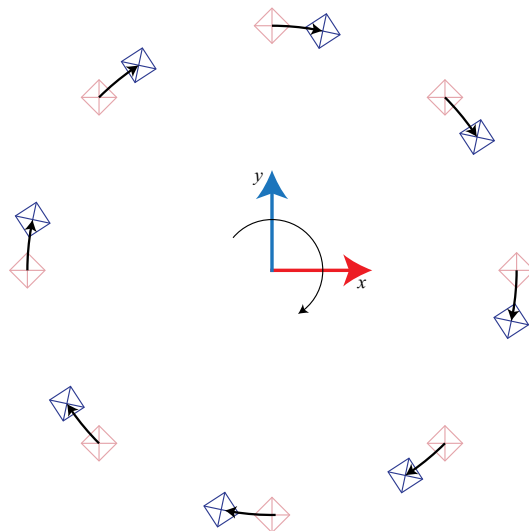
Degrees of Freedom

When solving a best-fit transformation, you can specify that the transformation may only be permitted in certain unlocked degrees of freedom, relative to the working coordinate frame. You can lock any com-

combination of X/Y/Z translation, X/Y/Z rotation, and scale. Any degree of freedom that is checked is unlocked—unchecking a degree of freedom locks it. Keep in mind that these degree of freedom restrictions are applied with respect to the working coordinate frame, therefore rotations are applied about the working frame origin.

As an example, suppose you have a bolt pattern measurement on a flange and you'd like to fit this measured pattern to a mating, nominal pattern. In this case, you would likely align the face of the measured points to the nominal flange face using perhaps a relationship fit, then you might wish to use the nominal bolt pattern to spin the measured flange about its axis until the bolt holes are aligned (Figure 12-5).

Figure 12-5. Performing a best-fit transformation allowing only rotation about the Z axis.



By locking all degrees of freedom except for rotation about the Z axis—and by placing the working frame at the center of the flange—the fit would simply clock the measured pattern to the nominals.

You might also restrict degrees of freedom if you have a leveled instrument (its axis is parallel to the Z axis) and you do not want it to rotate about the X or Y axes.

Notice that you can also set the instrument to a specified scale value for the fit, which allows you to apply temperature compensation to the measurements in situ.

Tolerance Coloring Zones





You can apply tolerance coloring to the deviations in the point list by setting the tolerances in the [Tolerance Coloring Zones](#) section.

Fit Controls

A series of buttons controlling the fit is displayed below the [Tolerance Coloring Zones](#) section:

-  **Re-fit button.** Recalculates the best-fit transformation

based on the current settings. A blue background on this button indicates that the current fit is not valid based on the current settings and a re-fit is needed.

-  **Edit Selected Items.** Brings up the *Best-Fit Item* dialog (page 355) for any selected items. More than one point can be selected at one time in the point list. This button allows you to change all of their properties at once.
-  **Quick Tips.** Displays a series of tips related to the dialog.
-  **Advanced Settings.** Displays the *Advanced Best-Fit Options* dialog, discussed below.
-  **Advanced View.** Toggles to a simplified version of the interface, hiding less frequently-used parameters.

Advanced Settings

The *Advanced Best-Fit Options* dialog (Figure 12-6) provides some advanced, less commonly-used settings for best fit transformations.

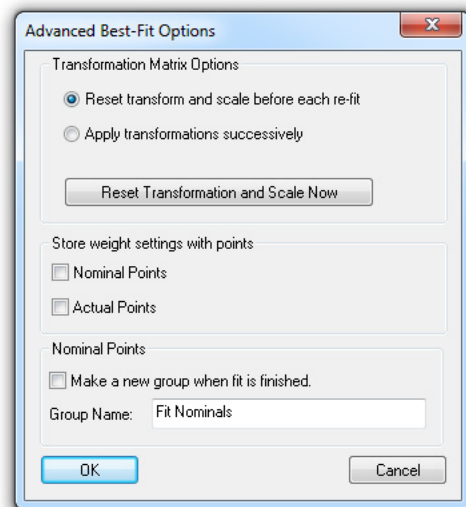


Figure 12-6. The *Advanced Best-Fit Options* dialog.

Descriptions of these parameters follow:

- **Reset transform and scale before each re-fit.** The transformation is solved from the same initial starting position and orientation each time the fit is performed. This is the default and is applicable to most fitting operations.
- **Apply transformations successively.** Each fit operation solves using the previous calculated transformation as a starting point. This allows you to apply iterative best-fit operations with different degree of freedom settings for each. For instance, you might wish to first solve the best-fit with some degrees of freedom locked, then solve again from that position with a different set of locked degrees of freedom.

- **Reset Transformation and Scale Now.** Resets the current transformation back to the “starting point”.
- **Store weight settings with points.** Fit weights can be stored with nominal or actual points such that if the points are used in another best-fit operation, the weights are recalled.
- **Make a new group when fit is finished.** Makes a copy of the reference (non-moving) points and puts them into a separate group with the specified name at the completion of the operation. If nominal coordinates were changed in the *Best Fit Item* dialog, this group will use those updated coordinates. This is not a common need.


Export to CSV

The **Export to CSV** button will export the fit results to a comma-separated .CSV file and import it into Excel (if installed).

Instrument-Based Best-Fit

To reiterate, an instrument-based best-fit transforms just an instrument and (of course) its measurements. Only one instrument can be transformed. As such, it is most appropriate when locating a single instrument to a set of reference points, or when relocating an instrument to a new station.

1. Measure a set of points into a point group. These will be the “measured” points.
2. Ensure that the measured point names match the corresponding point names in the reference group.
3. From the menu, select **Instrument>Locate (Transform to Part)>Best-Fit**.
4. If there is more than one instrument in the job file, select the instrument to locate (as prompted).
5. When prompted for the “nominals” group, select the reference group. This is the group containing the points that will not move during the transformation.
6. When asked for the “measured” group, select the group containing the measured points. These points should be measured by the instrument selected in step 4.
7. Choose the desired options from the *Best-Fit Transformation* dialog and perform the fit as described, then click **Apply Transformation**.

 **Note:** If your point names don't correspond, use **Edit>Rename>Points by Auto-Corresponding 2 Groups** to automatically rename them.

General Best-Fit

A general best-fit can be performed in which the calculated transformation is not simply applied to a single instrument that measured the

corresponding points. This is useful for applying point to point best-fit transformations when instruments are not involved, or when you'd like to move several instruments together as part of the fit. In the general best-fit, you can apply transformations to instruments and other objects, in addition to a few more options.

To Perform a General Best-Fit Operation

1. Select **Analysis>Best-Fit Transformation>Points to Points**, or press **Ctrl + B**.
2. Select the reference (non-moving) group.
3. Select the corresponding (moving) group.
4. In the resulting **Best-Fit Transformation** dialog, select the desired options for the fit and solve it (see “The Best-Fit Transformation Dialog” on page 352).
5. In the **Best-Fit Options** dialog (Figure 12-7), select the desired options (described below), then click **OK**.

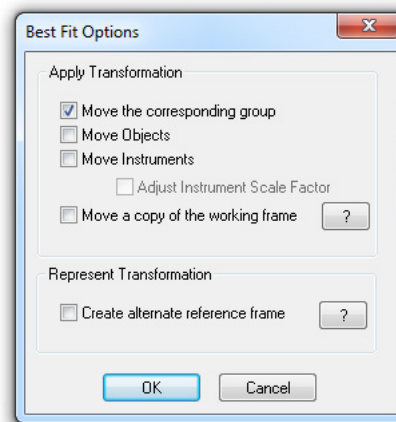


Figure 12-7. The options for a general best-fit.

Best-Fit Options

Toward the completion of a general best-fit operation, you are provided with a series of options that are not displayed in the single instrument version of the best-fit command. These options are described below.

- **Move the corresponding group.** Determines whether the transform is applied to the corresponding (moving) group. If unchecked, the corresponding group is not moved.
- **Move Objects.** If you'd like to transform other objects, select this option. After clicking **OK**, you'll be prompted to pick the additional objects to transform.
- **Move Instruments.** If you'd like to transform instruments, select this option. After clicking **OK**, you'll be provided the opportunity to select instruments to transform.

- **Adjust Instrument Scale Factor.** This option is only available when the scale degree of freedom is unlocked during the fit. When checked, the scale factor calculated during the fit operation is applied to the moving instruments.
- **Move a copy of the working frame.** When selected, the working frame will be copied, and the transform will be applied to the copy. This creates a frame that can be used to report the coordinates of the points as they were prior to the transformation.
- **Create alternate reference frame.** When selected, a frame is created that represents the inverse of the applied transformation. This allows you to transform the moving objects back to their position prior to the best-fit operation through the use of a frame to frame transformation.

Locate By Measuring Nominal Points

While locating an instrument by fitting to a set of nominal points (see “Instrument-Based Best-Fit” on page 358) is not a difficult task, you may find that using **Instrument>Locate (Transform to Part)>Measure Nominal Points** is preferable for typical in-process alignment in which more control is not needed.

Locating by measuring nominal points provides a few benefits over the instrument-based best-fit method:

- Point naming is handled automatically, eliminating the potential for typographical errors in measured point names.
- If the instrument is roughly aligned to the points already (for instance, it is being re-fit into the reference system after being bumped), it can easily be automatically pointed at the targets (if the instrument supports pointing). Single points or the entire set of reference points can be automatically measured with a single button click.
- The quality of the fit is displayed at the time of measurement—therefore any blunders in measurement (resulting in errors in the fit) can be discovered much sooner than if the fit were being performed later in the process.

There are a few things to keep in mind when using this command:

- It must be used at the time of measurement. If the measured points already exist, you must use one of the other two best-fit commands.
- You have much less control over the fit. You cannot control degrees of freedom (with the exception of scale), weighting, or advanced fit operations—and you cannot remove points from

the fit unless you delete them.

- You're presented with much less detailed information about the fit results.

For the first two measurements, insufficient measurements exist to perform the fit, so the delta deviations are not displayed. However, as each successive measurement is obtained, the fit is recalculated and the deviation values are updated. The *Locate Instrument by Measuring Nominals* dialog is presented in [Figure 12-8](#).

Figure 12-8. The *Locate Instrument by Measuring Nominals* dialog is useful when performing a best-fit points to points transformation during the measurement process.

Locate Instrument by Measuring Nominals

Instrument: Faro Tracker

Reference established by group: Station1

Group To Contain Measured Points: LocateInstMeas1

Scale

Vary Scale Initial = 1.000000

Level

Hold Level (Rx Ry off)

Closest Point

Use Closest Point (after initial locate)

Acceptance Threshold (0 = None) 0.0

Point	dX	dY	dZ	dMag
P1				
P2				
P3				
P4				
P5				
P6				
P7				

Measure Manually

Point At

Delete

Automatic Measurement

Single Point

Multiple Points

Insufficient data.
Measure more points.

Tolerance: 0.0 Apply

Finished -- Locate Instrument Cancel

A description of the options follows:

- **Group to Contain Measured Points.** This is the name of the group in which you'd like to measure the corresponding points. It is best to type this in before measuring any of the points, because if you change it after measuring any points, the already-measured points will not move into the new group. The **Apply** button will update the table based on the entered group, and the ellipsis button allows you to select an existing group to use.
- **Vary Scale.** When checked, scale is allowed to float during the fit. The initial, delta, and final scale values are listed next to this checkbox.
- **Hold Level (Rx Ry off).** This option can be if you do not want the locate operation to tilt the instrument. It restricts the degrees of freedom at are used in the alignment and was added

for use with gravity compensated instruments in mind.

- **Closest Point.** With this option enabled you still have to measure the first 3 points in order, but after that, you can measure the point in any order and SA will detect which point you measured automatically.
- **Measure Manually.** When clicked, the selected target in the list is measured. If the target already exists, an additional observation is added.
- **Point At.** Points the instrument at the selected target and attempts to lock on. (Applies only if the instrument supports pointing/target acquisition).
- **Delete.** Deletes the selected target.
- **Automatic Measurement - Single Point.** Attempts to point at, acquire, and measure the selected target.
- **Automatic Measurement - Multiple Points.** Attempts to consecutively point at, acquire, and measure all points in the list.
- **Tolerance.** You can specify a tolerance in this field. Any deviation magnitudes that exceed the tolerance will be highlighted red. Click the **Apply** button to apply the specified tolerance to the table.

To Locate an Instrument by Measuring Nominals

1. Ensure the instrument interface is running.
2. From the menu, select **Instrument>Locate (Transform to Part)>Measure Nominal Points**.
3. If more than one instrument exists in the file, select the instrument to transform, as prompted.
4. Select the nominal (reference) points.
5. The **Locate Instrument by Measuring Nominals** dialog is displayed.
6. In the *Group to Contain Measured Points* section, type in a name for the point group in which to place the measured points.
7. Measure points manually or automatically using the dialog as described above. Once three or more points have been measured, click the **Finished -- Locate Instrument** button at any time to accept the fit and complete the command.

Drift Check

Drift Check is an option easily accessed by directly right-clicking on the instrument in the graphics you want to check. It can also be found

it under the menu command **Instrument>Drift Check**.

The Drift Check dialog may appear familiar. The group name control and measurement controls are the same as those used in the **Measure Nominal Points** alignment option, and in many ways it's the same operation. It allows you to select a set of points and measure them in a fast and efficient manner.

However, the advantage of Drift Check is that it provides both a report of the current deviations and a preview of the RMS and Max deviation you would see if you were to re-fit, aligning using a Best-Fit to the nominal points. This is quite helpful because the deviations can be compared, and an informed decision can be made as far as how to proceed (Figure 12-9).

Figure 12-9. The Drift Check dialog which provides a streamlined means of re-measuring a set of points and either jumping station or recording the event.

Drift Check

Instrument: Faro Tracker
Static

Group To Contain Measured Points
DriftCheck1 Apply

Reference established by group: Station1

Point	dX	dY	dZ	dMag
✓ M1	-0.0001	0.0001	0.0000	0.0002
✓ M2	-0.0001	0.0000	0.0000	0.0001
✓ M3	0.0001	-0.0000	-0.0000	0.0001
M4				
M5				
M6				
M7				

Measure Manually
Point At
Delete

Automatic Measurement
Single Point
Multiple Points

Tolerance: 0.003 Apply

Drift Results RMS 0.0001, MAX 0.0002 In Tolerance

If you Relocate the instrument:

Best-fit RMS 0.0001, MAX 0.0002 Add New Instrument: Transform
Best-fit Scaled RMS 0.0001, MAX 0.0001 Add new Instrument: Transform and Scale

Finished -- Drift Acceptable Cancel

Drift Check results can be applied in either of 3 ways:

1. **Best-Fit (Add New Instrument: Transform).** Pressing this button will automatically add a new instrument model to the job file, disconnect the interface from the original model, reconnect it to the new model and perform a best-fit using the points measured in the drift check to align the new instrument station.
2. **Best-Fit Scaled (Add New Instrument: Transform and Scale).** Pressing this button will also automatically add a new instrument model to the job file, disconnect the interface from the original model, reconnect it to the new model and perform a best-fit using the points measured in the drift check to align the new instrument station. However, it will allow scale to float as part

of the best-fit and the new station will be scaled accordingly.

3. **Finish -- Drift Acceptable.** Pressing this button will record an event in the tree as a record but will not have any effect on the existing instrument station.

How to Perform a Drift Check Step by Step

1. Right-click on your live instrument model in the job file and select Drift Check.
2. Select a Point Group to measure as a reference. This should either be an imported reference group or set of prior measured reference points. ***See Key Points.
3. Measure each point in the list by selecting and manually measuring it (you can use Point At to find which point is which). If you have a set of targets already in place you can also use the Automatic Measurement option to measure all the points automatically.
4. Compare the drift results to the expected results if you were to relocate. Choose to Add a New Instrument and Transform if this difference is significantly greater than you would expect with your instrument under your conditions. ***See Key Points

Key Points: On How and When to Transform

- Drift Checks should be performed frequently, and on regular intervals. The frequency will need to increase the tighter the tolerances. Minor shifts in the instrument placement can add up and drift checks are the best way to make sure either: that things are still in a good condition, or to identify that you're instrument needs to be re-aligned. The sooner you realize an instrument was bumped the sooner this can be fixed.
- Get to know your instrument and your working environment to establish an expected threshold for a drift check. Drift checks should be quick and painless but in real life there will always be error. If you have an idea of how accurate your instrument is under your expected working conditions, it can become second nature to accept and continue or add a new instrument and re-align.
- You may not want to align to your reference group. It is very important to not create an error stackup situation. If you align one station to the next in sequence, errors can accumulate. You always want to align to your original nominal point set, but when these are imported from a CAD model the errors may be pretty big to start.

To prevent error stack up with imported nominal points , do the follow:

1. When you start the job you can measure the CAD points and align, noting the significant errors in alignment.
2. When you perform drift checks you will want to select the first measured point group so that the errors displayed in the Drift Check are 100% linked to the station placement.
3. Then when the instrument needs to be re-aligned you will want to again align to the original CAD points to prevent any error stackup. The Add New Instrument and Align option within the Drift Check dialog should only be used when the original measured group is the defining point group for the nominals within the job file.

Feature and Reference Point Alignments

Nominal Geometry Relationships

The function **Analysis>Best Fit Transformation>Nominal Geometry Relationships**, also available through the instruments right-click menu provides an easy way to locate using geometry relationships that are point reducible using a best fit. Point reducible relationships include:

- Point to Point
- Average Point
- Circles
- Slots
- Ellipses
- Spheres

This alignment tool provides an option to choose a selection of point reducible geometry relationships and perform a best fit using the centers of those features. The only requirements are that you have at least 3 point reducible relationships with nominals defined. Each feature will be listed as an entry in the Best Fit dialog and all the same controls are available (see “[The Best-Fit Transformation Dialog](#)” on page 352).

SpatialAnalyzer: How to Align an Instrument Using a Nominal Geometry Relationship - <https://www.youtube.com/watch?v=Wz0HcplvN10>

Reference Point System (RPS) Alignment

Much like a Best Fit, Reference Point System (RPS) alignments require a nominal and measured set of points for alignment. RPS is unique in that it provides the means to use a set of point reducible features (such as circle centers) as well as surface points to achieve an alignment. It then expects the input points to be used to constrain a single axis or set of axes for the alignment. This alignment can be found under **Analysis>Best Fit Transformation>RPS Alignment**.

Features can be added to the alignment list in either of two ways: they can be built directly through this dialog which will prompt for CAD selection or they can be selected from a list of existing compare to nominal geometry relationships or point to point relationships and added to the alignment list.

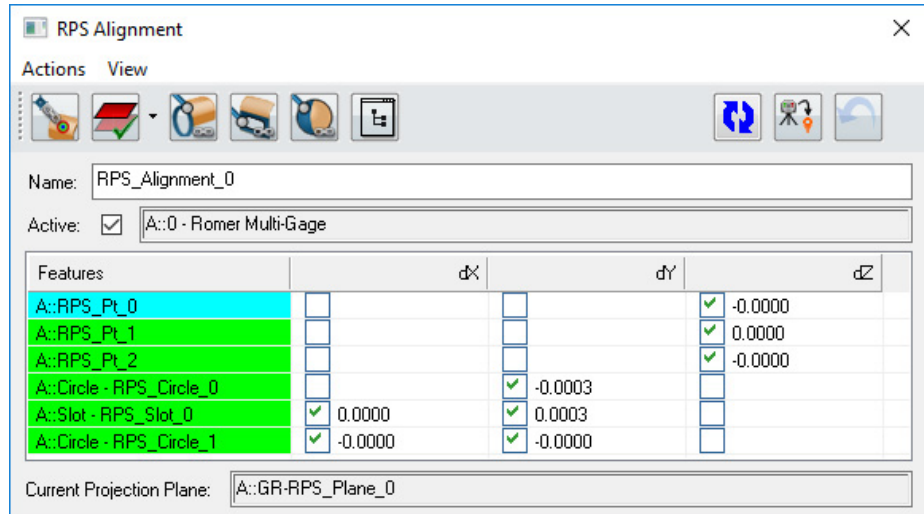


Figure 12-10. Basic RPS Alignment Dialog

Actions

A full list of available functions for adding and removing features from the RPS alignment is listed under the **Actions** menu (Figure 12-10). The primary functions are also available through the icon buttons, and include the following:

- **Add Feature.** This command initiates the addition of a feature to the inspection list. By selecting Point, Projection Plane, Circle, Slot, or Sphere, a built from CAD prompt will be initiated. It also provides access to selecting from Existing Features.
- **Delete Feature.** Removes the selected feature from the RPS list. This does not delete the feature from the tree.
- **Clear All Features.** Removes all features from the RPS list of features.
- **Run Fit.** Performs a fit computation, populating the summary table.

- **Apply Alignment.** Transforms the Active instrument. If no instrument is marked as Active it will prompt for objects and then instruments to move. This then provides the opportunity to align multiple instruments at once.
- **Undo Alignment.** Returns the instrument and objects moved to their starting location.

Features List

Each point reducible feature included in the RPS alignment is listed in the Features List as well as check marks for the dX, dY, and dZ components to include in the computation. What makes RPS unique is that it provides the ability to specify individually if a feature will contribute to the alignment in individual XY or Z components of the working frame. So if you specify a surface point and only use the Z axis then the measured point will influence the alignment only along that Z axis and will not influence the alignment along either X or Y. An example alignment then could include a set of 3 surface points to define a Z axis (or a top planar direction) and a set of two holes each constraining X and Y to lock the position of a part in space.

Additional information about each feature can be found by hovering over it, including projection plane assignment and additional tolerance information.

Projection Planes

Projection planes are not required but are recommended for features where the depth within a hole feature is not contained. A circle measured free-hand within the hole will have some degree of misorientation which will influence the fit and the resulting nominal comparison. However, projection planes are not part of RPS alignment. Surface points are expected to constrain a planar surface along an axis. When a projection plane is defined all features defined afterward that can take a projection plane as an input will include it.

View Menu

The display of the Summary and Advanced sections of the dialog can be controlled through the view menu. Each or both can be hidden if desired to save graphic space.

Summary Section

Figure 12-11. RPS Summary Section

Summary			
Max:	<input type="text" value="0.0003"/>	Status:	<input type="text" value="RPS fit over constrained"/>
RMS:	<input type="text" value="0.0002"/>	Tolerance:	<input type="text" value="0.0010"/>
Average:	<input type="text" value="0.0001"/>	Count:	<input type="text" value="6"/>

Features that are out of tolerance will be identified in the dialog and marked in red in response to the numerically value entered in the tolerance field in the summary section (Figure 12-11). Relationships will

additional tolerances set on them directly and who's additional tolerances are out of specification will be marked in Blue. When the RPS alignment computation is performed the Max, RMS, and Average will be displayed and reflect the individual component values.

The alignment Status , Tolerance and feature count are included in the second column of the Summary section. The Tolerance field controls the colorization of the relationships within the feature list.

Advanced Section

Figure 12-12. RPS Advanced Section

The screenshot shows a dialog box titled "Advanced". It contains the following controls:

- Scale Factor:** A text input field containing "1.0000".
- Scale:** An unchecked checkbox.
- Frame of Reference:** A text input field containing "A::WORLD".
- Show:** A section with three unchecked checkboxes:
 - Magnitude
 - Overall Weights
 - Component Weights

The Advanced section (Figure 12-12) provides a set of more advanced controls not necessary for most fits. These include the ability to add scale to the fit which in turn will scale the moving instrument to produce the best possible solution. It also provides a set of weight controls which will expose additional fields to adjust weights within the feature list.

Surface Normal Considerations

Building a surface point through the RPS dialog will result in a point to point relationship being built in the tree. This relationship will include the nominal point, the measured point, and the surface normal from the selected surface. Its important to realize, however, that the alignment is still a point to point alignment. If surface points are built from curved surfaces it is very important to measure the actual point very closet to the design location or some cosine error can result due to the change in curvature of the surface.

Performing a Basic RPS Alignment

Much like a best fit operation an RPS alignment is intended to provide a means to align point to their nominal locations. The only difference is that these points are the center locations of features or surface points and they are only used to adjust the fit along a single axis or set of axes. A basic fit to the NRK Demo part may go like this:

1. Add an RPS alignment using **Analysis>Best Fit Transformation>RPS Alignment** and import the CAD model into your job.
2. Build a set of surface points on the top of the CAD model to define the Z axis (these should be built from planar surfaces normal to the Z axis of the working frame but do not need to be co-planar).
3. Define a projection plane using the projection plane icon
4. Add 2 circles by first selecting the circle button and then clicking on the inner cylindrical surface of the CAD model to de-

fine nominal geometry relationships

5. Double check that the surface point feature only have the dZ component checked and the circles have both the dX and dY components check but not the dZ component.
6. Add an instrument, connect and measure these features.
7. Press the Run Alignment Icon and check that everything is within the specified tolerance (set within the Summary Section).
8. Press the Apply Alignment Button to locate the instrument.

Surface Alignments

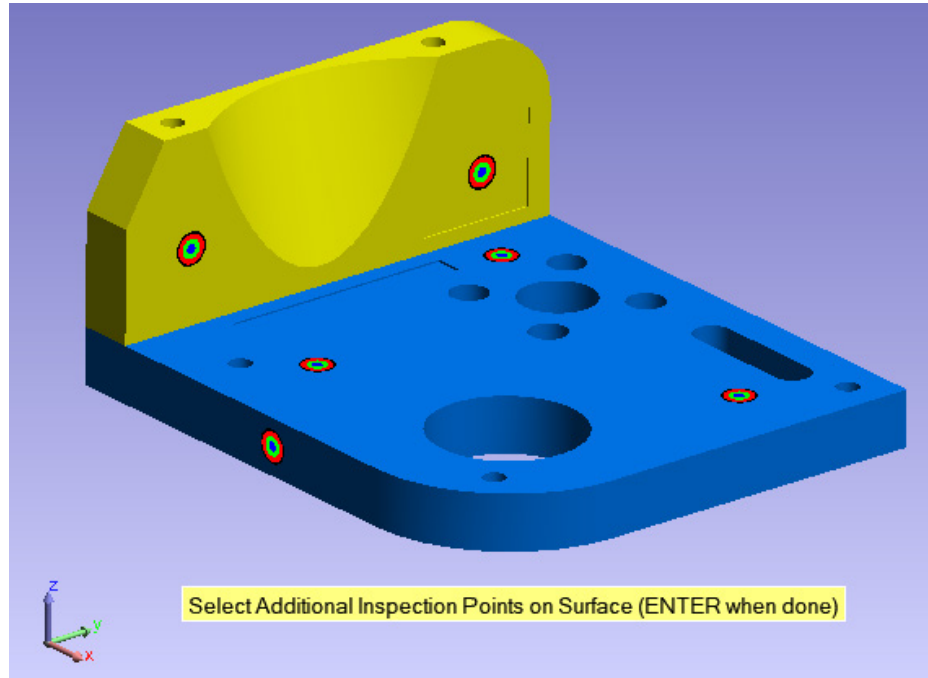
Quick Align to CAD

Quick Align to CAD is intended to be a quick and easy method to align to a CAD model and can be initiated either through **Analysis>Best Fit Transformation> Quick Align to CAD** or through the instruments right-click menu. The quick align dialog provides two functions: alignment designation and measurement guidance.

When the quick align dialog opens, it first expects you to specify points on the CAD model where you wish to measure and use for the fit. A minimum of six points are needed to align to a CAD model and these should be defined such as to fully constrain it in space. Starting with a 3-2-1 type laying is usually most effective.

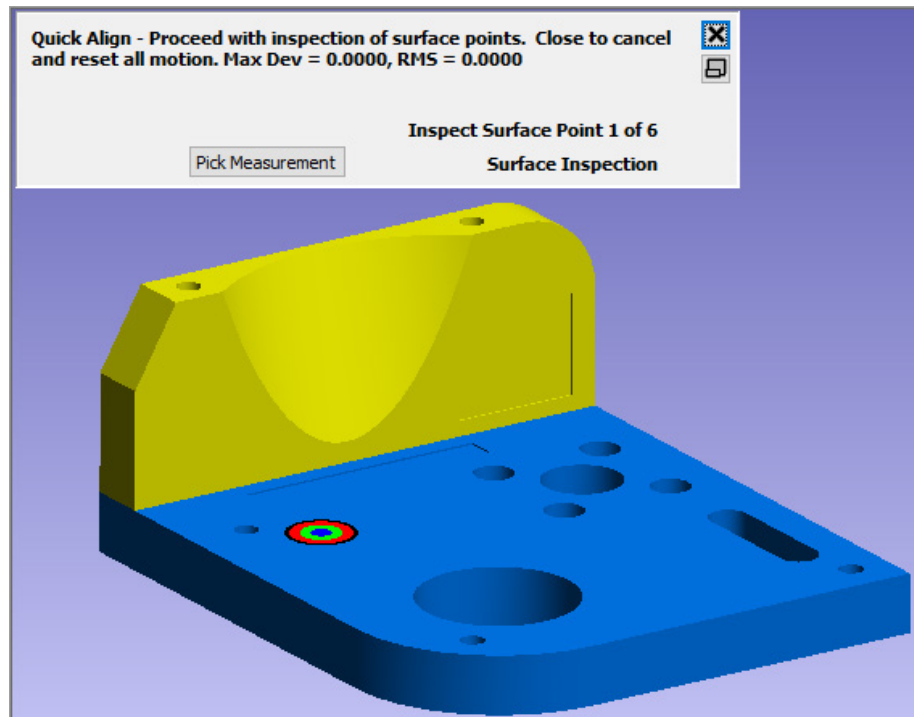
Point selection can be performed simply by clicking on the CAD model where you wish the measurements to be taken. These points should be laid out in the order, as you wish to measure them, and the view should be oriented such that it is clear where the points should be taken (both the view orientation and zoom depth is saved with each point). Red dots (bull's-eyes) are used to guide the user where to measure (Figure 12-13). The Backspace key can be used to remove unwanted points if you select an incorrect location, and the Enter key will end the selection process.

Figure 12-13. Point Placement for a 3-2-1 type Quick Align to CAD



As soon as Enter is pressed the inspection process begins. The user will be guided point by point through the list of inspection points. Measurements can be added in either of two ways. Any measurements taken by an instrument while the quick align process is active will be used for the alignment. Alternatively, the **Pick Measurement** button can be used to select pre-existing measurements either from the graphics or from within the tree to be used for the alignment.

Figure 12-14. Quick Align Measurement Guidance



Alignment computation options... [Analysis>Best Fit Transformation>](#)

- **Quick Align to CAD.** The quick align process performs a series of alignment operations. When the first point is measured the instrument is translated such that these points align. With the second measurement the instrument is also rotated to fit. With 3 or more measurements the instrument's points and nominal points are best fit together to show a reasonable graphical representation of the alignment. Once complete a final optimization process is performed where the measured points are then aligned directly to the CAD model looking to minimize the RMS error between the measured points and the surfaces.
- **Quick Align (Faces Only).** This alternative menu item is very much the same as a standard Quick Align to CAD, the only difference is that it is more restrictive. Where a Quick Align to CAD performs a final points to surface alignment, this option forces point deviations to fit to the specific surface faces used for the nominal designation. With small surface faces or larger errors in the measurements this option provides the means to ensure that the points are pulling the alignment toward the correct surface face even if the optimal RMS is reduced.

Best-Fit Points to Surfaces

Performing a linear least-squares fit between a set of points and one or more surfaces is a common task. Often, you want to know the deviation of an as-manufactured part or assembly from its perfect design condition, and it is common to have the nominal design represented in a CAD package as a set of surfaces. In cases where specific discrete reference datums for aligning measurements to the nominal design are not specified (or not desired), best-fit points to surfaces is one method for aligning your measurements to the reference system.

Best-Fit Points to Surfaces uses optimization methods to find the transformation that, when applied to the instrument (and therefore its points), minimizes the RMS error between the measurements and the surfaces, taking into account your specified constraints. Typically, the goal is to drive all the points (compensated for probe offset) to be exactly on the surface. SpatialAnalyzer extends that capability by allowing the operator to specify high and low tolerances for points and assign weights to those constraints.

Initial Condition

Like several other optimization methods, fitting points to surfaces

Basic Operation

For a standard best-fit, just leave the options (Figure 12-15) set to the defaults and then the points will be brought to the surface using a best-fit taking into account the target offsets of the points. Make sure all the surfaces you are using have their normals pointing in the right direction.

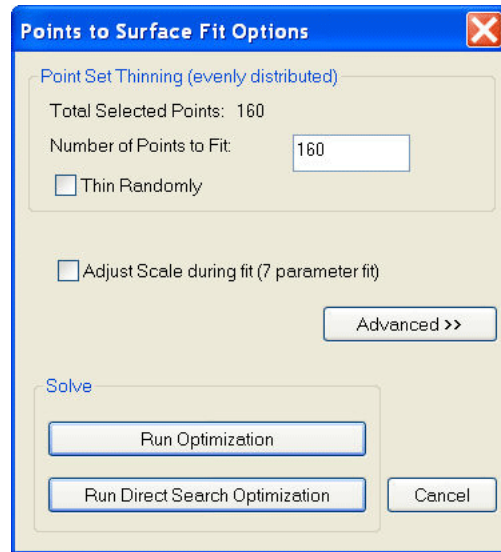


Figure 12-15. The Points to Surface Fit options.

Point Set Thinning

This option allows you to easily thin selected point sets for purposes of fitting to a surface. Suppose you have a scan with 10,000 points, but don't want to wait a long time for the best-fit. You can specify that you only want to fit 400 of the points and it will evenly sample 400 points from the selected point set.

Run Optimization

This is the default mode of operation. The optimization method performs a version of the steepest-descent algorithm on the system. This means it assembles a matrix that is 6 (unknowns) x N (number of points), inverts it and determines move directions. Though quick for reasonable data sets, it can take a long time when using a large set of points depending on the amount of RAM on the computer. The stopping tolerance is the amount that the objective function (RMS error) must repeat by in order to indicate a minima has been reached.

Run Direct Search Optimization

This optimization provides a powerful method for searching the solution space. It is also required when using the toleranced best-fitting options (in the advanced area) since they introduce non-linearities into the solution space. This method basically steps in a direction, evaluates the fit, then steps in another direction, etc. There is a lot of advanced logic built into the search, but at the core, it is a searching method. The advantage is that it does not have to assemble a large

partials matrix and invert it. This means that it can effectively deal with huge point sets. The disadvantage is that since it is trying to exhaustively search, it can sometimes take a long time to complete. At any time during the fit, however, you can cancel the operation and use the current location. To use this mode, you set step sizes. They are set as a ratio of the position and angular perturbation values in the SA units options page. The step size starts at the starting value then decays when it detects no improvement in any direction. Once it decays to the Ending step size, a minima is reached.

Recommendations

We have several recommendations for setting these options based on past experiences. As with many things, you may need to experiment to determine the best set of options for a particular combination of points and surface geometry.

The most important recommendation is to start with a good initial condition. The easiest way to do this is to use Quick Align to CAD.

- **Reasonable point sets without toleranced fitting.** Use only the conventional solution. It is not necessary to waste the time on the exhaustive search.
- **Huge point sets with a reasonable starting guess.** Use only the exhaustive search with appropriate start and stop perturbations.
- **Reasonable point sets with tolerances or unusual geometry.** Use both fit options.
- **Reasonable point sets, no tolerances, but very high accuracy required.** Use both fit options.

Advanced Options

By pressing the **Advanced** button, you can open the advanced side of the dialog (Figure 12-16).

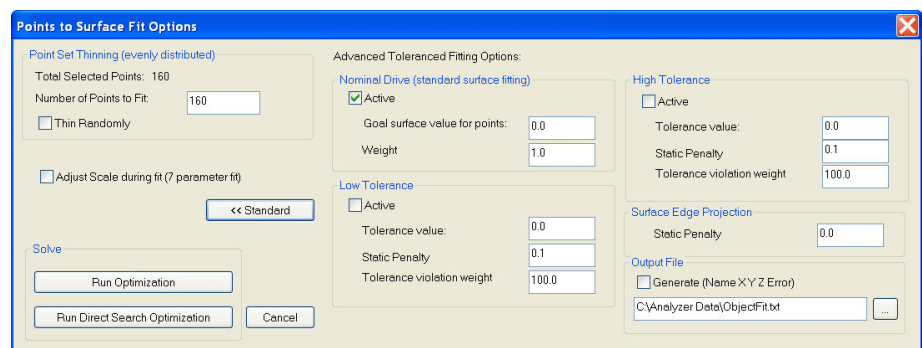


Figure 12-16. Advanced Best-Fit options.

These allow you to modify the constraints of the fit to achieve more complex fits. This may be used to avoid part to part interference in build operations, or other processes where the interaction between

parts is of interest.

Suppose we want to perform a standard best fit to the CAD surface, but we want to make sure none of the points are more than 0.030 below the surface. The parameters could be set as follows:

- Nominal Drive. Use default settings.
- Low Tolerance 0.030, and default weight values.

The result is that the optimization will drive the points to the nominal CAD surface but also exert extra pressure on points that fall below 0.030 on the negative side of the surface. Because the weight for the tolerance is fairly high, it will push the set of points so that the offending low tolerance points lie within the -0.030 band.

Nominal Drive (Standard Surface Fitting)

To perform a standard surface fit, use this option by checking **Active**. Set the goal value to 0.0 if you wish to drive the points to the surface, or you may use another value to use the offset surface. You do not need to set the weight unless you are using other tolerances in the fit.

Low Tolerance

These four variables control the low tolerance value. The tolerance value does not have to be on the negative side of the surface. You could, for example, have a low tolerance of 0.25" and a high tolerance of 1". The low tolerance is simply the lowest tolerance value.

- **Tolerance Active.** This flag allows you to easily turn on and off the low tolerance. Turning the tolerance off is equivalent to settings its weight to 0.0, except that it skips the computations necessary to determine the toleranced objective function value.
- **Tolerance Value.** This value is the tolerance for the point to surface fit of this FitPoint. If 0.0, then the point will be driven to the surface (after compensating for target thickness)
- **Static Penalty.** This value is added to the objective function in addition to the weighted tolerance violation. Think of this as a static offset applied any time a point moves beyond the tolerance. Initially, leave this value set to 0.0 and only use it when needed. When you use it, set its value low so it does not unnecessarily increase the nonlinearity of the system. Using large static offset penalty values will have the effect of stopping the optimization prematurely.
- **Tolerance Violation Weight.** This value is multiplied by the amount that the point violates the low tolerance. If the tolerance is -0.030 and the points projected distance is -0.040, the violation is 0.010 multiplied by the Weight.

High Tolerance

The parameters are identical to the Low tolerance except that they apply to the high tolerance.

Edge Projection Penalty

This value is a fixed penalty added to a point's objective function contribution if the point projects to the edge of a surface.

Network Alignments

Bundle Adjust

A bundle adjustment may be used to refine measurements acquired using a spherical measurement system. These include theodolites, laser trackers, laser scanners, etc. Basically, a bundle adjustment will attempt to find the instrument transformations (positions and orientations) that yield the minimum combined pointing error.

Initial Guess

Since the bundle adjustment algorithms use the current locations of the instruments as a starting guess, it is best to “rough” position the instruments. To do this, either enter the instrument transformations manually, or use the **Drag Instruments** function from the Instruments menu. It is best to place the instruments near their final location, as this will speed the bundle solution. If using theodolites with level compensators, you will use the option to force instrument vertical during the bundle adjustment. If this is the case, the virtual instrument models should be perfectly parallel to each other before initiating the bundle adjustment.

Scale-Bar Database

For instruments that measure angles and distances, it is not necessary to use scale-bars in the bundle process. They are required for theodolite networks, however, because without them, the instruments will scale freely.

SpatialAnalyzer contains a scale-bar database. This is where you can enter the various scale-bars you have placed into the measurement area. Essentially what this database does is it associates target names with a scale-bar distance and uncertainty.

To modify the database, select the **Scale-Bars** tab from the User Options page. The next window will list all of the current scale-bars and their information.

To add a scale-bar, press the **Add** button and a new record will be generated. Complete the information required and press the **Apply** button.

You should see the information in the list window update.

Since an accurate bundle solution for theodolites requires at least one scale-bar, make sure this database is populated before attempting a bundle. In addition, make sure the targets for the end of the scale-bar have been shot and included in the list of targets to be included in the bundle solution.

Note, however, that you may populate the scale-bar database at any-time before attempting to bundle adjust. It is not necessary to have this database in place before taking measurements.

Select **Bundle Adjust** from the Instruments menu. Next, select the instruments you wish to bundle using the dialog. The instruments that you select are the instruments that will be moved in order to refine the network solution. If you have 4 instruments, you will only have to bundle 3 because the first instrument may remain fixed while the others are adjusted relative to it. You may, however, select all of the instruments. This will make the solution take longer, however, because the entire network will be allowed to float in space. Next, the **Bundle Adjustment** dialog will appear. This dialog will allow you to configure the settings for the bundle, run the optimization, and view the results.

Selecting Targets to Bundle

Once the Bundle Adjustment window appears, you may select the targets to be included in the bundle. SpatialAnalyzer will default to including ALL of the targets. You may reduce the number of targets for computational efficiency or target accuracy reasons.

Scale-Bar Weight Factors

In the Bundle Adjustment window, you will notice a section for Scale Bar Weight factors. This section allows you to control the manner in which scale-bars are incorporated into the final solution.

You may change the state of the check-box to determine whether scale-bars are weighted based on their uncertainty. If this box is checked, the relative uncertainties in the Scale-Bar database will be used to weight those scale-bars with a higher accuracy more than those scale-bars with a lower accuracy (or a higher uncertainty). Basically, this check-box controls how multiplier scale-bars are compared to each other during the bundle solution.

In addition, you may enter a numerical weight that is applied to ALL scale-bars during the solution. This value defaults to 1.0, but may be increased or decreased at the user's discretion. Note, however, that this value is independent of the state of the check-box. You may weight scale-bars based on uncertainty and/or apply an overall weight. This is because the overall weight is applied after the uncertainty weight is applied.

Viewing System Errors

The Bundle Adjustment window displays both the Original shot errors and the adjusted shot errors. This gives you a measure of how much the bundle adjustment has improved the overall system error.

A maximum, average, and RMS error is displayed. These values characterize the combined pointing errors for each target that is included in the Bundle Adjustment. Each target has a pointing error, given in degrees. This is the amount that the measurements miss the best-fit computed target. The bundle adjustment attempts to minimize the combined pointing error by simultaneously adjusting both the position and orientation of the instruments.

Therefore, if there are 30 points in the bundle, the maximum error is the largest pointing error for all of the points, the average is the average of that set, and the RMS is the root-mean-square of the set.

If you wish to see the errors itemized by target, click the **Show Details** button, and you will see a list of the targets and their errors. You may wish to select this option before beginning the bundle to inspect the list.

Beginning Computation

Click the **Begin Computation** button, and the Bundle Adjustment will begin. As progress is made, the graphical display will be updated, and all of the numerical values in the Bundle window will reflect the system improvement.

When the algorithm has found a minimum, it will stop and issue a message to the user. Look at the details of the bundle to inspect the individual target errors as well as the scale-bar errors. If these are suitable, click the **Accept Results** button in the Bundle window, and the changes will be made active. The system is now bundled!

If, however, the results are not satisfactory, and you wish to start over where you began initially, select **Cancel** from the Bundle window. This will reset the instruments to their initial locations and re-compute the target locations.

Local Minima

Whenever you attempt to minimize a set of complex equations, the issue of local minima arises. Because you start from an initial guess and seek to minimize the error until you can minimize no more, you do not know that you are at the lowest possible minima for the entire solution space.

For this reason, it is possible for the bundle adjustment algorithms to find a solution that, while it is a minimum, is not the minimum you were searching for. SpatialAnalyzer contains a full 3-D graphical model of the measurement environment to help decipher some of the lo-

cal minima. It is important, however, to start the instruments near to their optimal location (i.e. use a good guess) to minimize the chance of a local minima.

Weighting

There are several weighting factors described above. These control the importance of different parameters on the overall optimization process. There are several other weighting factors outside this dialog that control the process as well.

In the User Options dialog, under the Units tab, there is a button for Angle-Distance Weights. This button will allow you to control the relative weight of angular measurements compared to distance measurements. This will only affect the bundle if you are using instruments that can measure distance (laser trackers for example).

Unified Spatial Metrology Network(USMN)

Our most advanced network alignment has its own chapter (see "[Unified Spatial Metrology Network](#)" on page 785).

Job Pre-Planning and Instrument Placement

How to Pre-Plan instrument placements

Measurement projects often need to be performed within a tight timeline while working in a crowded environment. For those applications it can be very helpful to pre-plan both how many instrument plants or station moves are necessary and where to place the instruments in order to maximize the number of targets that can be shot from each location.

The first step is to set up a nominal reference file.

1. Set up a job file with the nominal points you need to measure
2. Import CAD defining the structures in the building that need to be considered
3. Add any additional obstacles by building planes or cylinder defining those obstacles and then converting them too into surfaces (surfaces are better than basic geometry because they are bounded and define a specific surface area).

Add Instrument Stations to the file.

Method #1.

For many years SA has offered a function **Construct>Points>Copy Groups Excluding Obscured Points** this function allows you to select a point group measured by an instrument and build a new group that is composed of the points that are visible from that station. To use this method, you would:

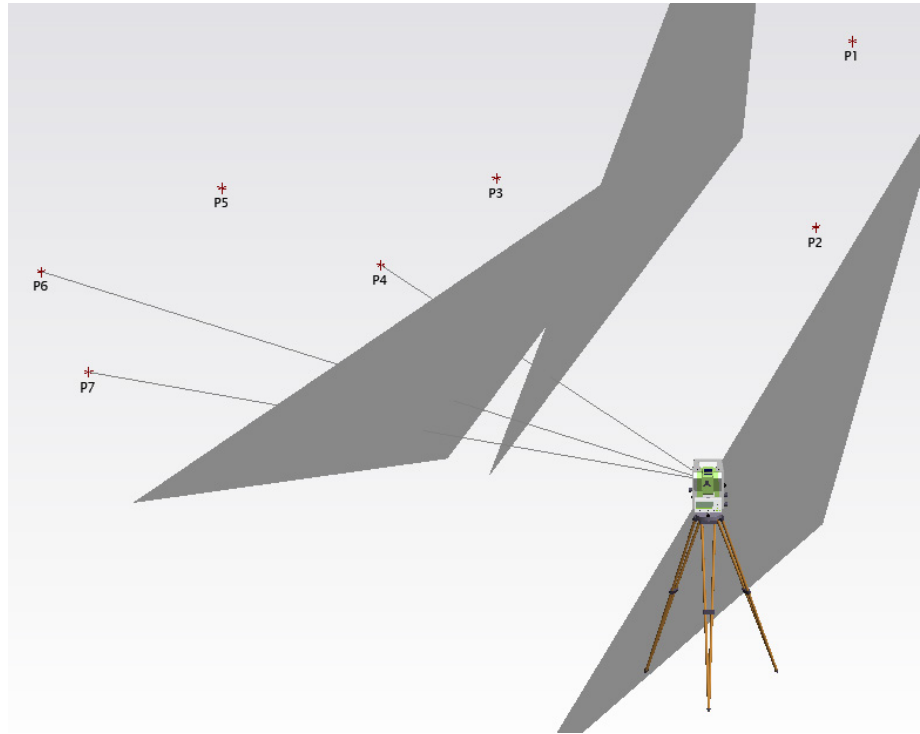
1. Add an instrument to the job and position it in a candidate position
2. Use the **Copy Groups Excluding Obscured Points** function to see if all the points are visible from that location.
3. Move the instrument if needed and repeat.

Method #2 (available in 209.11.21 or later)

There is now an option within the instruments Right-Click menu to **Show Obscured Shots**. This option simplifies the process quite a bit. Using this method, the workflow would be much simpler:

1. Add an instrument in the job and position it in a candidate position
2. R-Click on the instrument and enable **Show Obscured Shots**, selecting the groups to consider. Shot lines will be drawn graphically from the instrument to the hidden points showing where the interference is, making it easy to adjust accordingly
3. Use R-Click on the instrument and **Drag Instrument Graphically** to re-position the instrument in a more favorable location. The shot lines will update dynamically as you move the instrument making it easy to identify an optimal location.

Figure 12-17. Show Obscured Shots option in use displaying the 3 points not visible from the instrument.



4. To record the instrument position and available targets use **Construct Frame >On Instrument Base**, and use the **Copy Groups Excluding Obscured Points** command described above.


Watch Windows & Transform Tracking

CHAPTER

13

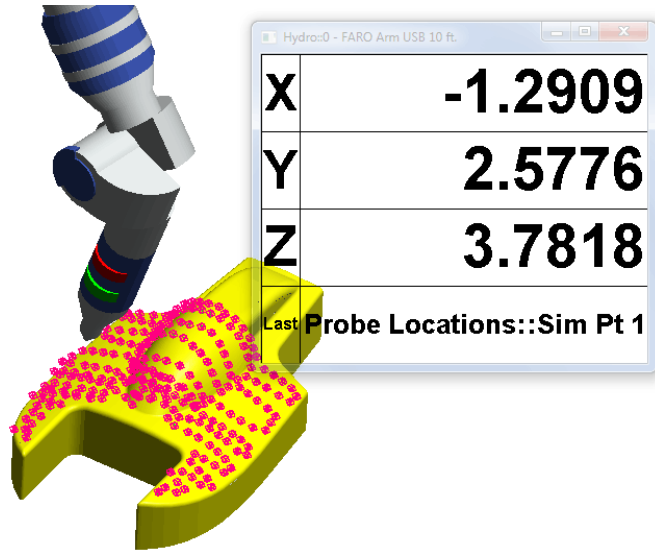
Viewing live measurement data is essential for tasks such as real-time assembly and tool-building. SA's Watch Windows provide a wide selection of digital readouts (DRO's) that can be configured to fit those needs.

SA's watch windows are live readouts of measurement data. You can view the coordinates of an instrument's measurement, watch the live delta transformation between two objects as they are moved, watch the deviation of a probe from a nominal surface, and more. This enables real-time assembly operations, in which you can watch a moving object's positional error in real-time and make corrections live. It also permits easy tool and fixture building, because you can watch a feature's position in real-time and check it as you make adjustments to the tool.

 **Tip:** If possible, you can use SA Remote (an iOS app for iPod/iPad/iPhone) to view watch window data remotely. See the chapter "Measurement" for more information.

A typical watch window is shown in [Figure 13-1](#). Depending on the type of watch window being viewed, you may be seeing raw coordinates (X/Y/Z values) or deviations (dX/dY/dZ values). Each value is displayed on a separate row in the window. The window may be resized or even maximized as necessary for easier viewing from a distance—the text will expand to fill the available window. You can open as many simultaneous watch windows as you'd like—they will all update automatically.

Figure 13-1. A typical watch window.



By default, watch windows (like almost everything in SA) show their data relative to the active coordinate frame. So an X value is the value in the working frame's X axis, and a dZ value is the delta along the working frame's Z axis. Settings for a given type of watch window are saved—meaning that if you, for instance, hide the X delta in a specific type of watch window, that value will still be hidden the next time you display that type of watch window.

How to Add a Watch Window

Watch windows are all about viewing the live readings of an instrument. So the first step is to connect your instrument and configure it to send updates to SA. Most instruments will always be sending position updates but some may need to be told to send updates.

With a live instrument you can add a watch window in either of 2 ways:

1. For more applications simply right-click on the instrument, point or object you wish to monitor and select **Add Watch Window** from the menu.
2. Watch windows are also available through the menus. Navigate to **View > Watch Window** and select the watch window type you would like to add.

Watch Window Types

There are a number of different types of watch windows that can be displayed in SA. They include:

- **Instrument's Point.** Shows the position of a live instrument's

- probe in real-time.
- **Point to Point.** Shows the deviation of the instrument's probe from a specific point.
 - **Point with View Zooming.** Compares the instrument's probe to a specific point, zooming in on the point as the deviation decreases.
 - **Closest Point.** Compares the instrument's probe to the closest of a set of points.
 - **Point to Objects.** Compares the instrument's compensated probe position to the closest of one or more objects.
 - **Three Point Frame.** A point measurement from each of three different instruments is used to define a coordinate frame, which is then viewed relative to the working frame.
 - **Two Point Axis Alignment.** Two measured points are used to calculate pitch and roll from a given coordinate frame.
 - **Theodolite Observer Point to Point.** Compares a point derived from two or more theodolite observations to any other point.
 - **Theodolite Observer's Point.** Displays a point derived from two or more theodolite observations.
 - **Theodolite Observer's Data.** Displays the data from the Theodolite Observer.
 - **Relationships.** Any relationship can have a watch window added in order to view that relationship's deviation.

Watch Instrument's Point

This watch window displays the coordinates of the *center* of the instrument's probe. It will update whenever updated data is received from the instrument.

To Display an Instrument's Point Watch Window:

- Right-click an instrument and select **Add Watch Window** from the context menu, or
- From the menu, select **View>Watch Window>Add Instrument's Point**.

Watch Point to Point

This watch window displays the deviation between the center of an instrument's probe and the selected point. Offsets are not used in this comparison.

To Watch Point to Point:

- From the menu, select **View>Watch Window>Add Point to>Point**, then select the point to compare, or
- Right-click a point and select **Add Watch Window** from the context menu.

Watch Point to Point With View Zooming

This watch window functions the same as a point to point window, but automatically zooms the graphical view and plays a sound whose pitch is proportional to the proximity of the target to the point. Offsets are not used in this comparison.

To Watch Point to Point with View Zooming:

3. From the menu, select **View>Watch Window>Add Point to>Point with View Zooming**.
4. Select the point to compare.

Watch Closest Point

A closest point watch window will automatically compare an instrument's probe position to the closest of a set of points. This set of points can be an entire group or a selection of points from a single group, or multiple groups. This watch window will automatically switch between points, displaying the name of the point to which it is comparing. Offsets are not used in point to point comparisons.

To Watch Closest Point:

1. From the menu, select **View>Watch Window>Add Closest Point**.
2. Select the point(s) to compare against.

To Watch a Point Group:

- Right-click the group and select **Add Watch Window** from the context menu.

Watch Point to Objects

A **Watch Point to Objects** watch window will display deviations between an instrument's probe position and either a single object or the closest of a set of objects. It can easily be added to any individual object by simply right-clicking on the object in the graphics and selecting **Add Watch Window**.

Like a query, a watch window will automatically account for probe offsets in its distance comparisons, and probe compensations are applied based upon the Projection Options set in the windows properties (see "[Watch Window Options](#)" on page 390).

Object Type Differences

Watch Objects - Lines:

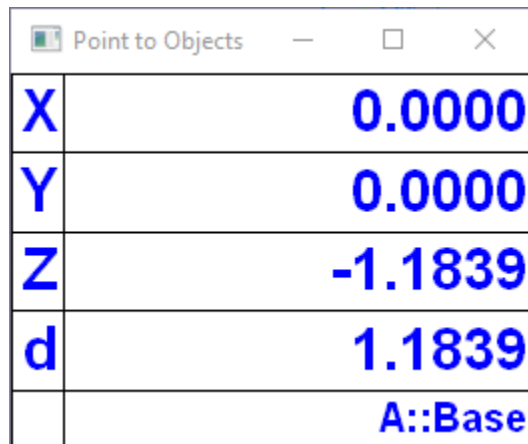
If the object you are comparing to is a line, SA will by default override target offsets to zero, since comparing the center of a probe to a line is far more common than comparing a compensated point to a line.

Watch Objects - Frames:

Adding a watch window to a Frame provides an added benefit if you have a 6D probe in that it will open a 6D watch window. This will report the relative distance and change in orientation of the instruments probe with respect to the selected frame.

Watching Multiple Objects At Once:

When multiple objects are selected, the closest object to the probe's current position is compared. As the probe moves, the object compared against may change, but the Collection and Object name of the object being compared to will be displayed in the Watch Window (Figure 13-2).



X	0.0000
Y	0.0000
Z	-1.1839
d	1.1839
	A::Base

Figure 13-2. Point to Objects Watch Window

To Watch Point to Objects:

1. From the menu, select **View>Watch Window>Add Point to>Objects**.
2. Select the objects to compare against.

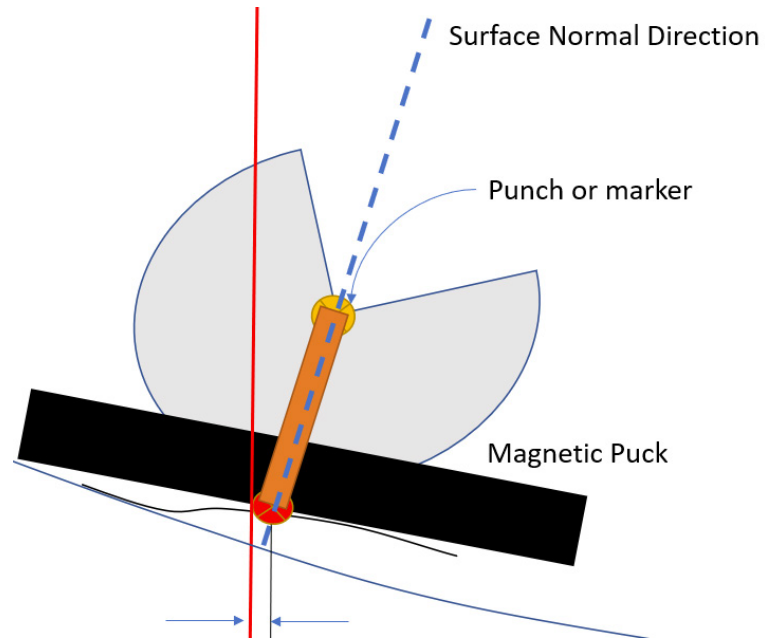
To Watch Point to a Specific Object:

- Right-click the object and select **Add Watch Window** from the context menu.

Watch Projected Point to Objects

This watch window is a variation on a watch point to objects in that it uses a second reference for offset compensation. The watch will project the current live point position to the reference object(s) by an amount corresponding to the live point offset value in the direction opposite the reference objects normal.

Figure 13-3. Example of using a projected point to object watch window to mark a point on a curved surface with respect to a nominal line or plane.



The data for this resulting point is then displayed relative to the measured object(s). This can aid greatly in situations where the curvature of the reference surface makes the center of the reflector a bad choice for comparison.

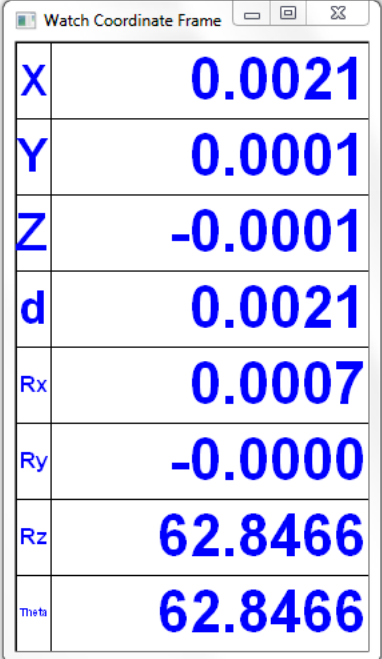
Watch Three Point Frame

Watching a three point frame is a fairly unique operation. It involves watching three 3-DOF devices simultaneously, wherein the device's points define a frame's origin, X axis, and XY plane. The watch window then shows the transform of this frame in the working frame.

For this command to be useful, the three instruments must be aligned to the same coordinate system—that is, they must be located to each other. Otherwise, the spatial positioning of the three points won't make sense.

If you have an object in which three points define a repeatable frame, you can use this command to easily determine the position and orientation of the object. If you then create and activate a corresponding coordinate frame representing a desired end state, this watch window represents the deviation needed to get to the active frame.

Figure 13-4. A three point frame watch window.



Parameter	Value
X	0.0021
Y	0.0001
Z	-0.0001
d	0.0021
Rx	0.0007
Ry	-0.0000
Rz	62.8466
Theta	62.8466

To Watch a Three Point Frame:

1. Ensure that the interfaces for three separate 3-DOF devices are running.
2. From the menu, select **View>Watch Window>Add Three Point Frame**.
3. Select the instrument whose point will define the origin of the frame.
4. Select the instrument whose point defines the X axis of the frame.
5. Select the instrument whose point lies in the XY plane.

Watch Two Point Axis Alignment

Often, two points defining the axis of an object must be measured and adjusted to either match a nominal orientation, or to match a prescribed pitch and yaw.

This watch window allows you to measure two discrete points with a single instrument. The watch window then indicates the pitch (elevation) and yaw (azimuth) of the measured axis relative to the prescribed axis (Figure 13-5).

The working coordinate frame defines the vertical direction—therefore its Z axis is used in the calculations.

To Watch an Axis Using Two Point Axis Alignment:

1. Measure or construct two points along the axis that serve as the nominal or “goal” points.

2. From the menu, select **View>Watch Window>Watch Two Point Axis Alignment**.
3. Select the instrument to use for measurements.
4. Select the first and second points, defining the X axis (from the first to the second). The measured X axis is along the direction from the first to the second points. The Z axis is clocked along the working Z axis, therefore if pitch to gravity is desired, this axis should be along gravity.



Figure 13-5. Two Point Axis Alignment.

5. The **Monitor Axis** dialog appears.
6. Specify the group name in the text field, if desired.
7. Click **Target 1** and **Target 2** to measure the points and associate them with the ends of the axis. The pitch and yaw values will update.
8. To automatically measure both targets (assuming you have at least two targets installed and a steerable instrument), click the **Cycle Targets** button. The instrument will point, search for, lock on, and measure each target.
9. As repeated measurements come in, SA will automatically increment the group name and put each repeated measurement in a separate group.

Note: **Cycle Targets** will not work if the axis points have moved beyond a reasonable search region for the instrument.

Watch Theodolite Observer Point to Point

When using the Theodolite Data Observer (see the “Measurement” chapter), this watch window compares the latest computed point to a selected point. For instance, if you have three bundled theodolites, and they are all focused on a common target, you can query the instruments with the Theodolite Data Observer to calculate the position of the target. This watch window will compare this position to a specified point. Note that target offsets are not used here.

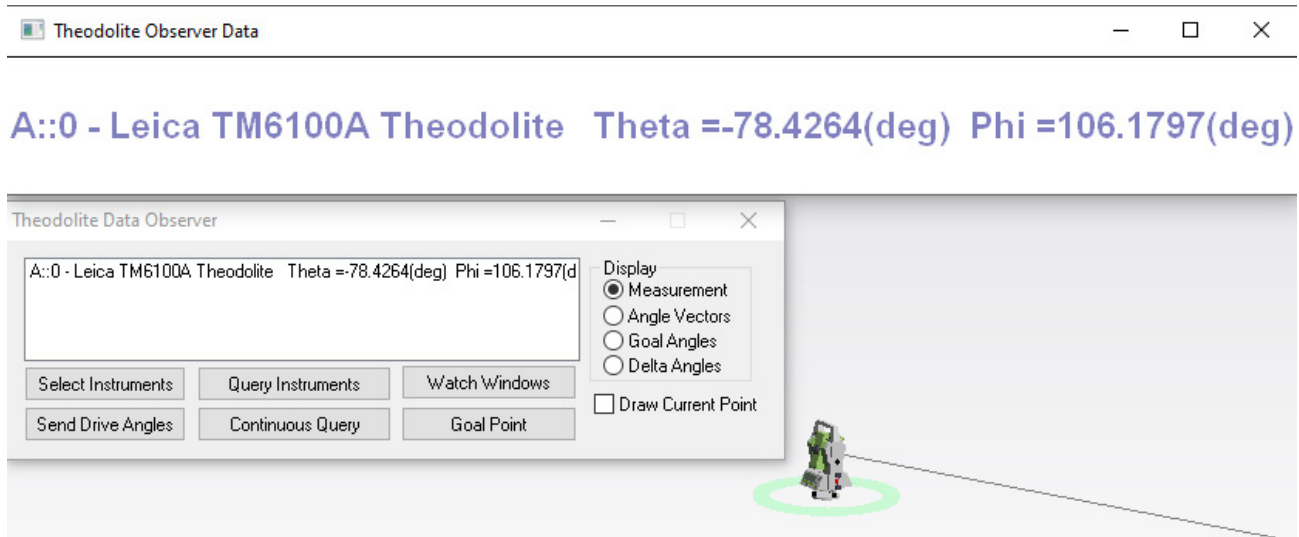


Figure 13-6. Theodolite Data Observer and available display options.

To Watch Theodolite Observer Point to Point:

1. Ensure that the Theodolite Data Observer is up and running, and that your instruments are bundled (see the “Measurement” chapter).
2. From the menu, select **View>Watch Window>Add Theodolite Observer Point to>Point**.
3. Select the point to compare against.
4. As updated data arrives from the theodolite data observer, the watch window will automatically update.

Watch Theodolite Observer’s Point

Similar to “[Watch Theodolite Observer Point to Point](#)” on page 388, except the point’s coordinates are displayed—no comparison is performed to another point. Note that target offsets are not used here.

To Watch the Theodolite Observer’s Point:

- From the menu, select **View>Watch Window>Add Theodolite Observer’s Point**.

Watch Theodolite Observer’s Data

This watch window displays all data from the Theodolite Data Observer, depending on the settings in the Theodolite Data Observer. For instance, it will display instruments and theta/phi angles from those instruments.

To Watch Theodolite Observer’s Data:

- From the menu, select **View>Watch Window>Add Theodolite Observer’s Data**.

Watch Window Options

Right-clicking a watch window will present a context menu containing a number of options for changing how data is represented and displayed. These options are discussed below.

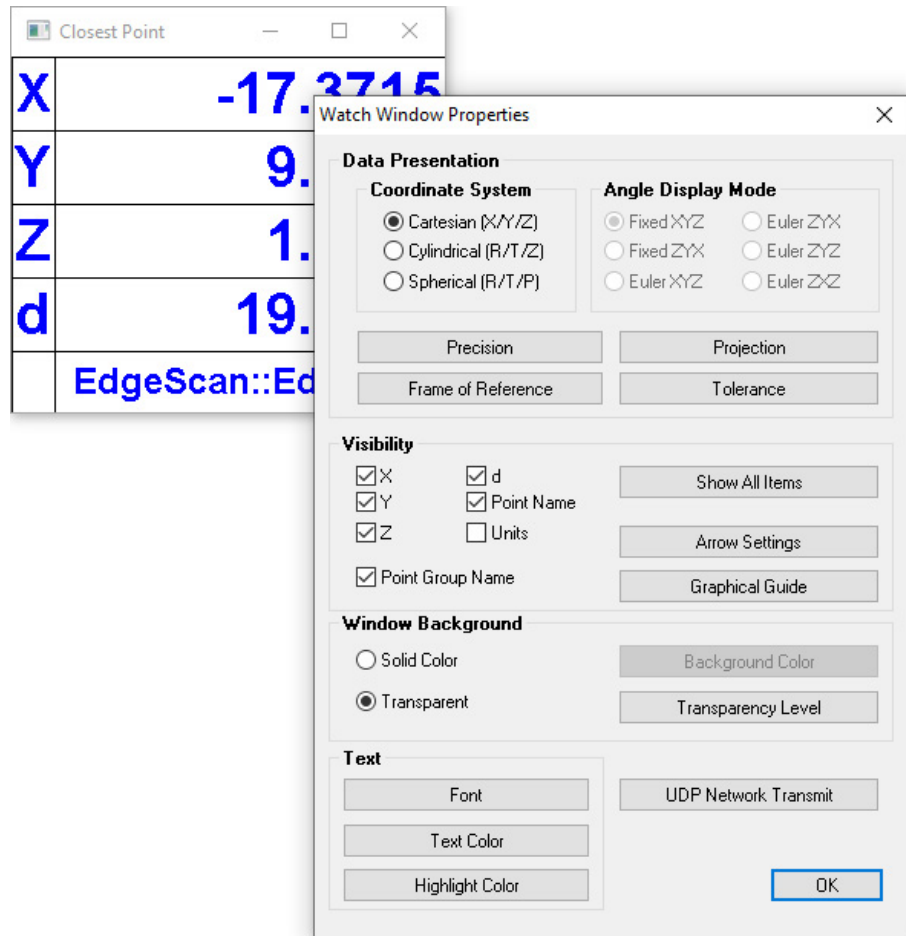


Figure 13-7. Example Watch Window properties dialog used for configuration

Data Representation

Coordinate System

Watch windows can display data in one of three coordinate system types: Cartesian, Cylindrical, or Spherical. In the cylindrical/spherical case, the XY axes of the coordinate frame define the θ plane. In the spherical case, the Z axis of the coordinate frame defines the $\phi=0$ direction. Values will therefore be displayed in either of 3 ways relative to the current reporting frame:

- XYZ coordinates in the default cartesian system

- R (radius or distance from the axis), T (theta or rotation about the axis), Z (height from the origin) in cylindrical coordinates
- R (radius or distance from the origin), T (theta or horizontal rotation), P (Phi or vertical rotation) in spherical coordinates.

Angular Display Mode

Angular rotation values can be reported in several display modes when full frame to frame transformation information is displayed.

Precision

The number of decimal places used to display watch window data can be controlled. Depending on the necessary accuracy and the user's tolerance for quickly-changing values, the default values may provide too much precision (revealing noise and resulting in quickly-changing numbers) or too little (yielding insufficient detail for the required accuracy).

By default, SA uses the default length and angle precision set in the User Options. However, you can use any preferred precision. The options when changing decimal precision are as follows:

- **Use SA Significant Digits.** When selected the watch window will use the software defaults. Deselect this to use custom precision settings for the watch window.
- **Length.** The number of decimal places to use for displaying linear values.
- **Angle.** The number of decimal places to use for displaying angular values.

To Change a Watch Window's Decimal Precision:

5. Right-click the watch window and select **Precision** from the context menu. The *Watch Window Significant Digits* dialog will be displayed (Figure 13-8).

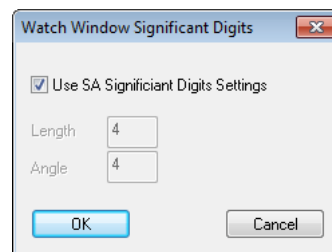


Figure 13-8. Changing decimal precision in a watch window.

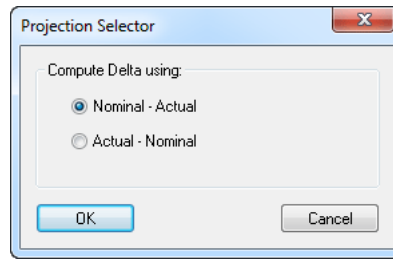
6. Apply your desired settings, then click **OK**.

Projection Options

As is the case with relationships and queries, you can control how deviations are depicted in a watch window. This is handled through the watch window's projection options.

The projection options for a point-based watch window are shown in [Figure 13-9](#):

Figure 13-9. Projection options for a point to point watch window.



- **Nominal-Actual.** Deviations show the delta from the “nominal” or reference point to the measured point. This presents the deviation from an inspection point of view—that is, how far is the measurement from nominal?
- **Actual-Nominal.** Deviations show the delta from the measured point to the reference point. This presents the deviation from a build point of view—that is, how far do I need to move the probe to get to nominal?

Projection options for an object-based watch window are shown in [Figure 13-10](#). There are four available indicator styles:





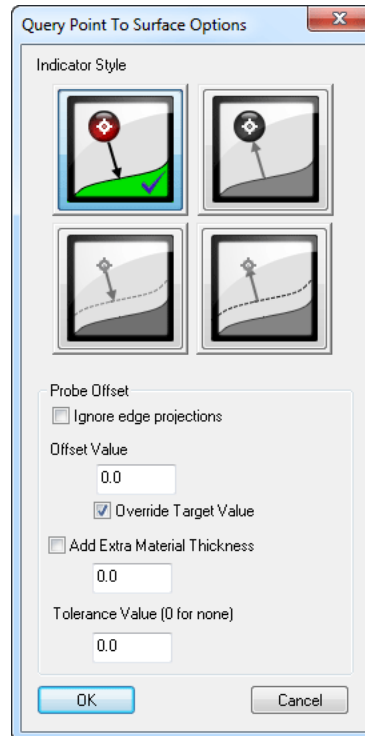
-  Deviations are from the measurement to the design—that is, they reflect the “build” point of view.
-  Deviations are from the design to the measurement—that is, they reflect the “inspect” point of view.
-  Deviations are the same as the “build” option above, but the arrows are drawn with their tails at the center of the probe.
-  Deviations are the same as the “inspect” option above, but the arrows are drawn with their heads at the center of the probe.
- **Ignore Edge Projections.** This option is ignored.
- **Offset Value.** The desired offset value when overriding the probe’s offset.
- **Override Target Value.** When enabled, the probe’s offset is overridden by the provided value.
- **Add Extra Material Thickness.** Adds virtual thickness to the objects being compared against. This is useful, for example, when you have an OML surface in your CAD model but you’re actually comparing to an IML, assuming a certain thickness for the material.

Figure 13-10. Projection options for an object-based watch window.



- **Tolerance Value.** This value is ignored.

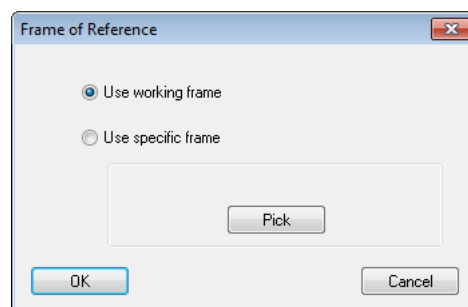
To Set Projection Options for a Watch Window:

1. Right-click the window and select **Projection Options** from the context menu.
2. Enter the desired settings, then click **OK**.

Frame of Reference

By default, watch window data is displayed with respect to the working coordinate frame. Sometimes, you may wish to display the data in a specific coordinate frame, regardless of the working frame (Figure 13-11). For instance, you may have the **World** frame active but want to view a specific watch window in **Local** coordinates.

Figure 13-11. Changing the frame of reference for a watch window.



Warning: This setting (along with others) is saved for a particular watch window type. If you forget to set this back, you may view a watch window at a later time with respect to the incorrect frame.

- **Use working frame.** Always represents the values relative to the working coordinate frame. This is the default.
- **Use specific frame.** When selected, a specific coordinate frame is used for data representation, regardless of the active frame.

The selected frame can be changed using the **Pick** button.

To Change a Watch Window’s Frame of Reference:

1. Right-click a watch window and select **Set Frame of Reference [FoR]** from the context menu.
2. Select the desired option, then click **OK**.

Tolerances

You can set tolerances for watch windows that provide deviation values. Depending on the type of watch window, this may just be a magnitude tolerance (for watching objects) or individual component tolerances (when watching points). The **Vector Tolerance** dialog, which allows setting component tolerances, is shown in **Figure 13-12**.

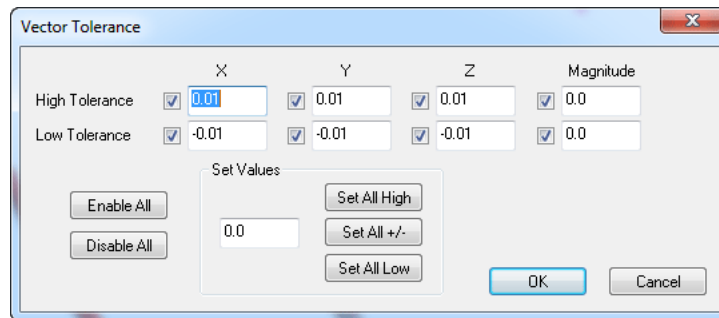


Figure 13-12. Setting component tolerances for a point to point watch window.

In this dialog, you can enable or disable individual component tolerances, set all high or low tolerances to a given value, or set them all to a symmetrical value.

To Set Tolerances on a Watch Window:

1. Right-click the watch window and select **Default Tolerance** from the context menu.
2. Set the desired options, then click **OK**.

Item Visibility

Showing/Hiding Rows

Individual rows in a watch window can be shown or hidden. For instance, if you are only interested in adjusting the XY position of a feature, you can remove the Z component from the window to prevent confusion and simplify the display.

Row entry visibility is controlled using the provided set of check boxes. Only the checked options will be displayed when the watch window properties dialog is closed.

Background Window

Background Color

The *Background color* button provides the ability to set the color of the watch window in *Solid Color* display mode such that it can be set to any color needed to help in visualization.

Transparency

Watch windows can be made transparent, so that their backgrounds are invisible. The window border can also be made partially transparent. This allows you to see what's going on in the graphical view while a watch window is being displayed across the entire screen.

To enable background transparency select the radio button in the properties dialog and use the *Transparency Level* control to adjust the window border transparency.

Text Controls

Font

Watch windows can be displayed in a custom font. You can change the font and style. (Although you can change the font size, this setting is ignored and the size of the window itself is used for determining text size).

To Change a Watch Window's Font:

1. Right-click the watch window and select **Font** from the context menu.
2. In the font dialog, select the desired typeface and style, then click **OK**.

Text Color

You can change the text color for the watch window on a row-by-row basis.

To Change Watch Window Text Color:

1. Right-click on the row in the watch window that you'd like to modify.
2. From the context menu, select **Text Color**.
3. In the color dialog, choose a new color, then click **OK**.

Highlight Color

If a prescribed tolerance for a watch window is exceeded, the out-of-tolerance values will be highlighted. You can control the color used for this highlight.

To Change the Watch Window Highlight Color:

1. Right-click on the row in the watch window you'd like to modify.
2. From the context-menu, select **Highlight**.
3. In the color dialog, choose a new color, then click **OK**.

Arrow Settings

When a watch window is displayed that is comparing to another entity, arrows are displayed in the graphical view as a way to visualize the deviations. You can control how these arrows are depicted using the options in the *Watch Window Settings* dialog (Figure 13-13).

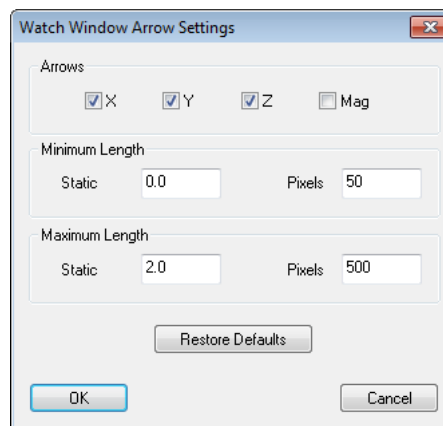


Figure 13-13. Watch window arrow settings.

- **Arrows.** Specifies which arrows should be displayed: X, Y, Z, and/or magnitude. The X/Y/Z arrows show the $dX/dY/dZ$ values. Magnitude shows the shortest distance between the watched item and the entity it's being compared against.
- **Minimum Length (Static).** The real-world minimum deviation below which an arrow does not get any shorter.
- **Minimum Length (Pixels).** The shortest length that any arrow can be drawn in the graphical view.
- **Maximum Length (Static).** The real-world maximum deviation above which an arrow does not get any longer.
- **Maximum Length (Pixels).** The longest length that any arrow can be drawn in the graphical view.

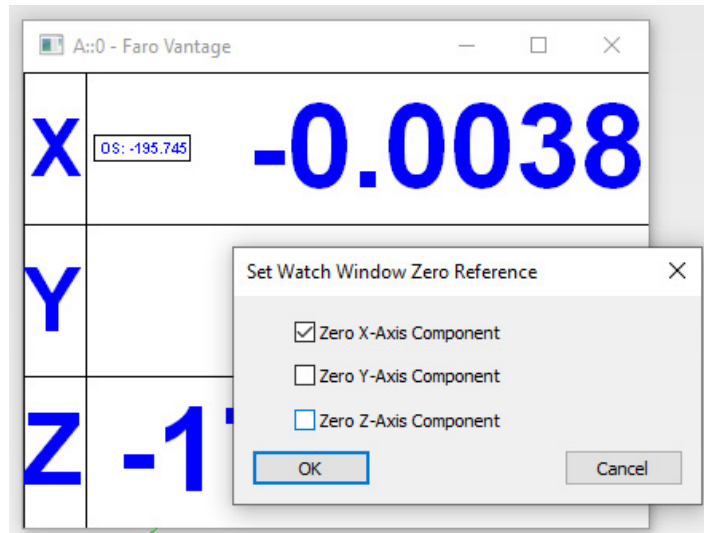
To Change Watch Window Arrow Display Settings:

1. Right-click the watch window and select **Arrow Settings** from the context menu.
2. Choose the desired options, then click **OK**.

Setting a Zero Reference

The X, Y or Z components in a watch window can be zero'd with respect to a particular reference position. To do so, place the reflector in the desired location and double click on the window to bring up the selection control. An offset reference can be added to any or all of the axes (Figure 13-14).

Figure 13-14. Setting a Zero Reference in a watch window

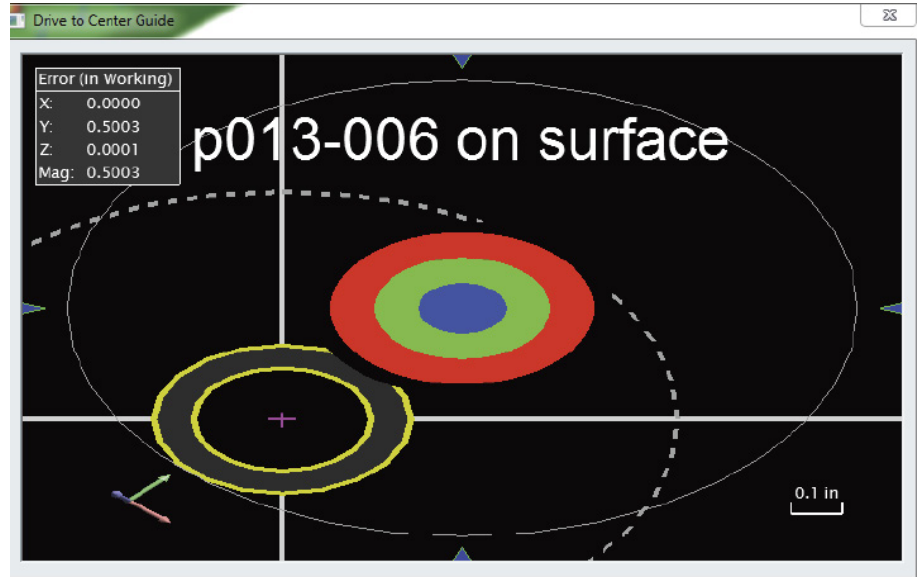


Once set, a display "OS: ###" will be added to the offset row so that this offset is clearly marked. To remove it, simply double click again and uncheck the corresponding zero component box.

Graphical Guide

The Graphical Guide (Drive to Center Guide), originally available through the proximity triggers utility, provides a visual indication of the probe's position with respect to a target in 2D. It is designed to make it easier for an operator to tell if they are centered on a target before taking a measurement, and does this by zooming in as you approach the target location (Figure 13-15).

Figure 13-15. Graphical Guide Display available for many watch windows and used to clearly find a location with high precision.

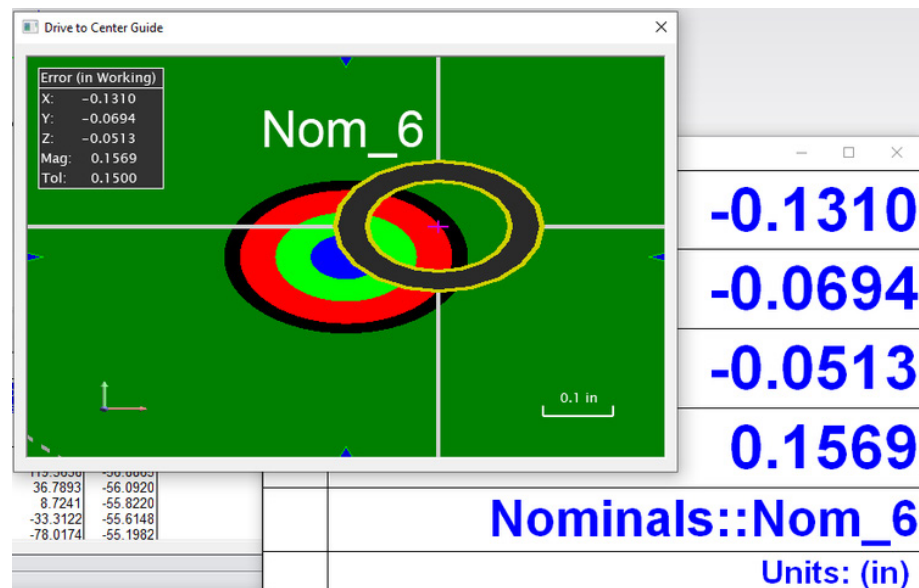


The Graphical Guide is available with **Instrument's point**, **Closest Point** and **Point to Objects** watch windows.

Watch window tolerances are not to be confused with the graphical guide proximity tolerance – they are independently set. A Right-click on the graphical guide window will provide the user an option to change the proximity tolerance. This is a 2D proximity tolerance set within plane of the view.

If you set a specific reference frame for the corresponding watch window, then that will be used to set the view orientation. If you set the reference frame to working frame, then the view will follow that of the main display. When in tolerance the background of the graphical view will turn green (Figure 13-16).

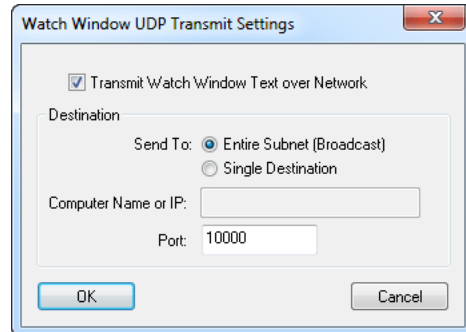
Figure 13-16. In Tolerance display using the Graphical Guide utility.



UDP Broadcast

Watch windows can be configured to transmit their data over TCP/IP using the UDP Protocol (Figure 13-17). This is useful for intercepting watch window data with your own custom code.

Figure 13-17. UDP Settings.



Options for UDP transmitting are as follows:

- **Transmit Watch Window Text over Network.** When enabled, data is transferred over TCP/IP.
- **Send To.** UDP packets can be transmitted to the entire subnet, which is referred to as a broadcast. Or, it can be directed to a specific destination (*Single Destination*).
- **Computer Name or IP.** If transmitting to a specific destination, the IP address or computer name of the recipient.
- **Port.** The port over which the data should be transmitted.

To Transmit a Watch Window's Data over UDP:

1. Right-click a watch window and select **UDP Network Transmit Settings** from the context menu.
2. Select the desired options, then click **OK**.

Transformation Tracking

For 3D instruments (e.g., Trackers) transforming objects in real time requires at least three on-line trackers. This functionality is also called Trans-Track (Figure 13-18).

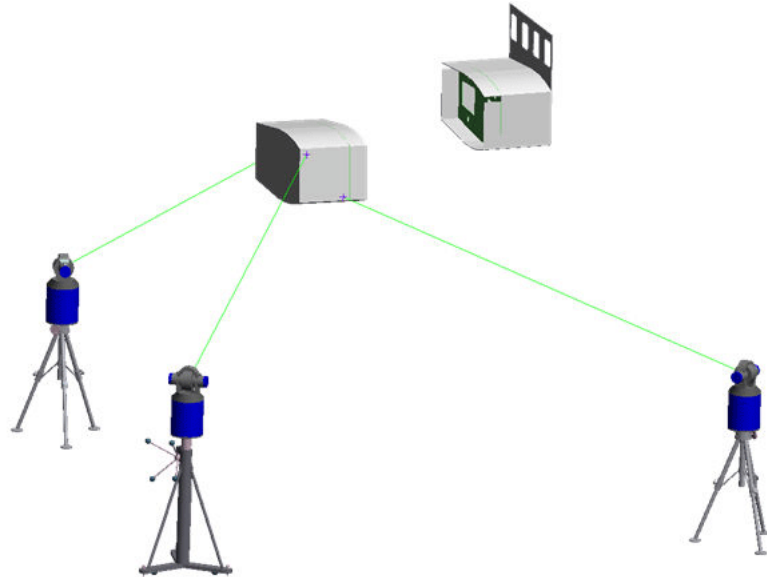


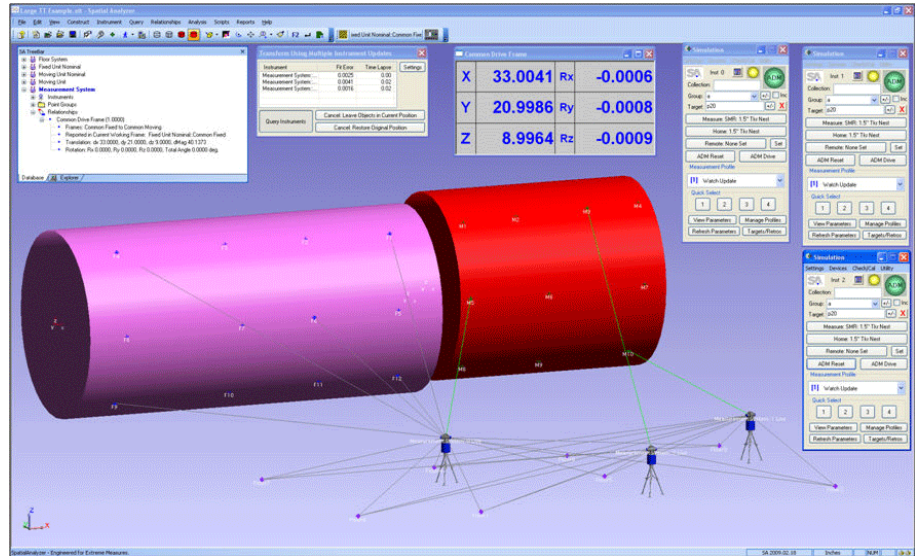
Figure 13-18. Transformation tracking.

Each instrument will monitor and update a point. The movements of the set of points are used together to update objects in 6D. Select one point per instrument. The point selection order must be consistent with the Instrument selection order e.g., select the points in the instrument sequence. After defining the instruments and tracking points; select the objects that need to move with the point set. The Trans-Track interface (shown below) indicates the instruments being monitored and the respective fit error for the point that it is tracking. The time lapse value indicates the time lapse between the measurements (i.e., synchronization error).

Relationships with Trans-Track

Relationships are dynamic by nature and work great with Trans-Track. Watch windows can be displayed for relationships which will allow real time monitoring while moving. Figure 13-19 shows an example of Trans-Track being used to join two cylindrical sections. A frame to frame relationship was used to monitor the position and orientation of the moving unit with respect to its final nominal location.

Figure 13-19. Using trans-track to join two cylindrical sections of an assembly.



Transform in 6D using Instrument Updates

Trans-Track can be accomplished with a single instrument if the instrument supports a 6D target. Examples are the Leica T-probe/T-Mac, API STS and Metris iGPS. This instrument can monitor and measure in 6D. The functionality is identical to Multiple Updates.

Translating Using Instrument Updates

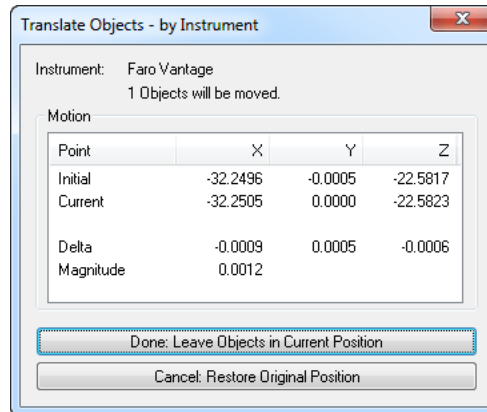
Updates from a 3 degree-of-freedom device such as a laser tracker can be used to translate one or more objects. Note that this is a pure translation--no rotations are applied.

When the command is initiated, the initial target position is recorded. As the target moves from this initial position, the translation is applied to the selected objects.

To Translate Objects Using Instrument Updates:

1. Select **Edit>Move Objects>Translate Using Instrument Updates**.
2. At the prompt, select the instruments (if more than one exists in the file) and the objects you'd like to move. These objects will move together as a rigid body.
3. The *Translate Objects - by Instrument* dialog will appear (see Figure 13-20).

Figure 13-20. Translating objects using an instrument's 3 degree-of-freedom updates.



4. When measurement updates are received from the instrument, the objects will be translated.
5. When finished, you can either click **Cancel: Restore Original Position**, which will put the objects back in their original location, or click **OK: Leave Objects in Current Position**, which will accept the transformation and apply it to the objects.

Transforming in 6D Using Instrument Updates

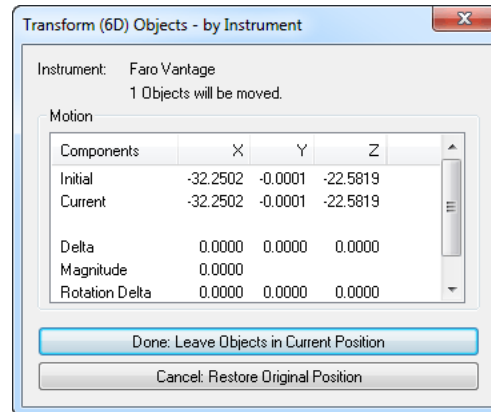
Updates from a 6 degree-of-freedom device (such as a T-Probe, portable arm, T-Mac, or STS) can be used to transform one or more objects. The positional and rotational delta experienced by the device is applied to the objects involved in the transformation.

When the command is initiated, the position and orientation of the device is recorded. As the device moves and rotates from this initial position, the transformation is applied to the selected objects.

To Transform Objects in 6 Degrees of Freedom Using an Instrument's Updates:

1. Select **Edit>Move Objects>Transform in 6D Using Instrument Updates**.
2. At the prompt, select the instruments (if more than one exists in the file) and the objects you'd like to transform. These objects will move and rotate together as a rigid body.
3. The **Transform (6D) Objects - by Instrument** dialog will appear (Figure 13-21).

Figure 13-21. Transforming objects using an instrument's 6 degree-of-freedom updates.



4. When 6-DOF measurement updates are received from the instrument, the object position and orientation will be updated automatically.
5. When finished, you can either click **Cancel: Leave Objects in Current Position** to accept the transformation, or **Cancel: Restore Original Position** to move the objects back to their original position.

Transforming in 6D Using Multiple Instrument Updates (also known as Trans-Track)

SA has the capability of synchronizing three degree-of-freedom measurements from 3 or more instruments and combining them to calculate a six degree-of-freedom transformation. This is commonly referred to as *Trans-Track*. For *Trans-Track* to function correctly, three or more instruments must first be tied to a common coordinate system. This can be achieved using any accepted technique, such as USMN, bundling, or best-fit. Once the instruments are tied together, each instrument must measure a single point, which becomes the reference point for that instrument.

The *Trans-Track* operation is then started, and the instruments are simultaneously polled for updates. A real-time transformation is then calculated, and that transformation is applied to any objects of choice.

To Transform in 6D Using Multiple Instrument Updates:

Tip: Spread reference points apart as far as possible to minimize angular errors in *Trans-Track*.

Tip: The accuracy of the *Trans-Track* operation is limited to the accuracy of the network, so try to tie instruments together as accurately as possible.

Tip: Be careful to place targets such that they will be visible throughout the range of motion of the moving object.

1. Ensure that you have three or more instruments tied together to the same reference system (usually a set of common points), and that their interfaces are all running in SA.
2. Affix three targets to the moving object, with one instrument watching each target--these will be your reference points.
3. Measure a single reference point from each of the instruments.
4. From the menus, select **Edit>Move Objects>Transform in 6D Using Multiple Instrument Updates**.
5. When prompted, select the instruments that will be part of

the operation. You must be explicit about the order in which you pick the instruments.

6. Now select the earlier-measured reference points that correspond with the instruments. The points must be selected in the same order as the instruments. (That is, select the first instrument's reference point first, then the second instrument's point, etc.)
7. When prompted, select the object or objects to which you'd like the transformation applied.
8. The *Transform Using Multiple Instrument Updates* dialog will appear (see [Figure 13-22](#)).
9. Click the **Query Instruments** button, and SA will begin polling the instruments and calculating the transformation. The dialog will list the fit error between the live set of points and the reference set. The time lapse among the measurements is also displayed.
10. When finished, press **Done: Leave Objects in Current Position** to accept the transformation, or **Cancel: Restore Original Position** to undo any movement.

Tip: Measurements cannot be perfectly synced, therefore slower movement will result in more accurate real-time transformations due to a smaller time delay among the instrument measurements. Allowing the targets to become stationary for a moment will remove sync-based error from the transformation.

The Trans-Track dialog also has a few settings that can be modified (see [Figure 13-23](#)):

- **Point Best-Fit Error Tolerance.** If a point's best fit error to the corresponding reference point exceeds this threshold, the point will be highlighted as a warning.
- **Time Lapse Tolerance (sec.).** If the time difference between the first point and another point exceeds this threshold, the point will be highlighted as a warning, indicating that the measurement sync was not acceptable. This can be ignored whenever the targets are stationary.
- **Delay after update (sec.).** The time delay between one set of measurements and the next.

Figure 13-22. Performing a Trans-Track operation with three laser trackers.

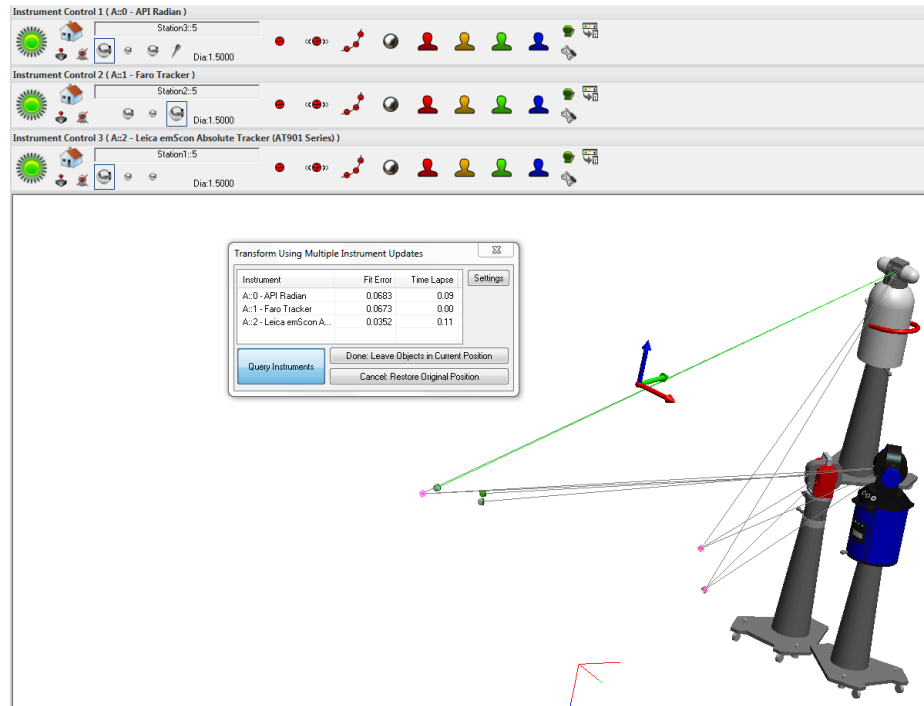
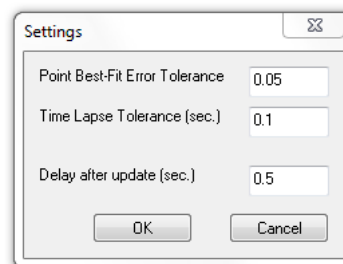


Figure 13-23. Trans-Track settings.



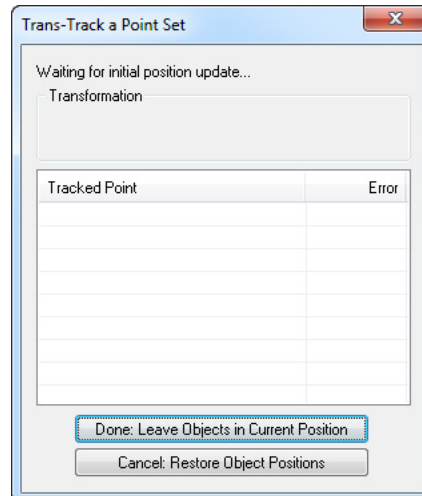
Transforming in 6D Using a Batch of Points from an Instrument

Some instrument systems, such as photogrammetry systems, are capable of tracking several points simultaneously. SA is capable of watching this batch of points and updating the transformation of one or more objects as the updates arrive. Since this command only involves one instrument, alignment among multiple instruments is not required as it is with **Transform in 6D Using Multiple Instrument Updates**.

To Transform Objects in 6D Using a Batch of Points From An Instrument:

1. From the menus, select **Edit>Move Objects>Transform in 6D Using a Batch of Points from an Instrument**.
2. Select the object(s) that you would like to transform.
3. The **Trans-Track a Point Set** dialog appears (Figure 13-24).
4. Before moving anything, measure a set of points. This will be used as the reference for the initial position.

Figure 13-24. The *Trans-Track a Point Set* dialog appears.



5. As each set of measurements are taken, the position of the selected objects will be updated, and the fit error will be reported in the dialog.
6. When finished, click **Done: Leave Objects in Current Position** to accept the transformation, or **Cancel: Restore Object Positions** to cancel the command and put all objects back into their original positions.

Guiding Objects in 6D Using Point Measurements

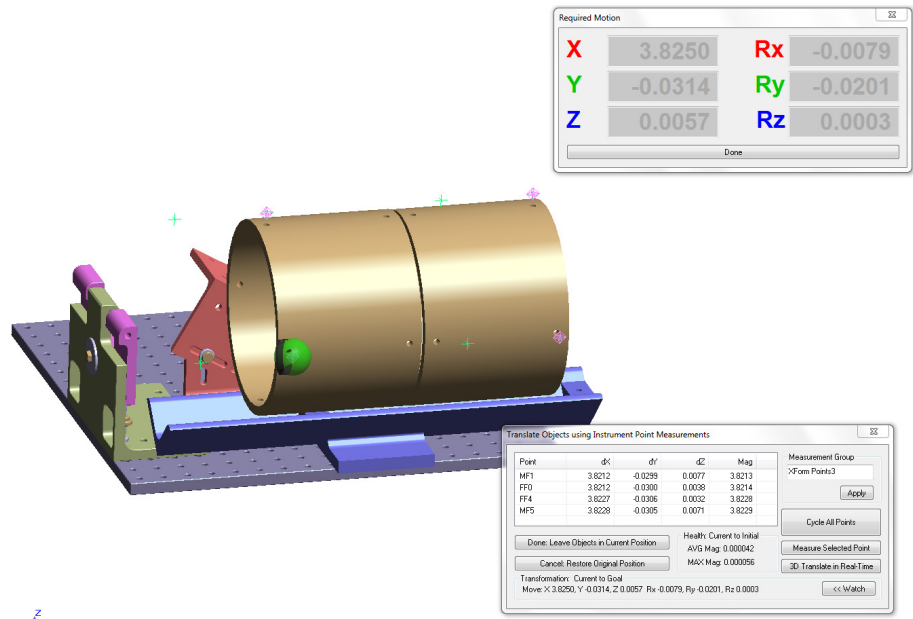
If you would like to continually monitor a 6 degree-of-freedom transformations to one or more objects, but only have access to a 3D device available, you still can. This is accomplished using a two step process.

The process involves measuring a set of reference points affixed to the moving object, and creating a corresponding set of “goal” points which represent these reference points transformed to the desired final location. This establishes as a starting point for the process.

Next you monitor a single point selected from the list as you move the part. During this step a simple translation and point to point distance is available.

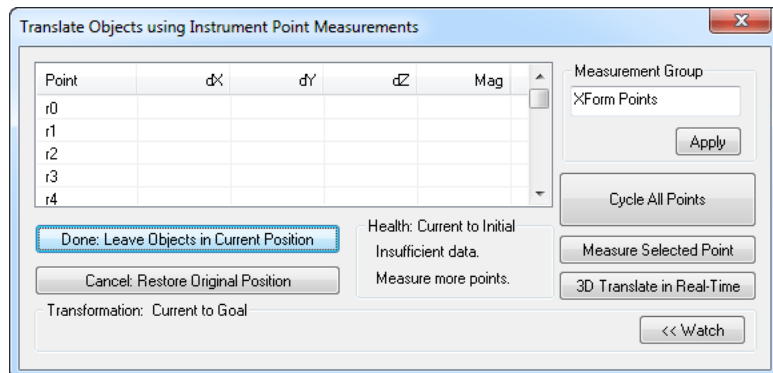
You then stop and repeat a measurement of the reference points on the moving object, and SA will update the position of the moving objects and also update a full 6D display or watch window indicating the required motion to transform from the new updated position to the final position (Figure 13-25).

Figure 13-25. Guiding objects using point measurements.



The controls for guiding in 6D are shown in [Figure 13-26](#).

Figure 13-26. The *Translate Objects using Instrument Point Measurements* dialog.




- Measurement Group.** Defines the name of the point group containing the measured points. If you change this, you must click the **Apply** button.
- Cycle All Points.** When clicked, this button cycles through all reference points, points at their last measured position, searches, locks, and measures each point (applies to steerable, targeted instruments only).
- Measure Selected Point.** Measures a point and associates it with the selected point in the table.
- 3D Translate in Real-Time.** Select a point in the table, then click this button. SA will track the target as this point and *translate* the moving objects. This is useful for getting real-time feedback on movement, without considering rotational information. Once you dial in the position, you can switch back to full 6 degree-of-freedom measurement of your reference


points by toggling this mode back off.

- **Watch.** Click this button to open a watch window showing the transformation, in the active frame, required to get from the current position to the goal position.

To Guide Objects in 6D Using Point Measurements:

 **Tip:** Just as with a best fit or trans-track operation, spread your reference points out to improve your baseline and reduce angular errors in the solution.

1. Locate into a coordinate system as necessary.
2. Measure a set of reference points on the moving object. Three are required at a bare minimum, more are better.
3. Create a set of corresponding goal points in the destination position. This is usually accomplished by copying the reference group and applying an appropriate transformation to the copy.
4. From the menu, select **Edit>Move Objects>Guide Objects in 6D using Point Measurements**.
5. At the prompt, select the destination (goal) group.
6. Select the reference group on the moving object.
7. Select any additional objects you'd like to transform.
8. When the dialog appears, select the appropriate point in the table and click **Measure Selected Point** to measure it. Or, ensure targets exist at each reference location and click **Cycle All Points** to measure them all automatically.
9. Using the reported transform, apply the desired movement to the moving objects, then repeat the measurement.
10. When satisfied with the final position, click **Done: Leave Objects in Current Position** to accept the transformation.

 **Note:** When using "Cycle All Points", movements must be small enough relative to the previously measured position to ensure that the instrument can still find the targets.

Big Data Window

The Big Data Window ([Figure 13-27](#)) allows you to view various measurement data as it arrives. This includes:

- Target Name
- Target Coordinate
- Component Measurement Uncertainty
- Pointing Error
- Apex Angle
- Group Comparison
- Theodolite Data Observer information

- Error Sensitivity
- Distance Between Last Two Points

The window is customizable: you can turn different data elements on and off, resize and relocate their position in the window, change fonts, and control colors, borders, and other visual elements.

The window also allows you to step forward and backward through recent measurements to view the associated data, copy the data to the clipboard, or take a graphical snapshot of the window.

To View the Big Data Window:

- From the menu, select **View>Big Data Window**.

To Enable/Disable Elements of the Big Data Window:

1. In the Big Data Window's menu, click **Customize**.
2. The *Customize* dialog appears. Select different elements from the dropdown, and check or uncheck their *Visible* option, then click **Apply**.

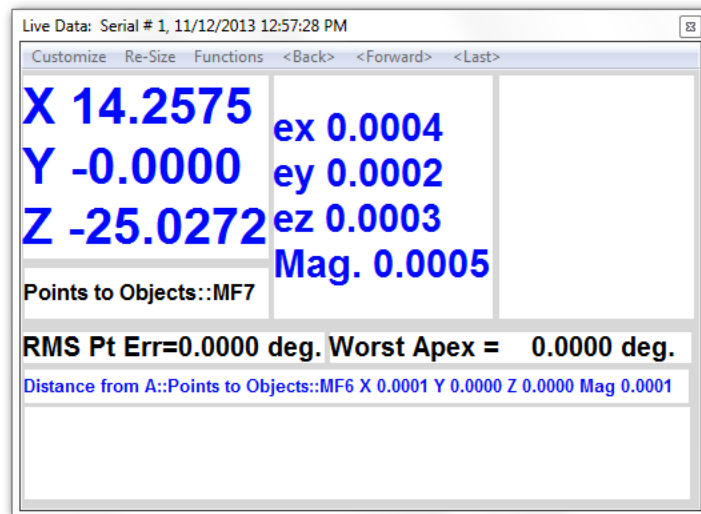


Figure 13-27. The Big Data window.

To Customize an Element of the Big Data Window:

1. From the big data window's menu, click **Customize**.
2. In the *Customize* dialog, select the element from the dropdown that you'd like to modify.
3. Choose among the below options to modify the element:
 - **Text Font.** The font used to display the textual data.
 - **Text Color.** The color of the font used to display textual data.
 - **Draw.** When enabled, draws a border around the element.
 - **Border Color.** Determines the color of the element's border.
 - **Fill.** When enabled, fills the element's background with the se-

lected color.

- **Fill Color.** The color to use for the element's background.
- **Highlight Color.** Sets the highlight color, if used by the element.

To Resize and Reposition Elements in the Window:

1. In the big data window's menu, select **Re-Size>Allow Sizing**. This puts the window into a resizing mode.
2. Click and drag an element to move it. Right-click and drag an element to resize it horizontally and vertically.
3. When finished, again select **Re-Size>Allow Sizing** to disable resizing mode.

To Copy the Big Data Window Text to the Clipboard:

- From the big data window's menu, select **Functions>Copy text to clipboard**.

To Copy the Big Data Window to the Clipboard as an Image:

- From the big data window's menu, select **Functions>Copy graphics to clipboard**.

To Move Through Your Measurement History:

- Use the **<Back>**, **<Forward>**, and **<Last>** menu items in the big data window to navigate through historical measurement data

UDP Broadcast

SA's instrument interfaces have the ability to send a stream of UDP packets when their measurements are sent to SpatialAnalyzer. To enable this for trackers, for example, go to the **Utilities** menu and select **UDP Data Stream** (Figure 13-28). Then configure the settings:

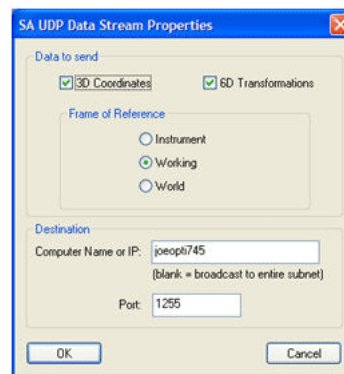


Figure 13-28. The UDP data stream properties.

Now, subsequent 3D and/or 6D measurements will also be sent as UDP packets to either the entire subnet or a particular IP address.

SA also contains a diagnostic utility so you can view this data stream

(Figure 13-29). Navigate file explorer to the SA install directory and run `SpatialAnalyzer UDPMonitor.exe`. Set it to the same port and hit Start. When measurements or updates happen, the data will flow into the monitor, as pictured at right.

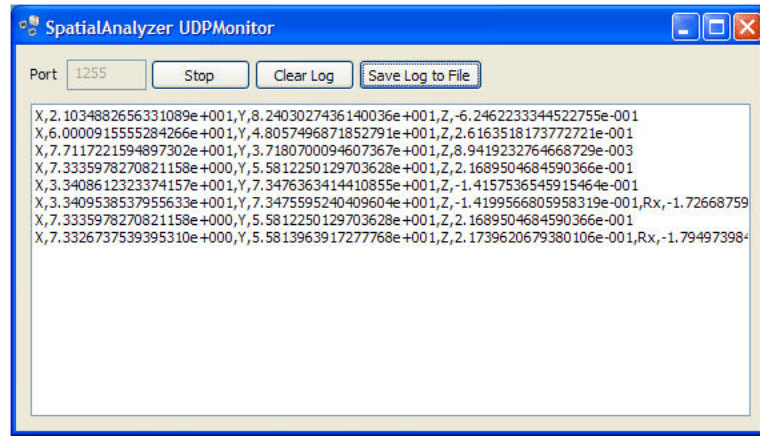


Figure 13-29. The UDP Monitor.

This output can be subsequently saved to a .CSV file.

SA Remote and Metrology Remote

New River Kinematic and SA now offers integration with two separate Apps for remote control of tracks and total stations. These include:

- **SA Remote.** For iPhone users
- **Metrology Remote.** For Android and Windows users. The Metrology Remote is currently available for download from the [Kinematics.com](http://kinematics.com) webpage. For more information on the Metrology Remote contact Support@kinematics.com. But its capabilities are much the same as those described below for the SA Remote.

SA Remote

SA Remote (Figure 13-30) is an iOS app that allows real-time viewing of 3-DOF and 6-DOF spatial data from SA and permits remote control of the SA Laser Tracker interface.

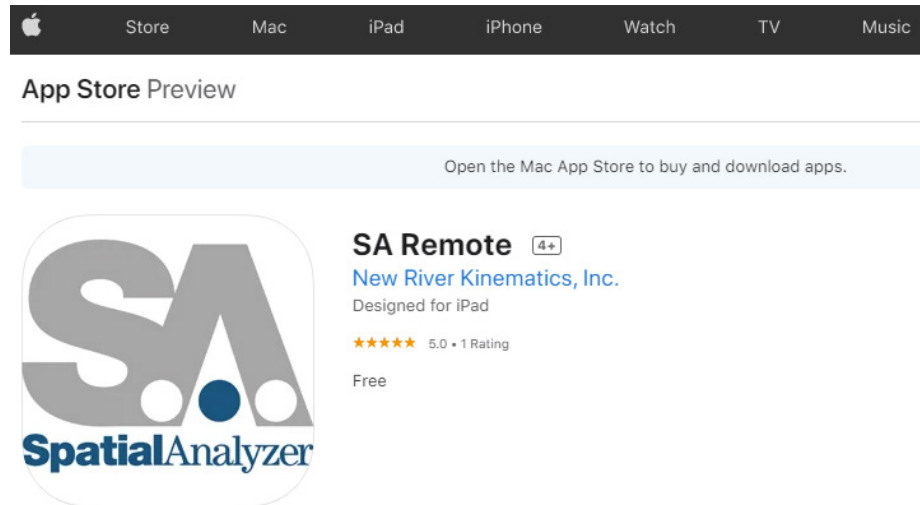


Figure 13-30. SA Remote App.

SA Remote is available as a free app on the Apple App Store for iPhone® and iPod® touch devices. SA Remote will also run on an iPad®.

SA Remote will receive UDP data for display from watch windows in SA or the Laser Tracker interface on port 10000. If multiple sources of watch data are available, the app allows switching between the sources allowing selection of which watch source on which to focus.

SA Remote can also view/control the SA Laser Tracker interface. This requires entering the IP/hostname of the computer running the SA Laser Tracker interface into the app in the settings page. Once this is done, SA Remote will provide display and control over the interface remotely.

Initial Setup

Downloading and installing the App

You should be able to open the app store and search for the “SA Remote” app by name to download and install. But if needed, the SA Remote app can also be found by here:

<http://itunes.apple.com/us/app/sa-remote/id473334282?ls=1&mt=8>

Connecting the Phone, Computer and Tracker

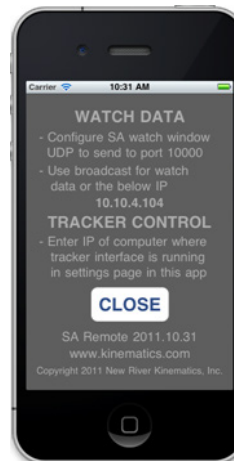
The upper half of the app screen is used to display watch data while the lower half is used to view and control the SA Laser Tracker interface. Initially, the display will likely look like [Figure 13-31](#) since no watch data is available yet and the SA Laser Tracker interface IP/hostname has not yet been entered.

Figure 13-31. SA Remote as it first appears on the iPhone.



Touch the small “i” info button (right) which is in the upper right corner of the screen to open the quick help screen (Figure 13-32).

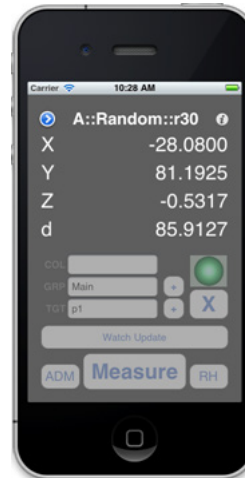
Figure 13-32. The SA Remote quick help screen.



The quick help screen provides a brief summary of the setup tasks for the app. It also displays the device IP address which can be used when setting up the UDP watch transmission settings in SA.

When running SA v2011.10.20 or newer, add a watch window and enable UDP broadcast. Specify port 10000. If SA is not running on the same subnet as the iOS device, you may need to choose directed UDP transmission entering the IP address of the device. The device IP address is available in the quick help page mentioned above. Once correctly configured, the top area of the app screen should start showing watch data as in Figure 13-33.

Figure 13-33. Watch Window information is now streaming to the device.



You can also enable UDP broadcast or directed transmission from the SA Laser Tracker interface.

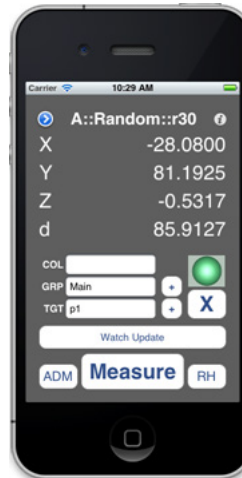
To configure communications with the SA Laser Tracker interface, touch the small detail button in the upper left corner of the screen to bring up the settings view. In this view (Figure 13-34), enter the IP or hostname of the computer running the SA Laser Tracker interface as well as the desired decimal precision and then tap **Close**.

Figure 13-34. Configuring the IP address.



Once the SA Remote app establishes communications with the SA Laser Tracker interface, the lower portion of the screen will enable and provide display and control of the interface as shown below.

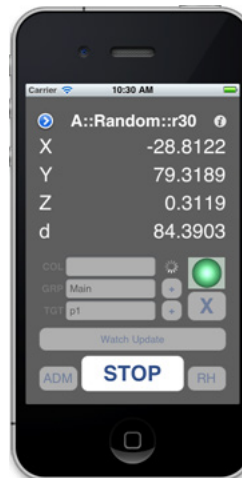
Figure 13-35. Tracker control is now enabled.



When the SA Laser Tracker interface is busy, the lower portion of the screen will be disabled for input, a busy animation will show, and the measure button will change to stop, allowing you to cancel the current operation as shown at right.

If multiple sources of watch data are available (multiple watch windows and/or instrument UDPs enabled) the label at the top of the screen will turn into a button indicating that you can tap it to select which source of watch data to view (Figure 13-36).

Figure 13-36. The button in the top left indicates you can select a different watch source.



Tapping the button will open the watch source selection view shown in Figure 13-37.

Figure 13-37. Selecting the desired watch source.



Select the desired source of watch data and then tap **CLOSE**. The selected source will then be displayed on the main screen.

Available Status/Commands

- The COL/GRP/TGT fields show the current collection/group/target fields from the SA Laser Tracker interface and you can touch them to change the values; when changed the new values will be reflected in the interface as well.
- The + buttons to the right of GRP and TGT will auto-increment the corresponding fields (decrement is not currently supported).
- The status light will reflect what is shown in the interface.
- The X button can be used to remove the last measurement.
- The long button above the measure button shows the name of the current measurement profile. When touched, it cycles through the quick-select profiles (1 – 4) and the name updates to reflect the current profile.
- The **ADM** button commands an ADM Reset.
- The **RH** button commands a Remote Home, if one is set.
- The **Measure** button initiates a measurement using the current measurement profile and changes to **Stop** when the interface is busy and can then be used to cancel the current operation.

Relationships are extremely powerful tools that have a number of uses: Dynamic inspection, alignment, virtual fit-up, real-time assembly, and optimization can all be driven by SA's relationships. Relationships provide optimization results that far surpass traditional methods, and without a complicated setup process.

What is a Relationship

Relationships are not physical objects in themselves but a defined link between objects. In the simplest case a relationship calculates the distance between two related entities and re-calculates this distance whenever either of the related items are moved. This linking process is quite efficient and can be expanded to include many points and many objects simultaneously. By building a relationship you can use this dynamic link in many powerful ways.

- **Reporting.** Relationships can be used for automatically updating dynamic changes between objects. Relationships can therefore be used directly to report deviations.
- **Build Operations.** Relationships can be used to update displays or watch windows to guide users to specific location for part alignment.
- **Dynamic Geometry Fitting.** Geometry Relationships are a special type of relationship that updates the properties of an object based upon the measurements. The relationship links the measurements to the resulting geometry making it dynamic.
- **Alignment.** Multiple Relationships can be used simultaneously and weighted independently for sophisticated alignment operations

Definition of Relationship Types

Points Based Relationships

The simplest relationship types are points based relationships. These relationships continually evaluate the distance between the associated points and objects and are used primarily for reporting and alignment purposes.

Point to Point

Reports deviation between two specific points.

Point to Object

Reports deviation between a specific point and object.

Points to Objects

Reports deviation between one or more points and the closest selected object from a list of one or more objects.

Groups to Objects

Reports deviation between multiple groups and multiple objects.

Point Clouds to Objects

Reports deviation between the points contained within one or more clouds and the closest object from a list of multiple objects.

Group to Group

Reports deviation between points of the same name in two point groups. Along with the Group to Nominal Group relationship it can be used for alignment through the right-click option **Execute Alignment**. Doing so triggers a best fit using the two associated groups, aligning the second group to the first.

Object Orientation Relationships

Object orientation relationships were added to monitor differences in the 6D orientation of one object relative to another object.

Frame to Frame

Reports a delta transform between two frames expressed in the working coordinates. This is not to be confused with the Frame Wizard relationship that drives the construction of a frame from a set of references.

Object to Object Direction

Reports the angle and mutual perpendicular distance between two objects.

Special Function Relationships

Group to Nominal Group

This relationship type functions like a Group to Group relationship but with added inspection guidance. It provides:

- Visual guidance with bull's-eyes identifying the unmeasured points
- Automatic renaming functionality by proximity
- Closest Point Watch window function
- Along with the Group to Group relationship it can be used for alignment through the right-click option **Execute Alignment**. Doing so triggers a best fit using the two associated groups, aligning the measured group to the nominal.

For more details refer to “[Group to Nominal Group](#)” on page 430.

Points to Points

This relationship type functions like a Group to Group relationship in that it provides a comparison between commonly named points. Its unique in that it gives the user the flexibility to select points from a combination of groups both as reference and as actual. All possible comparisons will be made between points in the reference list and those within the actual list by point name, regardless of collection and group designation.

Vector Group to Vector Group

Built with Sim and Gap applications in mind, this relationship requires a Points to Surface Face relationship and a corresponding vector group representing the far side of a gap. It then computes a delta vector group defining the current gap statistics and can be used for alignment.

For more details refer to “[Vector Group to Vector Group Relationships](#)” on page 431.

Points/Groups to Surface Faces

Points to Surface Faces and Groups to Surface Faces function just like Points/Groups to Objects relationship but are tied directly to CAD surface faces, providing greater control in comparisons.

Clouds to Surface Faces

Clouds to Surface Faces functions just like Clouds to Objects relationship but are tied directly to CAD surface faces, providing greater control in comparisons.

Point Cloud to Swatch

The Point Cloud to Swatch relationship can be thought of as a surface

point analysis tool. It is an even more specialized cloud to surface face relationship that also takes a reference point on that surface as an input. It then builds a mini cloud using only cloud points within a proximity to the reference point and a surface normal vector from a large point cloud. The resulting computed point uses the average distance values in the cloud swatch to compute the point offset.

Pipe Fitting

Pipe relationships are designed specifically for pipe alignment and cut calculations (see [“Move Collections by Minimizing Relationships”](#) on page 454).

MP Execution

This MP serves only a single purpose which is to provide a means to identify and run an MP as part of a inspection.

Geometry Relationships

Geometry relationships (or GR-Features) build dynamic geometry. They continually update the position, orientation and size of an SA object based upon the associated points or objects. They also report deviations between measured and nominal geometry.

For details on GR-Features refer to [“Geometry Fitting & GR-Features”](#) on page 457.

General Relationship Properties & Controls

Relationships function like dynamic queries. They calculate the spatial deviations of the associated points from the associated objects. Any time either a new point is added or the associated objects are moved, the relationships are recomputed. This ensures that the correct deviation statistics are always displayed.

Relationships provide many controls and the number of controls increase with complexity. This section provides an overview of the general property controls available for most relationships.

Visual Controls

Highlight Entities

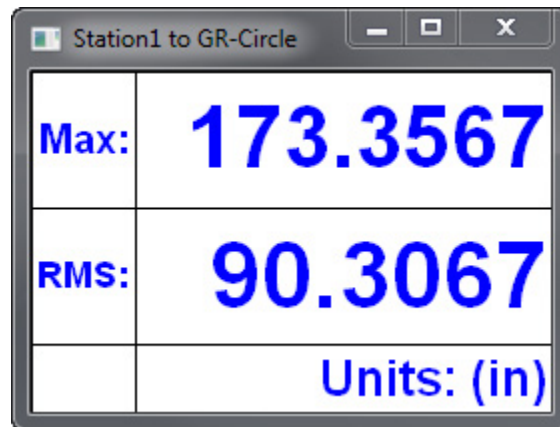
Highlights all the associated points and/or objects. This button functions as a toggle to turn on and off highlighting.

Add Watch Window

Adds a watch window to the relationship. The watch window type will depend on the specific relationship type but typically displays the

Max deviation and RMS for the relationship ((Figure 14-1).

Figure 14-1. Relationship Watch Window



Add Callout

This command does either of two things. If a callout view is active, it adds a callout for the relationship to the active callout view. On the other hand if no callout view is active it builds a new callout view and adds a callout.

Create Associated Inspection Callout

This builds a new callout view and it adds the relationship callout to it. It also activates the callout view when trapping is initiated.

Disable Associated Inspection Callout

This disables the link between the relationship and the inspection callout such that it is no longer indicated when trapping begins.

File Management Tools

Delete

This command deletes the relationship.

Copy

This command copies the relationship to the active collection. It will prompt the user to either keep the object/point associations or clear them.

Move to Another Collection

This command allows the relationship to be relocated to another collection within the job file.

Rename

The command provides a method to rename the relationship.

Projection and Fit Controls

Properties

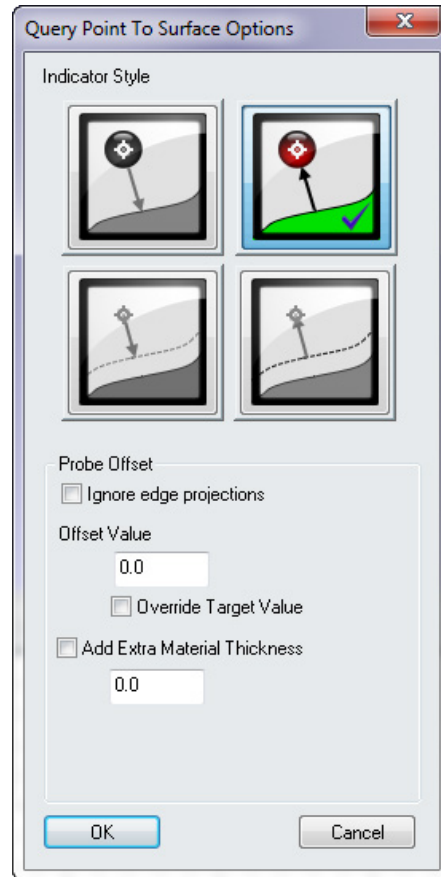
Some relationships provide a condensed set of primary controls in the Properties menu while others are controlled directly through the right-click menu for the relationship.

Projection Properties

Relationships have the same projection properties as a query. They provide the same probe compensation options with the added benefit of allowing changes to these settings at any time (Figure 14-2). These options include display and probe offset controls including:

- **Indicator Style.** Which defines the direction of the vector and the associated end point locations.
- **Ignore edge projections.** When enabled this check box causes point projections to be ignored from the relationship computation.
- **Offset Value.** When the Override Target Value check box is checked the point offsets will be ignored and the entered value will be used in place.
- **Add Extra Material Thickness.** When enabled this check box will add the entered additional material thickness to the offset compensation.

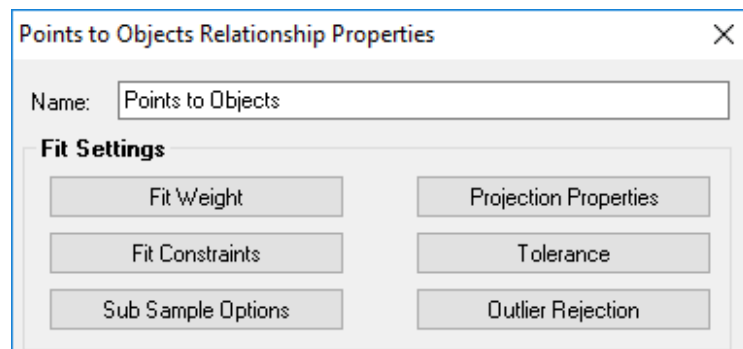
Figure 14-2. Relationship Projection Options



Relationship Fit Settings

There are a number of parameters that are available within a relationships properties that are directly used as part of a relationship fit or optimization. For more information on relationship fitting refer to (see “[Minimizing Relationships](#)” on page 444).

Figure 14-3. Relationship Fit settings.



Fit Weight

Allows more emphasis to be placed on the selected relationship. Often

times weighting can be tweaked to bring a part in tolerance.

Fit Constraints

When a fit constraint is added to a relationship it defines a “dead” region where the relationship has no influence. For example, a high limit can be set at +0.1 and low at 0. This will define a region from 0 to +0.1. When the solution reaches this region, the relationship will stop influencing the solution and the fit will only continue to optimize if other relationships are involved (see “[Minimizing Relationships](#)” on page 444).

Sub Sample Options

Sub sampling is useful for large data sets ([Figure 14-4](#)). Often all of the data is not required for the fit.

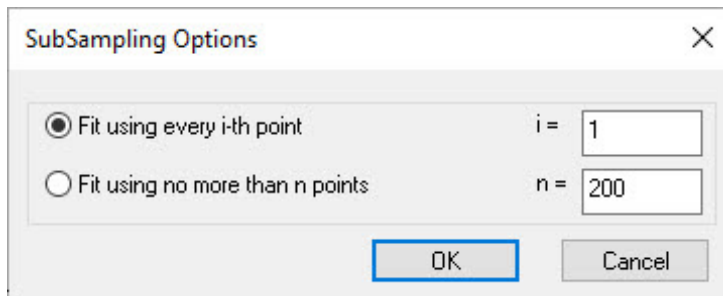


Figure 14-4. SubSampling options dialog.

Relationships provide sub-sampling controls to increase processing speed for large data sets. These include two categories:

- **Fit using every *i*-th point**

Using the *Fit using every *i*-th point* control provides the ability to display a reduced number of vectors that continues to increase in number as the data set increases in size. When $i=1$, all the point deviations are used for display and calculation. As i increases so does the number of skipped points.

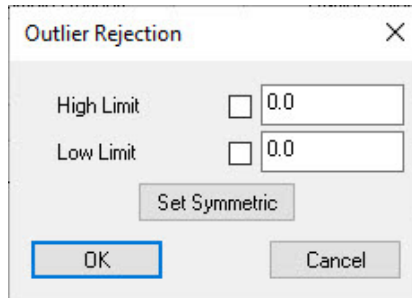
- **Fit using no more than *n* points**

Using the *Fit using no more than *n* points* control provides the ability to place a hard limit on the number of points used in the relationship for display and reporting purposes.

Outlier Rejection

Outlier rejection is a great way of removing stray data for the fitting solution. Stray points can often impose difficulties for the optimization.

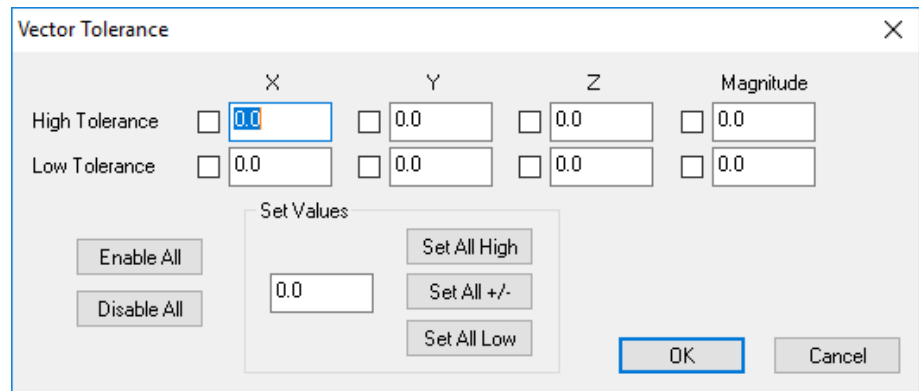
Figure 14-5. Outlier rejection options.



Tolerance Controls

Tolerances can be set for any relationship and apply to all the individual point deviations for both alignment and reporting purposes (Figure 14-6).

Figure 14-6. The Vector Tolerance dialog.



Individual tolerances can also be applied directly to nominal points and these point tolerances will be picked up and used within a Group to Group (and Group to Nominal Group) or point to point relationship. These individual tolerances will then be reported for the relationship report unless the override option is checked within the relationship properties dialog (Figure 14-10). Point to Object relationships will not use point tolerances as it is assumed that tolerances on measured points should not apply to a surface analysis.

Auto Update Vector Group Creation

Enable this option to create a dynamic vector group. Any time the relationship updates the vector group will update as well using the associated points and objects. This provides a very useful way to visualize deviations both during an optimization and later for reporting purposes.

Reporting Options and Controls

Include in Composite Quick Report

This option is determines if a relationship is included or excluded automatically in quick report.

Generate Quick Report

This command generates a quick report for the selected relationship only.

Add to Active SA Report

This command adds the relationship to the active report.

Reporting Options

Report Options control how the deviations will be reported and the format of the resulting relationship tables in a report (Figure 14-7).

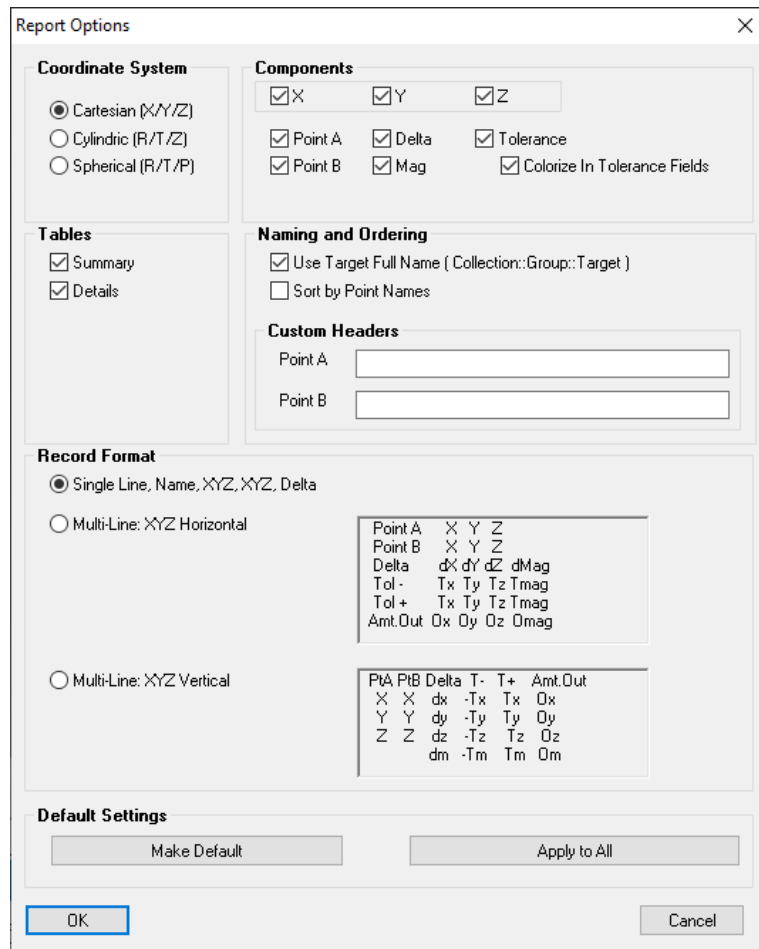


Figure 14-7. Relationship Reporting Options.

Controls within the Reporting Options include:

- **Coordinate System**

Report tables can display points in either of 3 formats: Cartesian (XYZ), Cylindrical (RTZ) or Spherical (RTP) coordinates. The reported

values are presented relative to the working frame unless specified in the reporting frame control.

- **Tables**

Relationships default to displaying both a summary table and a single line details table. When enabled the “Table per Object” option subdivides the details table into multiple details tables, one for each object associated in the relationship.

The summary contains a snapshot of the relationship (Figure 14-8). The complete options will display all the individual deviations.

Points to Objects Relationship Station1 to Station1 (Reported in A::WORLD)				
Statistic	dX (in)	dY (in)	dZ (in)	Mag (in)
Min	-0.0005	-0.0002	-0.0429	-0.0429
Max	0.0003	0.0005	0.0852	0.0852
Average	-0.0000	0.0000	0.0000	0.0000
StdDev from Avg	0.0004	0.0003	0.0578	0.0578
StdDev from Zero	0.0004	0.0003	0.0578	0.0578
RMS	0.0003	0.0003	0.0535	0.0535
Tol Range				
In Tol				
Out Tol				
Count	7			

Figure 14-8. Relationship Summary Table.

- **Components**

Each of the components of the details table can be turned on and off with these check box controls. The display of tolerances can be turned on and off, along with the tolerance color column.

- **Naming and Ordering**

Points can either be displayed using their individual names or the full Collection::Group::Target format to help identify points in the tree with similar names. The points are displayed in the order in which they appear in the tree by default. An option was added to sort the order in which they appear by their names.

Custom Headers can also be added. If a new heading is added they will replace the default Point A and Point B headings in the detail table.

- **Record Format**

Several formatting options are available for the details table including single line and 2 multi-line formats.

- **Default Settings**

The report settings can be applied to other relationships by either setting this relationship's configuration as default or by applying the

setting explicitly to other relationships.

Reporting Frame

By default relationships are reported relative to the working frame. In order to report relative to a different frame one must be specified and this option provides the control to do so.

Combined Relationship Reports

Relationships Report

Accessed either through the **Report** menu or through the right-click category. This Relationship report is designed to report details on all relationships except Geometry Relationships. It includes a summary table with basic statistics on each of the relationships in the selected collection and the details table for each of those relationships that has this details table enabled in its **Report Options** (Figure 14-9).

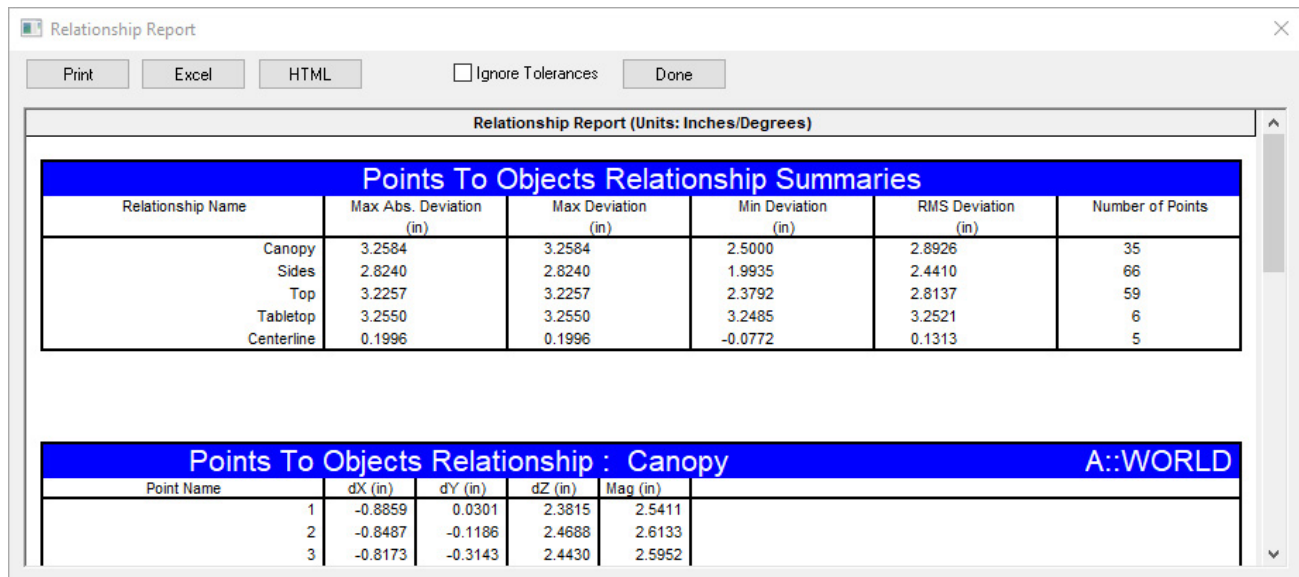


Figure 14-9. Relationship Report

Recomputation

Relationship Dormant Status

Relationships update continually with the job file. With a large number of relationships in the job this re-computation process can take valuable time. For this reason a flag was added to allow individual relationships to be temporarily set as "dormant". When marked as such the relationship will be marked with a snowflake icon ❄️ to indicate that the status is "frozen" and will not be refreshed until the dormant status is turned off. Be sure to turn off and refresh all the relationships before reporting results.

An additional menu option **Analysis>Re-Compute Relationships** was added

to aid in this process.

Measurement Associations

Associate Data

This control allows new data to be associated with a relationship. Depending on the type it provides options to pick points and/or objects and replacing the current list of objects with the new selection.

Clear Point Associations

This option clears the relationships associations.

Delete Point Associations

This option deletes the points associated with the relationship as part of clearing the relationship.

Measurement Trapping and Trapping Controls

For greater detail on trapping (see “[Why Feature Based Inspection](#)” on page 825).

Trap Measurements from an Instrument

This indicates trapping of measurements from a live instrument. When trapping is set any measurement obtained from the selected instrument will automatically be associated with the instrument.

Stop Trapping Measurements

This commands stops trapping

Trap Selections

Trap selection acts like trapping from an instrument but allows the operator to graphically select points to be added to the relationship. It is a great way to add additional points without having to reassociate all the points at once.

Configure Instrument

Instrument configuration provides options to control instrument settings when trapping is indicated.

Lock Out Trapping

Excludes the relationship from the list of relationships within the inspection tab of the toolkit (see “[Feature Inspection](#)” on page 832).

Change order in List

This changes the order of the select relationship within the relationship category in the active collection. The order in inspection tab mirrors the order of the relationships in the tree so being able to control its position in the list is important .

Properties & Controls - Special Function Relationships

Group to Nominal Group

This relationship type functions like a group to group relationship but with added inspection guidance.

Key Features:

This relationship assumes that the 1st group associated is the nominal or design group and uses this group for guidance in measuring the corresponding group.

Trapping Controls:

When trapping is started for a Group to Nominal Group relationship bull's-eyes are drawn on each of the points in the nominal group to guide the operator to measure those points on the part. Additional user guidance can be turned on and off within the properties dialog (Figure 14-10).

Group to Nominal Group Relationship Properties

Name:

Fit Settings

Override Individual Point Tolerances

Trapping Controls

Use closest point to match new measurements

Display closest point watch window

Use view zooming with proximity

Ignore points beyond threshold

Automatic Vector Group Creation

Update a vector group with relationship

Settings for Relationships of this Type

Figure 14-10. Group to Nominal Group Properties Dialog.

- **Update a vector group with relationship**

This function turns on an auto-vector group that provides a direct visual indication of the distance between the points in each group.

- **User closest point to match new measurements**

With this function enabled the point name control in the instrument interface is automatically set to a default “g2ngrTrap-p#”. As each point is measured it is automatically compared to the points in the nominal group and takes on the same name as the closest point. An initial alignment is helpful in ensuring the correct name is assigned.

- **Display closest point watch window**

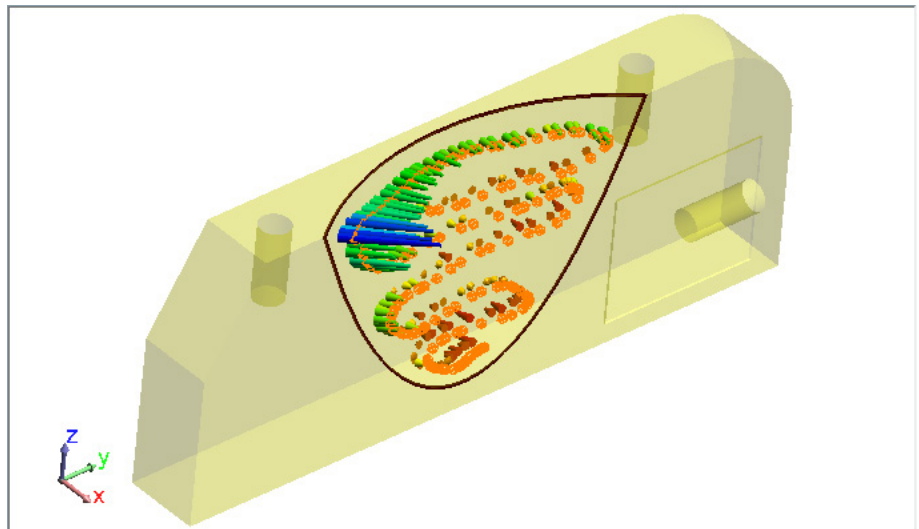
This option turns on a watch window that displays the distance between the probes current position and the closest point in the nominal group that has not yet been measured, and will update with each measurement.

Points/Groups to Surface Faces

Points to Surface Faces and Groups to Surface Faces relationships function just like Points or Groups to Objects relationship but are tied directly to CAD. By using these relationships individual CAD faces can be referenced directly where as before those faces had to be separated out as unique objects in the tree for reference.

CAD faces will also highlight during trapping for clarity (Figure 14-11).

Figure 14-11. Points to Surface Face relationship.



Vector Group to Vector Group Relationships

Built with Shim and Gap applications in mind, as well as comparisons between sequential measurements, this **Vector Group to Vector Group Relationship** (VG to VG Relationship) computes a delta vector group defining the difference between two reference vector groups.

A VG to VG relationship offers the following property controls (Figure 14-12):

Vector Group To Vector Group Settings

Name: Vector Group to Vector Group

Proximity Settings

Maximum Radial Offset: 1.0 Inches

Minimum Axial Offset: -10.0 Inches

Maximum Axial Offset: 10.0 Inches

Nominal Relative Polarity Between Vector Groups

Opposing Direction Same Direction

Gap Optimization Evaluation Settings

	Gap	Fit Weight
Minimum	0.0	10.0
Maximum	0.0	10.0
Nominal	0.0	1.0

Gap Linear Units: Inches

Gradient Steepness Factor: 50.0

Buttons: OK, Apply To Selected, Cancel

Figure 14-12. Properties Control for a Vector Group to Vector Group Relationship

Proximity Settings define a cylindrical proximity zone that is used to control which vectors in the second group are used for comparison with the first.

- **Maximum Radial Offset.** Maximum distance from the origin point of the reference vector to consider in comparisons.
- **Minimum Axial Offset.** Minimum distance from the axis of the reference vector to consider in comparisons.
- **Maximum Axial Offset.** Maximum distance from the axis of the reference vector to consider in comparisons.

Nominal Relative Polarity Between Vector Groups optional selection is used to control the mathematical comparison of the vector groups.

- **Opposing Direction.** Used for shim and Gap applications where measurements have been taken from two opposing surfaces
- **Same Direction.** Used for sequential surface measurement where the same surfaces is measured a second time and the

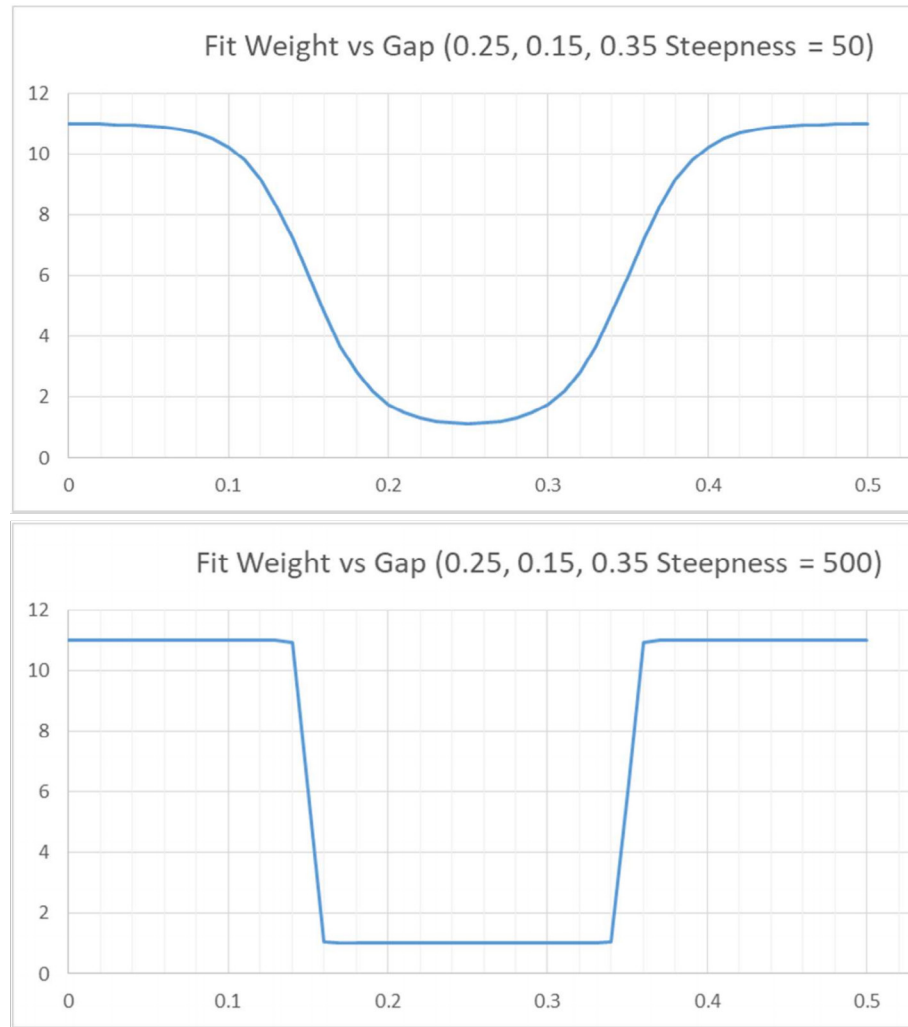
relative change in deviation is desired.

Gap Optimization Evaluation Settings are used to both control the fit setting used in a relationship optimization and the default tolerances passed to the resulting vector group for reporting purposes.

- **Minimum (Gap and Weight).** The *Gap* value is an absolute value used to define the minimum gap threshold and low vector tolerance while the *Fit Weight* is the maximum fit weight on the lower or negative side of the optimization.
- **Maximum (Gap and Weight).** The *Gap* value is an absolute value used to define the Maximum gap threshold and high vector tolerance while the *Fit Weight* is the maximum fit weight on the upper or positive side of the optimization.
- **Nominal (Gap and Weight).** The *Gap* value is an absolute value used to define the optimum or nominal gap threshold. The *Fit Weight* is the minimum fit weight used in an optimization.
- **Gradient Steepness Factor.** This value is used to control the rate of transition in Fit Weight applied when within the min or max thresholds but not at the nominal.

The **Gradient Steepness Factor** defines the how a relationship will influence the fit as it enters the desired gap threshold. Unlike a fixed constraint which is either on or off, the gradient control provides a means to control the transition rate of the fit weight as it moves from the max or min weight toward the nominal weight (see [Figure 14-13](#)).

Figure 14-13. Comparison of two example Gradient Steepness Factors



If the tolerance is very narrow, then the gradient steepness value must be set quite large (~500), providing a sharp drop off in influence. If the gap tolerance is larger then the gradient steepness can be set smaller (~50) which produces a slower transition. For more information on relationship fitting see “[Minimizing Relationships](#)” on page 444.

Basic shim and Gap application would go as follows:

Preliminary Setup:

1. Start with two nominal surfaces that are expected to mate together in some fashion.
2. Declare one of these surfaces to be the primary and define a **Points to Surface Face Relationship** to which points can be added to characterize the as-built surface relative to the nominal CAD surface. Use this relationship for a best-fit if desired and enable auto-vectors.
3. Declare the second of the two mating surfaces to be the sec-

ondary by defining a second points to surface relationship to characterize the second as-built surface. The collected points need to correspond to those points collected for the primary surface. Use this relationship for a best-fit if desired (if using a second instrument station) and enable auto-vectors.

Configuring the VG to VG Relationship

1. Create a “Vector Group to Vector Group” relationship and select the primary **Points to Surface Face** relationship auto-vectors as the primary input and then select the secondary **Points to Surface Face** relationship auto-vectors as the secondary input.
2. Open the properties for the VG to VG Relationship and set the proximity parameters as desired (Figure 14-12). The vectors in the first vector group will be used for reference and the closes (radial) corresponding vector will be found from the corresponding vector group. Also ensure that the **Opposing Direction** radio button is selected. **Same Direction** should be used when comparing points on the same surface.

Fitting with a VG to VG Relationship

VG to VG Relationships can be used in an optimization.

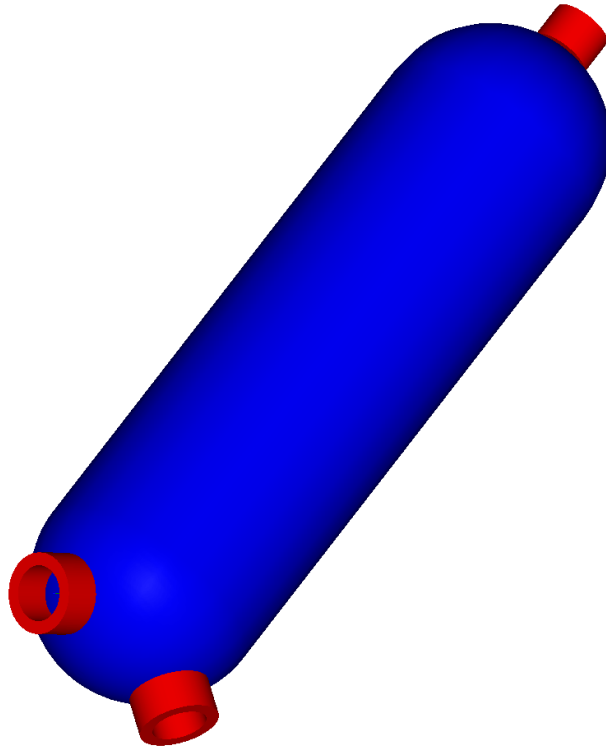
1. Set the properties for the “Vector Group to Vector Group” such that the nominal, minimum, and maximum gap values are set appropriately.
2. Enter an appropriate **Gradient Steepness Factor**.
3. Run the relationship fit by selecting *Relationship>Move Objects by Minimizing Relationships*

Pipe Relationships

In some industries, such as nuclear energy and shipbuilding, a common need is to align mating sections of pipe in order to calculate and lay off cut planes so that the pipes can be adjoined with a high-quality weld. In general, the goal is for these adjacent pipe sections to be as coaxial as possible and to share enough material overlap such that a valid cut plane can be calculated that enables a quality weld joint.

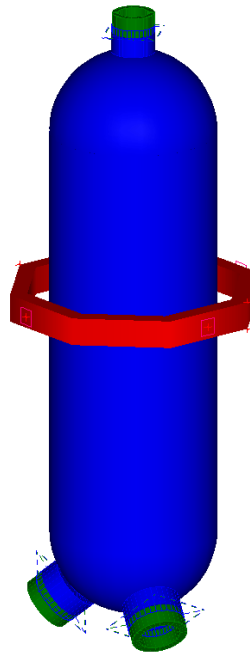
Consider the example of replacing a steam generator that has reached the end of its service life at a nuclear power plant. A nuclear steam generator typically has a number of nozzles or pipe openings—one variant design has a “hot leg” pipe for hot water, a “cold leg” pipe for cold water, and a pipe for steam (Figure 14-14).

Figure 14-14. A typical steam generator, with hot and cold legs (bottom) and a steam leg (top).



These piping interfaces are critical to the safety and performance of the nuclear power plant, and since they provide a direct barrier between the radioactive and non-radioactive sides of the plant, they must be aligned and cut with optimal precision.

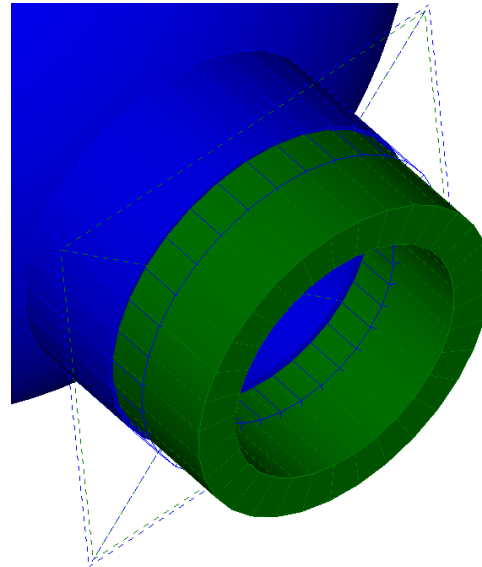
Figure 14-15. The steam generator after virtual assembly.



Since a nuclear reactor must be shut down in order to replace the steam generator—and since downtime at a nuclear power plant is enormously costly—anything that minimizes this downtime is extremely beneficial. Historically, cut planes were calculated manually

and refined with the steam generator in its final alignment. However, this proved to be a time-intensive and therefore expensive process. Instead, the steam generator can be digitally assembled and resulting pipe cuts can be calculated before the steam generator is in its final location (Figure 14-16). As a result, the pipes can be cut prior to installation—pipe relationships play a critical role in this process.

Figure 14-16. A cut plane calculated by a pipe relationship.



Using Pipe Relationships

The general process of using a pipe relationship is as follows:

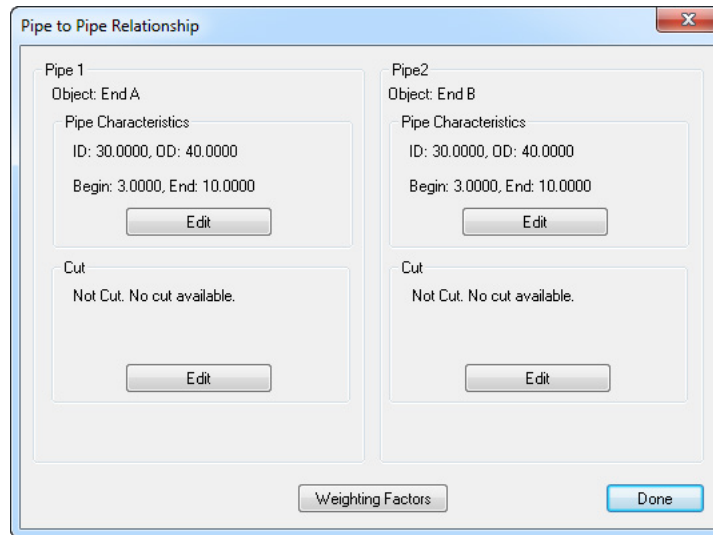
1. Define the pipe relationship between mating pipe/fittings.
2. Align the pipes using traditional relationships, pipe relationships, or a combination.
3. If the pipe relationship can find a valid cut plane, the proposed cut can be made. A frame can also be constructed on this cut for layoff purposes.
4. After physically cutting one end of the pipe, the actual cut plane can be fed into the solution, re-solving for a tweaked cut plane on the other end.
5. This tweaked cut plane can then be used to cut the mating piece.

Defining Pipe Relationships

Pipe Relationships can be defined between two objects that have positional and directional information. Typically, this means lines, cylinders, or planes. The particular type of object does not matter—as long as the object properly describes the pipe's position and orientation. When a pipe relationship is created, a tube representing the al-

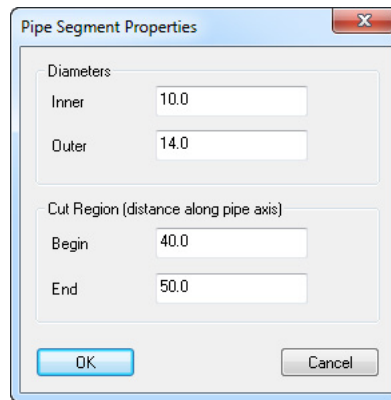
lowable cut region and ID/OD appears in the graphical view, and the *Pipe to Pipe Relationship* dialog is displayed (Figure 14-17).

Figure 14-17. The *Pipe to Pipe Relationship* dialog.



The first step is to define the properties of each pipe segment (Figure 14-18):

Figure 14-18. Properties of a pipe segment in a pipe relationship.



- **Inner.** The inside diameter of the pipe.
- **Outer.** The outside diameter of the pipe.
- **Begin.** The start of the permissible cut region, as a distance along the pipe axis from its origin. This value should typically be less than the End value.
- **End.** The end of the permissible cut region, as a distance along the pipe axis from its origin. This value should typically be greater than the Begin value.

A pipe relationship will supply several forcing functions to an optimization:

- It will try to make the pipe segments coaxial.
- It will try to make the pipe ends parallel.

- It will try to overlap the cut regions evenly.

After defining the characteristics of the pipe segments, the weighting factors for the relationship may be modified, if desired. Typically, the default values (Figure 14-19) are sufficient:

Figure 14-19. Weighting Factors for a pipe relationship.

- **Overall Weight.** The weight of the pipe relationship as a whole, which determines the influence of the pipe relationship in the overall optimization scheme.
- **Axis Offset.** The importance of minimizing the mutual perpendicular midpoint between the two pipe segments (i.e. how “coaxial” the two pipe segments are). Larger values attempt to bring the segments closer together.
- **Axis Alignment.** The importance of minimizing the angle between the two pipe segments. Larger values will force the pipe segments to be more parallel.
- **Center Pull.** The importance of overlapping the center of each pipe segment’s cut region (along the axis direction). Larger values assign more importance.
- **Weight.** The importance of having sufficient material for the weld. A value of zero implies that the solution does not care if there is sufficient material for the weld.
- **Static Offset.** A value that controls “how bad” the solution gets when there is insufficient material for the weld. Higher values assign a larger penalty to having insufficient material, effectively ensuring that material requirement is more crucial in the eyes of the solution.
- **Constrain Cut to Region at OD.** If checked, ensures that the cut angle at the outer diameter does not cause the cut plane to exceed the defined region. If unchecked, this restriction is ignored.

- **Constrain Cut so ID/OD overlap.** If enabled, the inner diameter/outer diameter values for the pipes are used to ensure that there is sufficient material based on the angle of alignment.

To Solve a Pipe Relationship:

1. First take a moment to think about the assembly. If there is more than one pipe segment that will be assembled (that is, if there will be more than one independently moving object in the optimization), then each moving object (pipe segment) should be placed into its own collection. This includes the objects defining the pipe endpoint, the pipe surface/cylinder, and anything else you want to rigidly move with that pipe segment. In this situation, you must use **Relationships > Move Collections by Minimizing Relationships** in step 7 below.
2. Create a primitive SA object with directionality (circles, lines, cylinders, or planes—but circles are preferred) to represent the end of each pipe opening in the assembly. Each object’s normal direction *must* face the mating pipe segment as depicted in [Figure 14-20](#).

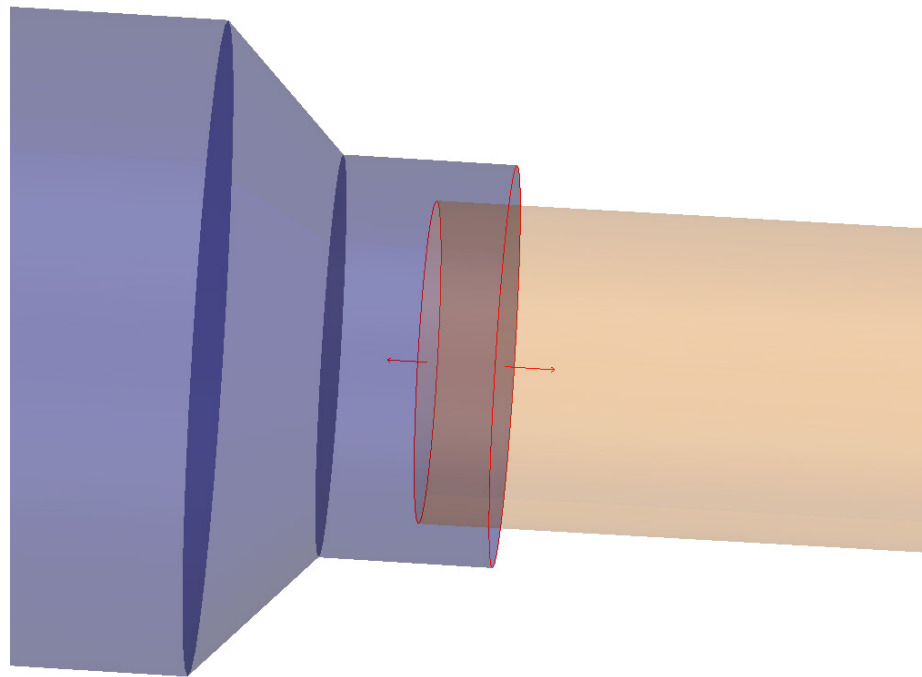


Figure 14-20. The circles defining each pipe opening have their normals pointing toward the mating pipe segment.

3. From the menu, select **Relationships > Pipe Fitting**.
4. Select the object (i.e. circle) describing the first pipe segment, then the second.
5. In the **Pipe to Pipe Relationship** dialog ([Figure 14-17](#)), click the **Edit** button for each pipe segment and define the ID, OD, and begin/end values for the allowable cut region ([Figure 14-18](#)).
6. If any pipe ends are already cut, create a frame representing

Warning: If the ID and OD values of mating pipe segments match, a valid cut plane will typically not be found unless you uncheck the *Constrain Cut so ID/OD Overlap* option.

that pipe end (usually **Construct > Frame > On an Object** is appropriate) and then click the **Force CUT to Frame** button.

7. Create any other conventional relationships governing the alignment, if applicable.
8. Solve the relationship using **Relationships > Move Objects by Minimizing Relationships** or **> Move Collections by Minimizing Relationships**, as applicable. Enable/disable and adjust the weights on all relationships as desired.

Tip: In general, pipe relationships are best solved by clicking the **Direct Search Optimization** button instead of the **Run Optimization** button.

Working With the Cut

If the pipe relationship can find a valid cut plane, it will display a dashed plane indicating the location and orientation of the cut plane.

Typically, the next step is to create a frame on the cut, which can then be laid off on the real pipe for cutting. This involves making the proposed cut, then constructing a frame on the cut plane. The XY plane of this frame now defines the perfect cut location on the pipe. Watch windows can be used to lay this cut off on the real pipe.

In some applications, it may be desirable to simply calculate the cuts and lay them all off at once, as described above. This would be a faster but less stringent approach. A more rigorous approach would involve laying off the cuts on one segment, performing the cuts on the actual pipe, then measuring the actual cut plane.

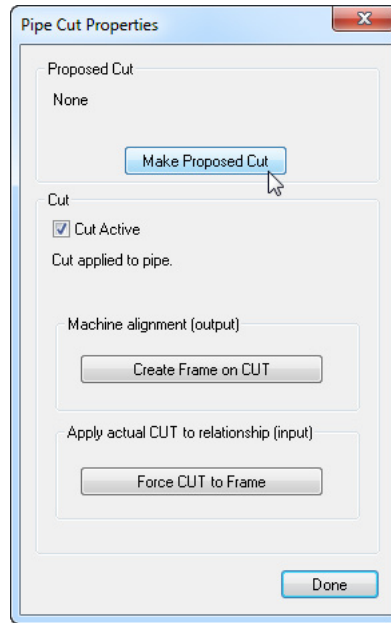
Once the actual cut plane is measured, a coordinate frame can be constructed that represents the location of the as-built cut. This as-built cut location can then be fed back into the pipe relationship, and the as-built cut can be used to tweak the alignment once more. After the alignment is further tweaked, a new tweaked cut plane can be laid off on the other pipe segment.

This process can be iterated, as desired, for each pair of pipe segments.

To Lay Off Cut Planes:

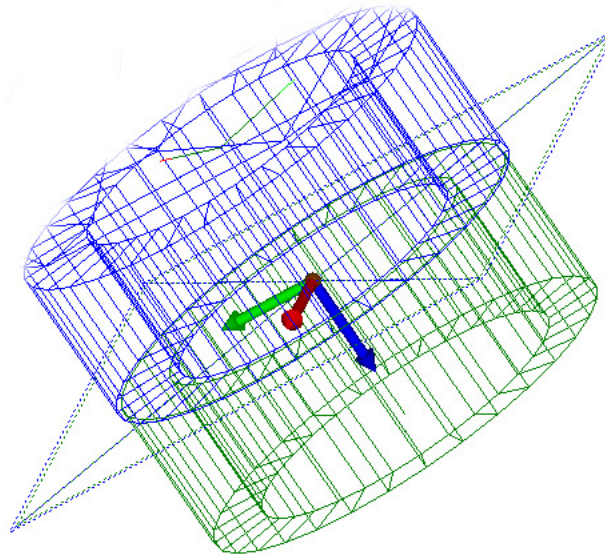
1. Once the relationship has been solved (see [“To Solve a Pipe Relationship:”](#) on page 440), double-click the pipe relationship in the tree. In the **Cut** section, ensure that “Valid Cut Available” is indicated.
2. Click the **Edit** button in the **Cut** section for one of the pipe segments. In the **Pipe Cut Properties** dialog, click the **Make Proposed Cut** button.
3. With the cut applied, click the **Create Frame on CUT** button to create a coordinate frame whose XY plane defines the cut plane.

Figure 14-21. The *Pipe Cut Properties* dialog.



4. Use a Watch Window to lay the cut plane onto the actual pipe.

Figure 14-22. A cut frame constructed on the cut plane.



The above steps can be repeated for the other pipe segment, if desired. However, if applying cuts sequentially, the process contains a few more steps.

Applying Cuts Sequentially

1. After cutting the first pipe segment, measure the actual cut and construct a frame on the resulting cut plane. Ensure that this data is aligned appropriately in the SA file.
2. Double-click the pipe relationship, and click the **Edit** button for the appropriate pipe cut.
3. Click the **Force CUT to Frame** button, and select the appropriate

- frame. This will force the cut to sit on the provided frame.
4. Re-solve the optimization. Based on the as-built cut location, you should see the solution get tweaked to account for the changes.
 5. Based on the results, make the cut on the other pipe segment using the steps in “[To Lay Off Cut Planes:](#)” on page 441.

Relationships with built in Alignments

Several relationships have built in alignment that can be performed at any time by using the **Execute Alignment** right-click option. If only a single instrument is associated with the measured group no additional prompting is necessary. However, if more than one instrument has measured points in the group an additional prompt will be displayed asking for which instruments to move.

Relationships with built in alignments include the following:

Best-Fit Applications

Group to group relationships build a link between the commonly named points in the two groups. This provides the necessary information to perform a best fit operation with the associated groups.

- **Group to Nominal Group.** Choosing **Execute Alignment** will bring up a best-fit dialog and allow you to align the measured group to the nominal group.
- **Group to Group.** A basic group to group relationship will also an align option and will align Group 2 to Group 1.

Frame to Frame Applications

Frame to Frame relationships also provide a defined alignment which, when executed will place one frame on top of the other.

- **Frame Wizard.** Frame wizard builds a frame from the associated points and or geometry. Choosing **Execute Alignment** will place the measured frame on the nominal frame.
- **Frame to Frame.** A basic frame to frame relationship can also be used for alignment. If you only have a single instrument in the job it will transform that instrument from the corresponding frame to the reference frame. With more than 1 instrument in the job it will prompt for selection of what instrument to move.

Minimizing Relationships

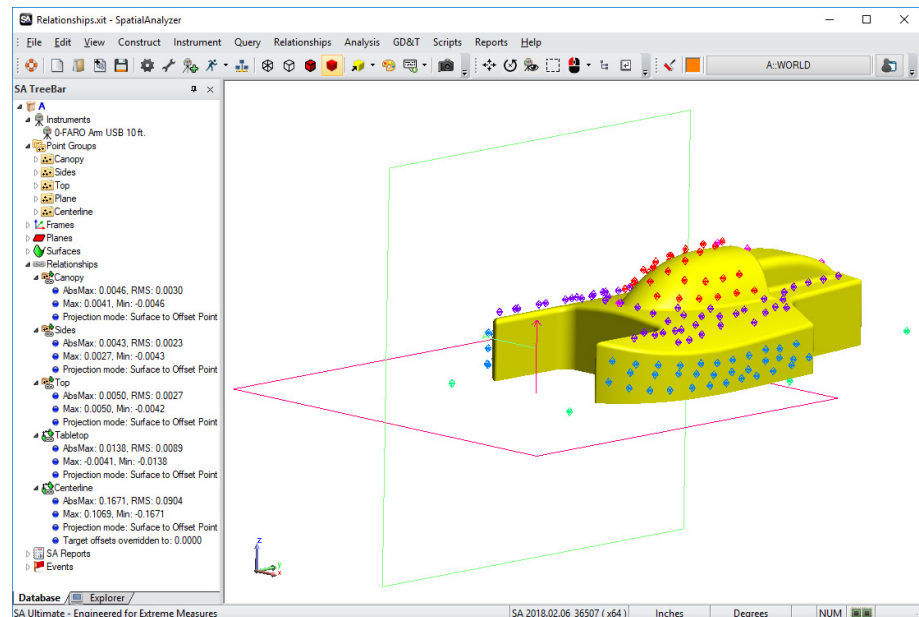
Overview

In most metrology applications the first step in analysis is to orient measured points to critical design features. This then provides the ability to perform an evaluation of the measured features. SpatialAnalyzer is able to help you accomplish that task by letting an optimization routine use relationships to drive a fitting process. This is a far more powerful process than a simple best fit because it dynamically establishes the best possible fit to potentially complex and curvy parts. The net effect is the distances defined between objects in each relationship are minimized.

By creating relationships between the measured points and the part's critical features you can use these dynamic features to find orientations that yield the best performing part. Since SA's relationships can be made between any object types you're able to define a comprehensive set of part features and then optimize their orientation.

Relationship optimizations are extremely powerful tools that are only fully available with a *SA Ultimate* license. A relationship fits can be performed with a single relationship using an *SA Professional* license but its real power comes from using multiple relationships at one time. This allows a user to balance the fit and optimize the alignment based on the tolerances placed on each feature.

Figure 14-23. Relationships set up in the tree.



Performing a Relationship Fit

Relationship Minimization is a powerful way to transform objects and instruments into alignment with design features. Relationship fitting can be performed as follows:

- **Move Objects by Minimizing Relationships.** Provides a means to select objects and or instruments and move them as a unit to alignment through an optimization process.
- **Sequential Alignments.** This utility offers the ability to align using one relationship after another in sequence controlling different degrees of freedom for each step.
- **Move Collections by Minimized Relationships.** Offers an option to move collections as separate units, optimizing each independently but all at one time. It is very helpful in large complex applications.

Moving Objects by Minimizing Relationships

In order to align using relationships you will want to do the following:

1. Build relationships or ensure that the relationships you would like to use are in the working collection.
2. Navigate to the Alignments tab and select **Relationship Fit** (available under Relationships menu in the classic menus).
3. Select the objects you would like to move. Note...

If you intend to move one or more instruments you can just press **Enter** without selecting anything. When you move an instrument the points measured by it and any associated objects (such as measured clouds, frames, or other geometry) will move with it automatically.

4. Select the instruments to move.

The *Minimize Relationships* dialog provides full control over the optimization process. It offers a list of available relationships in the active collection and provides direct access to their weighting and fit constraints.

The Minimize Relationships dialog should be considered in 3 sections from top down (Figure 14-24):

1. Weight and Degree of Freedom Controls across the top, which establish the overall fit conditions.
2. The list of individual relationships, which also provide direct access to the relationships fit settings.
3. The user controls and computation statistics are available be-

low.

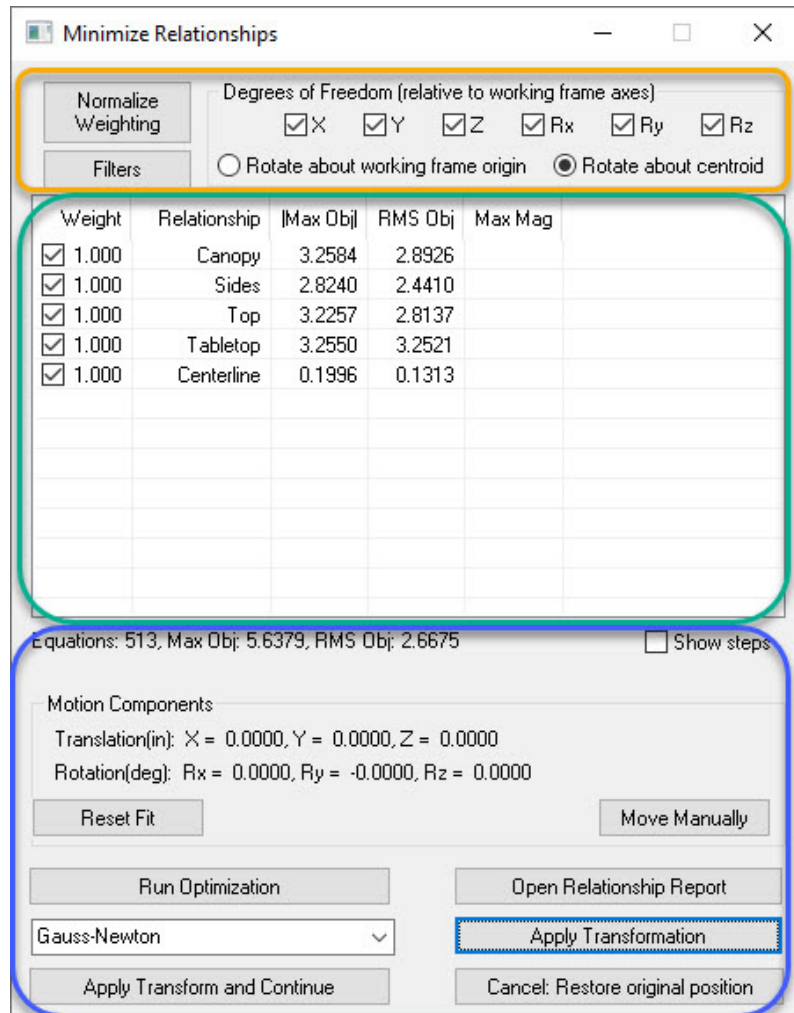
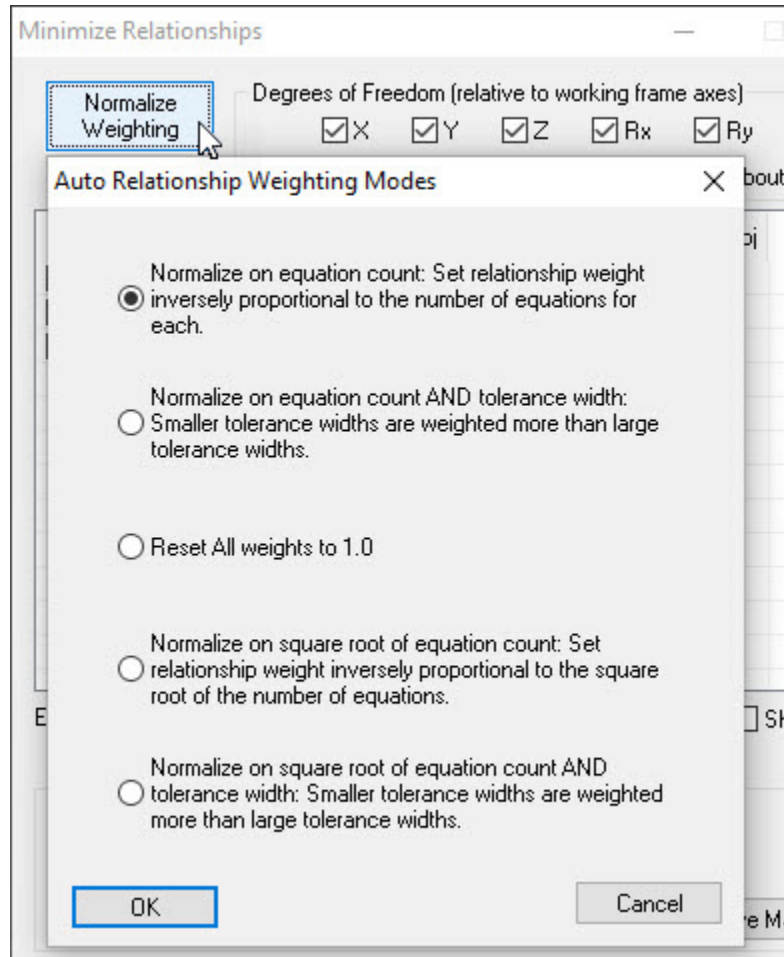


Figure 14-24. The Minimize Relationships dialog.

Normalize Weighting

The **Normalize Weighting** control provides a means to equalize the influence of relationships on the fit (Figure 14-25).

Figure 14-25. Options available to automatically adjust the relative relationship weights.



By default each relationship is weighted the same, an equal value of one. However these values can be adjusted to increase or decrease the relative “pull” within the optimization process for each relationship. Weighting can be controlled by 2 factors:

- **Equation Count.** A point to surface relationship is a single vector or equation, whereas a point to point relationship applies in X, Y, and Z and is in effect 3 equations. You can normalize the weighting between relationships such that the number of equations are equalized.
- **Tolerance Width.** This allows you to adjust the fit providing greater weight to features with tighter tolerances. Taking advantage of available tolerances may allow a part to pass that would fail when all features are handled equally.

Weights can be normalized based upon the following options:

- **Normalize on Equation Count.** This option sets weighting inversely proportional to the number of equations, equalizing the pull from a relationship with more points to that of a relationship with less.

- **Normalize on Equation Count and Tolerance.** This option takes into account both equation count and tolerance width and sets the weight accordingly.
- **Rest All Weights to 1.0.** This option allows the weights to be rest.
- **Normalize on Square Root of Equation Count.** The inverse of the \sqrt{N} where N = number of equations for relationship fit computation. This is very similar to the first option but uses the square root of N . There is some argument that this is a better weighting solution for balancing the pull from two relationships.
- **Normalize on Square Root of Equation Count and Tolerance Width.** Same as option but uses the square root of the equation count.

Filters

The **Filters** button offers access to options for hiding relationships from the display within the fit dialog (Figure 14-26).

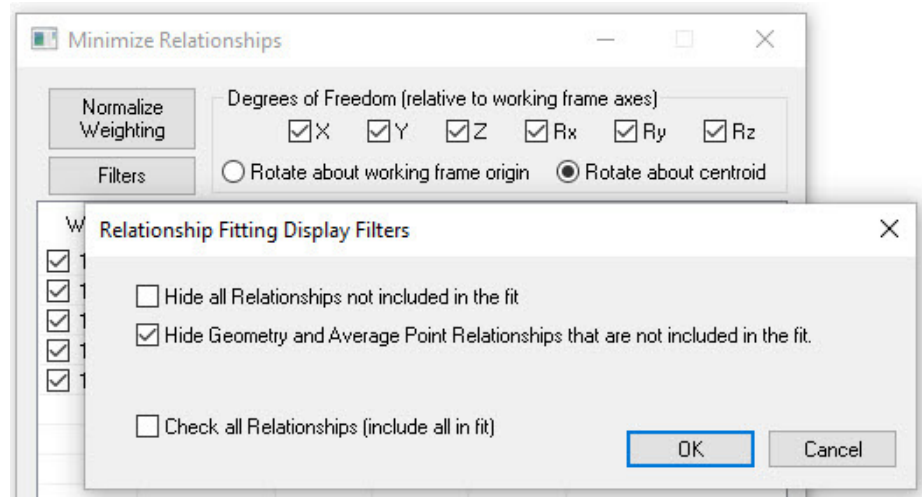


Figure 14-26. Display filters control used to hide relationships from the optimization dialog display.

These controls can be very helpful in large job files by allowing you to hide relationships not included in the fit.

Additionally, the individual criteria in any geometry relationship, frame to frame or average point relationship can be included in a relationship optimization. The display of these relationships is hidden by default because only specific criteria should be used.

In order to include criteria such as the *Avg Distance Between* value for a plane to its nominal, go to the relationship's properties, select the criteria of interest and enable the option to **Include in Optimization**.

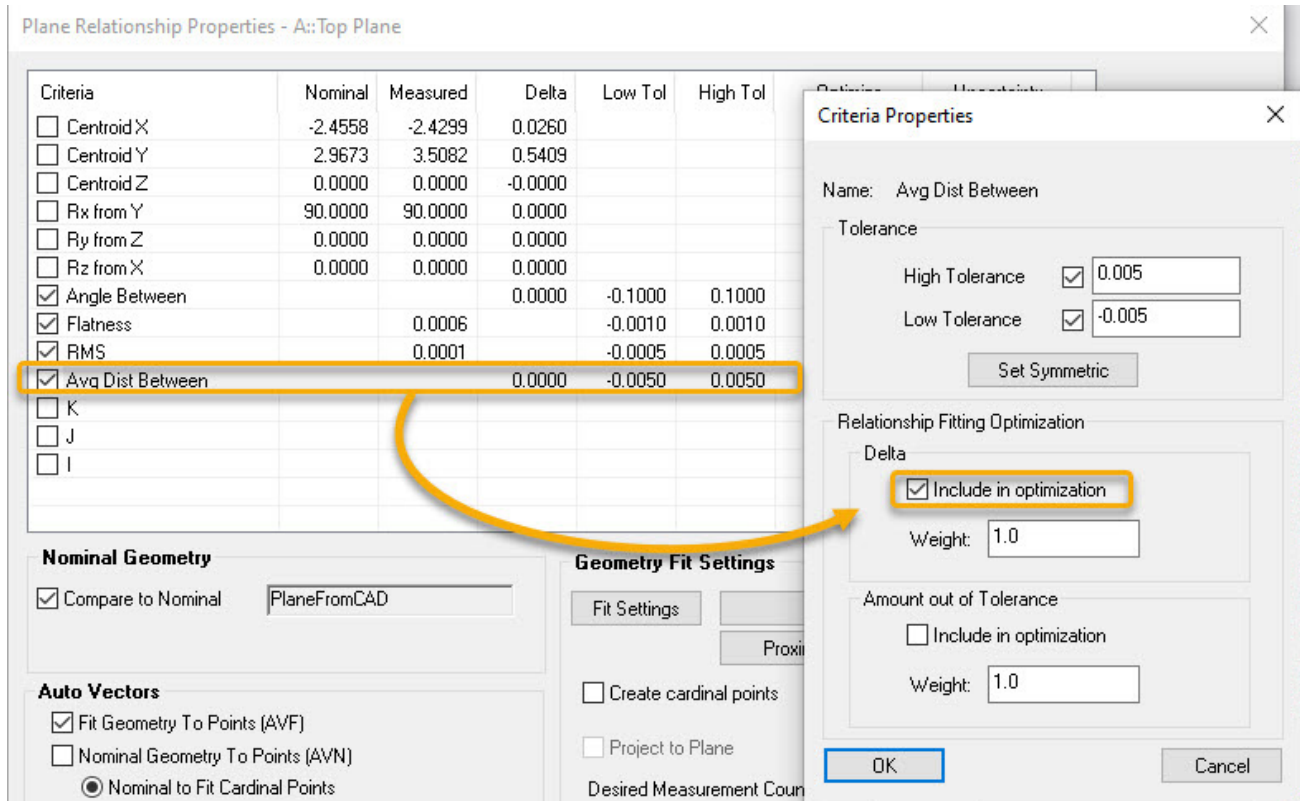


Figure 14-27. Enabling criteria from a GR-Plane to include in an optimization

Degree of Freedom Controls

The **Degree of Freedom (relative to the working frame axes)** controls across the top of the Minimize Relationships Dialog offer the ability to restrict motion in one axis or another with respect to the working frame, controlling the motion that is allowed.

Selecting X, Y or Z allow translation along these axes while unchecking the boxes prevents the solver from moving in that direction.

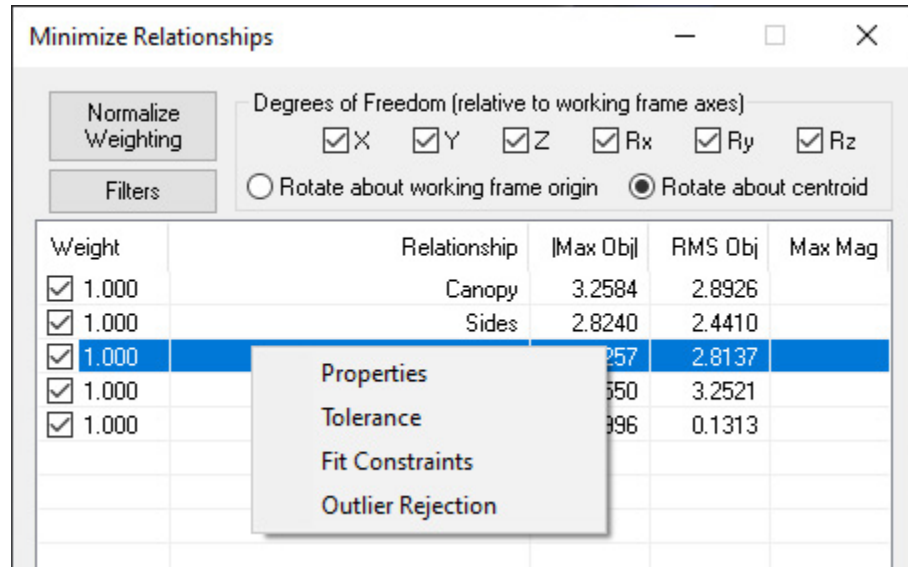
The Rx, Ry, Rz control rotation in the same manner. Un-checking a box prevents rotation about that axis.

The rotation centroid controls can be helpful if the working frame is very far from the part. In such a case the optimization process can be slowed due to inaccuracies caused by this rotation point. Selecting **Rotate about centroid** allows the fit process to rotate about an adjusted local axis but the Degree of Freedom reference controls are unaffected by this selection.

Relationship List

The relationship list within the Minimize Relationships dialog provides direct access to the individual relationship's settings (Figure 14-28).

Figure 14-28. Right-click access to the properties of a selected relationship.



The individual relationships weights can also be accessed in the relationship list. While the **Normalize Weighting** button can be used to normalize the weight values for all, they can be edited individually by just clicking twice slowly on the number. The edit field will open much like an excel entry field.

When an optimization is run the statistics for each relationship are updated dynamically and display the updated results for the following.

- **|Max Obj|.** Returns the absolute value for the maximum deviation within each relationship included in the fit.
- **|RMS Obj|.** Returns the RMS for each relationship included in the fit.

Reported Values and Starting Position

The current statistics for the relationship optimization are listed just below the relationship list window. These include:

- **Equations.** This line include the total number of equations included in the optimization. This will be roughly equal to the number of points included in the fit for simple optimizations but will adjust based on the inputs from other relationship types such as frame to frame or geometry relationship criteria.
- **Max Obj.** Returns the total absolute value for the maximum deviation within the fit.
- **RMS Obj.** Returns the total RMS (root mean squared deviation) for the fit.
- **Motion Components.** This transform reports the system motion from its origin with respect to the working frame. This origin point is either its starting point when the process began

or the new starting point after selecting Apply Transform and Continue, which will update the origin.

Move Manually

The **Move Manually** button offers an option to directly adjust the current position of the objects included in the fit through the use of a transform control. This can be helpful to shift objects in a direction to change the starting position of an optimization.

Reset Fit

The **Reset Fit** button provides an easy method to return the objects included in the optimization to their starting location without closing the dialog, allowing further analysis.

Run Optimization

The **Run Optimization** button begins the optimization process and the fit calculation based on your current selections. The default optimization method (Gauss-Newton) performs a version of the steepest-descent algorithm on the system. This means it assembles a matrix that is δ (unknowns) \times N (number of points), inverts it and determines move directions. Though quick for reasonable data sets, it can take a long time when using a large set of points depending on the amount of RAM on the computer.

The optimization continues until the stopping tolerance is reached, which is the amount that the objective function (RMS error) must repeat by in order to indicate a minima has been reached. This tolerance can be controlled from the Units tab of the User Options.

Optimization Methods

A selection of optimization methods are available to ensure that you can use the best solution possible for your particular application.

Gauss-Newton

In SA version 2023.2 additional optimization methods were added and the default **Run Optimization** option was more clearly identified as the Gauss-Newton method which is a well known solution for solving non-linear least squares problems. This solution is typically the fastest to solve and produces an excellent solution in most scenarios where the solution space is not too complex.

Levenberg-Marquardt

This optimization method helps to avoid solution problems that can arise in difficult fit scenarios. It interpolates between the default

Gauss-Newton algorithm and the method of gradient descent to provide a more robust solution without greatly increasing the solution time.

Gauss-Newton /w Gradient Search

This selection combines a Gauss-Newton solution with an exit strategy. After a fit is completed a gradient search is triggered as well which can also bypass sticking points such as local minima in an ill-conditioned solution space. The gradient search can be slow to solve, taking many iterations but as a safeguard it can ensure a good solution is found.

Direct Search

This optimization provides a powerful method for searching the solution space. It is also required when using the toleranced best-fitting options (in the advanced area) since they introduce non-linearities into the solution space. This method basically steps in a direction, evaluates the fit, then steps in another direction, etc. There is a lot of advanced logic built into the search, but at the core, it is a searching method. The advantage is that it does not have to assemble a large partials matrix and invert it. This means that it can effectively deal with huge point sets. The disadvantage is that since it is trying to exhaustively search, it can sometimes take a long time to complete. At any time during the fit, however, you can cancel the operation and use the current location. To use this mode, you set step sizes. They are set as a ratio of the position and angular perturbation values in the SA units options page. The step size starts at the starting value then decays when it detects no improvement in any direction. Once it decays to the Ending step size, a minimum is reached.

Finalizing the Optimization

Open Relationship Report

Offers an additional method to verify the fit results through directly opening the Relationship Report prior to accepting the results.

Apply Transform

The **Apply Transform** button accepts the current transform and closes the Minimization Dialog.

Apply Transform and Continue

The optimization is performed with degrees of freedom defined relative to the working frame. Rotations are applied with respect to this reference frame. There are cases where one it would be helpful to per-

form an initial optimization and then a subsequent optimization using a different set of degrees of freedom defined by a different frame. The **Apply Transform and Continue** button provides the ability to do this.

Cancel: Restore original position

This button offers an option to restore the original position of the objects included in the fit and closes the Minimize Relationships Dialog.

Sequential Alignment

The Sequential Alignment available under **Analysis > Best-Fit Transformation > Sequential Alignment** is designed to provide a streamlined means to apply one relationship optimization after another in a specific sequence. Individual degrees of freedom can be set with each relationship or group of relationships and are then used when that particular relationship is optimized (see [Figure 14-29](#)). This functions much like using the **Apply and Continue** button in the *Move Objects by Minimizing Relationships* option.

In addition, the reporting frame for that relationship can also be used to define different reference frames for the applied degree's of freedom.

This tool is ideal for GD&T style alignments, providing a way to align to curvy surfaces in a sequential way with defined degrees of freedom.

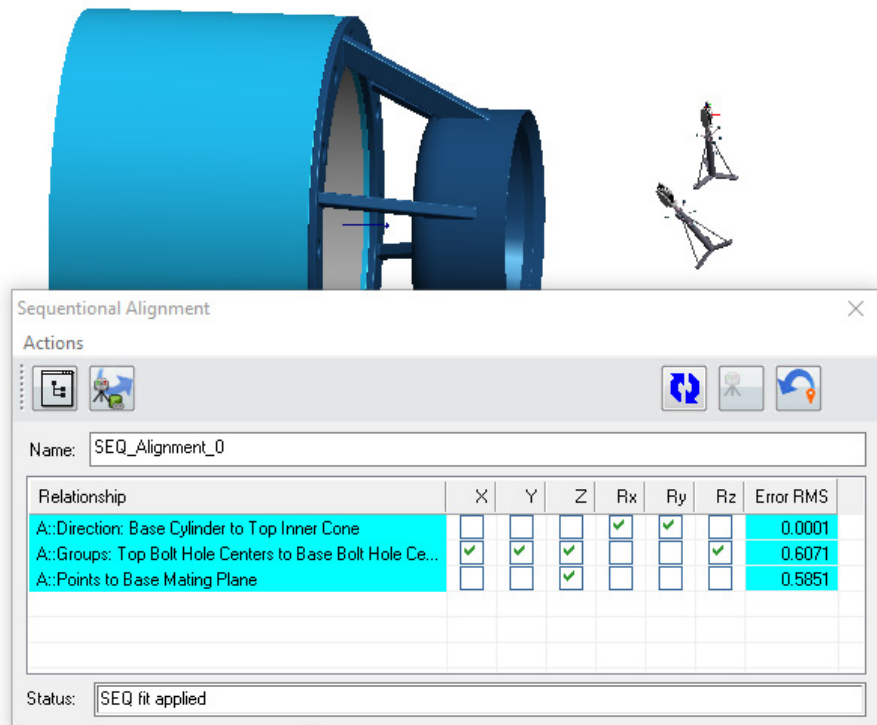


Figure 14-29. Sequential Alignment dialog

To perform an alignment do the following:

1. Press the **Select Existing Relationships** button and select the relationships you wish to use in the optimization process.
2. Press the **Select Objects and Instruments to Move** button and choose what you wish to move. Typically instrument selection is all that is required. Measured points and associated geometry will be moved with the instrument automatically.
3. Order the relationships in the relationship dialog in the order you wish to apply them and select the degrees of freedom you wish to influence through the relationship.

The Sequential Alignment process will use the Working Frame to define the degrees of freedom so a relevant frame should be built and set as working. If you wish to align using degrees of freedom referenced by a different frame you can set a **Reporting Frame** in the relationship right-click properties.

4. Once content that the alignment process has been defined correctly press the **Run (Re-Run) the Alignment Fit** button to compute a fit. This will compute the resulting Error RMS.
5. Finally when the numbers look good you can apply the fit by selecting the **Apply Alignment Fit** button. This will apply the alignment which can then be viewed in the graphics. You can also use the Undo Applied Alignment button to put the instrument back in its starting location.

Sequential alignments, along with RPS alignments are saved in the tree and can be re-opened and recomputed at any time.

Move Collections by Minimizing Relationships

Unlike the Move Objects by Minimizing Relationships option which moves a selection of items as a single unit, the Move Collections option allows you to use collections to define independently moving units and optimize the position of each independently at the same time. .

When you open the fit dialog you will see that each collection is listed at the top and has a unique degree of freedom control ([Figure 14-30](#)).

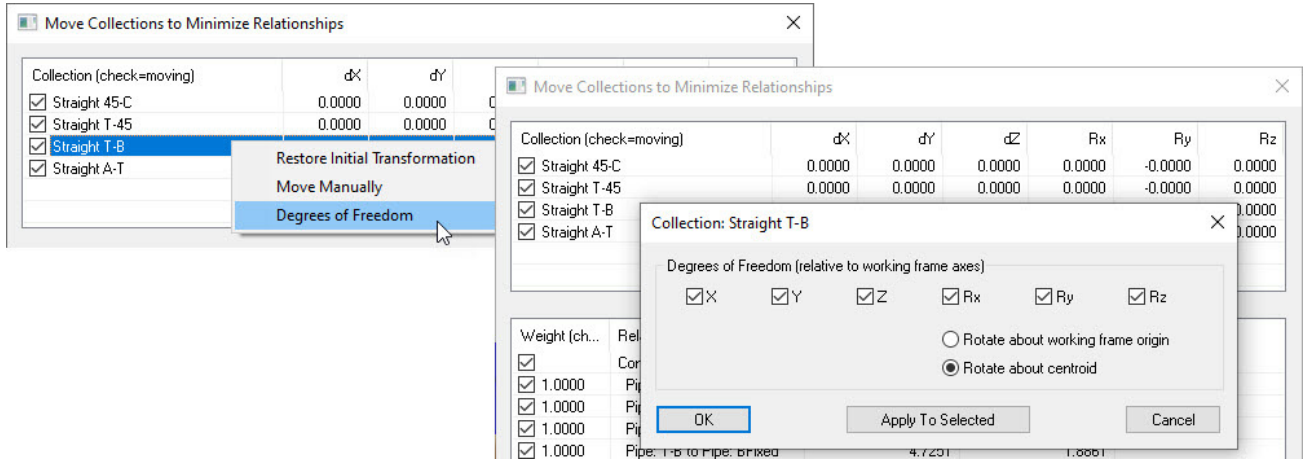
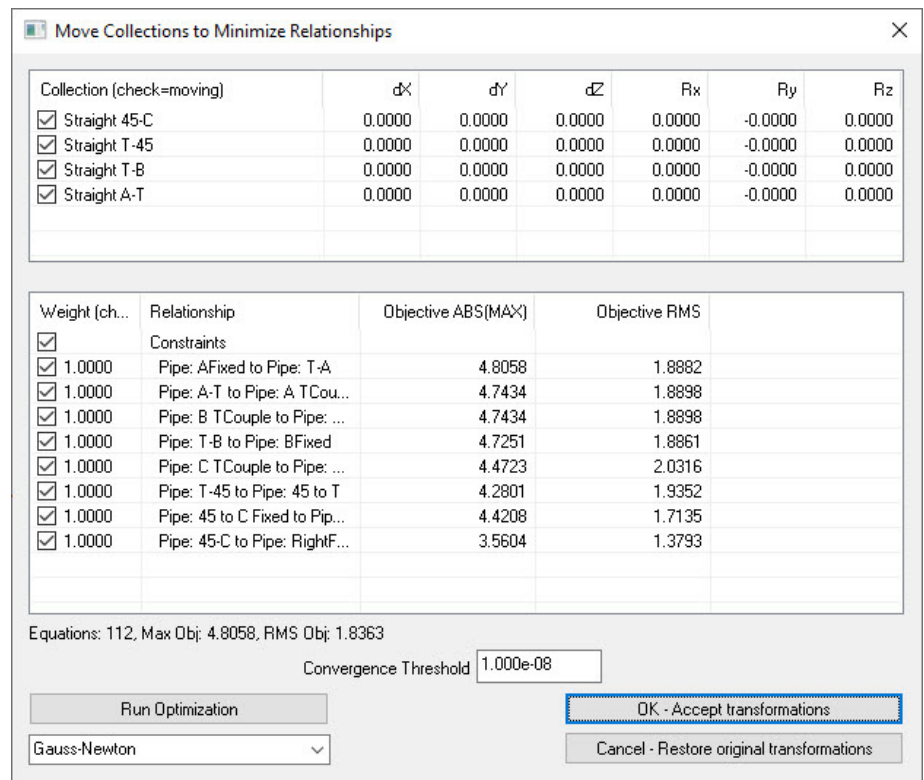


Figure 14-30. Individual Degree of Freedom Controls are available for each collection, each of which move independently.

The lower half of the dialog is much the same as the Move Objects by Minimizing Relationships dialog in that it offers a list of relationships to use and controls of the weights to apply to each relationship (Figure 14-31).

Figure 14-31. The full Move Collections by Minimizing Relationships Dialog.



The relationship optimization process is also the same, offering all 4 optimization strategies.

An independent **Convergence Threshold** is available to help control the speed and exit strategy for the optimization process.

This Page Intentionally Left Blank.

Geometry Fitting & GR-Features

CHAPTER

15

Geometry fitting using point measurements is the foundation of metrology.

GR-Features are relationships that bring geometry to life. They link measured geometry to nominal definitions and make features dynamic. They are the path to template construction and feature inspection within SA.

Geometry Fitting

Fitting geometry to measured points is a fundamental and critical aspect of almost any metrology analysis task. SA's has the capability to fit the following geometric shapes to any set of measured points or cloud points:

- Planes
- Circles
- Spheres
- Cylinders
- Lines
- Paraboloids
- Slots
- Cones
- Ellipses
- Torus



Note: When fitting slots, there are a few rules to follow. If measuring 5 points, obtain 3 in the middle (2 one side, 1 on the other other), and one on each end. If measuring 6 points, measure an extra point in the middle. For more than 6 points, distribute your points evenly around the slot.

Fitting geometry in SA is completely interactive. You can see and interact with the results of your fit settings in real-time as you modify them. You can interact with the graphical view, re-select points, change geometry types, and reverse offset and geometry directions, all while watching your geometry update in the graphical view.

Geometry can be fit to any number of measured points, constructed points, or cloud points--and it's fast. For example, you can fit a sphere to 640,000 cloud points in about 12 seconds. Unlike some metrology software, when you select large numbers of points the geometry is fit to all selected points rather than a subsample.

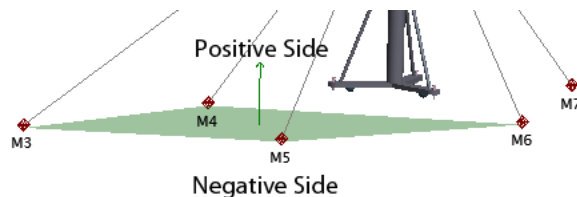
You can easily construct interactive vector groups while tweaking settings to visually see how they affect the fit. Easily create cardinal points, offset points, or graphs of fit deviations with a single button click.

Another powerful feature is the ability to project points to a plane before fitting the geometry. This allows you to measure several off-plane features and project them to the same planar surface.

About Planar Offset Methods

Planar offset methods determine what is considered the “positive” side of resulting geometry. The positive side of planar geometry (planes, circles, and ellipses) is the side toward which the normal faces, and in metrology it is typically thought of as the “outside” of the part or the side of the surface that was measured (Figure 15-1).

Figure 15-1. Geometry has a positive and negative side.

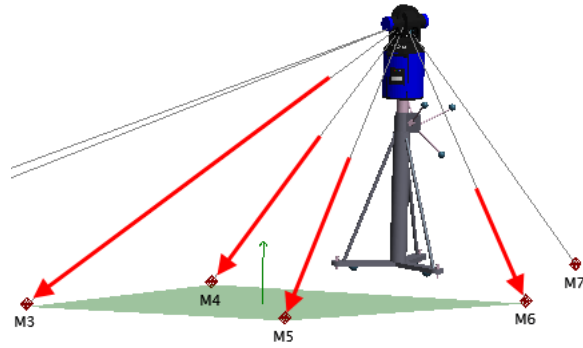


When fitting geometry to points, there are three offset methods that can be used to determine the positive side of geometry: Probing Direction, Working Origin Positive, and Right-Hand Rule.

Probing Direction

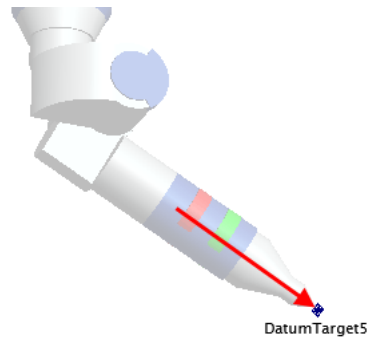
This is the direction from which the point was measured (Figure 15-2). For a line-of-sight instrument such as a laser tracker, total station, or laser radar, probing direction is the line-of-sight itself, as depicted by the red arrows below.

Figure 15-2. Probing directions.



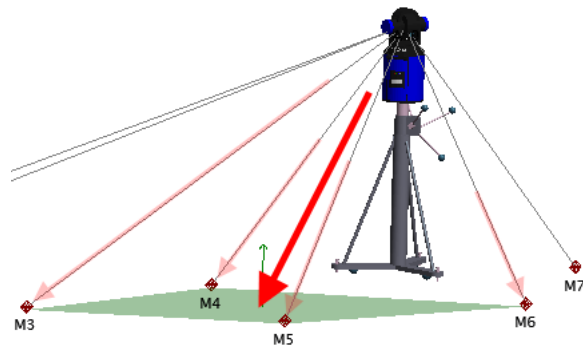
For non-line-of-sight devices, the probe direction is the vector of the device on which the probe is attached, as depicted by the red arrow in Figure 15-3.

Figure 15-3. Probing direction for a portable arm.



For the purposes of determining the positive side of geometry, the overall probing direction is the average direction of the probing directions for all measured points in the fit (Figure 15-4). Put another way, it is simply the average of all of the individual probing directions. The figure at right depicts the average probing direction for the plane fit from the laser tracker depicted earlier.

Figure 15-4. Average probing direction.



Note: Constructed points do not have probing directions. If no measured points are included in a fit, SA will fall back to using the Working Origin Positive method (described below). If at least one measured point is included in the fit, then the probing direction is used.

When probing direction is selected as the planar offset method, the positive side of the resulting geometry points generally in the opposite direction as the average probing direction. Put another way, the average probing direction points in the same direction as the negative direction of the geometry. For line-of-sight instruments, the result is that the positive side of the geometry points back toward the instrument, which is usually the desired result. This can be seen in the

previous figure: The thin green arrow depicting the positive side of the plane points back generally in the opposite direction as the average probing direction.

The advantage of this method is that for most situations, it describes the resulting geometry correctly--that is, the "outside" of the geometry is the side from which the measurement was approached.

Working Origin Positive

With this method, the geometry is defined such that the origin of the working frame lies on the positive side of the resulting geometry, as depicted in Figure 15-5.

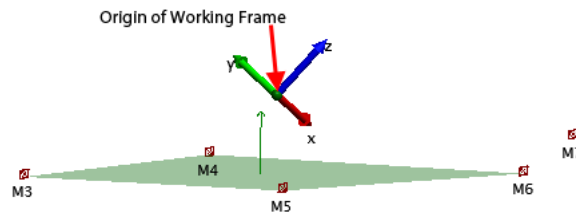


Figure 15-5. Working origin positive.

This is often used as the planar offset method in situations in which probing directions do not exist (for example, when using constructed or imported points).

Right-Hand Rule

This method allows you to define the direction of the resulting geometry through selection order--it uses the traditional "Right-Hand Rule" standard. If you curl the fingers of your right hand in the direction from the first selected point, to the second, then to the third, your thumb will define the positive direction. When using this method, the first three points in the fit should be individually clicked to define the selection order--then the remaining points can be selected as a whole using the marquee selection tool.

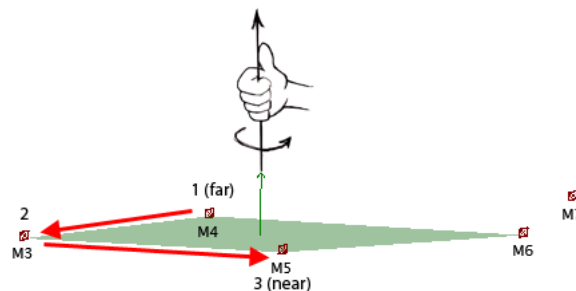


Figure 15-6. Defining offset direction using the right-hand rule.

In Figure 15-6, point M4 was selected first, followed by M3, then M5. M6 was selected after the first three points. Because of the orientation of the first three sequentially selected points, the resulting geometry points up in this image, as dictated by the Right-Hand Rule.

About Measured Side Settings

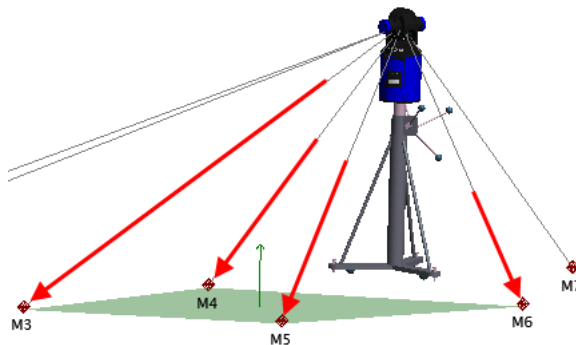
In order to properly offset geometry, SA must know on which side a feature was measured. For planar geometry (planes, circles, and ellipses), the planar offset direction has one of three options: Above Plane, Probe Center, or Below Plane.

The radial analogs to these settings, used in geometry such as circles, ellipses, spheres, and cylinders, are Inside, Probe Center, and Outside.

Above Plane

With this setting, the line-of-sight to the target (or the probing direction) points in the direction in which offsets should be applied. This is by far the most common measurement situation. In [Figure 15-7](#), the tracker measured several points on the floor. Since the resulting plane representing the floor should be offset away from the tracker, the Above Plane setting is used. In other words, the SMRs were located between the instrument and the measured feature.

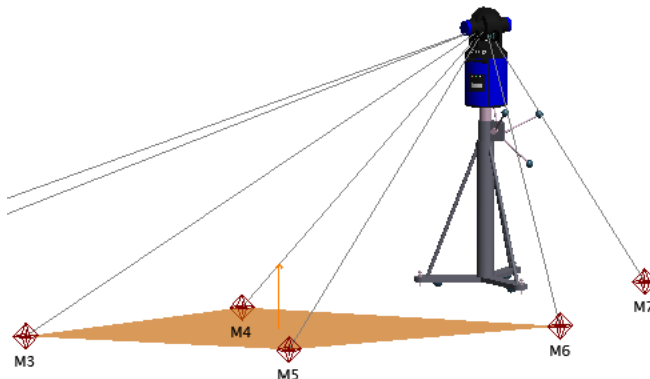
Figure 15-7. Offsetting along the probing direction using Above Plane.



Probe Center

This setting is used when it is preferred to ignore offsets and instead fit geometry along probe centers. This is not as common a setting. In [Figure 15-8](#), the Probe Center setting is active. Notice that the plane lies along the centers of the measured points--no offset is applied.

Figure 15-8. No offsets are applied when the Probe Center method is used.

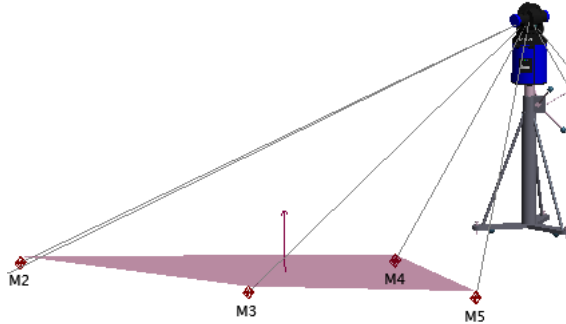


The radial version of Probe Center is equivalent--it implies that no offset is applied in the radial (inward or outward) direction.

Below Plane

This setting is used when the desired offset direction is in the opposite direction as the probing direction. With this setting, the geometry is shifted back toward the instrument (Figure 15-9).

Figure 15-9. Below Plane applies offsets opposite the probing direction.

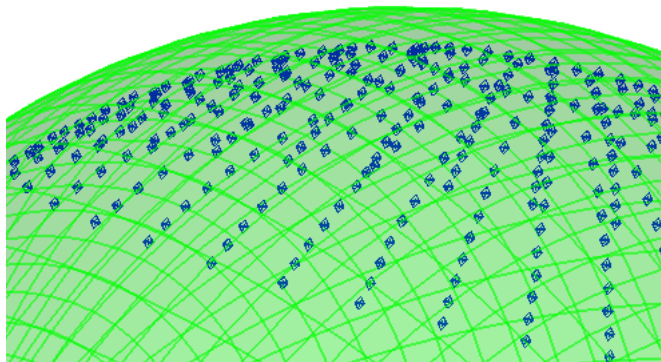


This is most commonly encountered with line-of-sight instruments in measurement situations where obtaining a proper line-of-sight to a surface is not obtainable, perhaps due to lack of space or the inability to move an instrument. As a result, the measured surface faces away from the instrument, and the probe is placed on the edge of the feature to obtain the measurement. This is often referred to as a backside measurement.

Inside

This is a radial setting for geometry such as circles, spheres, cylinders, and paraboloids. Inside (Figure 15-10) implies that the probe was placed on the inside of the geometry, and therefore the geometry should be shifted outward from the probe centers. The following figure depicts a sphere with the *Measured Side* set to *Inside*:

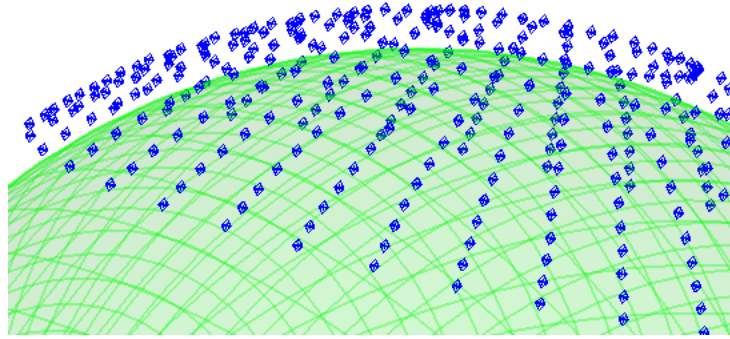
Figure 15-10. Measured side is set to *Inside*, since the measurements were taken on the inside of this sphere.



Outside

This is a radial setting for geometry such as circles, spheres, cylinders, and paraboloids. Outside implies that the probe was placed on the outside of the geometry, and therefore the geometry should be shifted inward from the probe centers. Figure 15-11 depicts a sphere with the *Measured Side* set to *Outside*:

Figure 15-11. Measured side is set to *Outside*, since the measurements were taken on the outside of this sphere.



Basic constructions and the Fit Interface

The Geometry fit interface can be accessed directly from the Construction tab of the ribbon (or under **Construct>Geometry by Fitting**)

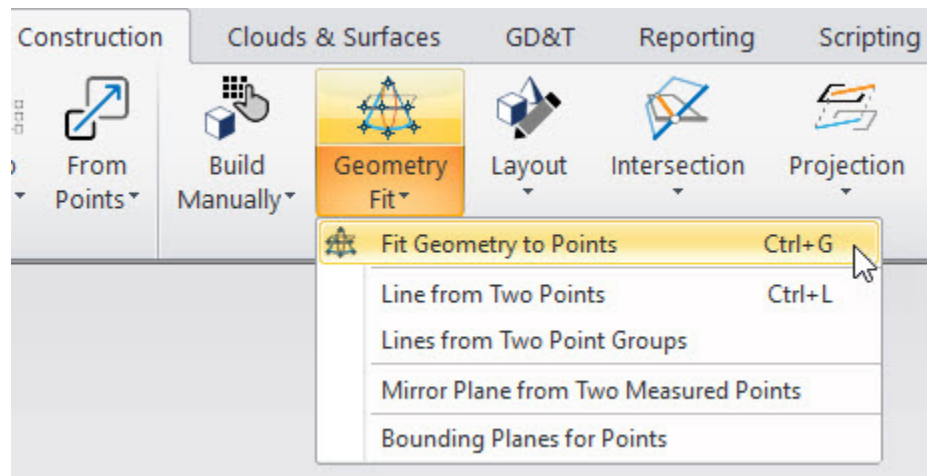


Figure 15-12. Geometry Fit Interface

The Fit Interface

The Geometry Fit Interface for a circle is pictured in [Figure 15-13](#) (the available options change slightly depending on the selected geometry).

GR-Features can be built from the Geometry Fit interface if you select the option to do so, or you can work directly with GR-Features from the Features tab. More information on geometry relationships is available starting with “[Definition of Geometry Relationship Types](#)” on page 472.

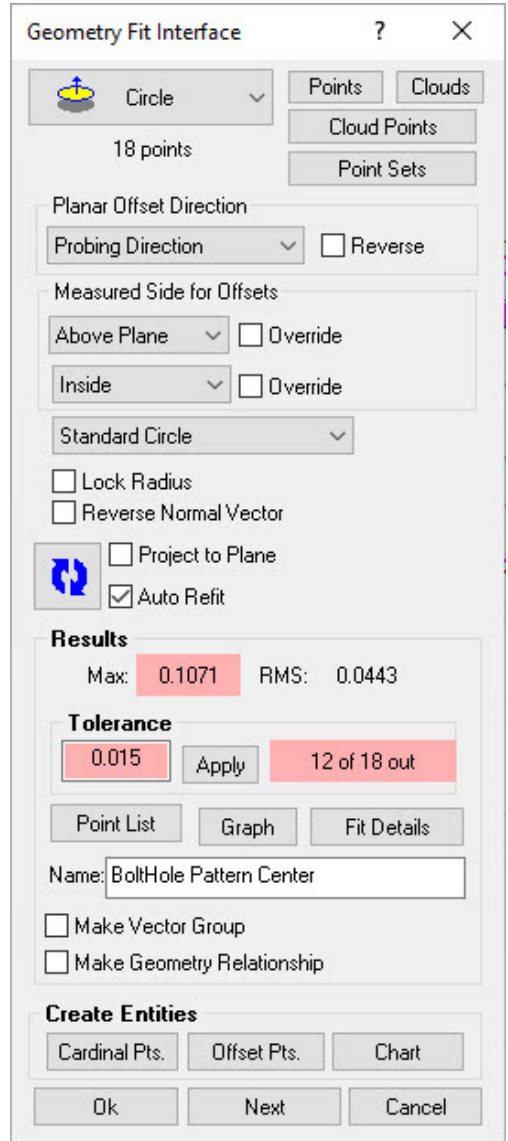
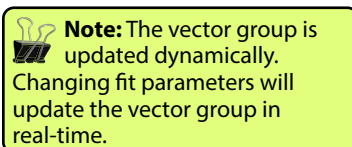
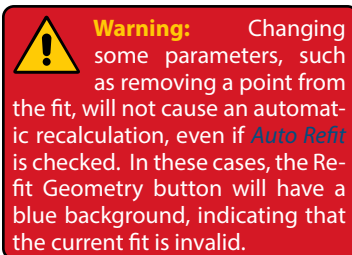
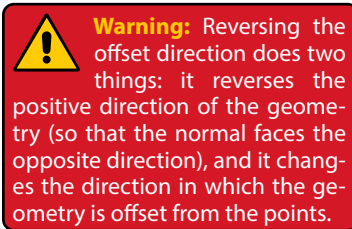


Figure 15-13. The Geometry Fit Interface

The options common to most geometry types will be discussed below, followed by geometry-specific options.

Common Settings

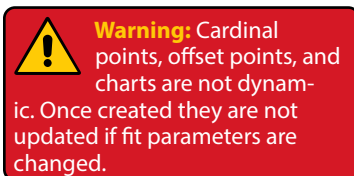
- **Geometry Type Selector.** Allows you to select the type of geometry to fit. This dropdown can be changed at any time to switch to a different geometry type.
- **Point Selection Buttons.** Choose from either Points or Clouds. Click the button, then select the points to include in the fit. The number of selected points will be displayed below these buttons--choose as many points as desired.
- **Planar Offset Direction.** Determines the planar offset method of the resulting planar geometry, using one of three possible methods: Probing Direction, Working Origin Positive, and Right-Hand Rule. This setting is used for plane, circle, and el-



- lipse fits.
- **Reverse.** Reverses the current offset direction. If the selected planar offset method does not offset in the correct direction, click this checkbox to reverse it.
 - **Measured Side Settings.** These settings allow you to specify on which side of the feature the specified geometry was measured. For planar geometry (planes, circles, and ellipses), there are planar settings of Above Plane, Probe Center, and Below Plane. For solid geometry, there are radial settings of Inside, Probe Center, and Outside.
 - **Override.** Checking this checkbox allows you to enter a value to override the selected probe radii and therefore use a different offset value (not applicable to the Probe Center option).
 - **Reverse Object Normal Vector.** Flips the normal direction of the resulting geometry, without changing other characteristics (such as position or radius).
 - **Refit Geometry.** Click this button to update the fit. If this button has a blue background, it indicates that the fit is not currently valid and needs to be recalculated.
 - **Auto Refit.** When checked, changing fit parameters will cause the fit to automatically recalculate.
 - **Project to Plane.** When checked, you can select a plane to project points to prior to performing the fit.
 - **Fit Summary.** The maximum absolute error and RMS error of the fit is displayed in this section.
 - **Fit Tolerance.** The current user-specified tolerance for the fit. To enter a new tolerance, type a new number into the field and then click *Apply*.
 - **View Point List.** Opens the Point List for the fit (see below).
 - **Graph.** Displays the Geometry Fit Graph for the fit (see below).
 - **Fit Details.** Displays a detailed, geometry-specific summary of the fit. For example, it shows absolute and signed errors, calculated radii or sphere centers, and other important information.
 - **Name.** Use this field to specify the name for the resulting geometry. The geometry will be placed into the active collection.
 - **Make Vector Group.** If checked, a vector group will be created that represents a comparison between the points and the resulting geometry from the fit. This is an excellent way to visualize errors in the fit.
 - **Make Geometry Relationship.** When selected the fit is per-

formed and a new GR-Features is built at the same time, creating dynamic geometry. For more information on GR-Features refer to "Definition of Geometry Relationship Types" on page 472.

- **Properties.** The Properties button opens the properties for the vector group representing the fit. This button is only displayed when **Make Vector Group** is checked.
- **Cardinal Pts.** When clicked, creates cardinal points (centroid, axis points, etc.) for the geometry.
- **Offset Points.** When clicked, creates points offset toward the geometry by an amount equal to the probe radius.
- **Chart.** When clicked, creates a histogram and run chart depicting point errors for the fit.



Geometry-Specific Settings

- **Lock Radius (Circles, Spheres, & Cylinders).** Allows you to specify a radius for the resulting geometry. SA will orient the specified geometry to minimize errors in the fit.
- **Lock Focal Length (Paraboloids).** Allows you to specify a focal length for a paraboloid. SA will orient the paraboloid to minimize errors in the fit.
- **Lock Included Angle (Cones).** Allows you to specify the apex angle for a cone. SA will orient the cone to minimize errors in the fit.
- **Standard [Geometry]/Max Inscribed/Min Circumscribed (Cylinders, Circles, and Slots).** Specifies a different type of fit for cylinders, circles, or slots. *Standard* performs a traditional fit by minimizing errors. *Max Inscribed* finds the largest geometry that fits inside the specified points, whereas *Min Circumscribed* finds the smallest geometry that fits around the specified points.
- **Round/Square (Slots).** Used to specify whether a slot should be round or square.
- **Guess Before Each Fit/Use Last Fit As Guess/Use Object As Guess (Cylinders, Paraboloids, Cones).** Specifies how the initial condition for a fit is determined. Due to the nature of these fit types, the initial condition for the fit can have an effect on the resulting generated geometry. For example, if there is not great axial or radial coverage of a cylinder, it is possible for the cylinder fit algorithms to get "confused". This option helps the fit algorithms find the proper solution. **Guess Before Each Fit** asks the algorithm to take a guess as to the initial size and orientation of the geometry before it progresses through the fit to

find the ideal solution, and is the default setting. **Use Last Fit As Guess** uses the current geometry fit result's size and orientation as the initial condition for the next fit recalculation. **Use Object As Guess** allows you to pick existing geometry of the same type to specify the starting condition geometry for the fit algorithm. In this way, you can provide a "hint" as to the resulting orientation of the geometry. If you find that your cylinder, paraboloid, or cone fits are not what you expect, try using this setting to help the algorithm with an initial guess.

- **Lock Focus Location (Paraboloids).** Prevents the current focus location from changing if a fit recalculation occurs.
- **Lock Vertex Location (Paraboloids).** Prevents the current vertex location from changing if a fit recalculation occurs.

Point List

The *Geometry Fit Point Listing* dialog (Figure 15-14) allows you to view individual points and their errors in a geometry fit.

The table contains all points selected for the fit, in addition to their individual deviation from the best-fit geometry. For circles and ellipses, planar and radial component error is displayed as well. To sort the list by a specific attribute (point name, magnitude error, radial error, etc.) click the associated column header. Click it again to sort the list in the opposite direction.

Any point can be excluded from the fit by unchecking its checkbox, and any excluded point can be included again by re-checking the checkbox. Unchecked (excluded) points appear with other excluded points in the list, so it is easy to find all excluded points.

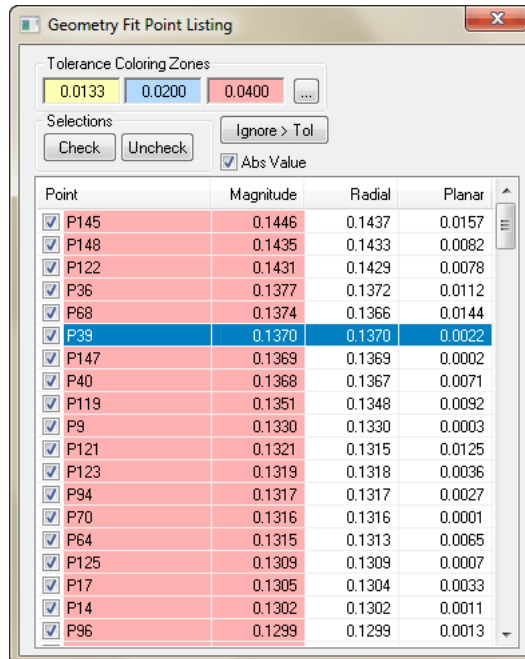


Figure 15-14. The Geometry Fit Point List.

Points that lie outside the specified tolerance in the main dialog are colored red. Points in the intermediate tolerance zones are yellow and blue, and points inside the first tolerance zone have a white background.

To Toggle Between Absolute Error and Signed Error:

- Check or uncheck the Abs Value checkbox.

To Remove Selected Points From the Fit:

- Select one or more points in the list.
- Do one of the following:
 - Click the **Check** or **Uncheck** button to include or exclude points from the fit, or
 - Right-click any of the selected points and choose **Check** or **Uncheck** from the context menu.

To Toggle the Included/Excluded State for the Selected Points in the Fit:

- Select one or more points in the list.
- Right-click any of the selected points and choose **Toggle**.

To Permanently Remove Selected Points From the Fit:

- Select one or more points to remove.
- Right-click any of the selected points and choose **Remove From List**.


To Exclude Points Beyond the Specified Error Tolerance:

- Click the **Ignore > Tol** button.

To Delete Selected Points from the SA Job:

1. Select one or more points to delete.
2. Right-click any of the selected points and choose **Delete From SA Job**.

To Change Limits or Colors for the Tolerance Zone Coloring:

- Click the ellipses  button next to the zone color swatches.

Tolerance Zone Coloring

The *Tolerance Zone Coloring* dialog (Figure 15-20) allows you to change the specified limits and coloring for three tolerance zones.

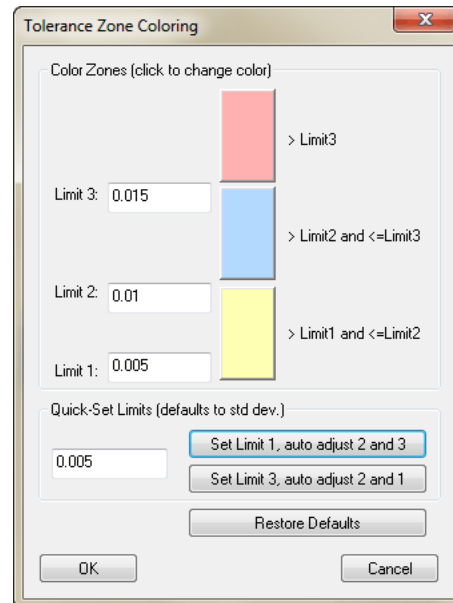


Figure 15-15. The Tolerance Zone Coloring dialog.

These zones allow you to characterize errors with three coloring zones. This is often useful for users who have a 1, 2, and 3-sigma tolerance. Each zone is bounded by limits. The first coloring zone starts at greater than *Limit 1* and ends at *Limit 2*. The second zone starts at greater than *Limit 2* and ends at *Limit 3*. The last zone starts at greater than *Limit 3*.

In the figure at left, points with errors $>0.005''$ and $\leq 0.01''$ are colored yellow; points with errors $>0.01''$ and $\leq 0.015''$ are colored blue, and points with errors $>0.015''$ are colored red. For users who only need one coloring zone, simply pay attention to the red zone and ignore the other two, or set all three zones to the same value.

The tolerance specified in the main Geometry Fit Interface is applied as the *Limit 3* value.

The *Quick-Set Limits* section allows you to set a tolerance for Limit 1 or Limit 3, and then set the remaining limit values based on 1, 2, and 3-sigma values. Click **Restore Defaults** to return the values to their default limits.

Geometry Fit Graph

The *Geometry Fit Graph* (Figure 15-16) allows you to view an interactive graph of the errors for all points in the fit. It is an excellent way to visually find outliers.

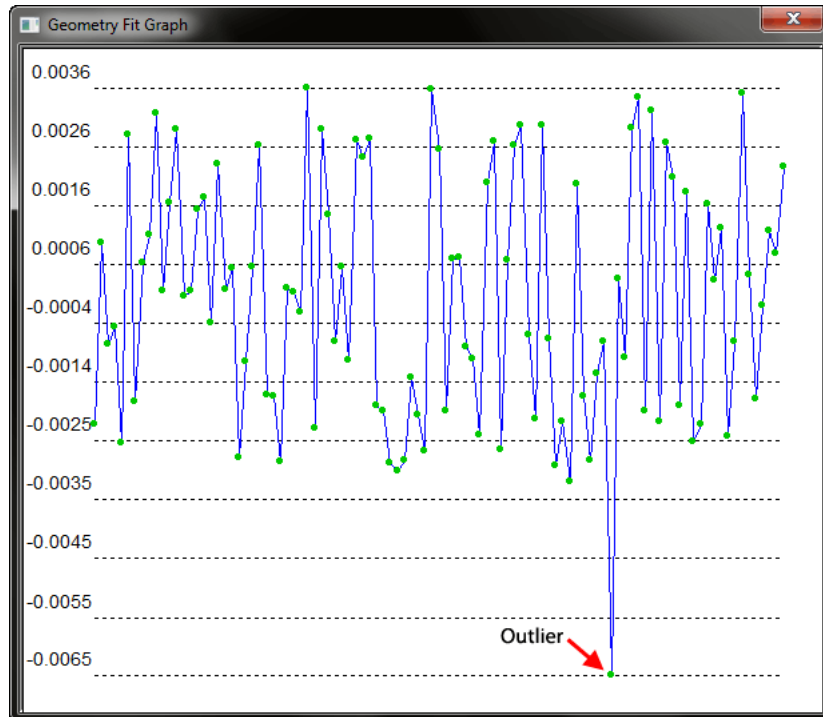


Figure 15-16. The Geometry Fit Graph makes detection of outliers easy.

Any point (outliers in particular) can be removed from the fit by way of the graph, and you can also find the point name and error for every point.

To Remove an Outlier from the Fit Using the Graph:

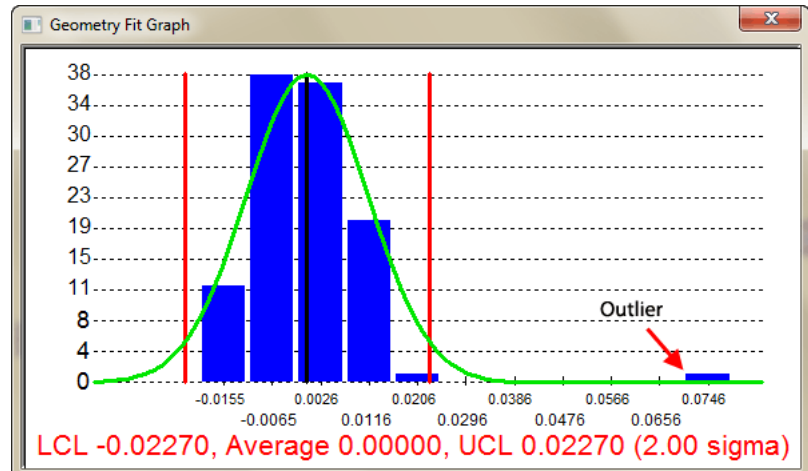
- Left-click the point on the graph and choose **Ignore Point**.

To View a Point's Name and Error:

- Left-click a point in the graph. The point name and error will appear at the top of the context menu.

In addition to a run chart of the data, you can also view a statistical histogram of the data (Figure 15-24).

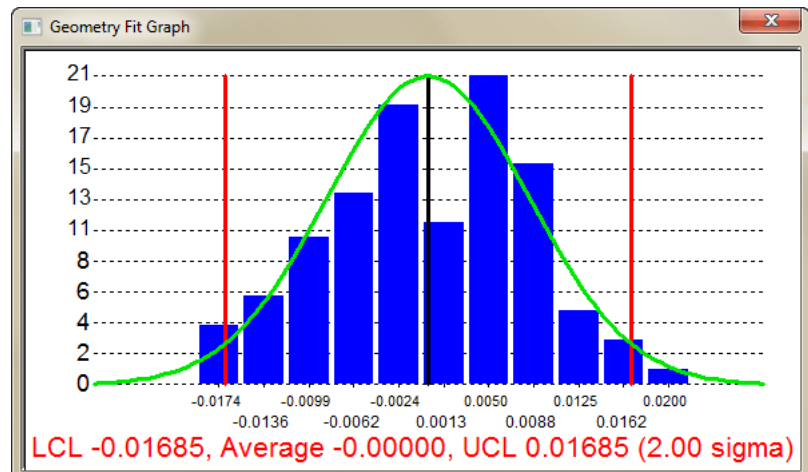
Figure 15-17. Viewing fit data in histogram form can also point to outliers or inconsistencies in data.



The histogram also includes the Lower Control Limit (LCL), average, and Upper Control Limit (UCL) at 2 sigma. As seen above, the histogram can also be used to detect an outlier, in this case located far outside the UCL.

By removing the outlier from the fit, the histogram reveals a more natural bell-shaped curve, as real-world data would typically indicate.

Figure 15-18. The outlier has been removed from the histogram, revealing a more natural Gaussian distribution of data.



Both the histogram and the run chart can be easily copied to the clipboard for pasting into a report or graphics program.

To Copy a Chart to the Clipboard:

- Display the desired chart, then right-click the chart and select **Copy to Clipboard**.

Definition of Geometry Relationship Types

Geometry Relationships

Geometry relationships build dynamic geometry. They continually update the position, orientation and size of an SA object based upon the associated points or objects. They also report deviations between measured and nominal geometry.

Geometry Comparison

This relationship type can be subdivided into 3 types:

- **Fit and Compare to Nominal.** This type provides a full fit and compare function between the measured points and the selected nominal geometry.
- **Fit Only.** Without a nominal assignment, Fit Only relationships build dynamic geometry but do not provide a comparison to a nominal feature (These relationships can easily be expanded to Fit and Compare to Nominal relationships with a nominal designation).
- **Compare Only.** These relationships only calculate and compare the differences between objects of the same type. They do not perform geometry fit functions.

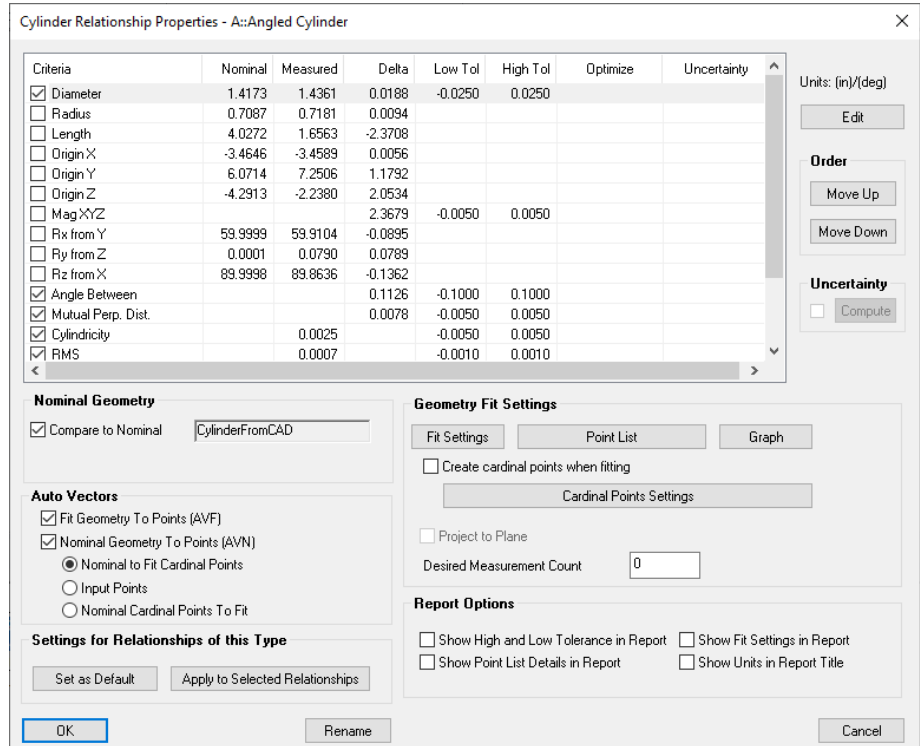
Properties & Controls in GR-Features

Geometry Relationships (GR-relationships) allow for dynamic fitting and comparison of geometry. Unlike standard geometry construction operations, GR-relationships will update the size and position of the geometry as measurements are either added or removed from the relationship and any comparison with a nominal geometry is also automatically updated.

GR Properties and Report Options

The Geometry Relationship Properties dialog provides control over all the individual controls for the relationship. Right-click the relationship and select properties, the respective properties dialog will appear (Figure 15-19).

Figure 15-19. Geometry relationship properties.



Report Options

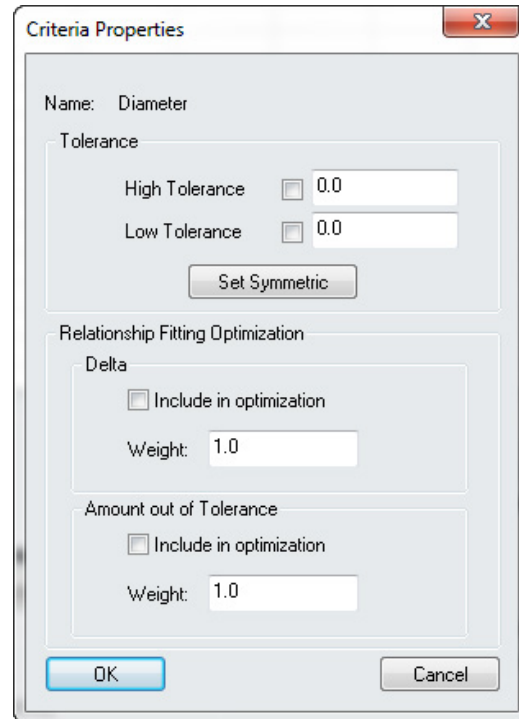
The report options for the relationship are composed of a list of specific *Criteria* (Figure 15-20). The check boxes on the left provide a means to include or exclude these computed values from the resulting report. The order of the criteria listed can also be adjusted by selecting a row and using the **Move Up** or **Move Down** buttons on the right.

Figure 15-20. Geometry relationship report options.

Criteria	Nominal	Measured	Delta	Low Tol	High Tol	Optimize
<input checked="" type="checkbox"/> Diameter	0.7874	0.7883	0.0009	-0.0100	0.0100	
<input type="checkbox"/> Radius	0.3937	0.3942	0.0005			
<input type="checkbox"/> Length	0.5906	0.2910	-0.2996			
<input type="checkbox"/> Origin X	2.5591	2.5589	-0.0002			
<input type="checkbox"/> Origin Y	4.3307	4.3306	-0.0001			
<input type="checkbox"/> Origin Z	0.7874	0.4301	-0.3573			
<input type="checkbox"/> Mag XYZ			0.3573			
<input type="checkbox"/> Rx from Y	-90.0000	89.9112	179.9112			
<input type="checkbox"/> Ry from Z	-180.0000	0.0066	180.0066			
<input type="checkbox"/> Rz from X	0.0000	85.7663	85.7663			
<input checked="" type="checkbox"/> Angle Between			179.9110			
<input checked="" type="checkbox"/> Mutual Perp. Dist.			0.0001			
<input checked="" type="checkbox"/> Cylindricity		0.0005				
<input checked="" type="checkbox"/> RMS		0.0001				

In addition, tolerance and relationship optimization settings can be accessed for each individual criteria by double clicking on a selected row (Figure 15-21).

Figure 15-21. Adjusting geometry relationship criteria.



Tolerance settings are absolute if no nominal reverence is available and no delta can be computed. When a nominal is available, they are relative and evaluated with respect to the delta value.

In addition to the criteria tolerance settings there are a set of check boxes controlling report table parameters including:

- Show High and Low Tolerance in Report
- Show Point List Details In Report
- Show Units in Report Title

Each criteria has the ability to be used in a relationship minimization. This allows a user to fine tune a relationship fit based on geometry components. An optimization can be performed using Delta and/or the Amount out of Tolerance.

Pressing the **Edit** button will open a master overview panel that facilitates editing of a selection of criteria at one time (Figure 15-22). It includes reporting settings, callout settings, and relationship fit parameters.

Criteria	Nominal	Measured	Delta	Low Tol	High Tol	Optimize	Uncertainty
<input checked="" type="checkbox"/> Diameter	1.4173	1.4361	0.0188	-0.0250	0.0250		
<input type="checkbox"/> Radius	0.7087	0.7181	0.0094				
<input type="checkbox"/> Length	4.0272	1.6563	-2.3708				

Name	Report	Callout	Nominal	Measured	Delta	Use Low Tol	Low Tol	Use High Tol	High Tol
Diameter	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	1.4173	1.4361	0.0188	<input checked="" type="checkbox"/> *	-0.0020	<input checked="" type="checkbox"/> *	0.0200
Radius	<input type="checkbox"/>	<input type="checkbox"/>	0.7087	0.7181	0.0094	<input type="checkbox"/>	0.0000	<input type="checkbox"/>	0.0000
Length	<input type="checkbox"/>	<input type="checkbox"/>	4.0272	1.6563	-2.3708	<input type="checkbox"/>	0.0000	<input type="checkbox"/>	0.0000
Origin X	<input type="checkbox"/>	<input type="checkbox"/>	-3.4646	-3.4589	0.0056	<input type="checkbox"/>	0.0000	<input type="checkbox"/>	0.0000
Origin Y	<input type="checkbox"/>	<input type="checkbox"/>	6.0714	7.2506	1.1792	<input type="checkbox"/>	0.0000	<input type="checkbox"/>	0.0000
Origin Z	<input type="checkbox"/>	<input type="checkbox"/>	-4.2913	-2.2380	2.0534	<input type="checkbox"/>	0.0000	<input type="checkbox"/>	0.0000
Mag XYZ	<input type="checkbox"/>	<input type="checkbox"/>			2.3679	<input checked="" type="checkbox"/>	-0.0010	<input checked="" type="checkbox"/>	0.0010
Rx from Y	<input type="checkbox"/>	<input type="checkbox"/>	59.9999	59.9104	-0.0895	<input type="checkbox"/>	0.0000	<input type="checkbox"/>	0.0000
Ry from Z	<input type="checkbox"/>	<input type="checkbox"/>	0.0001	0.0790	0.0789	<input type="checkbox"/>	0.0000	<input type="checkbox"/>	0.0000
Rz from X	<input type="checkbox"/>	<input type="checkbox"/>	89.9998	89.8636	-0.1362	<input type="checkbox"/>	0.0000	<input type="checkbox"/>	0.0000
Angle Between	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			0.1126	<input type="checkbox"/> *	-0.1000	<input type="checkbox"/> *	0.1000
Mutual Perp. Dist.	<input type="checkbox"/> *	<input type="checkbox"/> *			0.0078	<input type="checkbox"/> *	-0.0050	<input type="checkbox"/> *	0.0050
Cylindricity	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		0.0025		<input checked="" type="checkbox"/> *	-0.0050	<input checked="" type="checkbox"/> *	0.0050
RMS	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		0.0007		<input type="checkbox"/> *	-0.0010	<input type="checkbox"/> *	0.0010
J	<input checked="" type="checkbox"/> *	<input type="checkbox"/>	0.5000	0.5014	0.0014	<input type="checkbox"/>	0.0000	<input type="checkbox"/>	0.0000
I	<input checked="" type="checkbox"/> *	<input type="checkbox"/>	0.0000	0.0012	0.0012	<input type="checkbox"/>	0.0000	<input type="checkbox"/>	0.0000
K	<input checked="" type="checkbox"/> *	<input type="checkbox"/>	0.8660	0.8652	-0.0008	<input type="checkbox"/>	0.0000	<input type="checkbox"/>	0.0000

Figure 15-22. Criteria Group Editor.

Form Checks

Geometry relationships are designed to be computed quickly and repeatedly. For this reason, form checks do not use the full GD&T mathematical evaluation methods. Instead, they use a simplified computation based on the RMS fit object. Details are as follows:

- **Linearity.** The linearity of a line is similar to straightness but is computed as $2 \times (\text{max point deviation from the RMS fit line})$. In version prior to 2022.3 this was computed as the Max - Min deviation.
- **Flatness, Circularity, Cylindricity, and Sphericity.** The form of all other geometry types are computed as the Max - Min deviation with respect to the RMS fit object. This effectively adds the largest deviations on either side of the surface of the object to create a form deviation estimate.

These form results typically overestimate the form value by a small amount as compared to a full GD&T check result. A check that is true to the ASME or ISO specification can be added by linking a GD&T check to an existing GR-Feature (see “[Creating GD&T Annotations](#)” on

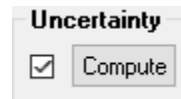
page 612).

Geometry Uncertainty Computation

Geometry Relationships offer the ability to compute uncertainty using the uncertainty of the points included in the relationship. In order to compute uncertainty perform the following steps:

1. Ensure that all of the points included in the GR feature have uncertainty fields computed. This can be done by using the function **Analysis > Coordinate Uncertainty > Create Point Uncertainty Fields**.
2. In the relationship properties check the Uncertainty check box and then press **Compute**. This will ask you for the number of samples to use and generate the uncertainty event, as well as adding the uncertainty statistics to the related GR Criteria

Figure 15-23. Compute Uncertainty Button



For more information on the geometry uncertainty computation process (see “[Geometry Fit Uncertainty](#)” on page 777).

Once uncertainty has been computed for a relationship the relationship is put into a Dormant status. This is done so that the uncertainty values are not removed as part of the normal recomputation process of a relationship. At any time the dormant status can be removed through the right-click option **Set/Clear Relationship Dormant Status**.

Cardinal Points

Cardinal Points can be turned on for any geometry. These points can be used as the input for other GR-Relationships to build dynamic geometry. For example, if you wish to build a line along the axis of a cylinder you can turn on the cardinal points for the cylinder and use those points to build line. As the cylinder updates the cardinal points will update and so will the position and length of the line.

Different numbers of cardinal points exist for different geometry types and the choice to turn on and off particular points and to control their naming is given when you turn them on ([Figure 15-24](#)).

Figure 15-24. Cardinal Point Name Control.

Automatic Vector Groups.

GR-Relationships offer a selection of Auto-Vectors to help visualize Point deviations and deviations between features. These include:

Fit Geometry to Points (AVF) - These vectors show the individual deviations between the measured points and the fit geometry. It helps visually display the quality of the fit.

Nominal Geometry to Points (AVN) - This Auto Vector type displays the deviations between the measured feature and the nominal feature. It can be shown as either:

1. **Nominal to Fit Cardinal Points** - The deviation of the cardinal points of the fit geometry from the axis or surface of the nominal geometry. This provides a minimal number of vectors to clearly illustrate the relative misalignment of the features. For planes the vertex points are used rather than the cardinal points.
2. **Input Points** - The deviations of the measured points as directly compared with the nominal feature as if it were the fit feature. This shows both deviations in form and location.
3. **Nominal Cardinal Points To Fit** - (Cylinders Only) This option more

closely reflects the results of a GD&T inspection in that it displays the deviations extrapolated beyond the fit cylinder to the extents of the nominal cylinder.

Set as Default and Apply to Selected

The default state of just about any setting can be defined for a geometry relationship and applied to other relationships of the same geometry type.

Geometry Fit Settings:

The Geometry Fit Setting will depend on the type of relationship. These types include:

1. Points and Point Group based relationships which provide greater control over individual point selection
2. Cloud based relationships which provide additional subsampling options
3. Intersection relationships which use objects as inputs.

Points based Fit / Fit and Compare relationships are performing a standard geometry fit operation using the associated points and updating that fit whenever the list of points changes. The geometry fit controls are the same as those offered for construction operations and include the following (Figure 15-25):

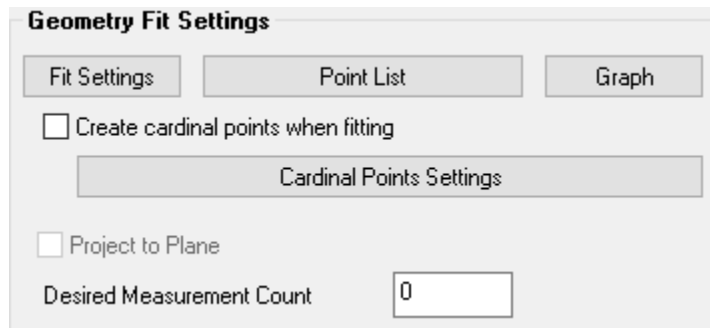


Figure 15-25. Points Based Relationship fit settings

Fit Settings - Provide basic geometry fit controls such as offset compensation, diameter override, Max Inscribed/Min Circumscribed, etc.

Point List - Shows a list of the associated points in the relationship and includes tolerances and point rejection controls.

Graph - Displays a graph of individual point deviations from the fit geometry and offers the ability to remove points graphically.

Cardinal Points - Cardinal or Design points are described separately below.

Projection Plane - 2D GR-Relationships offer the option to turn on a projection plane. Checking the Project to Plane check box will cause the user to be prompted to select the desired projection plane. Projec-

tion planes can also be assigned automatically when a relationship is built through designation in the toolkit.

Desired Measurement Count - The desired number of points can be designated for a relationship. This is used for inspection and will cause trapping to stop once the desired number of points has been reached during measurement.

Intersection Geometry don't offer any fit settings through the properties dialog other than cardinal point selection. The geometry selection process can be triggered through the relationship's right-click menu under *Associate Data*.

Working with Cloud Data

Geometry Relationships can also take point clouds as inputs. When a point cloud is associated with a geometry relationship the **Point List** button is replaced with a subsampling control (Figure 15-26):

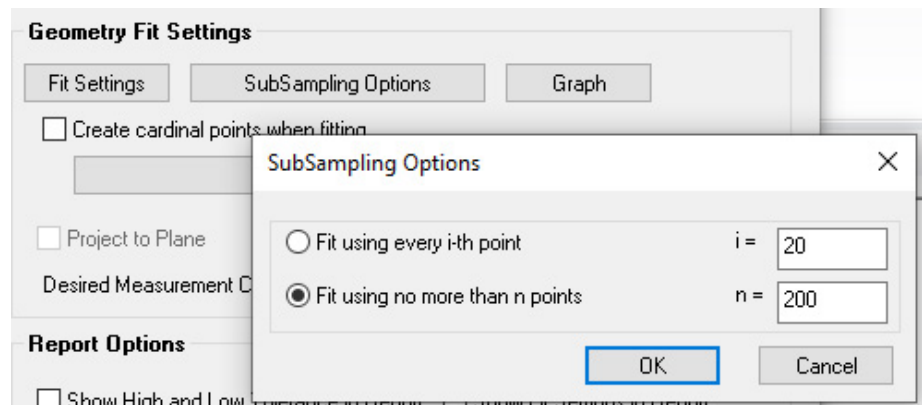


Figure 15-26. Point Cloud Based Relationship Fit Settings

By default geometry relationships take a random sample using a uniform distribution from the cloud data and only use 200 points. This can be seen when the AVN vectors are enabled. This allows for quick and efficient fitting.

Proximity filters can be defined for a Compare to Nominal relationship. The defaults for newly created relationships can be accessed directly through the **User Options>Display>Default Options>FR Proximity Filter**. But each relationship has its own settings which can be accessed through the feature's right-click menu when a nominal is assigned (**Configure Cloud Trapping Filter**) (Figure 15-27).

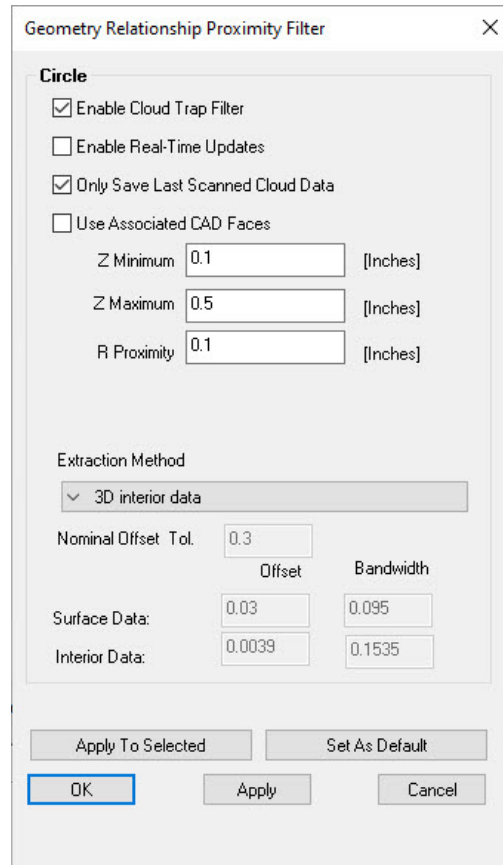


Figure 15-27. GR-Feature Proximity Filter settings for a circle.

These Proximity settings come into use in either of two way:

1. When you use the option in a features right-click menu: **Associate Data > Proximity Filtered Clouds**. You will be prompted to select a cloud and a new cloud will be built from the reference after applying the filter and associated with the feature.
2. Trapping while scanning. When actively trapping to a feature these proximity settings can be used. This limits the data acquired while scanning such that you only acquire points within the defined proximity to the nominal feature.

Proximity filtering is a great way to associate relevant cloud points with a particular feature and can easily be used in combination with a job level feature extraction (see “[Feature Extraction from Point Clouds](#)” on page 587).

Outlier Rejection

Outlier reject for GR-Features using points are available as follows:

- **Point List.** The point list can be used for outlier rejection by setting a tolerance and using the **Ignore > Tol** button to ignoring points beyond a threshold. The list headers can also be used to sort the data and selections of points can be excluded manually using Shift or Control Click selection.

- **Graph.** The deviation graph also provides direct access to included points and their relative deviation. Multi-select allows for quick removal of groups of points, all of which are set to “ignored” in the point list.

Outlier rejection for cloud based GR-Features are as follows:

- **Graph.** The Point List is not available for cloud based relationships. Rather than keeping track of individual cloud points a reference cloud is used to build an analysis cloud. The graph provides a means to remove points from this analysis cloud. The graph functions differently as well in that it provides a histogram display of the data and selection can be made of the data to keep with respect to sigma fit markers.

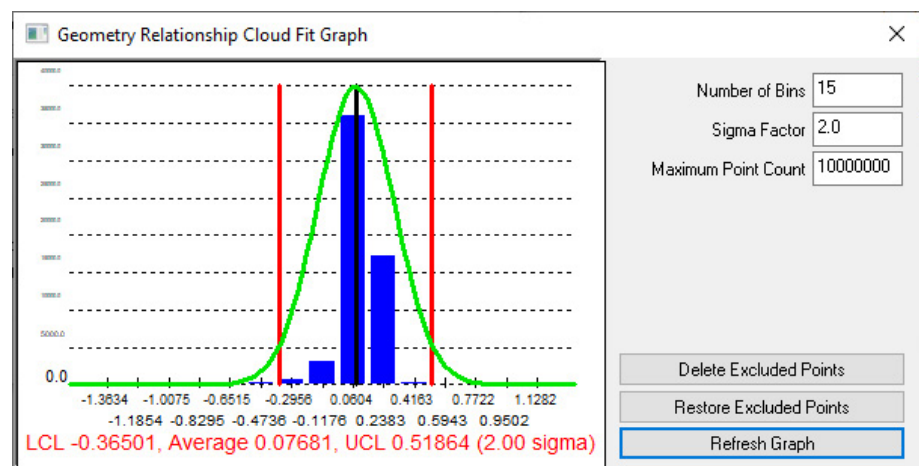


Figure 15-28. Cloud Point Fit Graph used for outlier rejection.

- **Proximity filter.** In addition to simple geometry extraction the proximity filter can be used to edit the cloud points included in the fit by adjusting the filter settings and pressing apply. This will effectively delete the existing filter cloud and re-extract from the original input cloud using the new settings.

Reporting Controls

Geometry Relationship Summary Table

By right-clicking on the Relationship category in the tree an option called **Create Geometry Relationship Summary Table** is available. This option prompts a user to select from a list of the geometry relationships in the job. It then builds a custom report containing the reporting criteria for each of the selected geometry relationships in a single table format.

Geometry Relationship Summary						
A::BoltHole Pattern1						
Criteria	Nominal	Measured	Delta	Low Tol.	High Tol.	Tolerance
Diameter	0.6457	0.6533	0.0077	-0.0050	0.0050	0.0027
Mag XYZ			0.0084	-0.0050	0.0050	0.0034
Circularity		0.0018		-0.0020	0.0020	Pass
RMS		0.0006		-0.0010	0.0010	Pass
Measurements						9
Proj. Plane						GR-Top Plane
A::BoltHole Pattern2						
Criteria	Nominal	Measured	Delta	Low Tol.	High Tol.	Tolerance
Diameter	0.6457	0.6542	0.0086	-0.0050	0.0050	0.0036
Mag XYZ			0.0084	-0.0050	0.0050	0.0034
Circularity		0.0013		-0.0020	0.0020	Pass
RMS		0.0004		-0.0010	0.0010	Pass
Measurements						9
Proj. Plane						GR-Top Plane
A::BoltHole Pattern3						
Criteria	Nominal	Measured	Delta	Low Tol.	High Tol.	Tolerance
Diameter	0.6457	0.6570	0.0113	-0.0050	0.0050	0.0063
Mag XYZ			0.0097	-0.0050	0.0050	0.0047
Circularity		0.0018		-0.0020	0.0020	Pass
RMS		0.0005		-0.0010	0.0010	Pass
Measurements						8
Proj. Plane						GR-Top Plane

Figure 15-29. Example Custom Geometry Relationship Table

Additional GR-Feature types

Average Point

These relationships are the points equivalent of a geometry relationship. The average point relationship calculates a points position based upon the associated input points and compares it to a nominal point.

Select Nominal Frame... Frame Wizard

This relationship is the frames equivalent of a geometry relationship. Like a Frame to Frame relationship it compares the position and orientation of two frames, but the relationships “measured” frame is built upon the associated points and features using the Frame Wizard selection utility(see “[Frame Wizard](#)” on page 259).

This frame can therefore be built upon other geometry relationships which in tern make it a dynamically updating frame. For this reason, geometry extracted from CAD is built as a placeholder geometry relationships that is then added to the tree, ready to be measured. This is a difference from the construction tool which builds a static frame from the nominal features extracted from the CAD model.

Frame Wizard relationships can also be used for alignment when a nominal frame is assigned. Using the **Execute Alignment** option in the right-click menu will perform a frame to frame alignment, aligning the measured frame to the nominal frame.

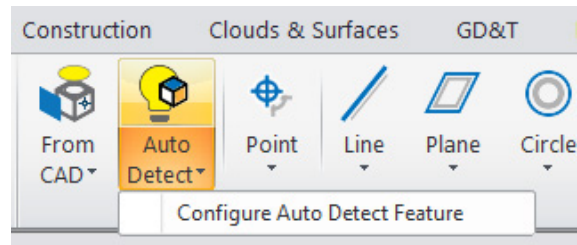
When “CAD Nominal” is enabled in the ribbon a special CAD only option enforced. This restricts selection to a CAD model and builds a

nominal frame as part of the same operation. This is perfect for defining a frame to frame or 3-2-1 type alignment to a job file that is dynamic and repeatable.

Auto Detect Feature

An Auto Detect feature is a placeholder that can be used to build a Geometry Relationship of the correct geometry type automatically based upon an RMS fit threshold to the associated points. It is intended to be used with Feature Trapping but can also be used with direct association.

Figure 15-30. Smart Feature Fit to Points as seen in the Ribbon Menu.



How it works

When trapping stops, by progressing to the next feature in the list, an attempt will be made to resolve the correct geometry type. This is accomplished through a sequential series of geometry fits to the data set, comparing the results to the RMS acceptance threshold.

The search sequence is: Line, circle, slot, plane, cylinder, sphere, cone.

Once an acceptable fit is found, the search stops and the geometry is built. At this point the “Auto Detect” feature in the tree will be converted into a GR-Feature with the type first found with an RMS below the specified threshold. This can also be triggered through a right-click menu **Resolve Smart Feature Geometry**.

Should an error occur and the geometry type be selected incorrectly this can be corrected through the GR-Features right click menu. Select **Reset Relationship Geometry** and pick the desired type. This menu option is hidden if the following conditions are not met:

- Measured geometry must have been created successfully.
- No nominal geometry can be associated
- No dynamically created geometry can be modified.

Depending on the number of steps in the solution it can take a short period of time to identify the geometry type. In 2023.1 an audible notification sound will be played when this step completes to help the user know when they can proceed. If points are acquired while the auto detect feature is being processed the newly measured points may not be associated.

Projection Plane Control

When enable, the active projection plane is set automatically as part of the Smart Feature resolve process. The idea being that each time plane is resolved it is set as the new projection plane. This works well when measuring lines/circles/slots on a sequence of different planes.

Note that a plane 90 degrees to the original will resolve as a line not a plane. So moving from one projection plane to another will work best if the planes are not perfectly orthogonal to the last.

This **Auto Reset Projection Plane** operation is optional and can be turned off in the configuration.

Detect Nominal from CAD

A search for the nominal feature is also included in the smart feature resolve process. This means that if you align to CAD first you can measure features as desired and record both the nominal and measured feature from the CAD as you go.

Single Point Geometry

A selection of geometry will become valid with only a single measurement. These include:

- **Average Point.** An average point feature will be valid with only a single point measurement.
- **Circle.** A GR-Circle will be valid with a single point measurement as long as a projection plane is defined, the point offsets are symmetrical, and a sphere defined by the center point and the radius of the probe offset intersects the projection plane. This intersection defined the size of the circle.
- **Sphere.** A GR-Sphere will be valid with a single point as long as it either has a fixed radius set in the fit settings, or a nominal is defined. The size of the sphere will be set by these settings in this order of priority.

Dynamic Geometry

Dynamic Geometry are a type of Geometry Relations that uses other objects or features for its computation. Like Comparison Only relationships they do not provide an option to select points but only evaluate the selected features for the computation.

General Computations

Dynamic Point

- **Mid-point of perpendicular to Two lines.** Use to establish the intersection point between two imperfect lines.

Dynamic Line

- Cylinder Axis
- Cone Axis
- **Bisect Two Lines.** Use to create a line at the center of two reference lines.
- Slot Center line along length

Dynamic Plane

- **Bisect Two Planes.** Use to create a line at the center of two reference lines.
- **Offset Plane.** Used to hold a specific distance and orientation relative to a reference plane (or the XY plane of a reference object).

Dynamic Circle

- **From Multiple Circles.** Used to build a circle through a set of reference circles such as a bolt hole pattern.

Intersections

These relationships compute and update a feature based upon the intersection of other geometries. The following intersection geometries are currently available:

Dynamic Points

- Intersection of a Line and Plane
- Intersection of a Cylinder (axis) and Plane
- Intersection of a Cone (axis) and Plane
- Intersection of Three Planes

Dynamic Line

- Intersection of Two Planes

Dynamic Plane

- Intersection of Two Cones (combinations)*
- Intersection of Cone and Cylinder (combination)*

Dynamic Circle

- Intersection of Cylinder and Plane (combinations)*
- Intersection of Cone and Plane (combinations)*
- Intersection of Sphere and Plane
- Intersection of two Cones (Best-Fit Plane)*
- Intersection of Cylinder and Cone (Best-Fit Plane)*

Dynamic Ellipse

- Intersection of Cylinder and Planes
- Intersection of Cone and Planes

Most intersections have a clear mathematical solution such as the intersection of a line and a plane. However others require more explanation and are marked with “*”.

Cone/cone and **cone/cylinder** intersections can be computed as a plane, ellipse or circle depending on the customers needs. These are computed as follows:

3. An exact analytical solution for 16 points spaced at uniform angular rotation about the axis of the first cone are developed for the intersection between the cone/cone or cone/cylinder.
4. These 16 points are used as input for determining a best-fit plane (approximate unless axes are coincident).
5. Ellipse is analytically constructed for the intersection of the first (only) cone with the best-fit plane.
6. Circle is constructed from ellipse by setting circle center to ellipse center and the radius = $\sqrt{\text{major} \times \text{minor}}$ [constant area]
7. Circle normal is set to best-fit plane normal, first cone axis, or second cone (cylinder) axis.

Its important to realize that, as cone/cone or cone/cylinder axes deviate further from coincident, the approximate solution degrades and as cone/plane or cylinder/plane deviate from perpendicularity, the solutions for circles created by intersection degrades. Therefore these intersections are intended primarily for situations where the axes are constrained such as measurements of shafts.

Comparison Only Geometry Relationships

Beginning in 2022 *Fit and Compare* geometry relationship can be converted to a *Comparison Only* relationships and vice versa.

To associate geometry:

1. Ensure that the geometry relationship has a nominal defined
2. Clear any prior point or cloud associations
3. Right-click and select *Associate Data>Measured Geometry*

There are a couple of differences between *Fit and Compare* and *Comparison Only* relationships which are as follows:

- AVF Vectors are disabled and AVN vectors are switched to Nominal to Fit Cardinal Points, if enabled.

- Projection Plane will no longer be an option
- Desired Measurement Count to -1. Trapping is currently not available for Comparison Only features.
- Point List and Fit reporting is also disabled.

Clearing the measured geometry association with re-activate this capability.

This Page Intentionally Left Blank.

Feature Inspection

CHAPTER

25

SA provides a streamlined set of tools which can be used for fast and effective feature inspection, include built in reporting and template creation. This process uses Geometry Relationships (GR) and/or GD&T Feature checks for standardized inspection. This chapter provides an overview of this capability.

Why Feature Based Inspection

Spatial Analyzer is a points based software. Each point is traceable back to the instrument that measured it and that link is fundamental to geometry fitting and instrument alignment. SA has always made it easy to measure points and fit geometry to those points, analyzing the deviations and reporting the statistics. However, part inspection is all about streamlined comparisons between nominal and measured geometry, reporting measured locations within specified tolerances and effectively reporting those results.

SA uses Geometry Relationships to effectively link nominal and measured geometry position and location information, and dynamically update and report on these comparisons. These features can be built in advance or after measurements are made and, once built, provide a template for future measurements

For more details on Geometry Relationships configurations refer to [“General Relationship Properties & Controls”](#) on page 420.

GR-Features also provide an ideal path to working with point cloud data. For more details on point cloud operations refer to [“Point Clouds”](#) on page 545.

Why GR-Features versus GD&T

Geometry can also be inspected using our ASME and ISO compliant Standardized GD&T inspection process. The datums and feature checks will also appear as stand along inspection features that also will updated dynamically.

The primary reason these evaluation processes are separate in SA are:

1. GD&T feature checks use the built in datum references to perform an internal alignment and then evaluate with respect to that alignment. GR features display deviations using the current job alignment which provide greater visual feedback.
2. GD&T feature use either ASME or ISO inspection processes that are designed to establish if a part is within specification. This means that it optimizes fits using bounding thresholds in its mathematical evaluation. GR features use an RMS fit to help identify measurement error.

Although these are separate approaches, both GR-feature inspection and GD&T can be used in tandem. More information on GD&T features and inspection can be found in the chapter on “GD&T” on page 611. But the GD&T Datums and Feature Checks will appear as part of the inspection process which can be conducted through the Inspection bar just like GR features.

Chapter Guide

The inspection chapter is broken into 3 separate pieces:

1. “[Building GR-Features](#)” on page 490 provides details on the buttons offered in the Ribbon Menu and how they can be used for feature construction.
2. “[Feature Inspection](#)” on page 500 offers details on how to perform an inspection with the inspection tab, and options for the measurement process.
3. “[Workflow Reference Guide for Points based Inspection](#)” on page 511 offers a set of suggested workflow options for basic inspection tasks.

Building GR-Features

Geometry Relationships (GR-Features) are built either through the **Relationships** tab of toolkit menu (and Relationship menu) or by using the **Features** page on the Ribbon Bar. The Ribbon Bar is the most advanced tool for this purpose and will be discussed here ([Figure 25-1](#)).

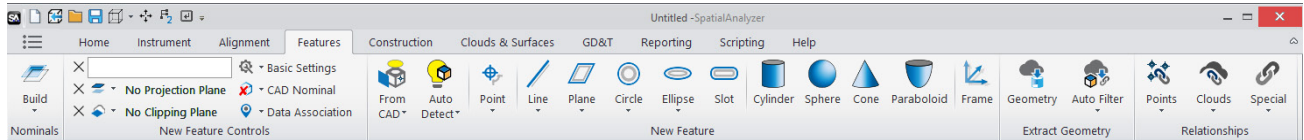


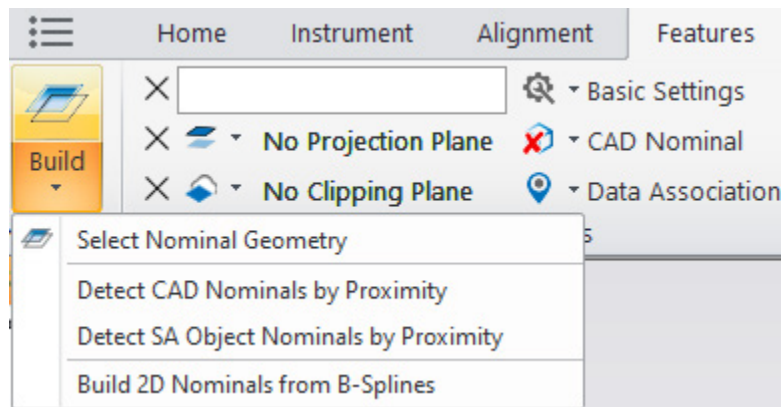
Figure 25-1. Features Page of the ribbon bar

While GR-Features) are fully configurable after they are built, there are a number of tools that can be used in the **Features** page to configure features so that you get the desired settings applied when they the GR-Features are built.

Nominals

The Build Nominals section of the Feature tab provides an easy way to convert existing nominal geometry into GR-features and extract or detect nominals from fit features with a proximity to the nominals (Figure 25-2).

Figure 25-2. Build Nominals options.



Select Nominal Geometries

This function takes the specified geometry and converts them into geometry relationships such that they can be used for trapping and direct comparison with a measured feature.

Detect Nominals

Used to quick gather nominal features based on proximity.

- **Detect CAD Nominals by Proximity.** This function looks at a CAD model, extracts the nominal features from it and associates them with the measured features.
- **Detect SA Object Nominals by Proximity.** This function looks for existing objects and both sets them as the nominal object for the geometry relationship and renames the measured feature using the nominal names.

The Detect Nominal options for both CAD and SA Objects only con-

sider the visually displayed features. If you hide parts of the CAD model or SA Objects they will not be considered for association. This provides the user the ability to control which objects are detected.

Build 2D Nominals from B-Splines

2D Compare to Nominal geometry relationships such as lines and circles can be build directly from B-splines with this tool. Selection of a spline provides a nominal reference for the new GR-Feature.

New Feature Controls

Feature Name Field

A **Name** field allows you to enter a specific name for a relationship prior to building it. This speeds construction by allowing you to name features in advance. If this field is left blank, then the default geometry type name will be used.

Defining Projection Planes

The projection plane setting (Figure 25-3) options provide simple on/off control in order to switch from 3D to 2D projected geometry. The projection plan can be set as follows:

- **None.** Does not use a projection plane for the feature.
- **Selected Plane.** You choose a specific plane which will be used as a projection plane. When this option is checked you will be prompted to choose the desired planar feature. The name of the select plane will then be shown.
- **Plane for Each Feature.** When selected, each additional feature added to the inspection routine will also have a projection plane feature added, so that you first measure the projection plane, then the feature to project to it.

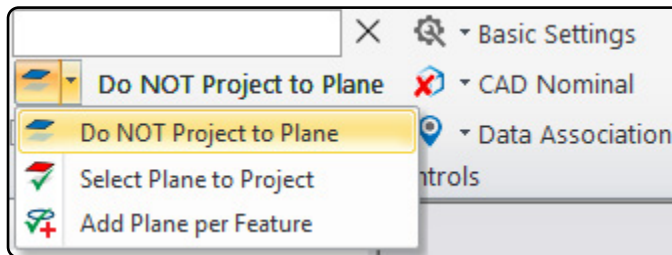


Figure 25-3. Projection Plane and feature name controls

Basic Settings

A selection of basic settings can be applied to newly built features as part of the Basic Settings tool. This function allows you to define:

- **Auto Vectors.** This option allows you to build Fit vectors (AVF) by default with newly created features.
- **Cardinal Points.** Enabling this option will enable Cardinal

points by default for newly created features.

- **Desired Measurement Count.** The specified measurement count will be applied to the created feature.
- **Measurement Trapping.** The **Repeat Last Feature for Inspection** option can be used for repeated feature measurements. A new feature of the same type will be built each time you press next in the inspection tab, or the desired measurement count is reached.

This is particularly useful if you have to measure a large number of similar features, such as 100 holes. Simply enable the Repeat option, and then define the first hole. As you advance past the first hole a new circle features will be created for you automatically.

CAD Nominal

The CAD Nominal button provides a toggle to choose between nominal selection options when a feature is built (Figure 25-4). This selection then directly influences the operation of the feature buttons when a new feature is defined:

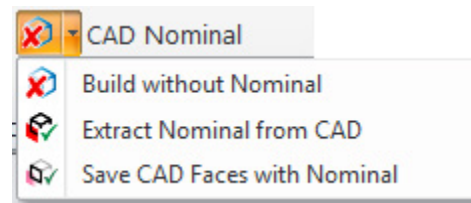


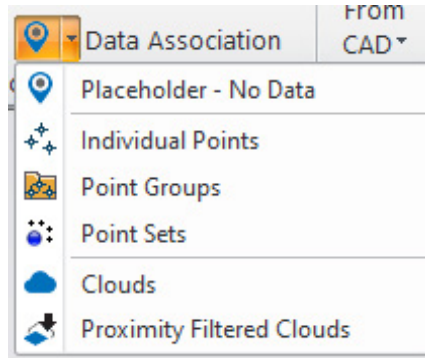
Figure 25-4. Nominal selection and CAD interaction controls.

- **Build without Nominal.** This setting is to be used without CAD and builds a basic fit only feature without pre-assigning a nominal feature.
- **Extract Nominal from CAD.** With this option selected you will be prompted to select the nominal feature by clicking on the desired surface face. The newly built nominal feature will be assigned to the new feature.
- **Save CAD Faces with Nominal.** It can be helpful to save the selected CAD faces when building nominal features from CAD. When you do so, these faces will highlight along with the highlighted relationship and they can then be used during auto filter operations.

Data Association

In many cases a feature may be built after data has already been acquired. The Data Association toggle button provides an option to be directly prompted to select a certain data type as features are built. The available options are shown below (Figure 25-5):

Figure 25-5. Data Association option



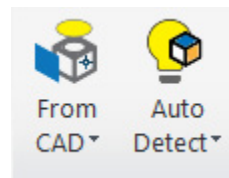
- **Placeholder - No Data.** This is the default setting and is used when data has not yet become available to build a placeholder relationship which is ready to be measured.
- **Individual Points.** With this option selected you will be prompted to select points when a new feature is built.
- **Point Groups.** With this option selected you will be prompted to select point groups for association with the newly created feature.
- **Point Sets.** With this option selected you will be prompted to select point sets for association with the newly created feature.
- **Clouds.** This option allows selection of entire clouds for use in a feature.
- **Proximity Filtered Clouds.** This option is only available with features that have a nominal pre-defined (or defined as part of the newly built feature). It will prompt you to select a cloud that is then filtered by proximity to the nominal feature, creating a new sub-cloud that is associated with the relationship.

Building New Features

Adaptive Features

The first two options in the feature tab are adaptive ([Figure 25-6](#)).

Figure 25-6. Adaptive Feature construction tools



- **From CAD.** The from CAD option provides a means to construct features of any type directly from CAD by selecting a CAD face.
- **Auto Detect.** The auto detect option detects the feature type that fits the input data type within a required RMS (see [“Auto Detect Feature”](#) on page 483).

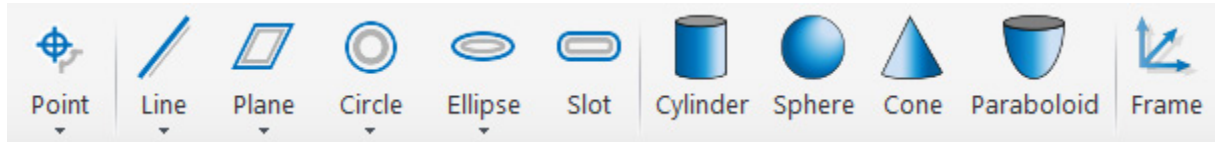


Figure 25-7. New Features

Point

Creates a GR-Point feature that dynamically builds a point from the average of all associated points. The points can be projected to a plane before averaging.

The following additional Intersection Points can also be created:

- **Line and Plane Intersection.** Computes a point from the intersection of a line and a plane.
- **Cone and Plane Intersection.** Computes a point from the intersection of a cone's central axis and a plane.
- **Cylinder and Plane Intersection.** Computes a point from the intersection of a cylinder's axis and a plane.
- **Three Plane Intersection.** Computes the intersection point from 3 roughly orthogonal planes.
- **Mid-point of perpendicular to two lines.** Builds a point at the center of a line constructed perpendicular to two selected lines. Also called a mutually perpendicular line, this defines the closest point to two lines that nearly cross each other.

Line

Creates a GR-Line feature that dynamically builds a line from the associated points.

The following additional Intersection and axis Lines can also be created:

- **Cone Axis.** Computes and builds the axis line of a selected cone.
- **Cylinder Axis.** Computes and builds the axis line of a selected cylinder.
- **Two Plane Intersection.** Computes the intersection line between two planes.
- **Bisect Two Lines.** Builds the bisection line or average line defined by two selected lines.
- **Slot Centerline Along Length.** Computes and builds the the central axis of a slot.

Plane

Creates a GR-Plane feature that dynamically builds a plane from the associated points.

- **Prepare to Projection Plane.** This option both builds a GR-Plane feature and sets the measured plane from it as the active projection plane as part of the build process.
- **Prepare to Clipping Plane.** This option both builds a GR-Plane feature and sets the cloud clipping plane from it as the active clipping plane as part of the build process.

The following additional dynamic planes can also be created, “[Dynamic Geometry](#)” on page 484 provides greater detail on the construction methods:

- **Offset Plane.** Builds a dynamic plane that is offset by a user selected amount from a reference plane (or the XY plane of a reference object). The plane then dynamically maintains that offset distance.
- **Bisect Two Planes.** Computes and builds a plane that is the bisection or average of the two selected planes.
- **Two Cone Intersection Options.** Computes and builds a plane from the intersection of two cones.
- **Cone and Cylinder Intersection Options.** Computes and builds a plane from the intersection of a cone and a cylinder.

Circle

Creates a GR-Circle feature that dynamically builds a circle from the associated points.

Single point circles allow you to lay a probe in a hole (assuming the hole diameter is smaller than the probe diameter) and create a circle with a single measurement, based on how far the probe sinks into the hole.

To create a single point circle geometry relationship, create a circle relationship as usual, and enable a projection plane. If the relationship has exactly one point, a single point circle will be created automatically. If you continue measuring additional points, a regular circle relationship will be created instead. Note that the measured point must be on the positive side of the projection plane.

The following additional Intersection circles can also be created, “[Dynamic Geometry](#)” on page 484 provides greater detail on the construction methods:

- **Sphere and Plane Intersection.** Computes and builds a circle

from the intersection of a sphere and a crossing plane.

- **Cylinder and Plane Intersection.** Builds a circle from the intersection of a cylinder and a plane based on the selected method.
- **Two Cone Intersection.** Builds a circle from the intersection of two cones.
- **Cone and Cylinder Intersection.** Builds a circle based on the intersection of a cone and a cylinder
- **Cone and Plane Intersection.** Builds a circle based upon the intersection of a cone and a plane.

Slot

Creates a GR-Slot feature that dynamically builds a slot from the associated points. Selection between square and rounded ends are available from within the slot fit settings.

Ellipse

Creates an GR-Ellipse feature that dynamically builds an Ellipse from the associated points. Feature extraction from CAD is not available for this feature type.

The following additional Intersection ellipses can also be created, “[Dynamic Geometry](#)” on page 484 provides greater detail on the construction methods:

- **Cylinder and Plane intersection.** Computes and builds an ellipse at the intersection of a cylinder and a plane.
- **Cone and Plane Intersection.** Computes and builds an ellipse from the intersection of a cone and a plane.

Sphere

Creates an GR-Sphere feature that dynamically builds an Sphere from the associated points.

Single point sphere allow you to build a sphere using a single point measurement. To create a single point sphere GR-feature, create a sphere relationship as usual, and either set a locked radius or set a nominal sphere. The size will be established using these settings in this order of priority.

Cylinder

Creates an GR-Cylinder feature that dynamically builds an Cylinder from the associated points.

Cone

Creates an GR-Cone feature that dynamically builds an Cone from the associated points.

Paraboloid

Creates an GR-Paraboloid feature that dynamically builds an Paraboloid from the associated points. Feature extraction from CAD is not available for this feature type.

Frame

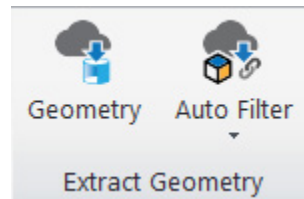
Creates an GR-Frame feature that dynamically builds an Frame from the associated objects.

The Frame Wizard relationship provides the same controls as the classic frame wizard utility. It still provides object and point selection to define a frame but it continually monitors those objects so that if any of the input features change so does the computed frame. In addition it provides a dynamic comparison to a nominal from in with a similar look and feel to other Geometry Relationships.

Extract Geometry

In addition to direct data association there are several direct extract and proximity filter options available (Figure 25-8).

Figure 25-8. Extract GEometry Section of the Features Tab



Geometry

This function is used to extract geometry from a point cloud without any nominal reference. It uses a click point as a starting point to search through the cloud and identify features.

For more information on feature extraction and auto filter functions refer to “[Feature Extraction from Point Clouds](#)” on page 587.

Auto Filter to Nominal

These functions provide a means to gather data within proximity to the nominal features and associate that data with the relationship.

- **Auto Filter to Nominal 3D.** Used to detect either point or cloud data within proximity to geometry (see “[Auto Filter to Nominal Geometry >3D Extraction](#)” on page 587).
- **Auto Filter to Nominal 2D.** Used to detect holes in surface scan data (see “[Auto Filter to Nominal Geometry >2D Geom-](#)”).

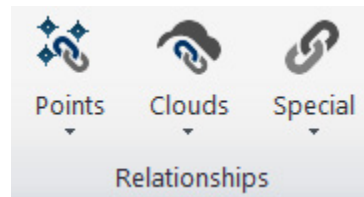
etry” on page 597).

- **Auto Filter to Nominal Line.** Used to detect cloud points within a specified proximity to a line.

Relationships

The Relationships section of the Features tab provides buttons to build relations of types other than geometry relationships (Figure 25-9). These include the following:

Figure 25-9. Relationships Section of the Features Tab.



Points

- **Point to Point.** Creates a point to point relationship used to link two specific points together.
- **Points to Points.** Like a Group to Group relationship, it provides a comparison between commonly named points. Its unique in that it gives the user the flexibility to select points from a combination of groups both as reference and as actual.
- **Point to Object.** Creates a point to Object relationship used to defined the placement of a single point relative to an object.
- **Points to Objects.** Creates a Points to Objects relationship used to compare a set of selected points to a set of objects.
- **Points to CAD Faces.** Creates a Points to Objects relationship where the specific faces of a selected CAD model are used for the comparison.
- **Groups to Objects.** Creates a Groups to Objects relationship used like a points to objects relationship but without the need for selection of individual points. All the points in the selected groups will be used.
- **Groups to CAD Faces.** Creates a Groups to Objects relationship where the points in the selected point groups are compared to a set of selected surfaces faces from a CAD model.
- **Group to Group.** Creates a group to group relationship used to compare the commonly named points within the two selected groups.
- **Group to Nominal Group.** A Group to Nominal Group relationship functions like a group to group relationship but with

some added features for inspection. When trapping begins a set of bull's-eyes are displayed on the points contained within the nominal group. When a point is measured, it will automatically be renamed to the closest nominal point. And the bull's-eye removed.

Clouds

- **Cloud to Objects.** Creates a clouds to objects relationship used to compare the cloud points from the selected point clouds to the selected objects.
- **Clouds to CAD Faces.** Creates a clouds to objects relationship used to compare to a selection of CAD faces.
- **Cloud to Swatch.** Builds a swatch relationship used for surface point analysis. It then builds a mini cloud using only cloud points within a proximity to the reference point and a surface normal vector. The resulting computed point uses the average distance values in the cloud swatch to compute the point offset.

Special

- **Frame to Frame.** Creates a simple frame to frame relationship used to compare two selected frames.
- **Object to Object Angle.** Creates an objects to objects direction relationship used to compare the angle between the Z-axes of the selected objects.
- **Vector Group to Vector Group.** Creates a relationship that computes a delta vector group defining the difference between two reference vector groups, for more information refer to "[Vector Group to Vector Group Relationships](#)" on page 431.
- **Pipe Fitting.** Creates a pipe relationship used to compute a cut plane, for more information refer to "[Move Collections by Minimizing Relationships](#)" on page 454.
- **Inspection MP Execution Node.** Used to incorporate an embedded MP into an inspection process such that iteration through the feature list will trigger the MP execution.

Feature Inspection

Feature Inspection is most effective through the use of either the **Inspection Bar** menu or **Inspection** tab of the **Toolkit**. These inspection panels provide a focused view of the features to measure in the tree and provide easy navigation controls for effective workflow, without having to navigate the full tree.

To access the **Inspection Bar** menu from within the Ribbon Menu, navigate to the Home page and press the Inspection Bar button in the inspection Controls section (Figure 25-10).

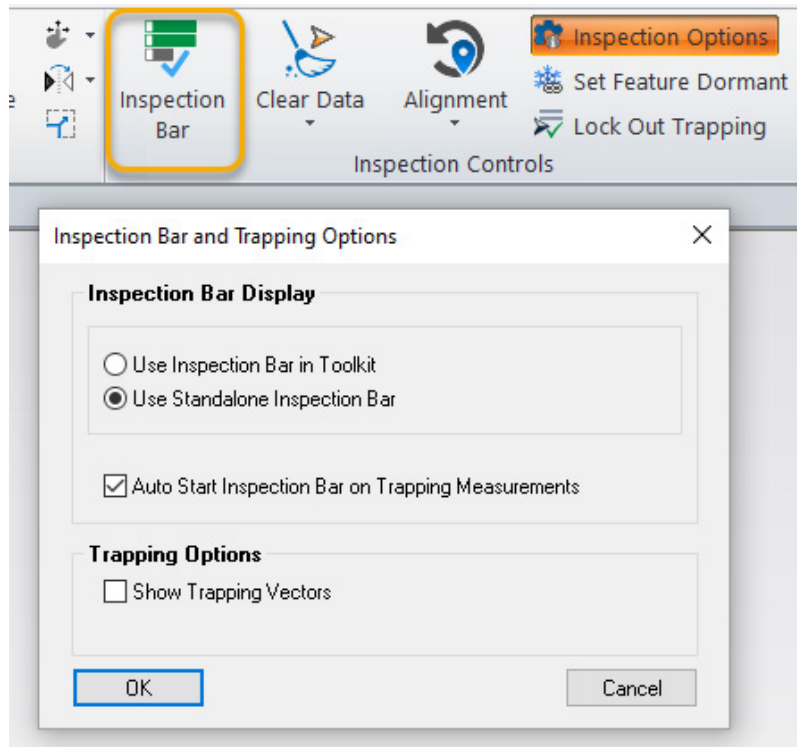


Figure 25-10. Inspection Controls and Inspection Bar access button from the Home Page.

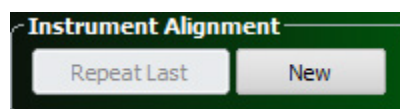
There are a selection of Inspection Options available.

- **Inspection Bar Display.** The Inspection Bar button will open either the Stand alone Inspection Bar or the Toolkit Inspection bar depending on the user selection.
- **Auto Start Inspection Bar on Trapping Measurements.** With this option enabled the selected inspection bar will open automatically when trapping begins.
- **Show Trapping Vectors.** Enabling this option will automatically display fit vectors for the active trapping item (see “[Enable Trapping Auto Vectors](#)” on page 507).

Instrument Alignment

Convenient access to various alignment options are available under the **Alignment** button (Figure 25-10) as well as at the top of the Toolkit Inspection tab (Figure 25-11).

Figure 25-11. Instrument Alignment section of the Inspection tab.



Repeat Last Alignment

Executes the last repeatable alignment used in the job file. Repeatable alignments include:

- Measure Nominal Points
- Quick Align to CAD
- Frame Wizard
- Nominal Geometry Relationships

This **Repeat Last** function can only be used after an alignment has already been performed. Therefore it is intended as a benefit to those looking to either align a second instrument to the same features or as a way to build a template file that can be used again and again.


New

The **New** alignment button provides that same list of alignment functions available by right-clicking on the instrument and selecting **Locate** from the menu. This button is only accessible if a live instrument is connected within the job file. If you wish to set up an alignment for repeated use as a template file you will need to first connect to the instrument in simulation and perform the alignment. This alignment will then be available for use with the **Repeat Last** function.

Measurement Trapping

The **Measurement Trapping** page (Figure 25-12) provides a convenient list of all trappable features in the job file. This list can be rearranged and edited as needed to provide a list of features to inspect. Trappable features include:

- Geometry Relationships or GR Features
- Standard Relationships
- GD&T Datums
- GD&T Feature Checks

Trapping simply means direct data association. When you are actively trapping to a particular feature it will be marked as active . This indicates that each measurement taken is automatically associated with that active feature.

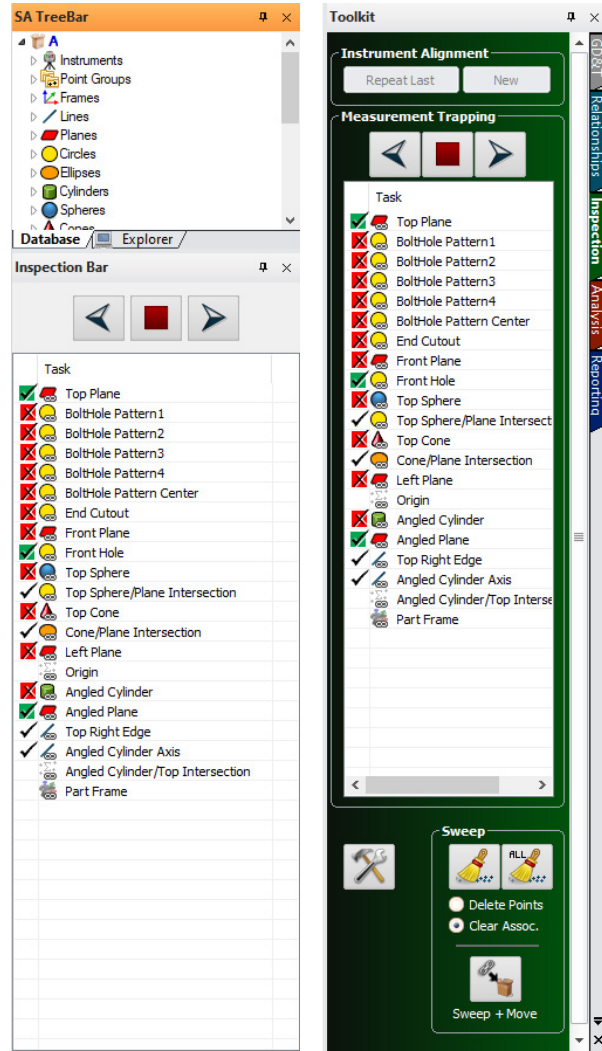
There are currently two separate inspection toolbars that can be displayed (Figure 25-12).

When trapping begins, the default inspection bar embedded in the tree will be displayed. The stand alone inspection bar provides a larger list area and a cleaner look and was added to work with the ribbon menu and its built in tools.

The Inspection tab of the Toolkit can still be shown using the Inspec-

tion Bar button on the Home page. The Inspection toolkit tab provides more controls for things such as Instrument Alignment, Feature properties and Sweep functions.

Figure 25-12. Measurement Trapping can be displayed either within the automatic inspection Bar or within the Toolkit Inspection Tab



The navigation controls provide a means to navigate through a list of features:

Figure 25-13. Inspection Trapping Controls



➤ The Next button advances trapping. Ctrl+Right click provides a means to move a feature down in the measurement task list. A simple Right click will open the Multi-Feature Cloud Association option and access to Simultaneous Feature Trapping.

■ The Stop button halts feature trapping.



The Previous button moves trapping up to the prior feature in the list. Ctr+Right click provides a means to move a feature up a slot in the measurement task list.

In addition to directly trapping measurements taken from and instrument relationships provide the ability to trap select points already in the job file simply by clicking on those points. To start **Trap Selection** right-click on the relationship in either the tree or the inspection tab and select Trap Selections.

Simultaneous Trapping

Multiple features can be selected at one time when scanning and used as scanning targets. This process is initiated with a Right-click on the Next button in the Inspection Bar ([Figure 25-14](#)).

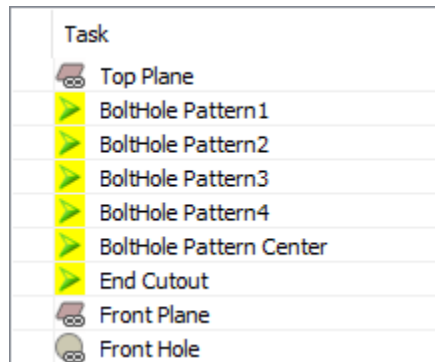


Figure 25-14. Simultaneous Trapping function in action

Simultaneous trapping is very helpful when trying to obtain data on specific features. It pre-filters the incoming scan data while measuring such that data is only recorded within the proximity limits set for each feature (see [“Working with Cloud Data”](#) on page 479). Once trapping is stopped, the cloud is divided and automatically associated with each feature.

Batch Proximity Cloud Association

The multiple features selection process (right-clicking on the Next button in the Inspection tab) can also be used when a live instrument is not connected. This same selection option offers the ability to associate a cloud with a group of selected feature for processing using the proximity cloud association option.

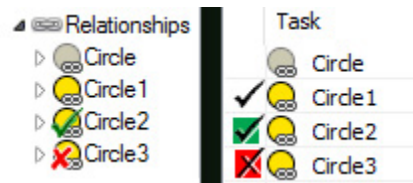
Inspection Status Icons

The status of features in the toolkit can vary depending of if tolerances are set or if measurements have been made of particular features. To more clearly identify the status of these features different icons are used in both the tree and in the Inspection Tab of the Toolkit. This includes 4 states:

4. Unmeasured

5. Measured
6. Tolerance set - Pass
7. Tolerance set - Fail (or mixed status)

Figure 25-15. Status Icons showing that status of a circle



The status icons are displayed slightly differently in the tree and the Task bar (Figure 25-15). In the tree the icon is condensed by placing the check mark on top of the feature icon. In the task bar these are separated out to make it easier to see that status.

Measurement Trapping and Navigation

To begin trapping to any individual feature in the inspection tab you can double-click on it. As long as there is at least 1 live instrument trapping will begin.

At the top of the Measurement Trapping section are a set of navigation buttons. When actively trapping to a particular feature, these buttons provide a convenient means to move to the next or previous feature in the list or stop trapping.

Reordering Features in the Inspection Tab

By right-clicking in the navigation buttons at the top of the Measurement section you can change the order in the list of the currently selected feature. Moving that item either forward or backward in the list. The inspection task list is a direct reflection of the tree. By changing the order of the features in the inspection task list you are also re-order the entries in the tree within their respective category. Likewise by reordering the entries in the tree you will also change the order of the entries in the inspection list.

Lock/Unlock Trapping

The availability or display of any feature in the Inspection tab can be controlled on an individual or feature type basis. To exclude or include features in the inspection tab right-click on that feature in the tree and turn on or off **Lock Out Trapping**. Likewise, by right-clicking on the Relationship, Datum, or Feature Check categories you can select **Lock/Unlock Trapping** and control the display of multiple features from that category at one time.

Helpful Tools for Inspection

Measurement Counts

Measurement Counts are a property of any feature within the inspec-

tion tab. They are used to set the desired number of measurements for a feature. When trapping to that feature you can measure until the desired count is reached and then the trapping will be automatically iterated to the next feature in the inspection list.

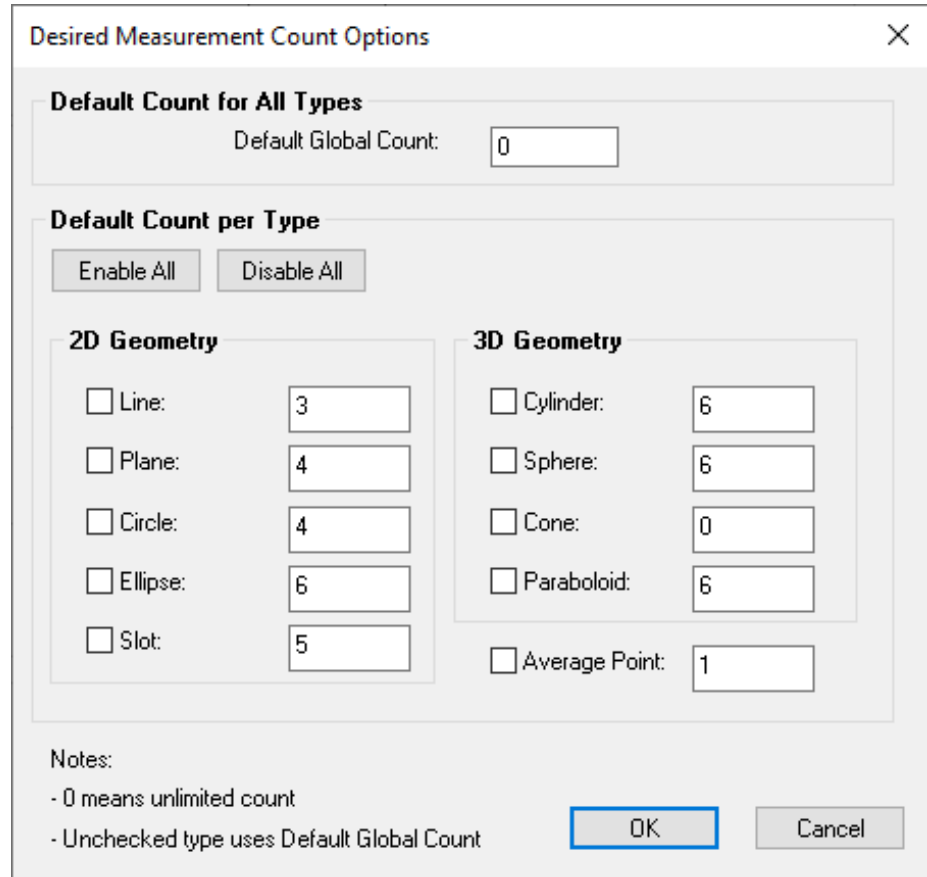


Figure 25-16. Desired Measurement Counts.

Measurement Counts are off by default (when 0), but global default controls (Figure 25-16) can be set for all features within the **User Option > Analysis Tab** or within the properties of each Feature individually.

Associated Inspection Callouts

Callout views can be directly linked to relationships for use in inspection. Once linked when you start trapping to that particular relationship the associated callout view is activated. This opens up some very powerful scripting tools. Activating a callout can:

- Set the viewpoint and render mode
- Set the working frame
- Set the visibility layer (such that only objects of interest are shown)
- And display any individual callouts you wish to display.

To associate a callout with a relationship you can do either of the following:

1. Right-click on the relationship in either the tree or the inspection task list and choose **Create Associated Inspection Callout**. This will build a new Callout View with the particular relationships callout added to it. Edit the properties of the new Callout as desired.
2. Go to and existing callout and use **Add Relationship**. Each relationship added to callout will then be linked to that callout view.

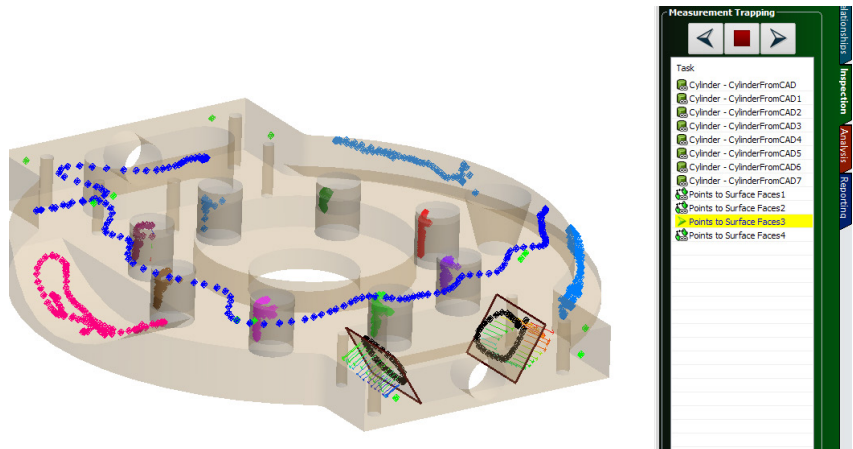
Show Only Selected Relationship

Within the right-click relationship category is an option to **show only the selected relationship**. This option hides all relationships in the graphics except for the one currently selected in either the tree or with trapping focus in the inspection bar. When used with trapping this can make the inspection process much clearer.

Enable Trapping Auto Vectors

Within the relationship category is an option to turn on/off trapping auto vectors (Figure 25-17). This function automatically turns on an auto vector group for the for the feature that is actively being trapped to. It builds a vector group called “Trap Vectors: [Relationship Name]”.

Figure 25-17. Enable Trapping Auto Vectors displays vector deviations for the currently feature.



- Trapping vectors display the point to Measured Object (AVF) vectors for geometry relationships or the point to object vectors.
- When the toolkit is advance to the next feature in the inspection tab this vector group is automatically replaced by a new one representing the current feature. When trapping stops the vector group is then deleted.
- Should an auto vector group already exist for a particular relationship in the tree it is updated and then hidden when you move to the next feature.

Set/Clear Relationship Dormant Status

There are times such as when performing a relationship optimization process or working in large job file when it would be helpful to postpone the recomputing process inherent in a relationship. This dormant status is used for that purpose. Any relationship marked as dormant will stop updating until its status is changed.

Feature Trapping Controls and Automation

Within the right-click menu for a relationship there is an option *Trapping Controls*. This control provides full access to all the operations that can be performed when trapping begins for a specific feature.

These controls include:

- **Lock Out Trapping.** Which hides the feature from the inspection process completely
- **Use Inspection Callout View.** Offers the ability to link a callout view with a feature such that the callout is displayed when trapping begins.
- **Link Embedded MP Script.** If greater control is needed, an embedded MP script can also be triggered to run as part of the trapping process.

Taking Measurements Automatically

A user has the ability to configure an instrument automatically when trapping begins for that feature (Figure 25-18).

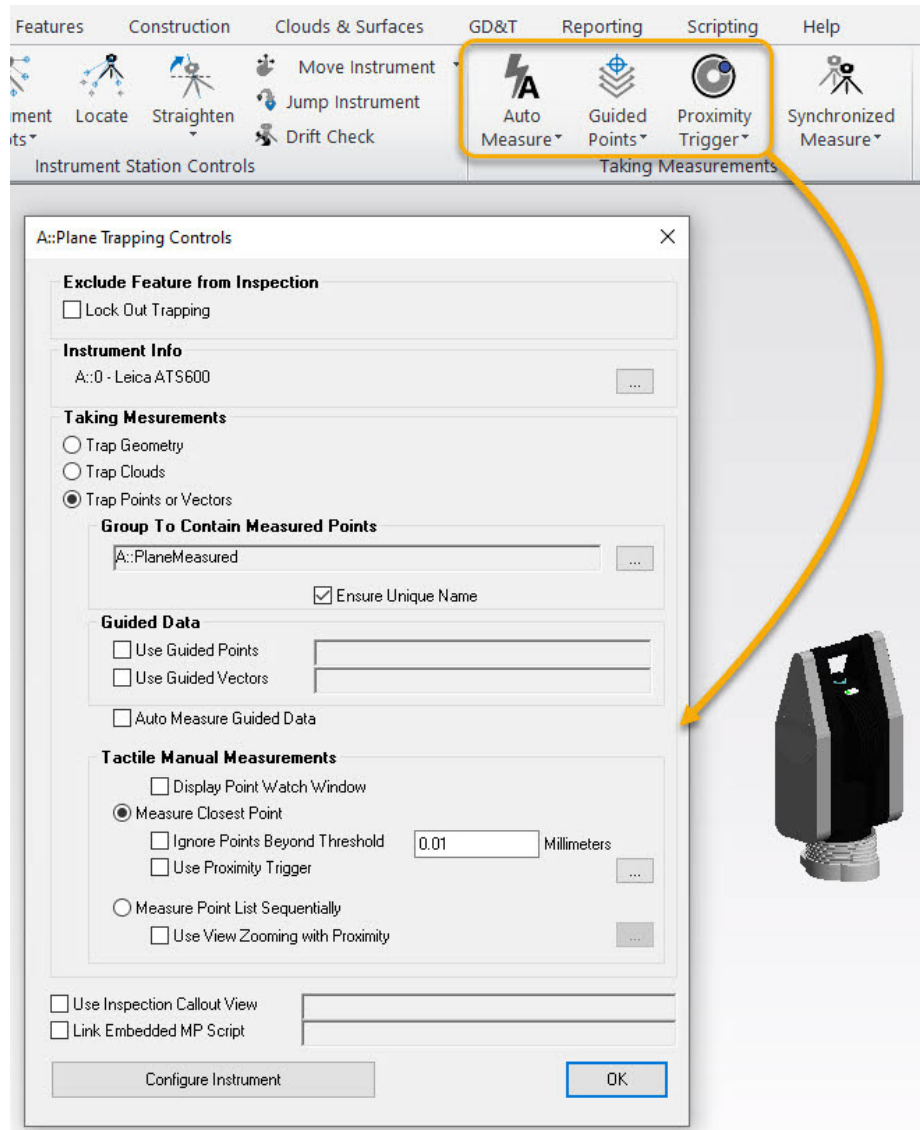


Figure 25-18. Trapping Controls Dialog

Select the features you would like to enable as part of the measurement process.

- **Instrument Selection.** Many operations are instrument type specific so the instrument to be used should be selected first.
- **Group to Contain Measured Points.** This control defines where measured points will be saved.
- **Guide data.** Choose either points or vectors to guide the measurement operation.
- **Auto Measure Guide Data.** This check box will cause the instrument to perform an auto measure using the selected guide data, using either auto measure points or surface vector intersections.
- **Display Point Watch Window.** This open will open a closest point watch window when guide points are selected.

- **Ignore Points Beyond Threshold.** This option compress the measurements to the nominal reference points and will prevent association of points that are further than the specified threshold from the guide points.
- **Proximity Triggers.** Proximity Triggers can be used as part of a feature measurement (see “[Proximity Triggered Points](#)” on page 325).
- **Measure Point List Sequentially.** This option allows you to walk through the list of guide points in order.
- **View Zooming with Proximity.** Offers guidance to a particular point using the Measure Guide Points Option (see “[Guided Points](#)” on page 324).

Configure Instrument

The settings available will depend on the instrument type selected. For example, a laser tracker can have both its measurement profile and measurement target set, and a even the increment of a spatial scan can be set ([Figure 25-19](#)).

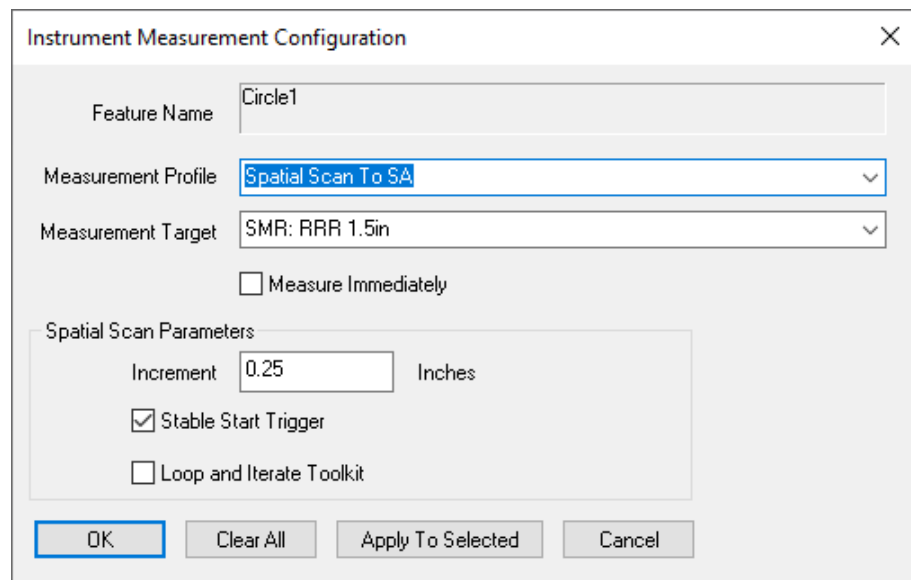


Figure 25-19. Configure Instrument Dialog.

The Configure Instrument control is also directly tied to the *Auto Measure >Batch* of Features command (see “[Measure Batch of Features](#)” on page 323).

Toolkit/Toolbar Keyboard Shortcuts

Keyboard shortcuts and T-Probe/4-button remote functions have been added for the SA Toolkit and Instrument Toolbars:

- F6 (‘A’ button). Toggle measurement in the instrument toolbar.
- F7 (‘B’ button). Go to the next feature in the toolkit.
- Ctrl + F7. Go to the previous feature in the toolkit.

- F8 ('C' button). Remove last point in the toolbar.
- Ctrl + F8. Stop trapping in the toolkit.
- F9 ('D' button). Iterate the measurement mode in the toolbar.
- F10. Iterate the active target in the toolbar.
- F12. Find the reflector (target search) in the toolbar

Workflow Reference Guide for Points based Inspection

The following suggested workflow options for inspection are available:

- [“How to Measure Features \(Auto Detect\) Using a Probing Device.”](#) on page 511
- [“Building Geometry from Existing Measurements.”](#) on page 514
- [“Building Center Points and Intersections from Existing Geometry.”](#) on page 516
- [“How to build a coordinate system using measured features”](#) on page 521.
- [“How to measure a set of reference points.”](#) on page 524

How to Measure Features (Auto Detect) Using a Probing Device.

SA can automatically detect what feature type you are measuring using the points you measure. To begin:

1. Connect your instrument.
2. Navigate to the Features Tab of the Ribbon, enter a name for your new feature and then add an **Auto Detect Feature** (Figure 25-8).

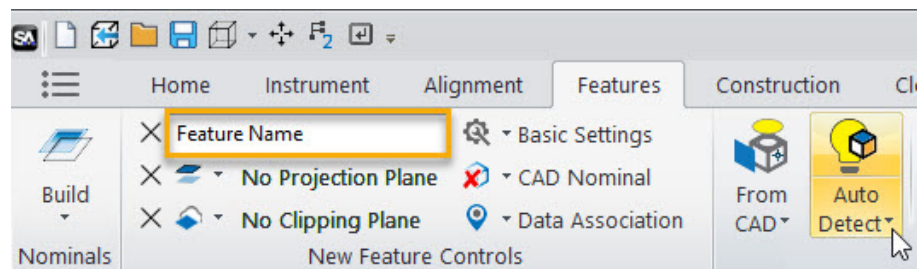
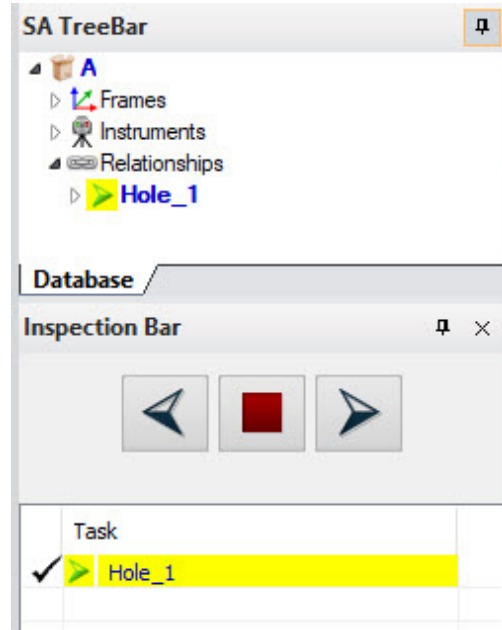



Figure 25-20. Auto Detect Feature

When you add the **Aut Detect** feature to the job, trapping will begin like this (Figure 25-11). You will see trapping indicated both in the SA Tree bar and in a popup Inspection Bar, which also gives you navigation controls. Settings for this behavior are on the Home tab of the ribbon under **Inspection Options**.

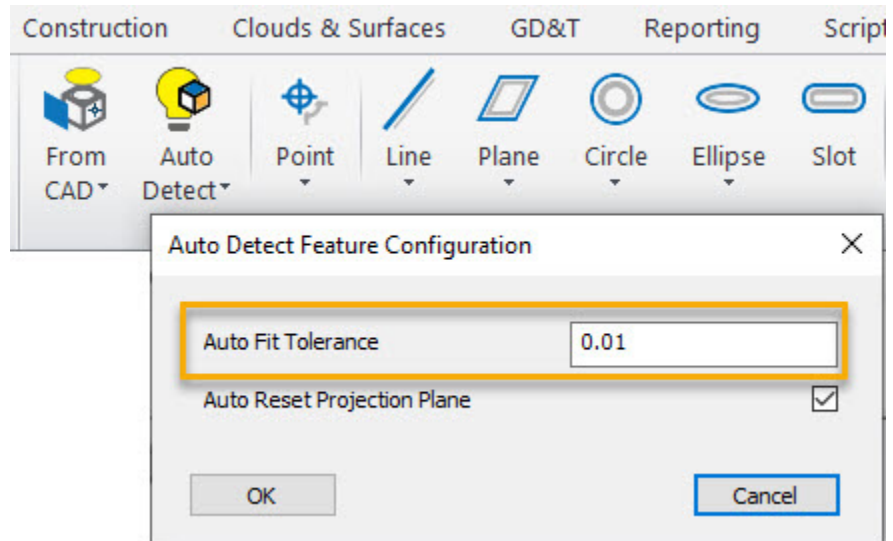
Figure 25-21. Trapping has begun



3. When trapping is active measurements will be captured and used by the selected feature. When you have measured all the points you want for this feature press the next button 

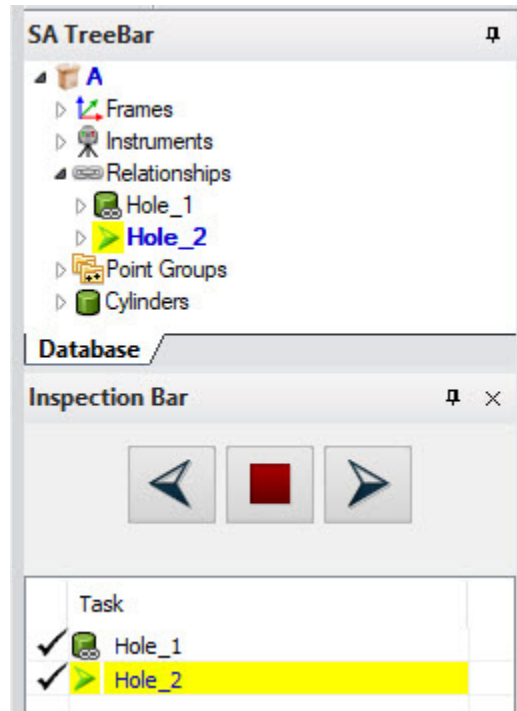
When you press next SA will cycle through the available fit options and settle on the simplest feature it can find that fits with an RMS (root mean squared deviation) under the Auto Fit Tolerance you have set in the **Auto Detect** feature settings (Figure 25-12).

Figure 25-22. AutoDetect feature Settings



A new placeholder feature will also be added for you so that you can continue to measure (Figure 25-67). An audible beep will sound when this new feature has been added and is ready to be measured (which can be helpful when the screen is hard to see).

Figure 25-23. Measuring the next feature

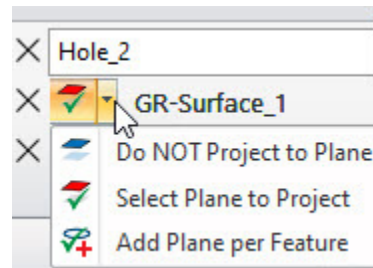


Continue to measure pressing **Next** after each feature has been measured.

Troubleshooting/Additional Details

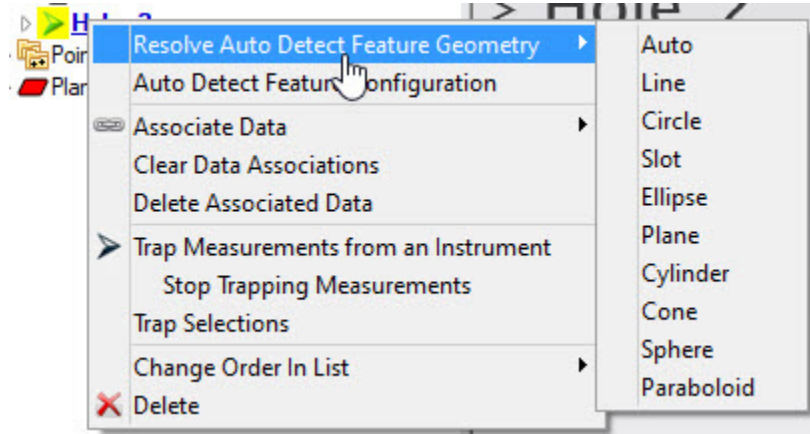
If you measure a plane with the **Auto Reset Projection Plane** option enabled (Figure 25-12) then the newly measured plane will become the active projection plane. This can be seen in the Features Tab and can be adjusted from there if need be (Figure 25-68).

Figure 25-24. Projection Plane Control



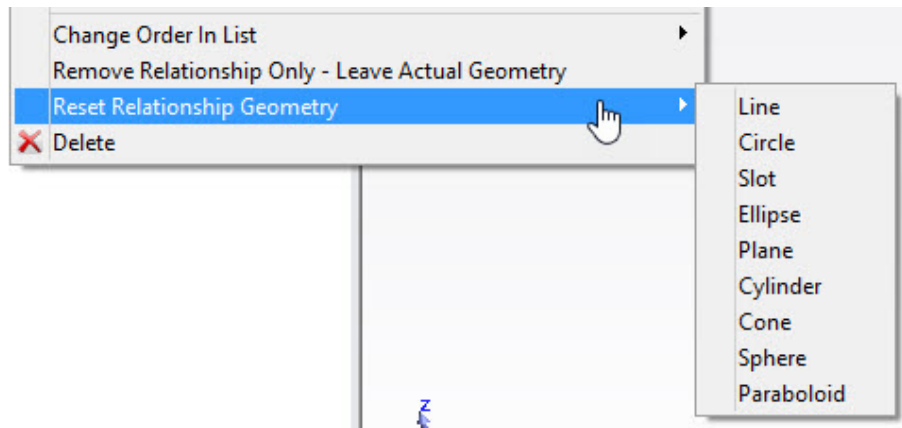
Should the Auto Detect Feature fail to resolve a fit within the RMS tolerance you have set it can be manually assigned through the right-click menu (Figure 25-69).

Figure 25-25. Assigning the Feature type within an Auto Detect Feature



Alternatively if a solution was found but it was not the desired geometry then the newly built geometry's type can be changed through its right-click menu .

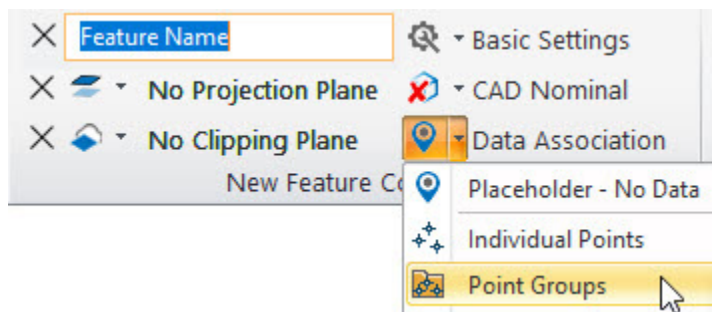
Figure 25-26. Correcting the geometry type.



Building Geometry from Existing Measurements.

1. Navigate to the Features Tab of the Ribbon and enter a name for your new feature in the Feature Name entry field.
2. Choose a selection mode (points, or groups) from the drop down list(Figure 25-71).

Figure 25-27. Preparing to build a feature



If you would like to build a 2D feature like a circle or slot projected to a plane, the projection plane should also be set (this is not neces-

sary when building geometry like planes and cylinders). A projection plane can be selected from the drop down list under the feature name field, if it already exists, or you can build a new one first by selecting that option from the Plane's drop down list (Figure 25-73).

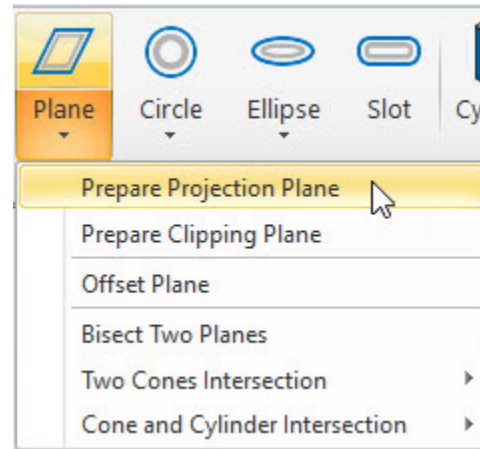


Figure 25-28. Prepare Projection Plane

3. Select a feature type to build. This new feature will take the name you enter and then prompt you to select. You can select a feature type directly or use the Auto-Detect feature to have SA figure out a desired feature type (Figure 25-70).

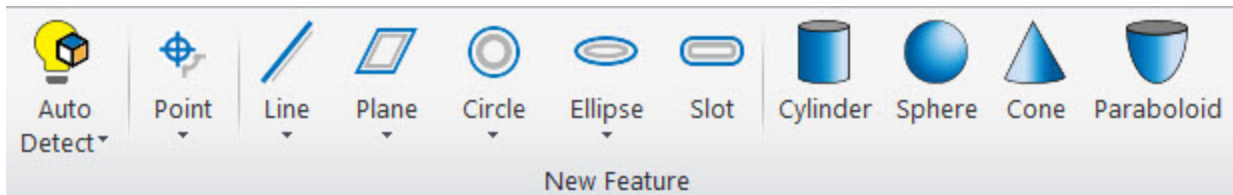


Figure 25-29. Features list you can build

SA expects you to define the feature you want to build before you begin selecting anything. When you add the feature by clicking the desired feature button, a feature of the type you select will be added to the tree using the name you entered. It will then prompt you to select either Individual Points or Point Groups (all the points in a group) based upon your selection in the drop down (Figure 25-71).

Selection can be performed in multiple ways:

- You can single click on items in the graphics
- You can hold down the Shift key + Left click and hold with the mouse button to drag a dotted rectangle around the items you want to select.
- You can also select items in the Tree Bar Select Items panel.

Troubleshooting/Additional Details

If you forget to enter a name, the feature can be renamed through its

properties. These can be accessed by double clicking on it the tree or right-clicking and selecting properties. Then the **Rename** button.

If you forget to set the Data Association drop down menu the feature will be created but you will not be prompted to select anything. To fix this you can right-click on the placeholder geometry and select **Associate Data** (Figure 25-72).

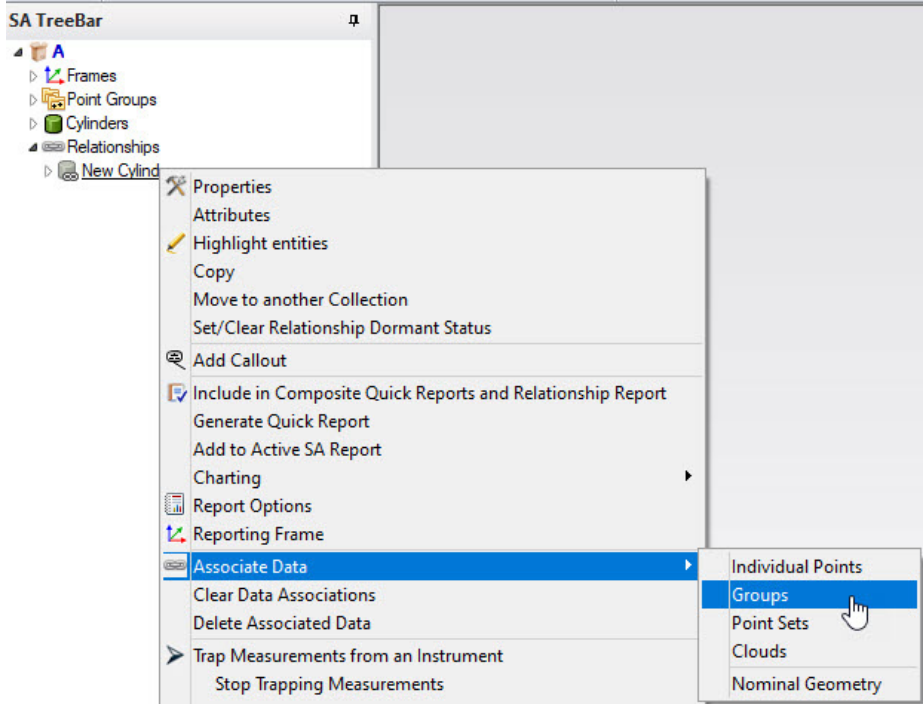


Figure 25-30. Direct Data Association.

Building Center Points and Intersections from Existing Geometry.

Lets say you have a patter of 4 holes you have already measured. You would now like to fit geometry, build the center points from each hole, and determine the diameter of a circle fit through those center points (Figure 25-74). The following are the steps to do so:

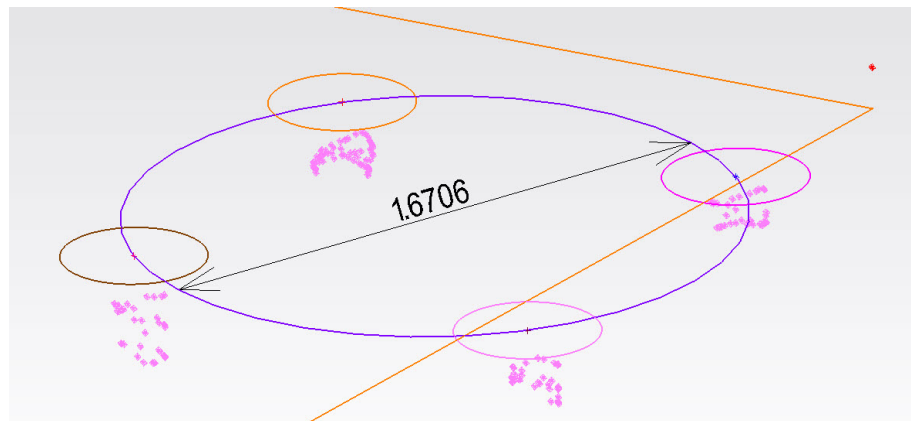


Figure 25-31. Pattern of circle with a circle fit through the center points.

1. Build a projection plane. Circles and slots are typically mea-

sured by placing a probe inside the hole and measuring the wall of the feature. Therefore the first step is to build a projection plane that will define the plane for the circles. To do so, first enter a feature name, select the data to use, and add a feature like this:

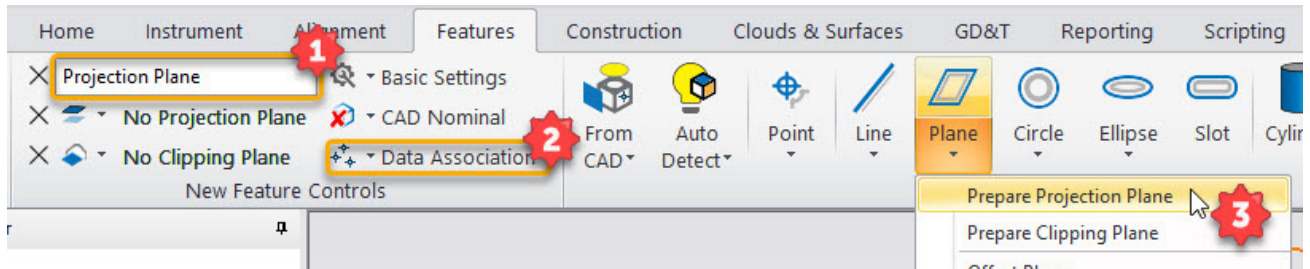


Figure 25-32. Steps to build a projection plane

2. Now you can build circle by editing the name and then selecting the circle icon, and then selecting the points to use in the circle fit.

Don't be surprised if you fit a circle as a pin rather than a hole, we can fix that later. As you can see below, the circle is projected to the plane and fit as a pin inside the measured points, compensating automatically for the point offsets ([Figure 25-75](#)).

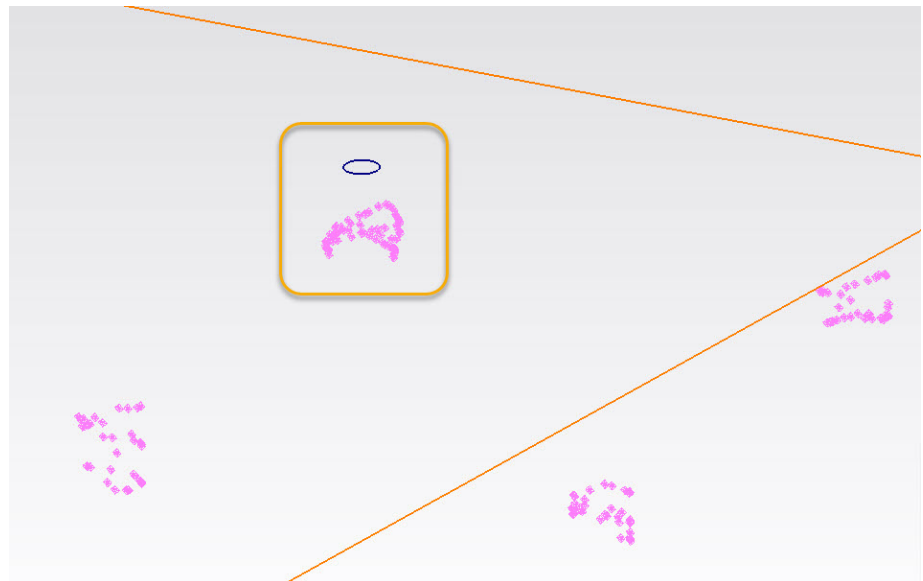
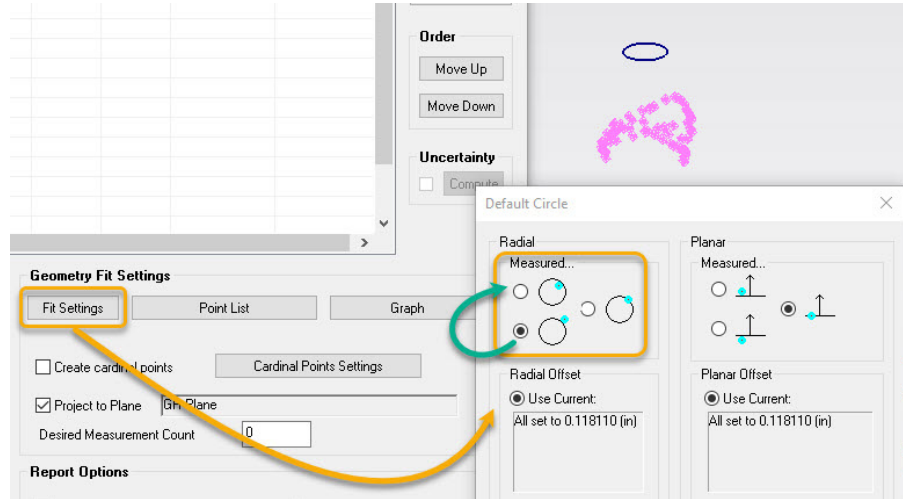


Figure 25-33. First circle

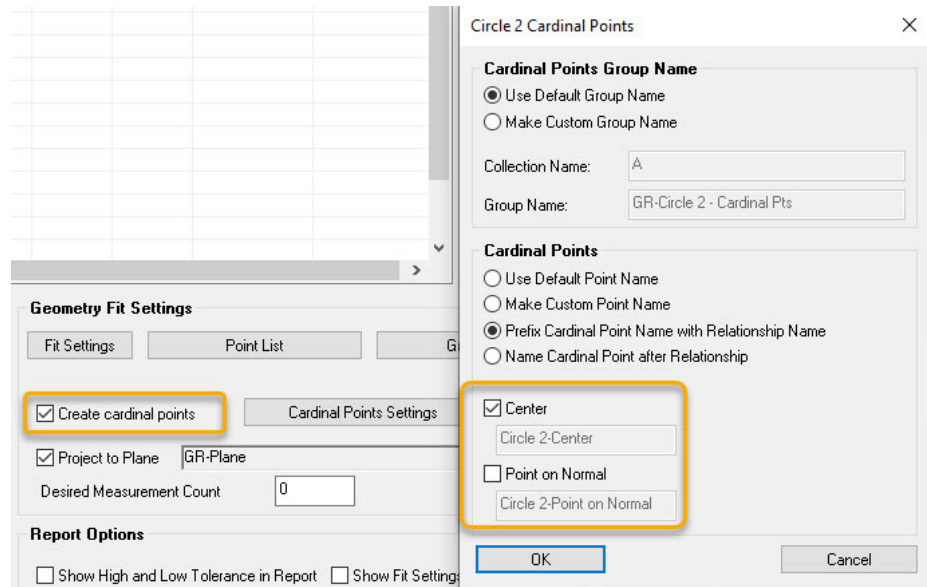
3. Go ahead and build 3 more holes.
4. Open the properties of one of the circles and fix the geometry fit settings, such that the points define a hole measurement ([Figure 25-17](#)).

Figure 25-34. Changing the Geometry fit settings from pin to hole



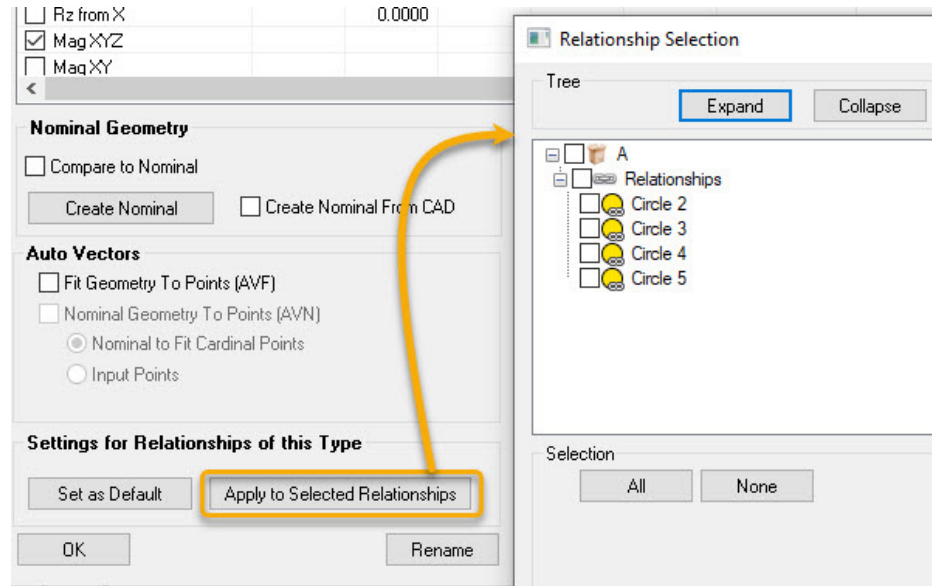
5. Turn on the **Cardinal Points**. The circle center points are included as part of the cardinal points. To turn them on first edit the settings as desired (in this case you probably want to turn off the point on normal) and then check the box to turn them on (Figure 25-16).

Figure 25-35. Enable the center cardinal point



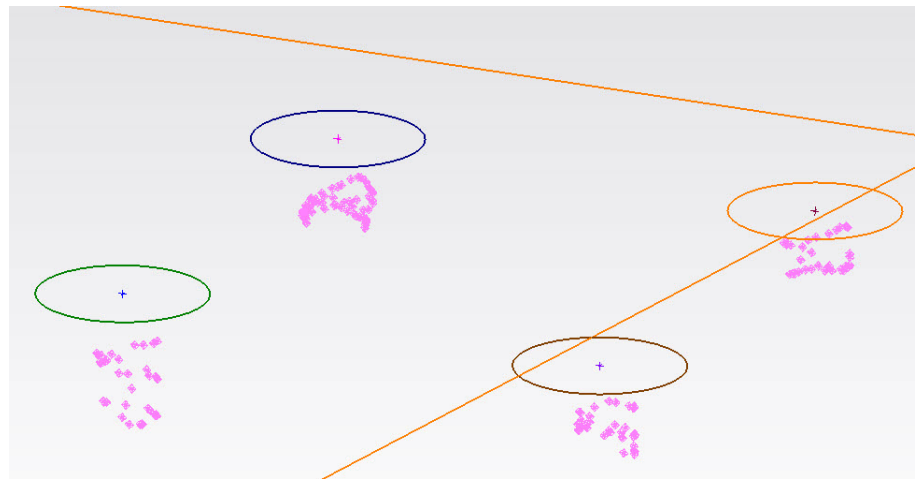
6. Apply your settings to the other circles. As soon as you make changes to a circle you will see the graphics update accordingly. You can apply your settings to other circles using the **Apply to Selected Relationships** Button (Figure 25-36).

Figure 25-36. Applying your settings to the other circles.



You will see a fairly complicated dialog offering you the ability to select only particular settings to apply, but just select all and your settings will be applied to the other circles (Figure 25-37).

Figure 25-37. Final fit circles.



7. Build an additional circle through the existing circle centers. This can be done by selecting the center points but there is an easier option. Select **From Multiple Circle** from within the circle drop down list. Geometry intersections of all types can be found in these feature drop down lists (Figure 25-15).

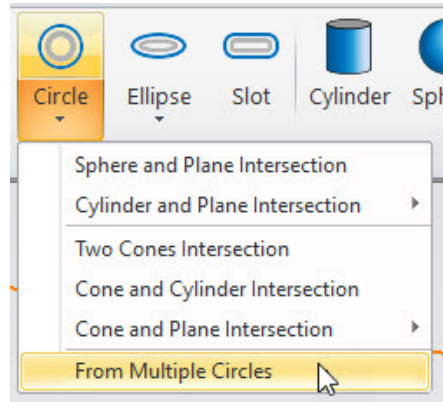


Figure 25-38. Selecting Circle from Multiple Circles

8. When prompted, select all 4 circles.

Note that from the tree you cannot select the GR-Feature under the relationship category directly, but you can expand it to select the measured circle within. GR-Features tie together all the contributing pieces so that you can easily build reports (Figure 25-4).

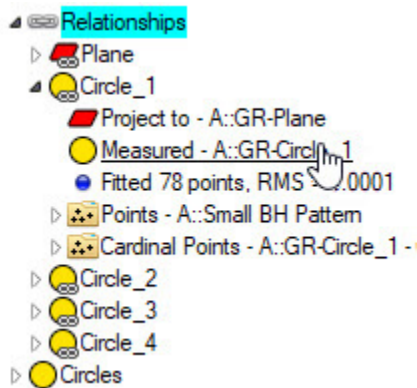


Figure 25-39. Selecting the measured circle within the GR-Circle Feature

9. Finally, add a diameter dimension to the circle you just created by navigating to the Reporting tab and selecting the diameter dimension from the dropdown list

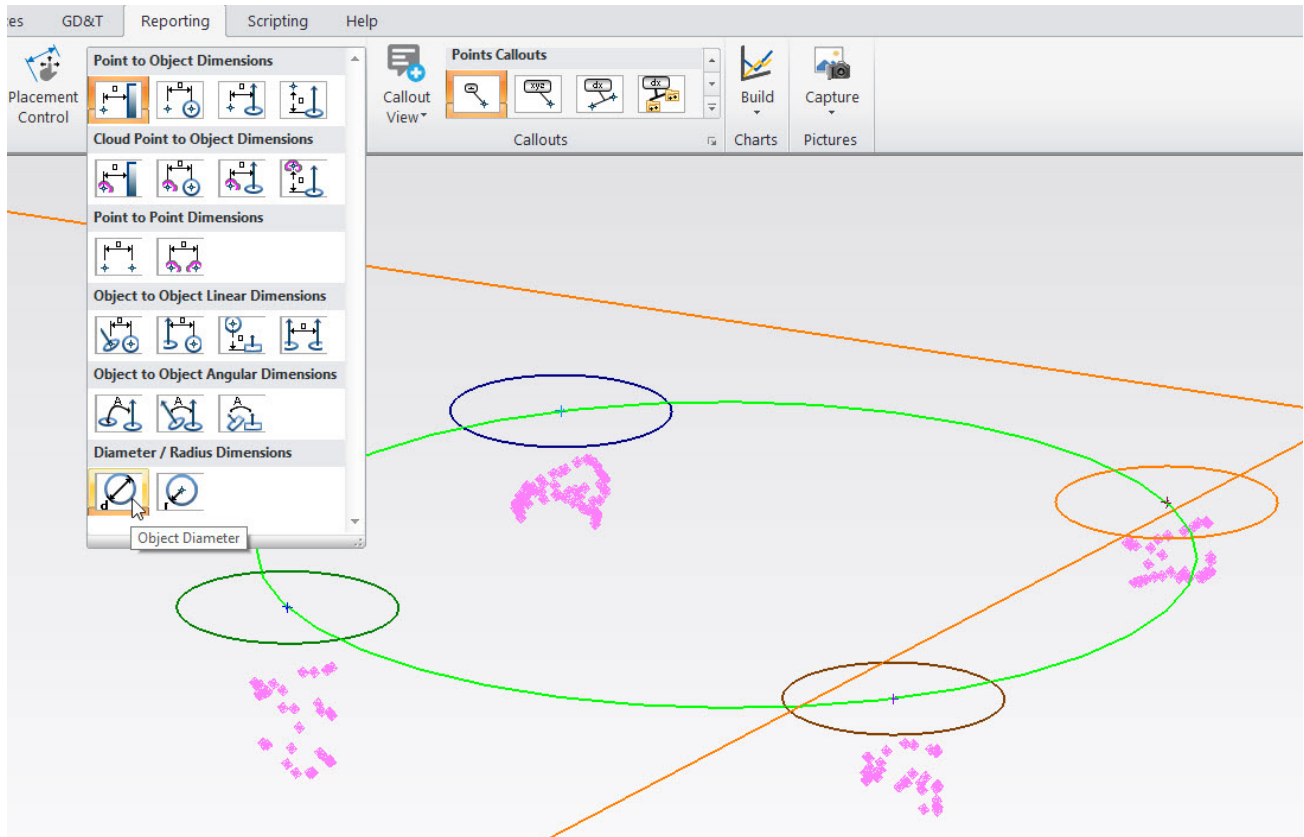


Figure 25-40. Adding a diameter dimension to the circle from multiple circles

How to build a coordinate system using measured features

Taking the prior example, let's build a new coordinate frame from the projection plane and this circle pattern (Figure 25-5).

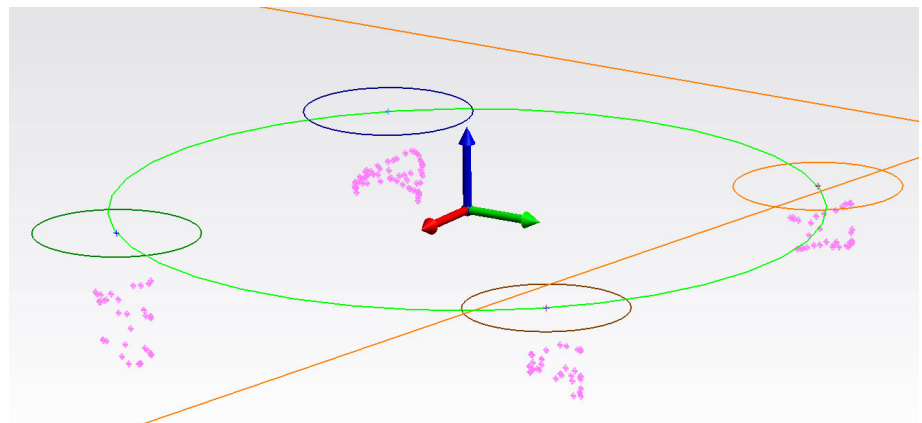


Figure 25-41. Coordinate Frame built from a bolt hole pattern

To do so, do the following:

1. Switch to the Features Tab of the ribbon, enter a name for this new frame and then press the Frame button like this (Figure 25-9).

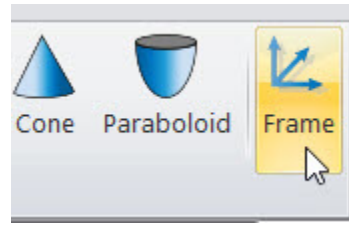


Figure 25-42. Adding a frame.

Frame are defined based upon a 3 inputs which can be entered in any order:

- **Origin.** This defines the base coordinate of the frame
 - **Primary Axis.** This axis is help perfectly to the selected reference
 - **Secondary Axis.** This axis is fit as well as is possible to the reference while maintaining the primary axis alignment.
2. Lets pick the origin first. In this case, we want to build a frame centered on the circle through the patter, or the center of the patter. To do so use the **Select Object** button and select this circle as the Origin like this(Figure 25-1).



Figure 25-43. Selecting center circle as the reference for the new frame.

3. Next lets define the Primary Axis. In this case everything is being projected to the same plane, so lets use that. To do so select **Object** and select the projection plane.

With each new selection the wire frame placeholder frame will update give you a preview of the frame you are creating(Figure 25-65).

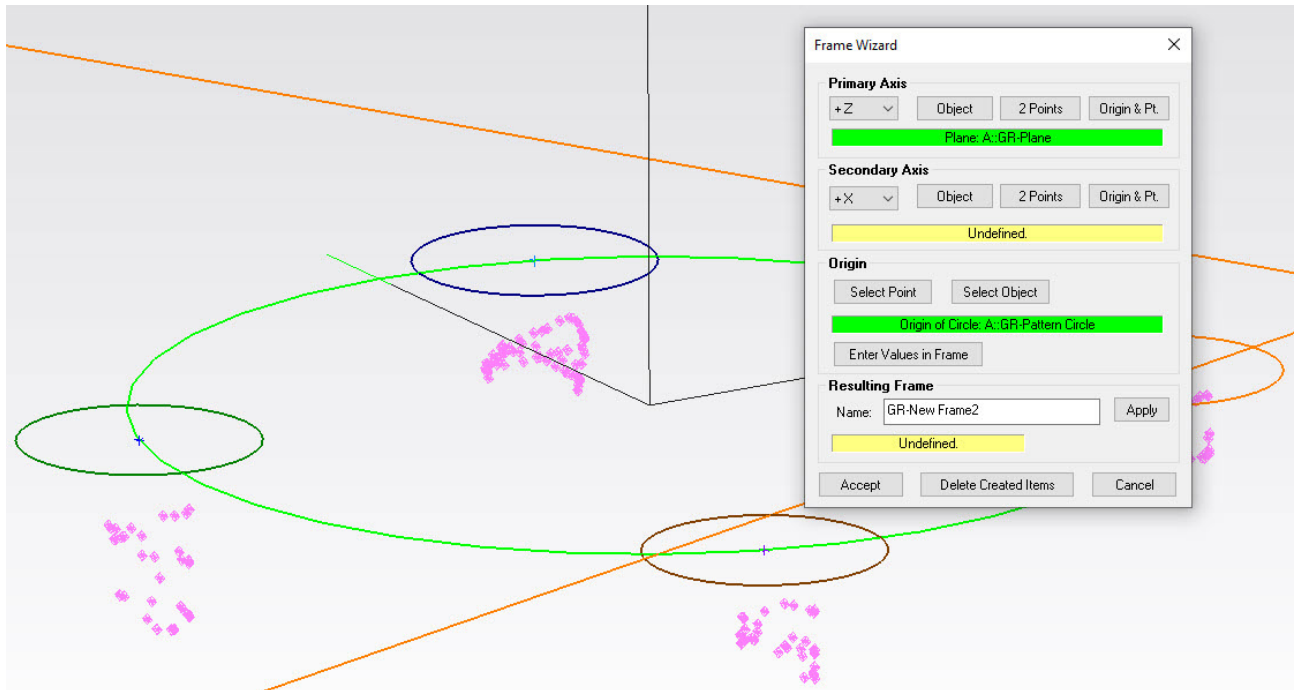


Figure 25-44. Origin and Primary Axis selection

If you notice, the definition for each axis can be changed. The primary axis is +Z by default but it can be edited as desired (Figure 25-63)

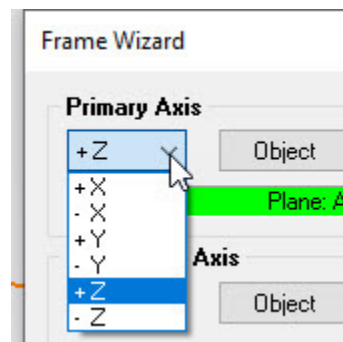


Figure 25-45. Axis definition

4. Finally we want to clock the frame such that the secondary axis points from the center point of the first circle to the 4th circle. Select the **2 Points** button and select the circle points in the order you want them to be used.

If you are defining the +X axis for example select the first point as the start and the second point as the direction from that starting point.

5. Accept the frame and pick a nominal reference frame if you wish. Doing so will provide a comparison between the frame for reporting purposes.

How to measure a set of reference points.

Another common measurement task is to measure a set of reference points using a probing device. This could be any set of defined point coordinates exported from another file or saved in a text file.

To begin lets import a set of points from an ascii file.

1. Drag and drop an ascii file with your nominal points into SA.

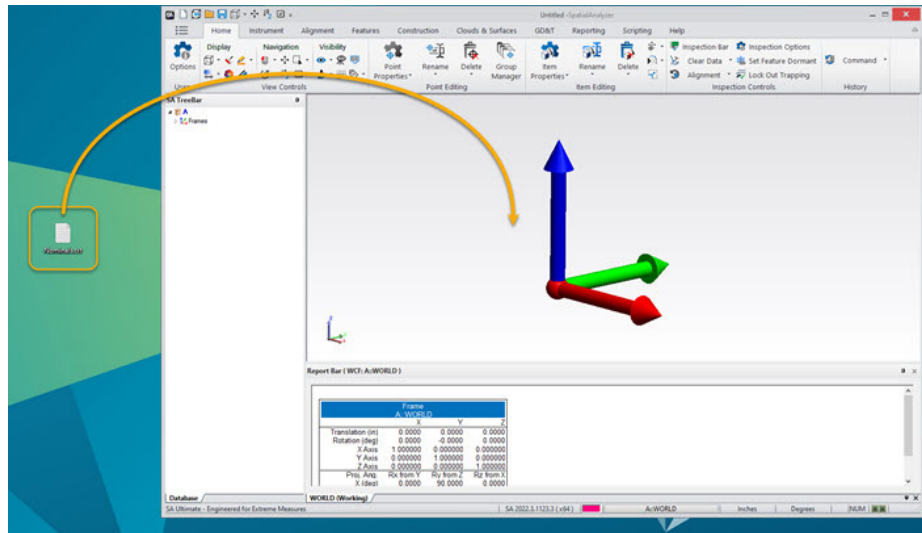


Figure 25-46. Importing a file into SA using drag and drop.

This can be any *.txt file with a standard format such as Point Name, X,Y,Z format. When you import it, our Ascii import dialog will open and you can choose the format that matches your file (Figure 25-2).

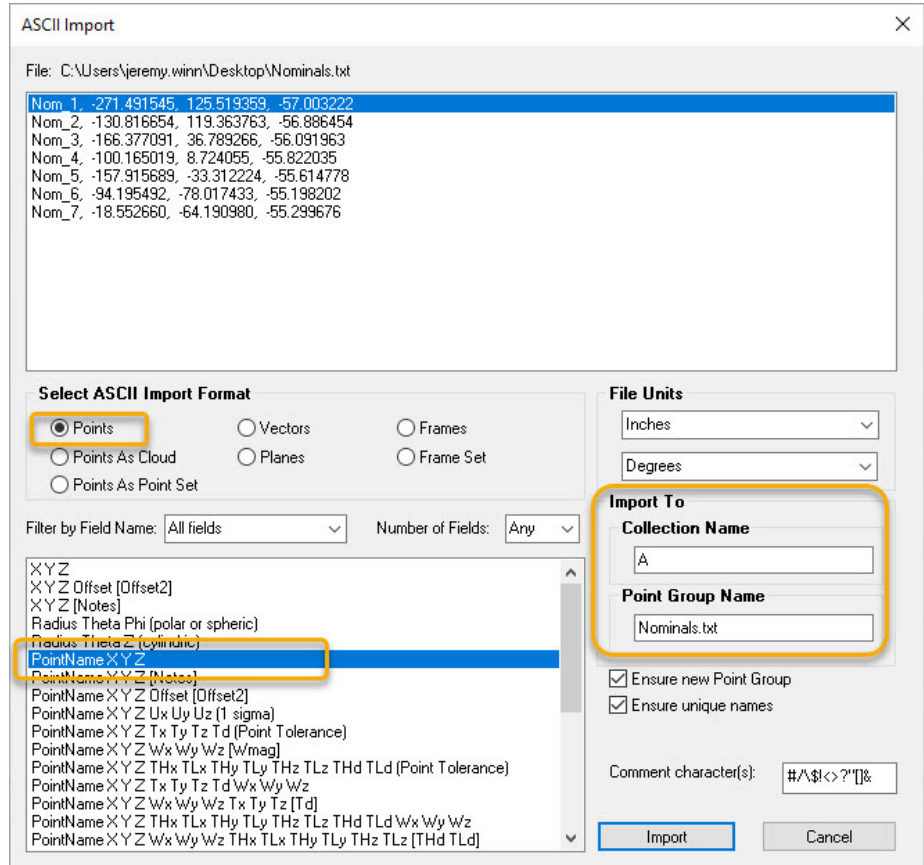


Figure 25-47. Ascii Import

2. Select the Points option for what you are importing, the format your file is in from the list of formats, and double check the collection and group names as to where you want the data to be stored.
3. Navigate to the Features tab of the ribbon and build a **Group to Nominal Group** relationship from this new point group (Figure 25-6).

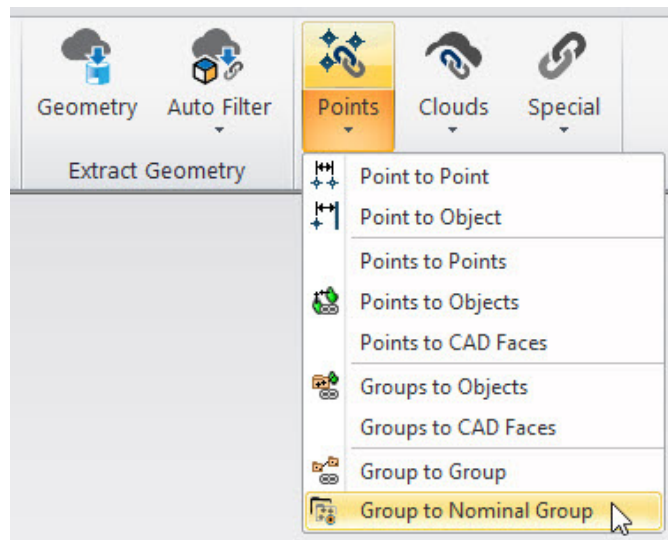


Figure 25-48. Build a Group to Nominal Group relationship

This will build a relationship or dynamic link between the imported created group you selected and a new placeholder group that is ready to be measured.

4. Double click on the relationship in the tree to open its properties, and select the options you would like to use, closing the properties dialog when satisfied.

Group to Nominal Group relationships offer a set of options which can be enabled while you measure for greater user feedback. These are optional and include:

- **Use closest point to match new measurements.** This option is on by default and will rename the measured points to match the nominal points based on proximity
 - **Display closest point watch window.** A watch window will open when you start trapping that shows the distance to the closest point.
 - **Use view zooming and proximity.** This option enables the view tracking and zooming as you get closer to the reference point.
 - **Ignore points beyond threshold.** This option prevents you from measuring if you are further than the preset distance from the reference point.
 - **Update a vector group with relationship.** This option creates auto-vectors to show you how far off your measurements are from the reference after you have taken the measurement.
5. Verify your instrument is connected and ready to measure.
 6. Right-click on the relationship in the tree and select **Trap Measurements from an Instrument (Figure 25-64)**.

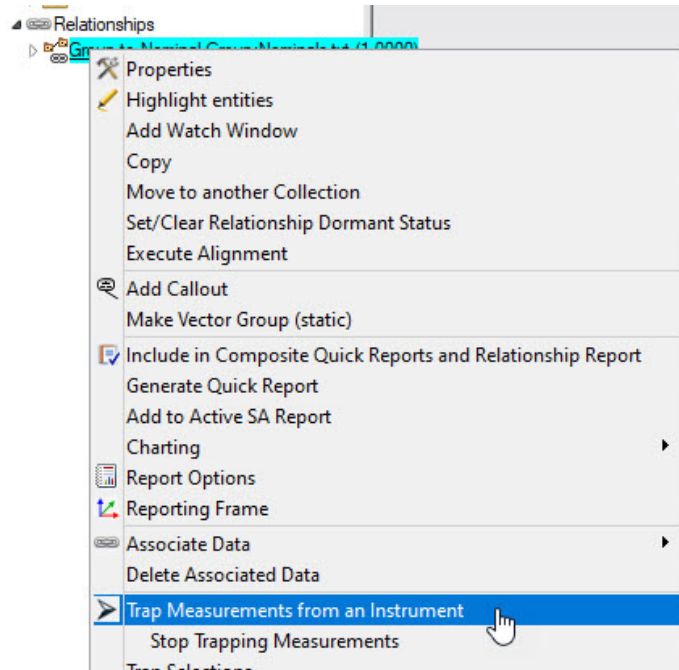


Figure 25-49. Start Trapping


7. Measure the first 3 points from the list to locate the instrument. If you are roughly aligned and measure the points in the same order as the reference group then you can continue measuring. But if not,

Cloud Based Part Inspection with an Arm and CAD.

This workflow walks a user through the process of importing CAD, adding an arm, and scanning a Part on a Tabletop. It can also be following by opening a sample file “GD&T with Point Clouds” from the *Help>Open Sample files* option.

Compare to CAD

To begin, open a new SA job and import a CAD model (You can also just open the sample file mentioned above and skip to the Alignment process).

1. Import CAD by going to **File>Import>CAD Formats** . CAD import can be performed at any point in the measurement process and can be performed by simply drag-dropping a file into the graphics.

Connect to the Arm

2. Connect to your instrument and initiate a scanning mode. This process will differ slightly depending on the arm but typically scanning button on the Instrument toolbar will initiate the scan ([Figure 25-50](#)).

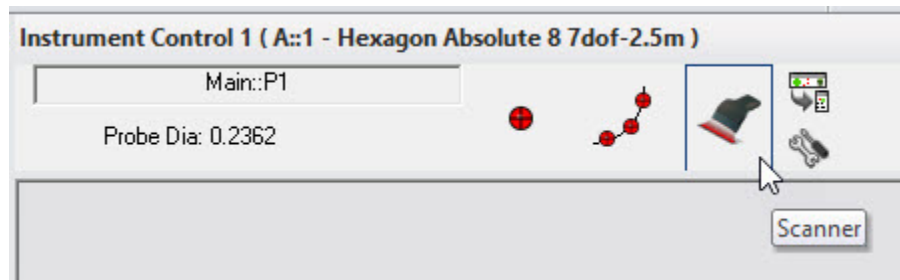


Figure 25-50. The Scanner button on a connected arm in Toolbar mode

Preparing to the Scan

Before scanning the part there is an easy way to prevent extraneous measurements of the table top, which is to define a clipping plane. Assuming you have a part on a flat surface, this surface can be measured as a plane and used for cloud data “clipping”.

1. From the Features Tab, in the plane drop-down list, select **Prepare Clipping Plane** ([Figure 25-51](#)).

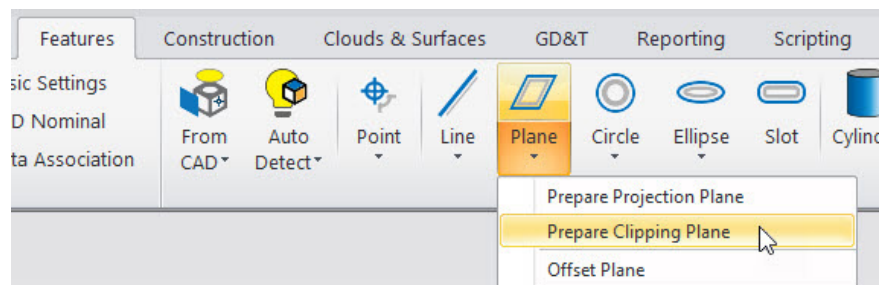


Figure 25-51. Selecting Prepare Clipping Plane.

2. Scan the table top or surface you are using to define the clipping plane (the surface your part is sitting on). The Cloud Clipping Plane we added should be actively trapping and collect the measurements, automatically fitting a plane. This will work both through scanning and simple point capture as show in (Figure 25-52).

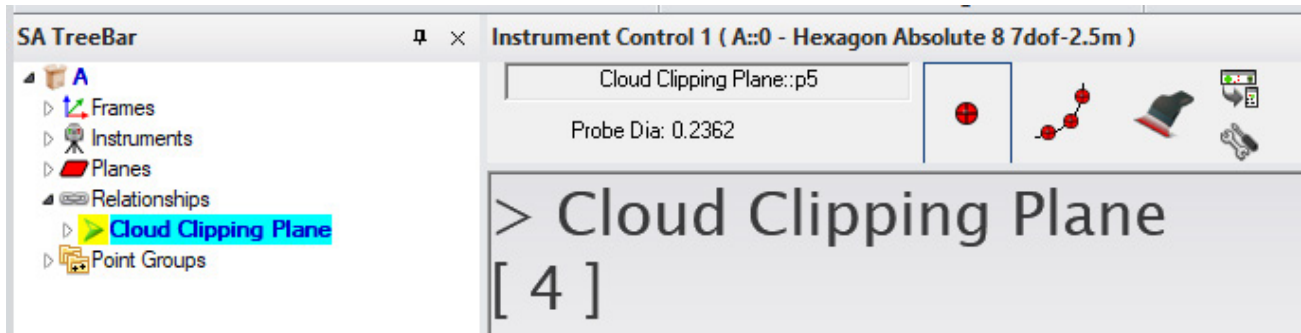


Figure 25-52. Trapping into a Clipping Plane feature

3. Press the **Next** button on the arm to stop measuring the projection plane and continue on to scanning your part. This can also be done from the controls on the Inspection bar or by directly right-clicking on the Clipping Plane in the tree.

Configuring the View and Scanning your part

The most important thing when scanning is to make sure you have good scan coverage. The easiest way to see that is to enable the instrument view and zoom in tight.

1. From the **Home** tab in the *View* section select *Enable Viewpoint from Instrument Updates*.

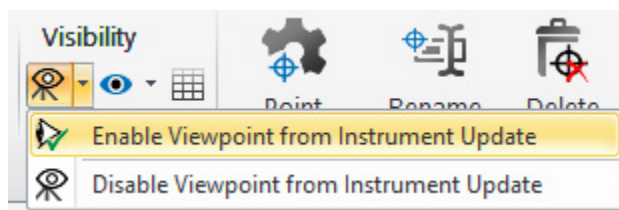
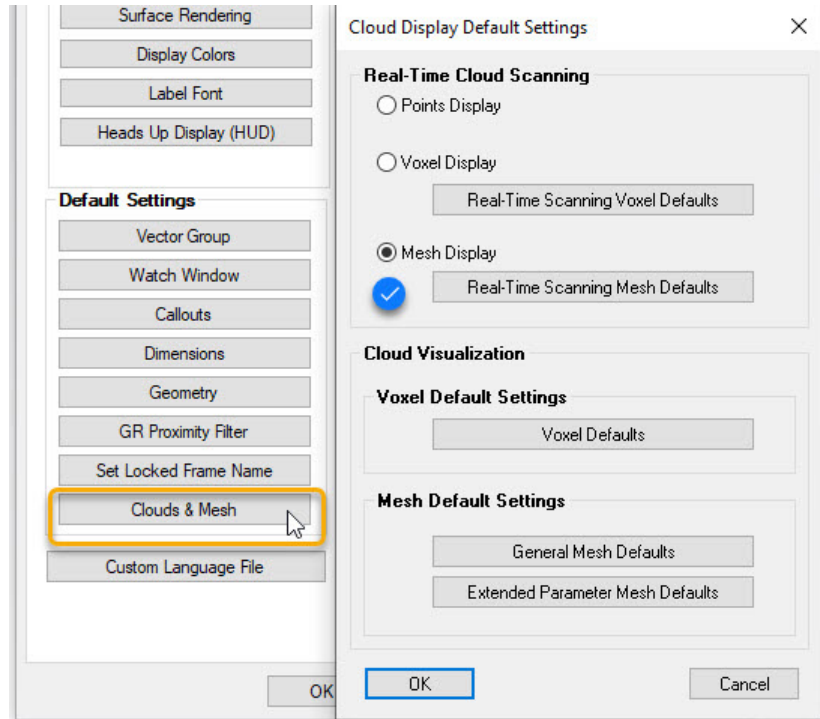


Figure 25-53. Enabling viewpoint from Instrument.

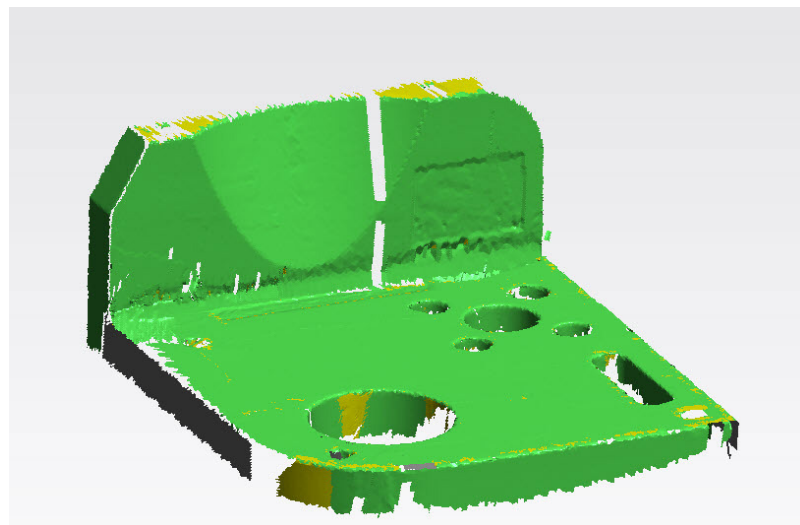
2. Enable Real-Time Meshing (optional). The most effective visualization of the cloud is through the use of a Real-time mesh. This can be enabled through the default setting in the **User Options > Display Tab > Clouds & Mesh**.

Figure 25-54. Enable Real-Time Mesh display



3. Scan the part making sure that you get good coverage. A green mesh indicates good data, where as a yellow section of a mesh indicates data of lower. You can also go to the properties of the cloud and select *“Re-Generate Mesh”*.

Figure 25-55. Example scan of a part in process, showing the Real Time Mesh display.



Align to CAD

If you are working with the sample file, right-click on the arm in the job file and select **Drag Instrument Graphically**. This will allow you shift the scan off the part so that you can test the alignment process below.

1. If you are using the sample file or had trouble with the clip-

ping plane there will be extra data from the table top. This extra data should be removed to aid the alignment. You can use *Ctrl+Alt D* to delete unwanted cloud points (hold Shift while left-click dragging with the mouse to select a region of data to remove).

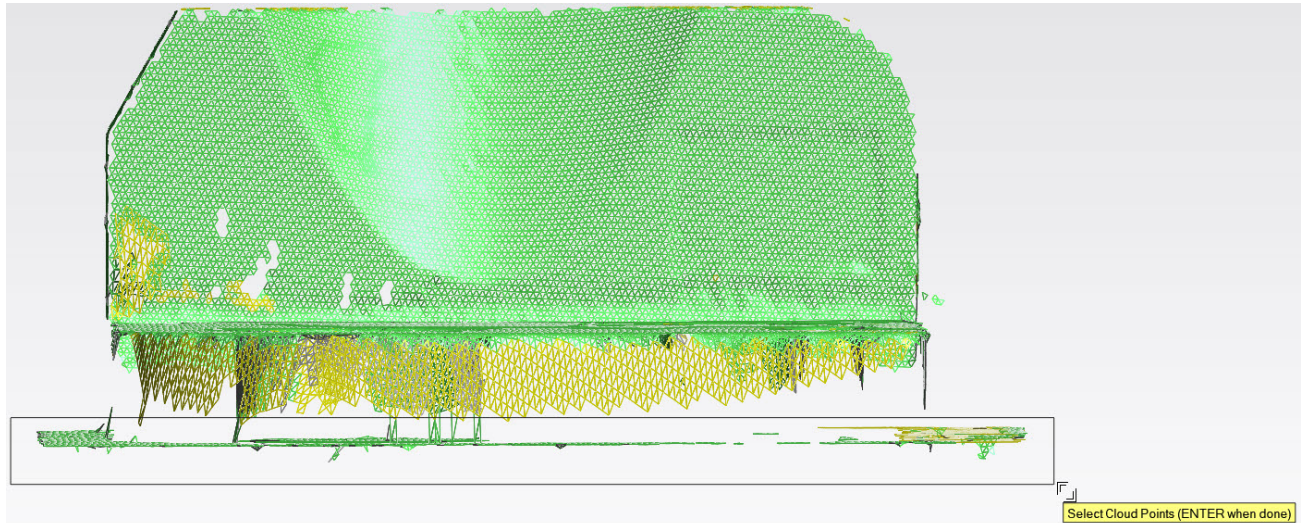


Figure 25-56. Selecting Data to Remove

2. Align your scan to the CAD file by right-clicking on the cloud and selected **Align to CAD**. When prompted select the CAD model(s) you have scanned and press **OK**.

You will see an alignment report like this (Figure 25-57). If you are happy with the results press **Yes** to accept. But if the fit looks questionable press the **Perform N-Point Alignment** button (see Optional Advanced Align to CAD below).

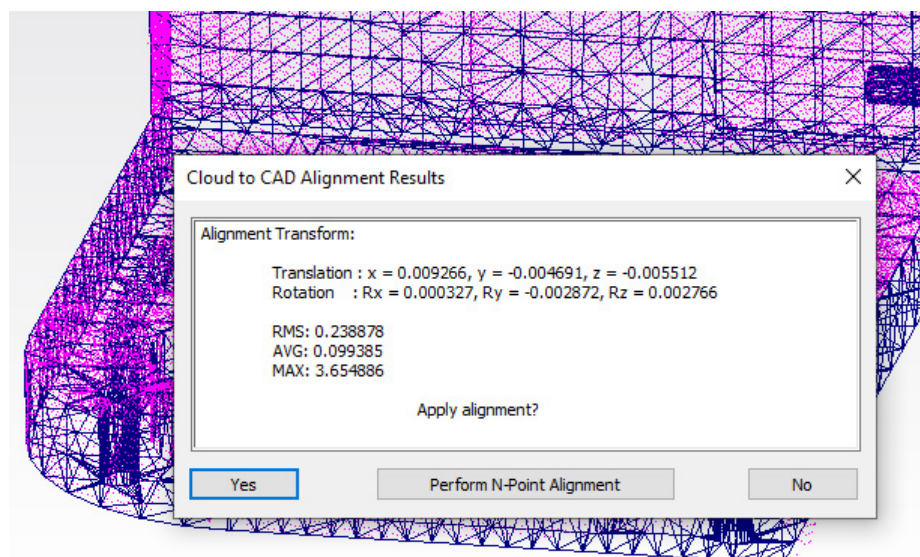


Figure 25-57. Cloud to CAD Alignment Results.

The alignment process will build a Cloud to CAD relationship for you

which can be used to display the overall deviation color map (Figure 25-58).

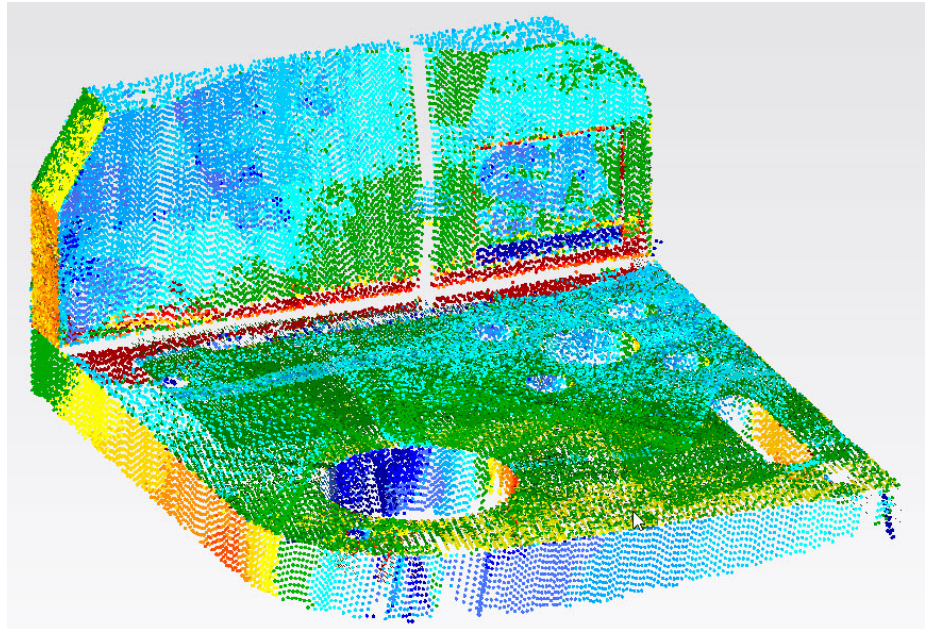


Figure 25-58. Deviation Color Map

The color map can also be enabled by opening the cloud's properties and switching from *RT Mesh* to *Voxel Cloud Display*.

Optional Advanced Align to CAD

If only a section of your CAD model can be measured or if your CAD only represents a section of the scan then the option to Align using an N-Points should be used as it defines seed points for the process.

1. Arrange the second graphic window and Align Cloud to CAD dialog so that you can see both the CAD model and cloud scan to facilitate selection of corresponding points such as the below (Figure 25-59):

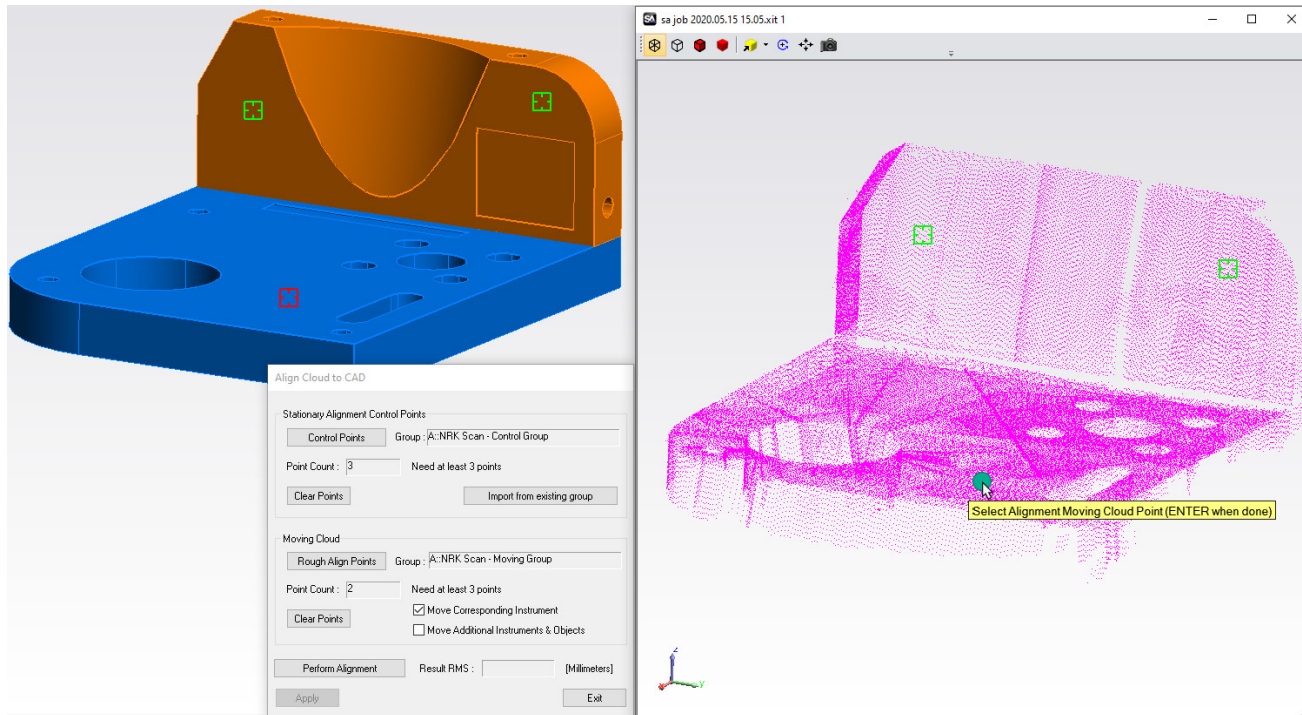


Figure 25-59. Align Cloud to CAD windows and arrangement.

2. Press the **Control Points** button in the **Align Cloud to CAD** dialog and select at least 3 points on the CAD model. These points should be on surfaces with good scan coverage.
3. Press the **Rough Align Points** button and graphically select a corresponding cloud point from the axillary view for each of the control points you placed on the CAD.
4. Press **Perform Alignment** and accept the alignment by pressing **Exit** when you are happy with the results.

Feature Extraction from CAD

The next step in the inspection process is to report the position of features on the part. To do so, the following steps can be used:

1. Navigate to the **Features** tab of the ribbon and select **Auto Filter**. This will open the Nominal Geometry Auto Filter dialog (Figure 25-60).

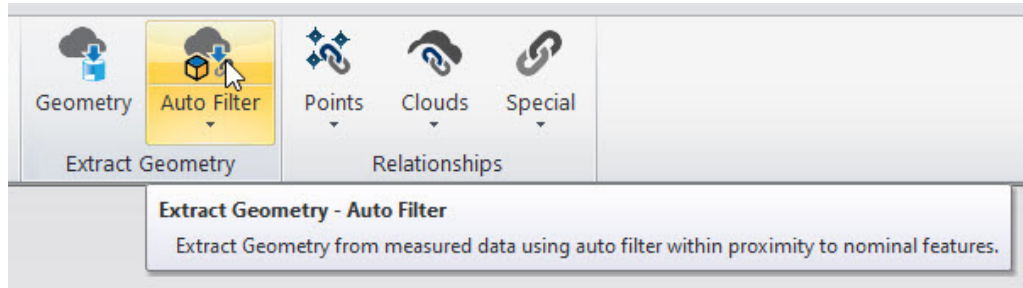


Figure 25-60. Auto Filter Option on the Features Tab

This option offers the ability to build features from a point cloud based on filtering within a proximity to nominal features. It also has some built in automatic extraction tools we will take advantage of. Ensure the following options are enabled:

- Live Auto Filter - under Relationships
- Build Callouts (optional)
- Auto Detect - under Input Clouds

2. Move the Auto Filter dialog aside but don't close it, and then also select the desired From CAD extraction method (3D for Planes, Cylinders, Cones and Sphere's while you will select 2D for circles and slots) (Figure 25-61).

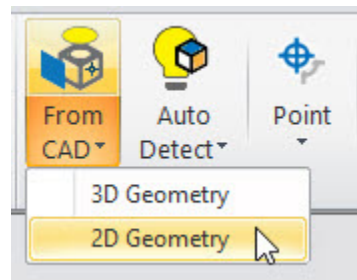


Figure 25-61. From CAD options

3. Click faces on the CAD model to analyze. This selection will extract a nominal geometry from the CAD, extract cloud data within a proximity to that nominal feature and fit a measured feature to the data in one click.

SA Toolkit (Classic Menu Only)

The SA Toolkit (Figure 25-62) contains five tabs that provide convenient access to a subset of the most powerful tools in SA. These tabs include GD&T, Relationships, Inspection, Analysis, and Reporting.

GD&T.

What was formerly known as the GD&T bar has been brought on as a tab in the Toolkit bar. This tab is an ideal tool for building GD&T Annotations. For more information on the GD&T Tab and its use see the **GD&T** Chapter.

Relationships.

The relationships tab enables one-click definition of features (defined by CAD or using primitive geometry) primed for easy inspection using geometry relationships, with or without projection planes. Other relationships can be easily defined as well.

Inspection.

The inspection tab presents all trappable features from all collections—relationships, datums, and feature checks—in a single list. This allows you to easily advance through an inspection routine, trapping measurements for each feature, and instantly providing inspection results against nominal data.

Analysis.

Some of the more commonly-used analysis tools, such as the frame wizard and geometry fitting, are presented together in the analysis tab.

Reporting.

Functionality for creating reports, making dimensions, and creating/annotating callout views is gathered together in the reporting tab. To display the Toolkit bar, right-click on the menu/toolbar/status bar area of the SA interface and select Toolkit.



Figure 25-62. SA Toolkit

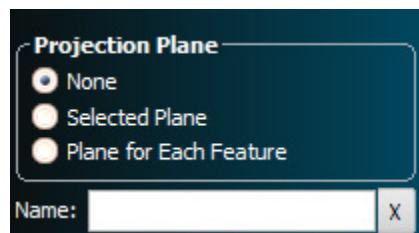
Tutorials On Using the Toolkit for Inspection

For instructions on using the toolkit refer to the Tutorials section of this manual.

Relationship tab

Functionally the Relationship tab of the Toolkit works much like the Features page of the ribbon but with a slightly different organization. The data association options are not available but the projection plane and name controls are (Figure 25-63).

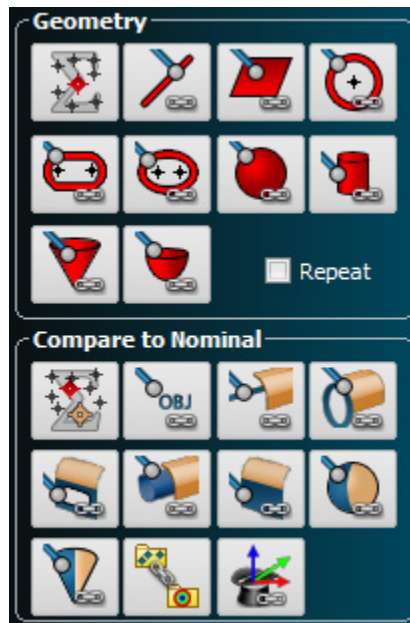
Figure 25-63. Projection Plane and feature name controls



The **Geometry** section (Figure 25-64) provides access to relationships available through the **Relationships>Geometry Comparison>Fit Only** menu, while the **Compare to Nominal** section provides access to the **Relationships>Geometry Comparison>Fit and Compare** menu. These relationships are functionally the same. The only difference is that the **Compare to Nominal** section prompts for nominal selection as part of the build

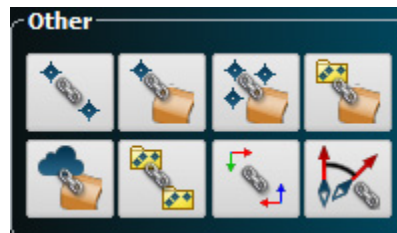
process. Nominal features can be assigned or edited at any time.

Figure 25-64. Geometry and Compare to Nominal Sections of the Relationship tab of the Toolkit



The Other section of the Relationship tab (Figure 25-8) provides buttons to build relations of types other than geometry relationships.

Figure 25-65. Other section of the Relationship tab.



The Utilities section of the Relationship tab provides quick links to a selection of very useful tools needed for data association. These include:

Figure 25-66. Utilities section of the Relationship tab.



Detect Nominals

Used to quick gather nominal features based on proximity.

- **Detect Nominals by Proximity.** This function looks at a CAD model, extracts the nominal features from that CAD model and associates them with the measured features.
- **Detect SA Object Nominal Geometry by Proximity.** This function looks for existing objects and both sets them as the nominal object for the geometry relationship and renames the measured feature using the nominal names.

Select Nominal Geometries

This function takes the specified geometry and converts them into geometry relationships such that they can be used for trapping and direct comparison with a measured feature.

Auto Filter to Nominal

These functions provide a means to gather data within proximity to the nominal features and associate that data with the relationship.

- **Auto Filter to Nominal 3D.** Used to detect either point or cloud data within proximity to geometry (see [“Auto Filter to Nominal Geometry >3D Extraction”](#) on page 587).
- **Auto Filter to Nominal 2D.** Used to detect holes in surface scan data (see [“Auto Filter to Nominal Geometry >2D Geometry”](#) on page 597).
- **Auto Filter to Nominal Line.** Used to detect cloud points within a specified proximity to a line.

Extract Geometry From Cloud

This function is used to extract geometry from a point cloud without any nominal reference. It uses a click point as a starting point to search through the cloud and identify features.

For more information on feature extraction and auto filter functions refer to [“Feature Extraction from Point Clouds”](#) on page 587

Sweep and Sweep & Move

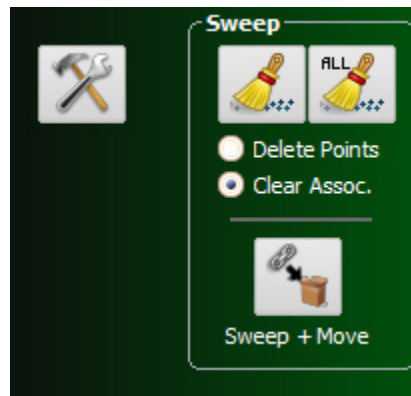


Figure 25-67. Sweep and Sweep & Move sections of the Inspection tab.

Sweep

The sweep controls ([Figure 25-67](#)) are composed of two button icons and two radio buttons. This provides a 4 way control.

- Sweep/Clear Assoc. - clears the actively selected feature
- Sweep All/Clear Assoc. - clears all the features

- Sweep/Delete Points - deletes the points from the actively select feature
- Sweep All/Delete Points - deletes the points from all features

Sweep & Move

Sweep & Move was added to allow collection management and easier repeat measurement of common features in separate collections. This function only works with relationships not GD&T features. When you press the Sweep & Move Icon you will see the following dialog:

By utilizing the Sweep & Move feature, a user may choose to either *copy* or *move* relationships from the active collection to a destination collection. By selecting *COPY*, a duplicate set of the relationships will be built in the new collection, constructed point associations will be maintained while clearing measured points from these new relationships. By choosing *MOVE*, relationships will be moved into a destination collection, leaving the measured points and constructed geometry only in the original collection.

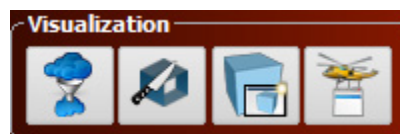
Properties Button

A properties button is also available next to the sweep section. This button provides direct access to the properties dialog of the selected feature in the inspection tab.

Analysis Tab

The Analysis tab consists of a set of direct shortcuts to powerful and commonly-used features also available through SA's main menus. A brief synopsis of the function of each button is provided here.

Figure 25-68. Visualization section of the Analysis tab.



Visualization

The Visualization section of the Analysis tab (Figure 25-68) provides a set of useful graphical controls and object identification tools.

Cloud Thinning

This button brings up the Cloud Display control dialog. It allows you to limit the number of cloud points rendered on-screen, and control the size of the rendered cloud point, in pixels.

Clipping Planes

This button brings up the Clipping Planes control dialog. It allows you to define, edit and turn on and off the display of clipping planes for visualization.

Auxiliary View

This button opens an Auxiliary View dialog. Auxiliary graphical views can be rotated, positioned, and rendered independent of the primary graphical view, and they also support selection.

Hover Window

This button opens the Hover window which will indicate the collection, name, type, and location of the item in the tree, the coordinates of a point (in Cartesian, Cylindrical, or Spherical coordinates, the deltas for a vector, or the number of measurements associated with an instrument.

Figure 25-69. Construction section of the Analysis tab.



Construction

The Construction section of the Analysis tab (Figure 25-69) provides a set of useful construction tools.

Frame Wizard

The Frame Wizard is used to create a coordinate frame based on different combinations of points or features. It allows control of axis components and origin placement based upon these features. For more information see the second on Frame Wizard in the Analysis Chapter.

Objects from CAD

This button provides direct access to a collection of geometry extraction tools. It allows you to pick a geometry type and then graphically select faces on a CAD model, building the nominal geometry from the selected face.

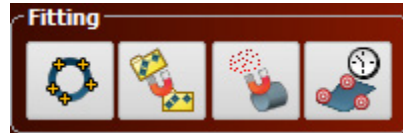
Circle Center Points

This button triggers the **Construct>Points>Center>Circles** function.

Colorized Graphical Mesh

This button triggers the **Construct>Colorized Graphical Mesh>From Proximity Points** command. It will prompt for selection of points and surfaces and build a colorized mesh from the point proximity. Right-click to build a mesh using point clouds.

Figure 25-70. Fitting section of the Analysis tab.



Fitting

The Fitting section of the Analysis tab (Figure 25-70) provides access to a selection of powerful alignment tools.

Geometry Fit

This button will open the **Construct>Geometry by Fitting** dialog.

Best-Fit Transform

This button will open the **Analysis>Best Fit Transformation>Points to Points** function.

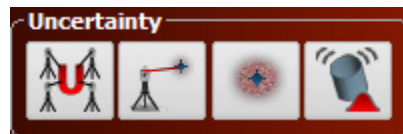
Relationship Fitting

This button triggers the **Relationships>Move Objects by Minimizing Relationships** function.

Quick Align to CAD

This button triggers the **Analysis>Best Fit Transformation>Points to Surfaces/Objects>Quick Align to CAD** function.

Figure 25-71. Uncertainty section of the Analysis tab.



Uncertainty

The Uncertainty section of the Analysis tab (Figure 25-71) provides a set of powerful analysis tools that take advantage of the uncertainty information contained within measured points.

USMN

This button triggers the **Analysis>Coordinate Uncertainty>Unified Spatial Metrology Network** function.

Simulate Measurements

When you wish to link constructed points with an instrument station and treat them as measurements you can use this button which triggers the **Analysis>Measurement Simulation>Fabricate Measurements** function.

Make Uncertainty Fields

This button calls the **Analysis>Coordinate Uncertainty>Create Point Uncer-**

ainty Fields function.

Geometry Fit Uncertainties

When you wish to determine the uncertainty in fitting a particular type of geometry to a selected set of points you can use this button which triggers the **Analysis>Coordinate Uncertainty>Fit Geometry Using Uncertainty Fields** function.

Figure 25-72. Scaling section of the Analysis tab.



Scaling

The scaling section of the Analysis tab (Figure 25-72) provides a set of controls to help with scaling points and controlling scale of instruments and their measurements.

Temperature Compensation

This button triggers the **Analysis>Temperature Compensation** command which prompts to choose a reference frame and then the point groups to scale, allowing for scaling of constructed or imported points.

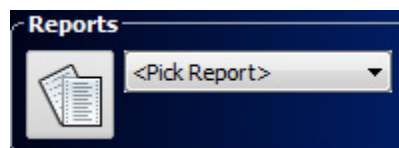
Scale Bar Measurement

Make Scale Bar

This button triggers the **Construct>Scale-Bar>Pick 2 Points** command.

Reporting Tab

Figure 25-73. Reports section of the reporting tab.



Reports

The reports section of the Analysis tab (Figure 25-73) provides access to the SA reports in the job file.

Add New SA Report

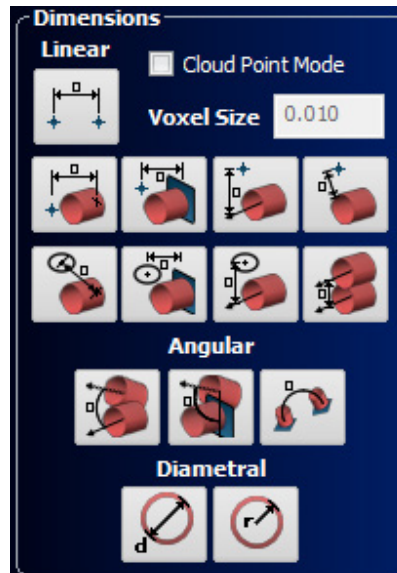
This button builds a new SA Report.

Pick Report

This drop down menu provides access to a list of all the SA reports

in the Job file. It provides a quick way to open the desired report by clicking on it in the list.

Figure 25-74. Dimensions section of the Reporting tab.



Dimensions

The Dimensions section of the Reporting tab (Figure 25-74) provides a set of quick format buttons for adding dimensions. Dimensions are defined as one of 4 different dimension types:

- Point to Point
- Point to Object
- Object to Object
- Diameter/radius

The individual icons provide not only the ability to add these dimensions to an SA job file as you would be able to do through the Analysis menu, but they provide additional pre-formatting options depending on the selected icon.

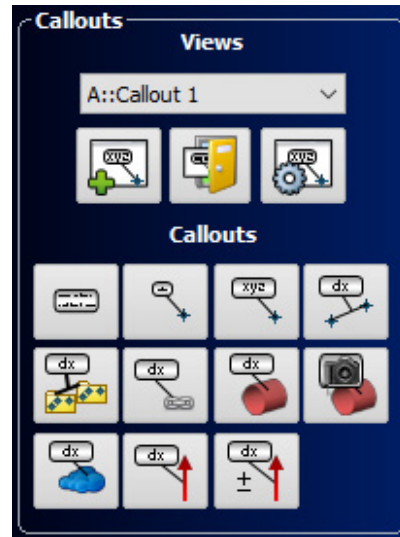


Figure 25-75. Callouts section of the Reporting tab.

Callouts

The Callouts section of the Reporting tab (Figure 25-75) provides access to both the callout views in the job and the individual callouts that can be placed in any callout view.

Views

Selection List

This drop down menu provides access to a list of all the callout views in the Job file. It provides a quick way to activate the desired callout view by clicking on it in the list.

Add New Callout View

This button adds a new callout view to the job file in the active collection and prompts for a name.

Close Callout Views

This button deactivates all the active callout views.

Callout Options

This button provides access to the active callout view's right-click menu options.

Callouts

This section provides a set of buttons to build callouts within the active callout view of various types.

Point Clouds are a collection of compressed point locations in a format that is optimized for working with large amounts of data. This chapter addressed the fundamental principles necessary for effectively working with Point Cloud data within SA and the differences between cloud points and regular points.

Point Clouds

Point Clouds are unique from Points and Point Groups because they are optimized for data storage and minimum memory usage. All cloud points are treated as a rigid body so point offsets and meta data are only recorded on the cloud level not with the individual points. Point Clouds are therefore ideal for large amounts of data, such as data measured with a laser room scanner or arm scanner, but can also be used to reduce the size of scanned measurements from a laser tracker with offsets.

Point Cloud Formats

Standard Clouds Point Clouds

The standard point cloud format is the simplest and most generic format of point clouds. Each cloud point consists of nothing more than an XYZ position and an index. The cloud contains targeting offsets (applied to all included cloud points) and visual controls.

Scan Stripe Clouds Scan Stripe Clouds

Scan Stripe Clouds were added to provide a missing piece of information necessary for meshing. In this format a transform representing the instrument probing direction is saved per scan stripe. Scan Stripe Clouds therefore contain slightly more information but are otherwise the same as Standard Clouds. For this reason Scan Stripe Clouds can be converted to Standard Clouds but Standard Clouds cannot be converted to Scan Stripe Clouds.

Cross Section Clouds Cross Section Clouds

Cross Section Clouds were added as a convenient way to process a cloud or group of clouds in cross sections. It provides access to a set of filtering controls and the cloud visualization tools for cross sections.

Point Cloud Visualization

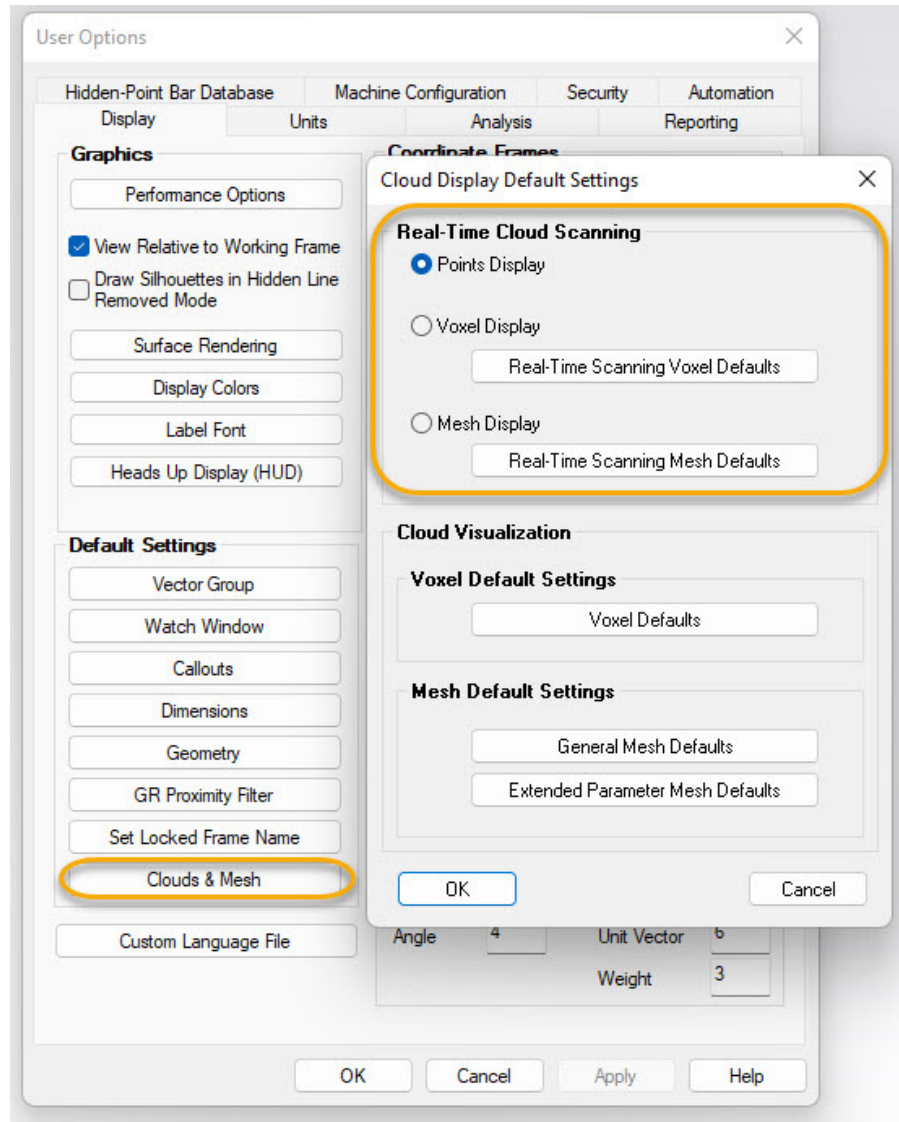
One of the more challenging aspects of working with point cloud data is finding the best way to visualize the information. When actively measuring (in Real-Time) it may be better to have one view format and a different one when processing data.

Real-time Data Display

There are 3 primary ways to visualize cloud data as it is being measured and these default settings can be controlled from the Users Options on the display tab ([Figure 26-1](#)).

1. **Points Display.** The raw point as measured can be shown using this as the default. This method is also important to use in order to see color or intensity values saved with points (see [“Point Cloud Formats”](#) on page 545).
2. **Voxel Display.** A more efficient method of display that shows a single point per volume of a specified size (see [“Voxel Cloud Display”](#) on page 558).
3. **Mesh Display.** This option does not show the measured points but uses a mesh to display the data (see [“General Mesh”](#) on page 602).

Figure 26-1. Default display settings available in the Users Options.



Working with Raw Point Cloud Data

Display Control

The number of cloud points displayed graphically can be controlled using the Cloud Display Control (**View>Cloud Display Control Ctrl+T**). This utility provides the ability to adjust a global thinning factor which reduces the overall number of cloud points displayed as well as a cloud point size control.

As a Point Cloud is rotated or translated in the graphic view, the display resolution of the cloud will reduce automatically and will be restored once the movement is stopped. Control over the extent of this additional thinning is also provided as part of the Cloud Display Control (Figure 26-2).

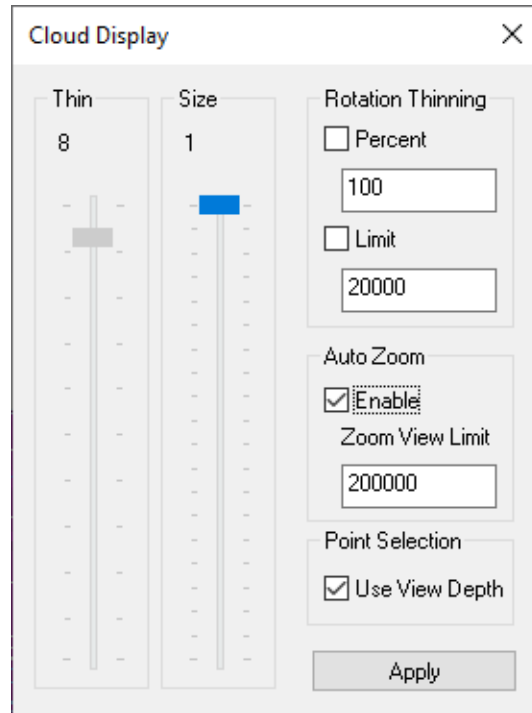


Figure 26-2. Cloud Display Control Dialog

- **Thin.** The visual thinning factor applied. If set to the 8, for example, only 1 point in 8 will be displayed.
- **Size.** The number of pixels used to display a point cloud point.
- **Rotational Thinning.** This section controls the additional thinning during graphical moves and rotations. Either or both of these rotation limits can be applied:
 - **Percent.** If set, Rotational Thinning is controlled by a percentage value from 0-100 where 100 is no additional thinning.
 - **Limit.** If set, Rotational Thinning is controlled by an upper limit of the number of cloud points displayed. In other words no more than the Limit number of points will be displayed graphically.
- **AutoZoom Enable.** Enables/Disables a zoomable display in which a limit can be set for how many cloud points are displayed graphically within the viewing window. As you zoom in more points will appear until the cloud is displayed at full resolution, while zooming out reduces the total visible in a single area in favor of holding the max display value.
- **Point Selection.** This option is used to help select points that are closest to the viewpoint when selecting points from a cloud with multiple layers.

Dynamic Cloud Thinning

When a live instrument is connected to in the job the total number of

cloud points visible at one time is limited to 200,000 points. This prevents potential loss of data due to slow downs resulting for graphic display issues. Cloud thinning can be regulated below this threshold but can not be increased beyond the threshold until the instrument is disconnected.

Auxiliary Cloud Thinning Factor

In addition to the global cloud thinning controls each individual cloud can have its own thinning factor set independently through its properties. This control decreases the total number of points displayed at any time. This allows clouds of very different densities to be equalized for display purposes.

Cloud Bounding Box

In order to more readily visualize the extents of a cloud a bounding box can be displayed by checking the option in the cloud properties dialog. This option provides the minimum volume box that contains all the points in the cloud. This box can be quite helpful in visualizing a cloud when you have less points displayed or in detecting outliers in a cloud (Figure 26-3).

- **Axis Aligned Box.** This option defines a box that is oriented with the World coordinate system in job file. Its the default because it and requires little initial computation.
- **Minimum Orientation Box.** This box defines the minimum volume necessary to contain the data.

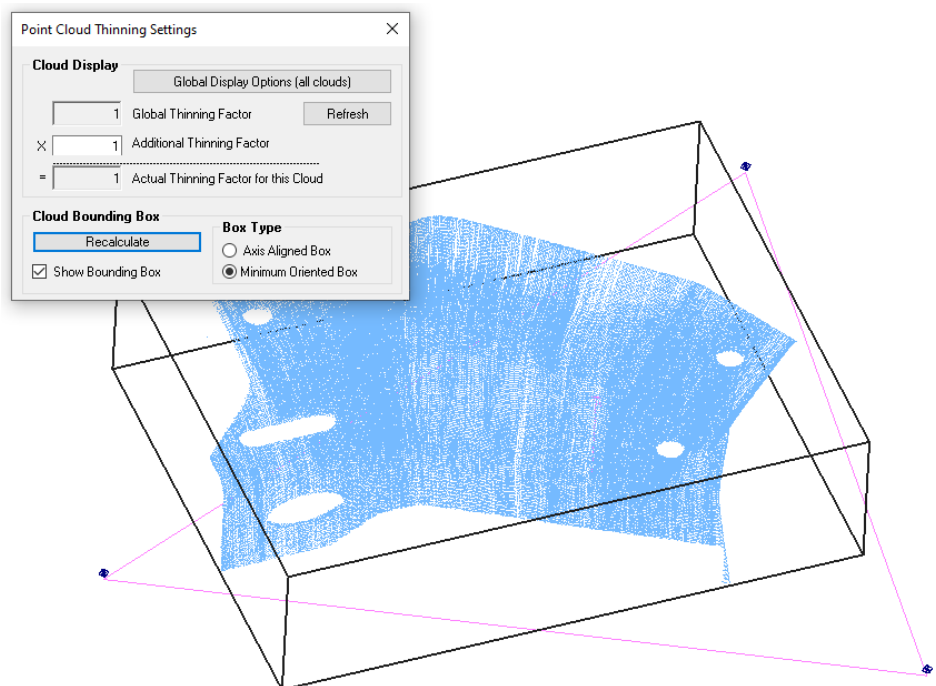


Figure 26-3. Example Bounding Box control found through the Thinning Settings button

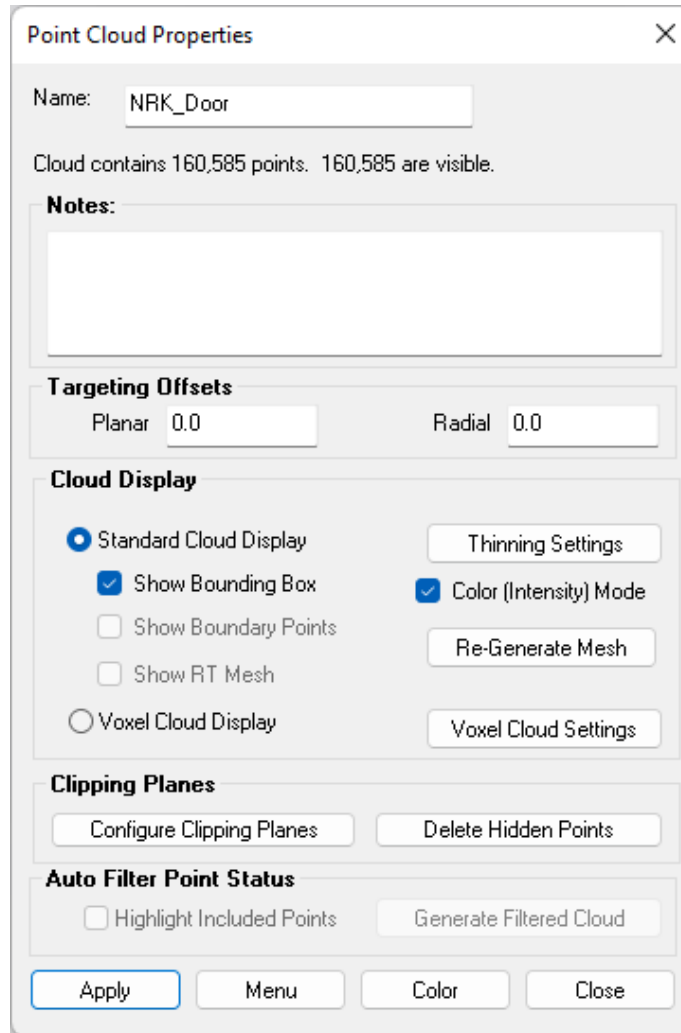


Figure 26-4. Cloud Properties Dialog

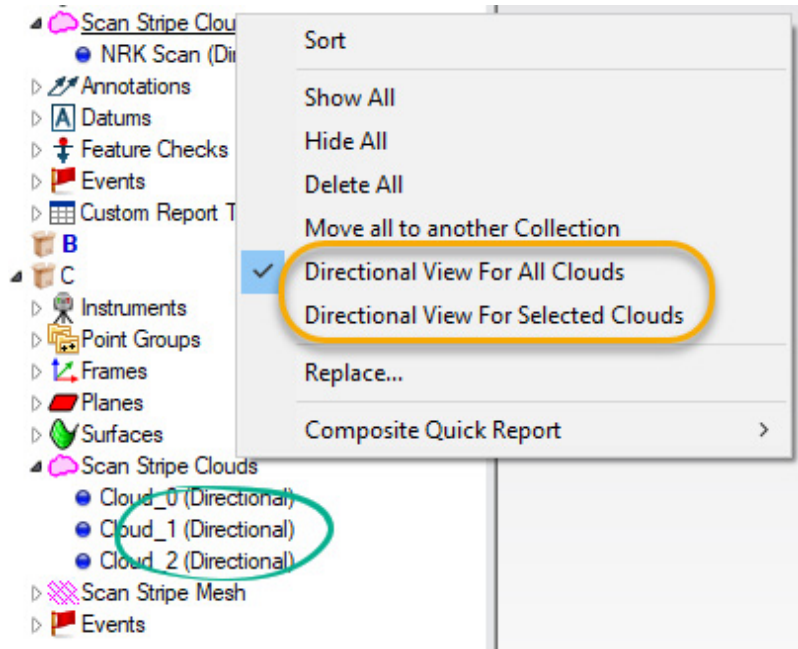
Directional View Control

Scan stripe clouds have an additional directional display control that can be enabled or disabled. This mode allows view of cloud points only when aligned within 60 degrees of the probing direction. This hides cloud points measured on the backside of a part making view of front side coverage much easier.

This mode can be enabled for an individual point cloud through its Right-click menu. It can also be set for all clouds, or a selection of clouds, through the Scan Stripe Cloud category right-click menu.

When enabled the cloud will have a (Direction) identification appended to the cloud name (Figure 26-5).

Figure 26-5. Directional cloud display controls through the Scan Stripe Clouds category R-click menu.



Color (Intensity) Mode

Both RGB or Intensity can be saved as part of a point cloud. When this information is available, it can be used to colorize each individual cloud point while in the standard cloud display mode. This color information will replace the standard cloud color and can be enabled or disabled through the use of the **Color (Intensity) Mode** check box in the cloud properties menu (Figure 26-4). Only certain instruments and import formats (such as E57 or PTX) currently include RGB color or Intensity information.

Cloud Point display can also be edited based upon the cloud point color and intensity information. This is done by right-clicking on the cloud and selecting **RGB Filtering**. This will open a filter control dialog that will allow you to limit the displayed cloud points based upon a max and min thresholds that can be applied to Red, Green and Blue independently (Figure 26-6).

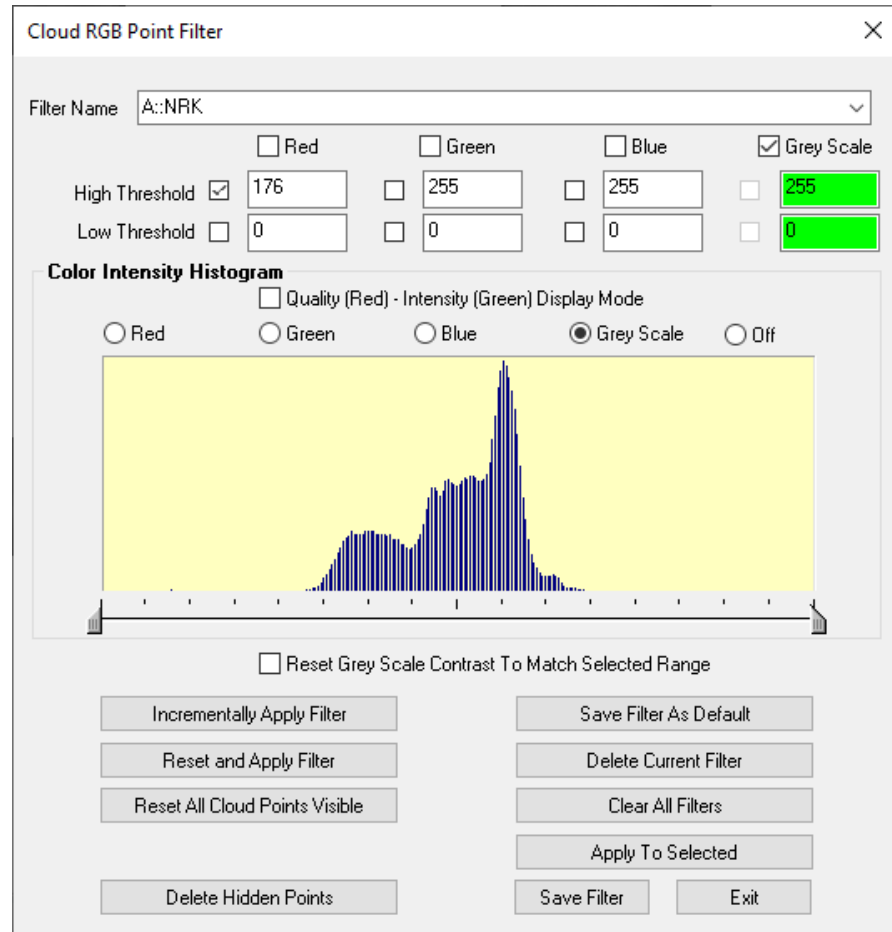


Figure 26-6. RGB Filter controls used to control cloud display based on saved cloud point parameters.

The check boxes next to the individual color components can be used to temporarily isolate the graphical display of the colors for the visible cloud points. This can be helpful for an instrument like the Leica ATS600 that saves the incidence (broadening) value in the Red color scheme while using Blue and Green to identify intensity. By unchecking the Blue and Green components the points will be displayed only using the red component making it easier to determine the ideal threshold to set in order to isolate the display of the desired cloud points.

Display of the select color or combination of colors will be updated when the Apply Filter buttons are pressed.

The individual controls function as follows:

- **High/Low Threshold.** These check boxes and numerical values define the filter threshold that will be applied to the cloud points. These values should be between 0 and 255.
- **Incrementally Apply Filter.** The selected settings are applied to the cloud as it is currently displayed without any consideration for hidden points.
- **Reset and Apply Filter.** The selected settings are applied to

the entire cloud regardless of its current display status.

- **Reset All Cloud Points Visible.** This option will reset all the cloud points such that they are visible regardless of color.
- **Save Filter As Default.** This option applies the current filter settings as a Default filter available to all clouds.
- **Delete Current Filter.** Removed the current filter from the list of available filters.
- **Clear All Filters.** This option removed all the saved filters such that only the default filter will remain.
- **Apply to selected.** This button operates as a batch function allowing the selected settings in the current cloud to be applied to a selection of other point clouds.
- **Save Filter.** This option allows you to apply the current filter settings to the currently selected Filter. Remember that at any time new filters can be created by typing a name into the **Filter Name** field and saving the current settings.

A simple example of how to use this filter is as follows. Assume you have scanned a black and white target and you want to use this filter to identify the center point of the target ([Figure 26-7](#)):

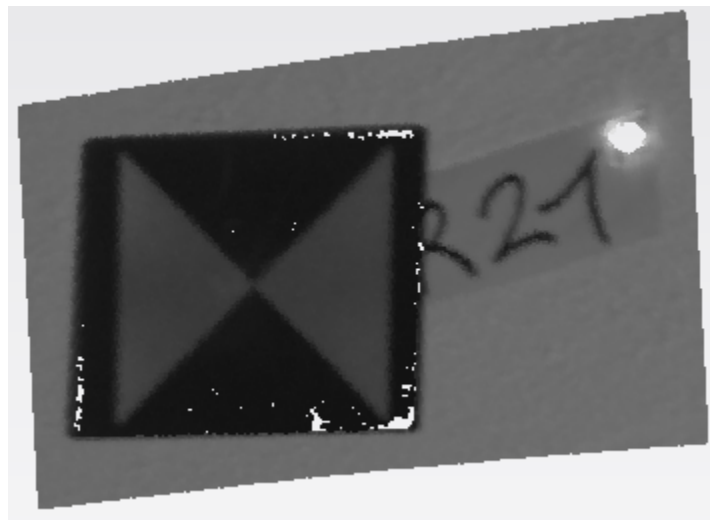


Figure 26-7. Scanned black and white location target

4. Right-click on the cloud in the graphics (or in the tree) and select **RBG Filtering**.
5. Check the *Grey Scale* check box and the *Low Threshold* check box below it.
6. Enter a mid range value (45 worked here) to separate the dark or lower return points and hide them by pressing the **Reset and Apply Filter** button. When you do the dark points will be hidden (see [Figure 26-8](#)). The threshold can be adjusted and the display of the cloud points edited as desired to extract clean

lines on the target.

Once isolated each triangle can easily be selected graphically and the center point constructed.

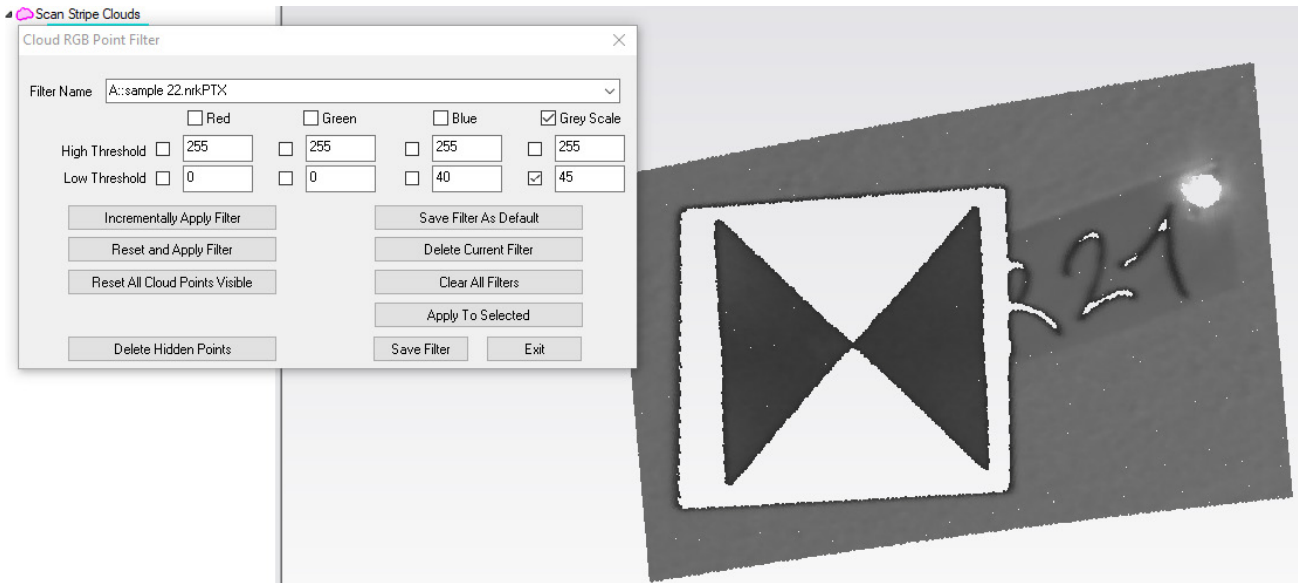


Figure 26-8. Scanned Target with the specified filter settings applied.

The **Color Intensity Histogram** was added to help in selecting the correct color values to use in the filtering process. It also has a *Quality* and *Intensity* mode which is more intuitive for new users than editing the green or gray scale values depending on the instrument (Figure 26-9).

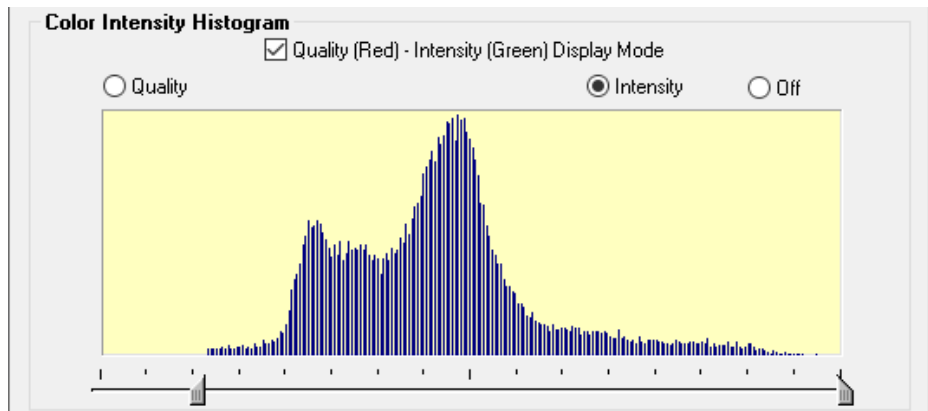


Figure 26-9. RGB Histogram

The **Quality (Red) - Intensity (Green) Display Mode** check box can be used to switch the controls for direct RGB selection to either Quality or Intensity only. The slider can then be used to position the cut off threshold logically based upon the frequency distribution of the data.

Adjusting Contrast

The contrast in the image can also be adjusted to the extents of the selected data range. This can be very helpful when trying to read text

or identify visual cues from the intensity data. To do so, check the **Reset Grey Scale Contrast To Match Selected Range** check box at the bottom of the histogram available in Grey Scale mode (Figure 26-10).

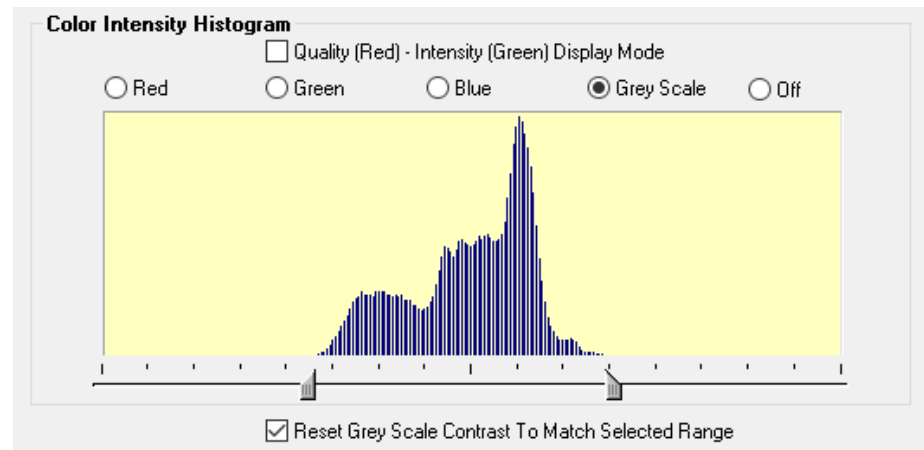


Figure 26-10. The Reset Grey Scale Contrast option can be found when in Grey Scale mode.

The benefits of this mode can be seen here when trying to detect chalk markings on a part (Figure 26-11).

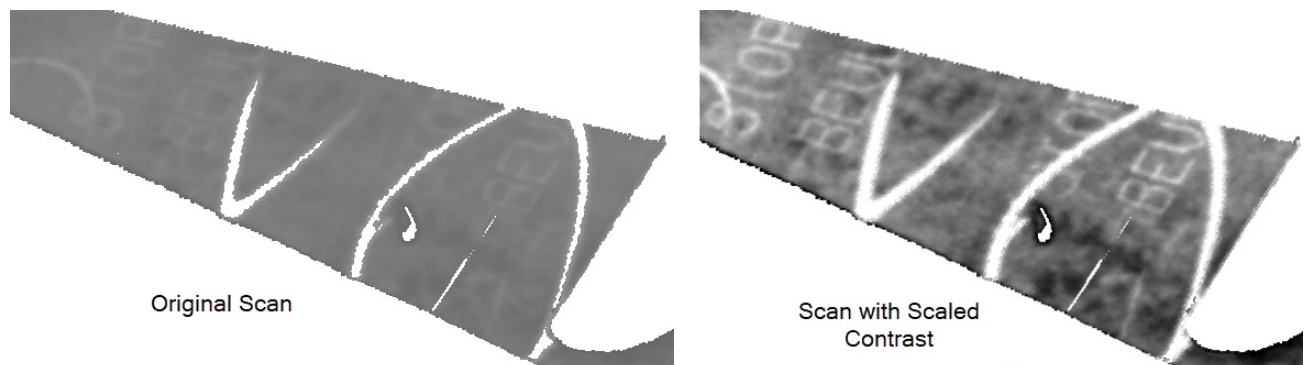


Figure 26-11. Scaled Contrast

Using RGB Filtering with Leica ATS600 in SpatialAnalyzer -

<https://youtu.be/m11ufArEkJU>

Boundary Points

Boundary Points are an additional cloud display option within a cloud's properties menu. This display mode includes only specific cloud points that define breaks in the scan line. For example a scan line that crosses a hole will be filtered to include only the two points on either side of the hole, where data was last measured. These boundary points can be used for feature extraction or edge detection. These can be viewed individually or in addition to the real-time mesh display.

Within the right-click menu for a Scan Stripe cloud is an option to *Generate Boundary Points Cloud*. This option will build a new cloud from the original

Real-Time Meshing

Real-Time Meshing

A real-time mesh can be enabled during scanning. This can be quite useful as an easy to see coverage indicator and provides a surface like rendering of the data (Figure 26-12).

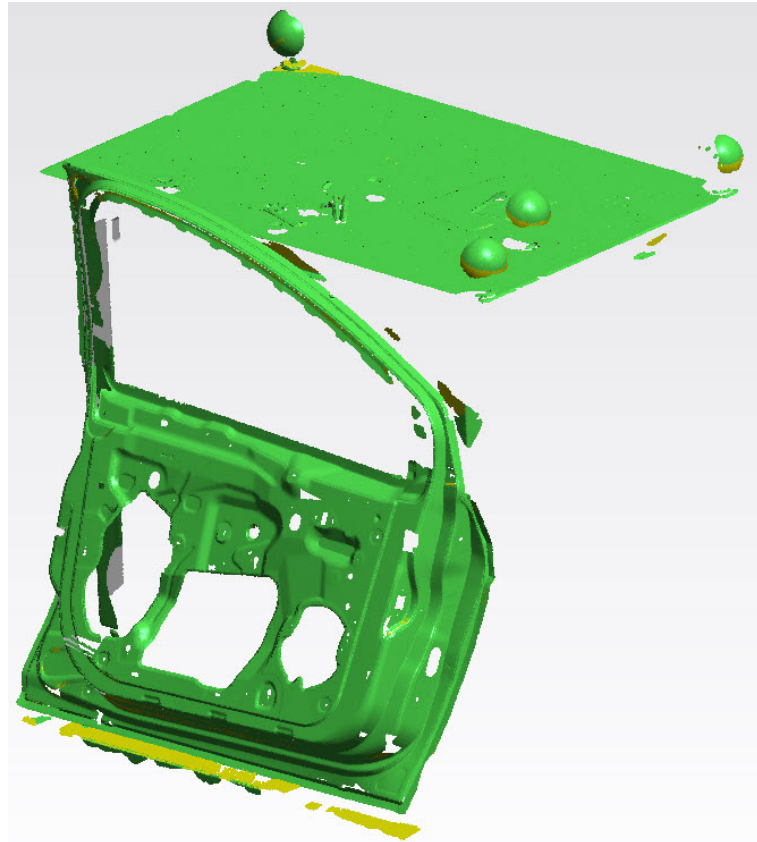


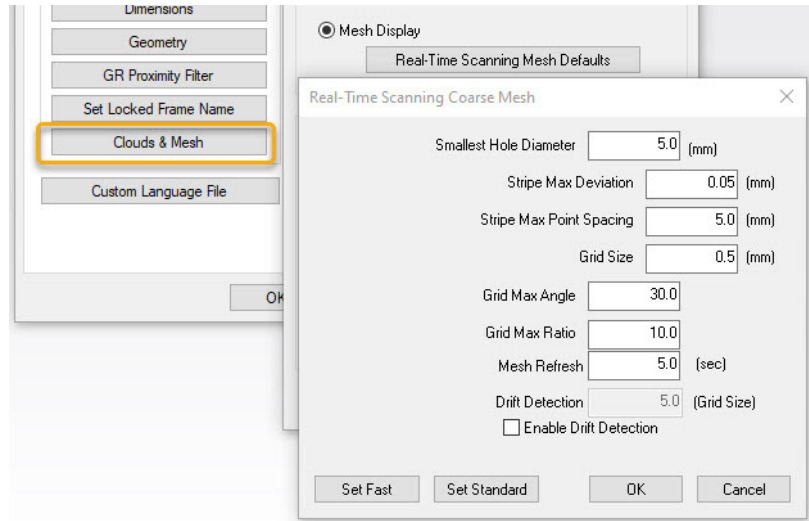
Figure 26-12. Example Real-Time mesh.

In order to enable the Real-Time Mesh go to the *User Options* and on the *Display* tab select *Clouds & Mesh* (Figure 26-13). Within this default control is an option to enable the *Mesh Display*.

The mesh is displayed in two colors.

- **Green.** Indicates good mesh quality
- **Yellow.** Indicates the scan was at a sharp angle relative to the mesh grid. The mesh is colored yellow when the scan to surface angle is greater than $\frac{2}{3}$ the **Grid Max Angle**.

Figure 26-13. Default Real-Time Mesh Controls within the User Options



Defaults can be set using the predefined *Fast* or *Standard* options. Or your settings can be customized with the following controls:

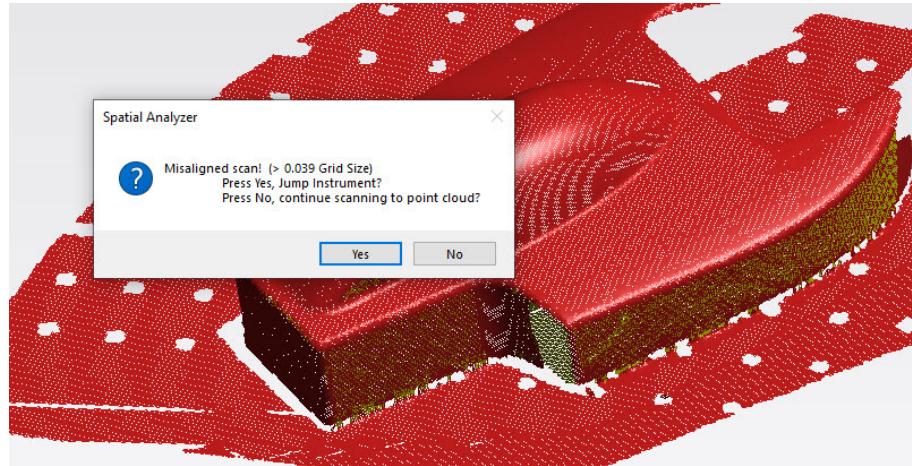
- **Smallest Hole Diameter.** Defines the expected hole size in the mesh. Setting too large a value with could cause holes to be filled. This field can be used as a mater control for the following three settings (*Stripe Max Deviation*, *Stripe Max Point Spacing* and *Grid Size*).
- **Stripe Max Deviation.** The maximum strip distance to be connected when building a mesh.
- **Stripe Max Point Spacing.** The maximum distance within a scan line to bridge within the mesh.
- **Grid Size.** Defines the average triangle edge length to use in the meshing process.
- **Grid Max Angle.** This setting works as a rejection threshold beyond which data with too steep a probing vector will be either rejected or marked as suspect.
- **Grid Max Ratio.** Maximum ratio of the triangle area with respect to an equilateral triangle. Basically how long and thin a triangle can get without being broken into smaller triangles.
- **Mesh Refresh.** The mesh refresh rate controls the rate at which new mesh segments are added to the graphics during measurement.
- **Drift Detection.** The drift detection threshold controls notification of possible drift during scanning. Expected values are between 0 (no notification) and 10 or 10 times the size of the mesh grid.

At any time you can also right-click on a scan stripe cloud and generate a finalized.

Real-Time Scanning Drift Check

When enable, the Real-Time mesh also offers a live scanning drift check notification. This offers the ability to Jump Instrument to New Location if the newest measured mesh segment differs in alignment from the rest of the mesh by a given amount. This **Drift Detection** threshold is set within the Real-time mesh settings.

Figure 26-14. Scanning Drift Check will notify you if a part has moved by highlighting the segment in red and displaying a warning.



- **Yes.** Pressing **Yes** will jump the instrument to a new location, split the last scan pass from the original cloud and build a new one associated with the new instrument ready to be re-aligned.
- **No.** Pressing **No** offers an option to ignore the warning and continue scanning.

If you press No its quite possible that there is a misalignment between different scan passes. SA also offers an option to align scan passes within a cloud which may improve data integrity in a case like this.

Note -- If you want to disable this notification process for any reason you can do so by entering "0" for the Drift Detection [...](Grid Size) value.

Voxel Cloud Display

Voxelization (enabling the voxel processing) provides an alternative to using the Cloud Display Control to help visualize large point cloud files by providing a 3D grid or volume based sub-sampling option. It also provides an alternative to vectors for displaying cloud to object deviations or other Surface Analysis displays (see [Figure 26-15](#)).

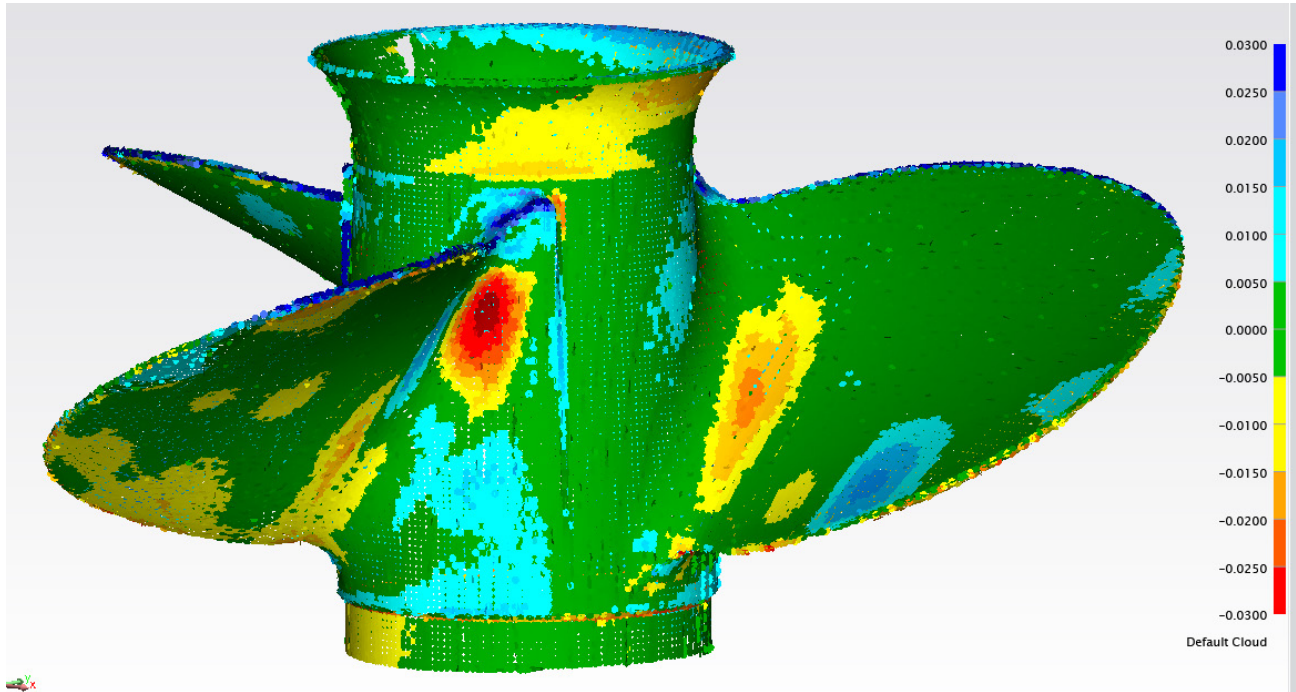


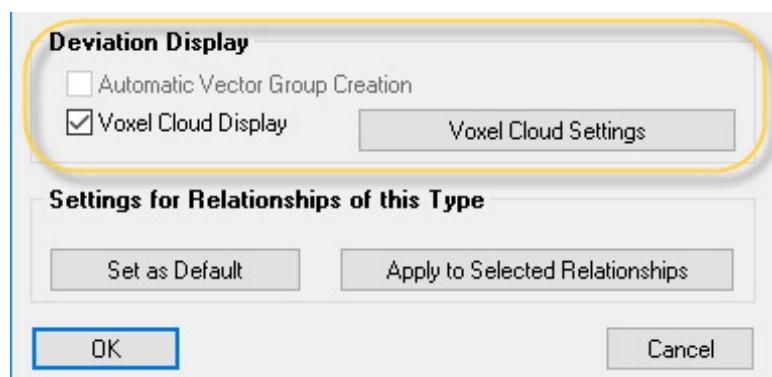
Figure 26-15. Voxel display of deviations from a scan of a propeller.

How it works is you specify a grid size or volume and then only the point closest to the average for a specified volume, which we call a “voxel”, is displayed. Displaying a cloud in this way creates a roughly uniform density display of cloud data cross the entire scan region, and the size of the voxel point (displayed as a screen oriented disc or blotch) can be adjusted as a percentage of the volume used in its computation.

Enabling Voxel Display for an Existing Cloud

The Voxel cloud display can be enabled either through a cloud’s properties (Figure 26-4) or as an alternative vector display option from within a Cloud to Objects relationship (Figure 26-16).

Figure 26-16. Voxel Cloud Settings available through both the relationship and cloud properties dialogs.



The Voxel Setting dialog provides the ability to adjust the display as

needed with the following controls:

- **Voxel Size.** This entry field defines the length of a single side of the 3d volume used to identify each voxel. Much like defining a grid for what data to keep, this control defines the data density. The **Auto Detect** button provides a convenient way to compute an estimated best size to balance detail and screen performance.
- **Minimum Point Count Per Voxel.** This value defines the minimum number of cloud points required in a volume in order to compute a voxel. If there are fewer cloud points than the specified number the voxel will be ignored. This provides a means to ignore outlier points.

Traditional vectors reports consider cloud point deviations no different than points. You can sub-sample and you can ignore edge projections but its still just considering the points directly. Voxels are different. They are volume oriented. In each voxel volume the cloud points are processed and only the single cloud point closest to the average in that volume is considered. This dramatically down samples the cloud statistics and distributes the analysis much more equally across the part. In addition there is some built in outlier rejection. If a considered volume only has 1 or 2 points it, it will be ignored, which is ideal for ignoring noise from a scanner. This is an adjustable threshold (**Min Point Count Per Voxel**) that defaults to 3. If you set this value to 6 for example there has to be 6 points in that volume or no voxel will be built in the zone. The advantage in this method is that it is not just smoothing by a distance but instead it preserves the actual scanned data. If there is a blemish in the part then removing outliers by a distance threshold may hide those bad places on the part. Voxels will not.

If you change the Min Point Count Per Voxel to 1 and turn the option to ignore edge projections off you should see the Total and Valid point counts are equal (see [Figure 26-17](#)).

Figure 26-17. Voxel Cloud Deviation Report displayed for a Point Cloud to Object Relationship

Voxel Cloud to Objects Relationship NRK Scan to Objects (Reported in A: WORLD)	
Min (mm)	-7.4954
Max (mm)	32.6474
Average (mm)	1.5977
Min Abs (mm)	0.0109
Max Abs (mm)	32.6474
Average Abs (mm)	1.8508
RMS (mm)	4.3426
Total Point Count	15560
Valid Point Count	15560
Invalid Point Count	0
Projection Mode	Surface to Offset Point

This statistic is basically telling you how many voxel volume's were processed and ignored. The Voxel Size defines the volume considered so with larger point clouds you will want a larger volume. For quick graphics displays it seems that keeping the total displayed number of voxels to roughly 200K is a good rule of thumb for a reasonable laptop. The Auto Detect button tries to do that for you but is generally a little conservative.

Voxel Display Settings

The Voxel Display Settings are used to control the color and size of the blotch used to display the voxel.

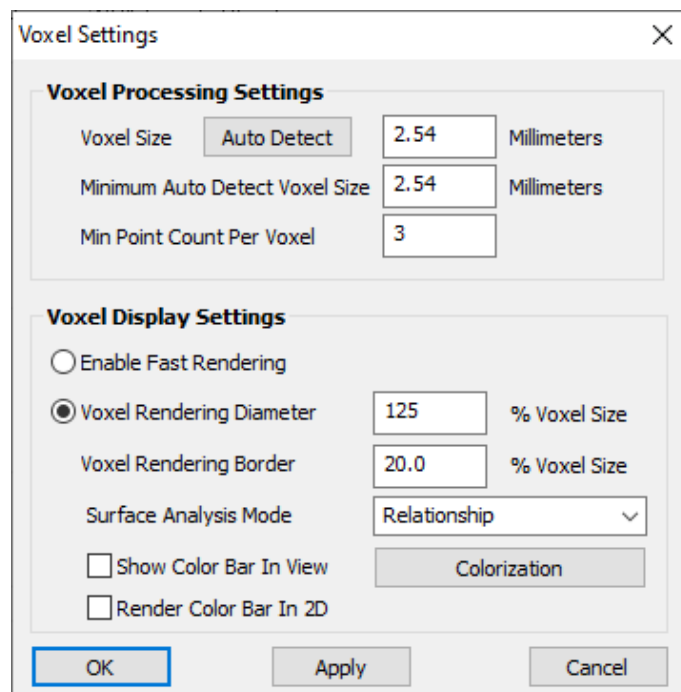


Figure 26-18. Voxel cloud settings dialog

- **Enable Fast Rendering.** This check box provides a method to disable blotch display and show voxels as regular cloud points only.
- **Voxel Rendering Diameter.** Voxels are displayed as blotches or round discs oriented toward the screen, unless Fast Render mode is enabled. The size of these blotches is controlled as a percentage of the **Voxel Size**. This value can be adjusted to fill gaps or more clearly separate individual voxels. For a smooth colored display a slightly larger value is best such as 125%.
- **Voxel Render Border.** The depth and 3D nature of a voxel display is dependent upon a black border around the voxel. The thickness of this border can be adjusted if the voxels overlap greatly and the color is overly black.

Using Voxel Colors for Deviation and Analysis

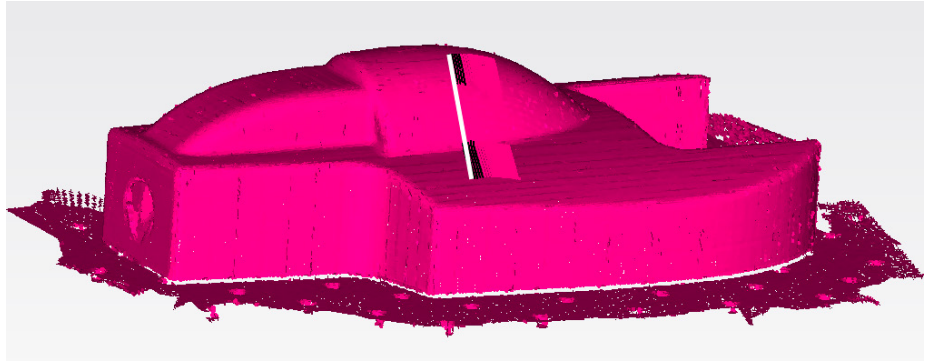
Surface Analysis Mode

Each voxel point can be colored to display different information using the following settings:

- **None.** Displays using the cloud color. This color is set when the voxel display is enabled so if the cloud color is changed after voxel display is set then the voxels will need to be recomputed for the color to change.
- **Relationship.** In this mode, each voxel is colored based upon the distance information in the relationship. A single cloud to object relationship needs to be defined using this cloud for this mode to be available. A single relationship can only control the colorization of a single cloud, and if a second relationship is added this mode will no longer be available.
- **Normals.** Displays the XYZ direction vector for the voxel in RGB color based upon the working frame.
- **Curvature.** This setting will colorize the voxels based upon the depth or extent of the data within each voxel. Changes in the depth distribution of the voxel relative to a fit plane for that same data can help identify transition zones.
- **Deviation RMS, Max, Avg.** Provide a way to display the voxel colorization based upon each voxel's statistics such as: RMS, Max and Average values independently. The extents for each voxel can then be compared to the surrounding voxels.
- **Pseudo Surface.** This option was added to help visualize the cloud data in 3D. Each voxel point is oriented based upon a plane fit using the surrounding voxel points and colorized using surface lighting (Figure 26-19).
- **Cloud Comparison Deviation.** This mode is only available in a cloud generated from a cloud to cloud query. In this case the voxel colors display the deviation results (see "Cloud to Cloud Query" on page 417).

The voxel processing takes place following the scan data arriving in SA so you will see a scan pass line appear first and then be converted to a surface voxel as you scan.

Figure 26-19. Pseudo Surface voxel display



Basic Cloud Display Controls & Part Comparisons In SpatialAnalyzer - <https://youtu.be/lzTV3IK4rVg>

Simple Surface Deviation Display Process

To visualize the deviations between a cloud and a CAD model using Voxels follow these steps:

1. Import the CAD Model, scan the part and align to it. Then disconnect from the instrument.
2. Build a Cloud to Objects relationship between the cloud and the CAD model.
3. Open the Relationship properties and enable the Voxel Cloud Display. Select Voxelized Cloud Settings to adjust coloring or tolerance settings (Figure 26-16).

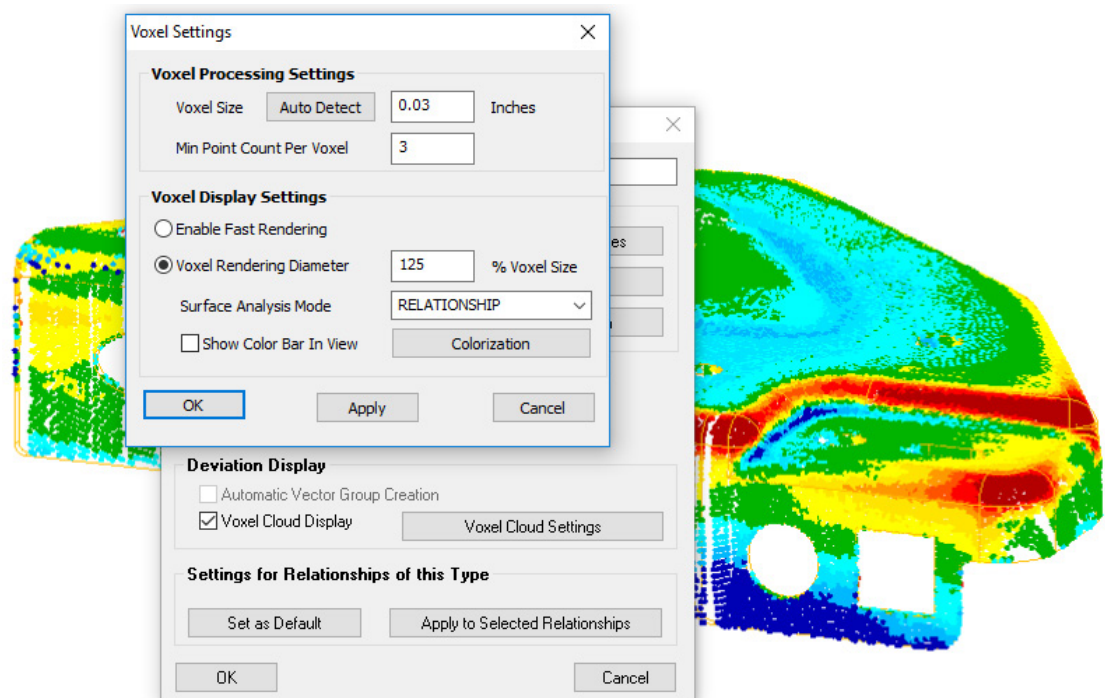


Figure 26-20. Voxel Settings for Surface Deviation Display

- To add callouts displaying voxel deviation at particular locations, right click on the relationship and select Add Voxel Callout (Figure 26-21).

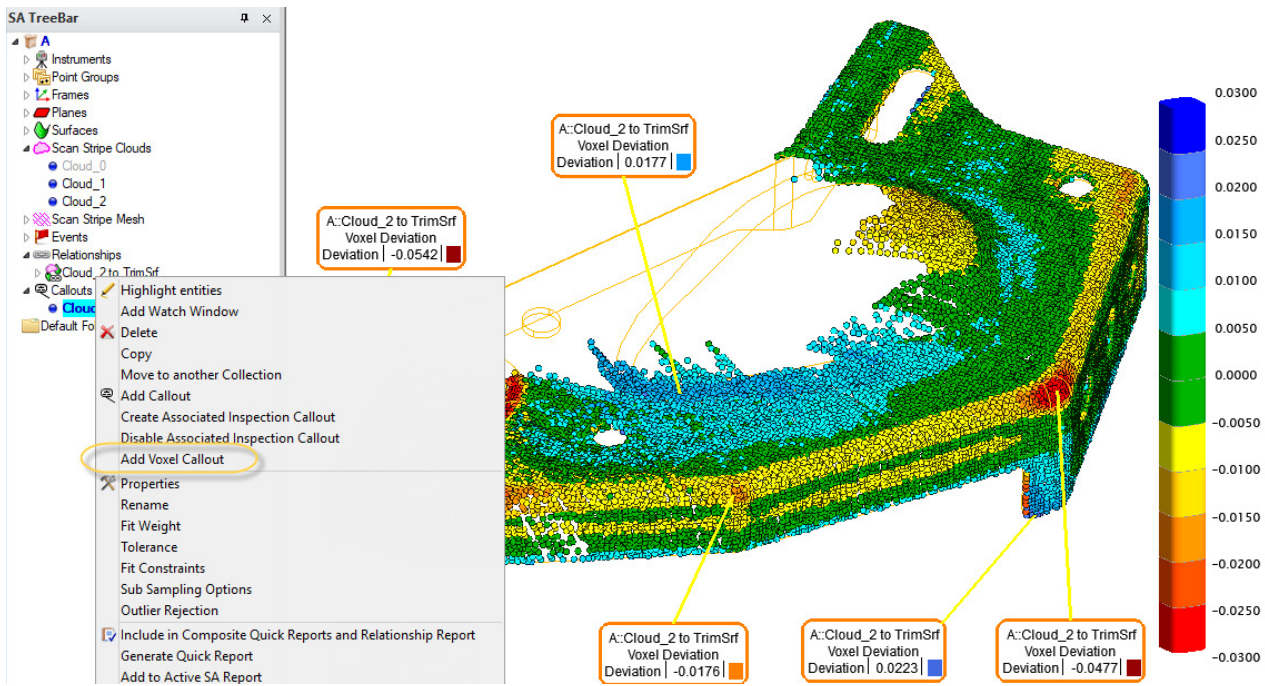
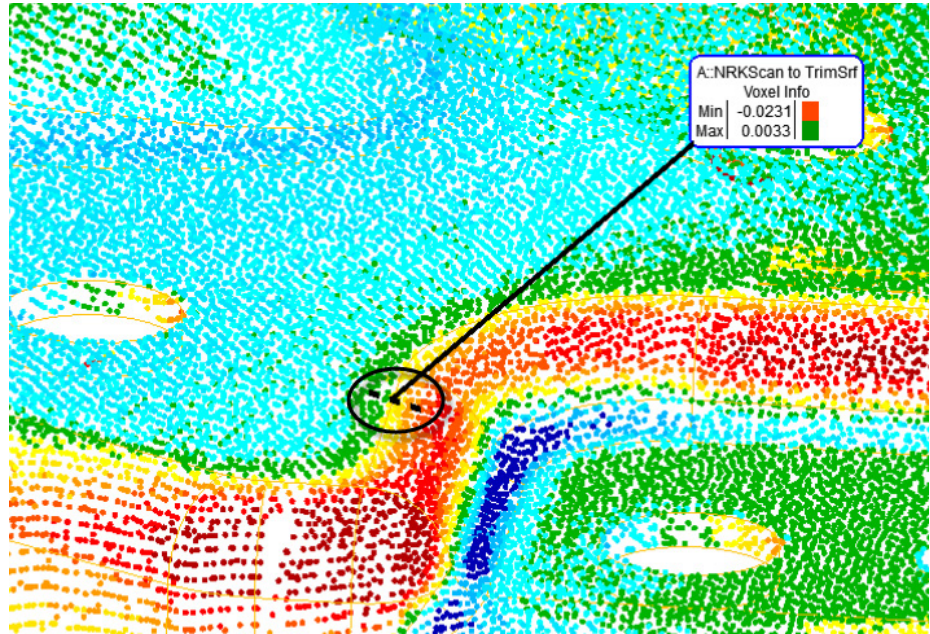


Figure 26-21. Adding Voxel Callouts

Voxel callouts can be added through the right-click properties of a Cloud to Objects relationship or through the standard Callout drop down list. When you add a voxel callout you select a zone of voxels depending on how closely you are zoom in or out. The callout will by default identify the highest and or lowest voxel value for that region assuming all the voxel deviations are the same sign. If your selection bridges a zone where there are positive and negative values in that selection the callout will also display both the high and low values (Figure 26-22). The leader line will attach to the specific voxel being reported if selected directly or include a region based upon the zoom extent at the time the callout was added.

Figure 26-22. Voxel Callout including multiple voxels over a span bridging zero



Clipping Planes

Clipping planes can be extremely useful in removing unwanted data from a scan. For example, in many cases a part is set upon a flat surface and then scanned and the splash measurements of the flat surface are included with the measurements of the part. Clipping planes provide a way to easily removed measurements of the flat surface leaving only the measurements of interest.

Clipping planes can be set globally but are defined through a point cloud's properties and must be edited there as well. These clipping planes can be edited at any time to be used in 2 ways:

- **Defined in advance.** If a clipping plane is defined for a cloud prior to scanning the data obscured by the clipping plane will be deleted automatically.
- **Defined after scanning.** If a clipping plane is defined after the scan has been conducted then the hidden data is only hidden and not removed until the **Delete Hidden Points** command is used.

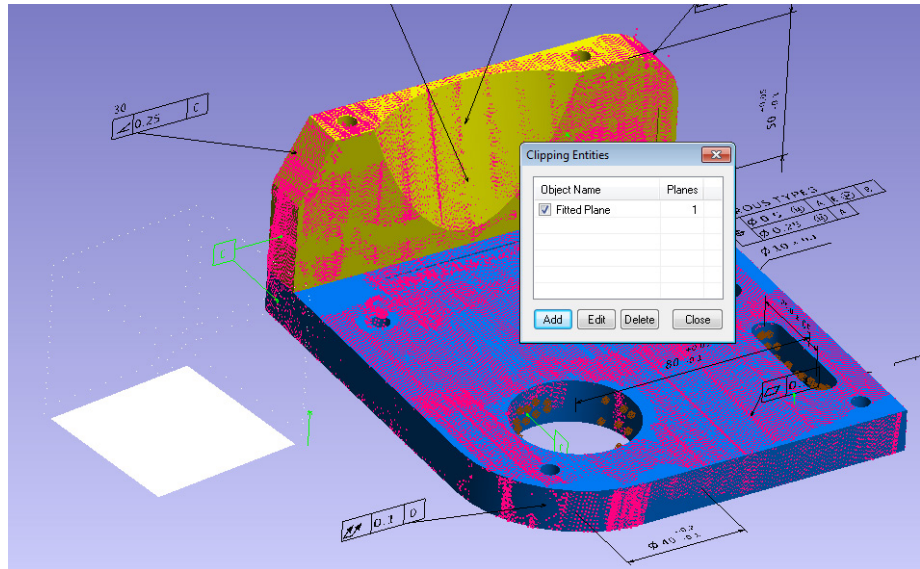
To define a clipping plane do the following:

1. Open the clouds properties dialog
2. Press the **Configure Clipping Planes** button
3. Press **Add** and select an object.

Each object in the Clipping Entities dialog (Figure 26-23) defines the base frame for a particular clipping volume. By default the XY plane of this object's base frame defines the clipping plane used. Therefore if

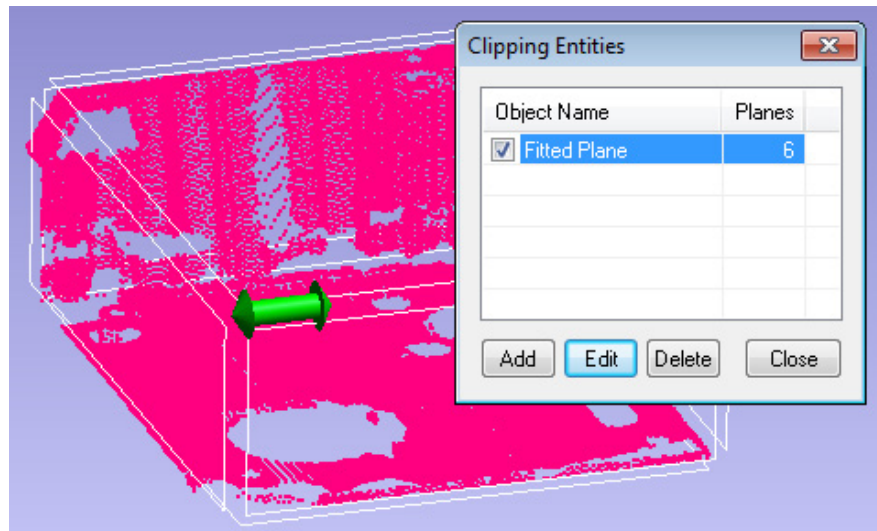
you choose a plane directly it will be used as expected. However, you can choose any object as the reference object to define this plane.

Figure 26-23. Clipping Plane



Each Clipping Entity defines a coordinate system which defines not only a clipping plane but a full clipping cube or volume. Any of the 6 different clipping planes or sides of a cubic volume can be enabled and used as clipping plane to fully enclose a specific region of data. Therefore, up to 6 clipping planes per object can be used and as many objects as you want can be added all of which can refine the clipped volume. With multiple clipping entities defined just about any specific volume can be partitioned to either eliminate or isolate cloud points from the cloud (Figure 26-24).

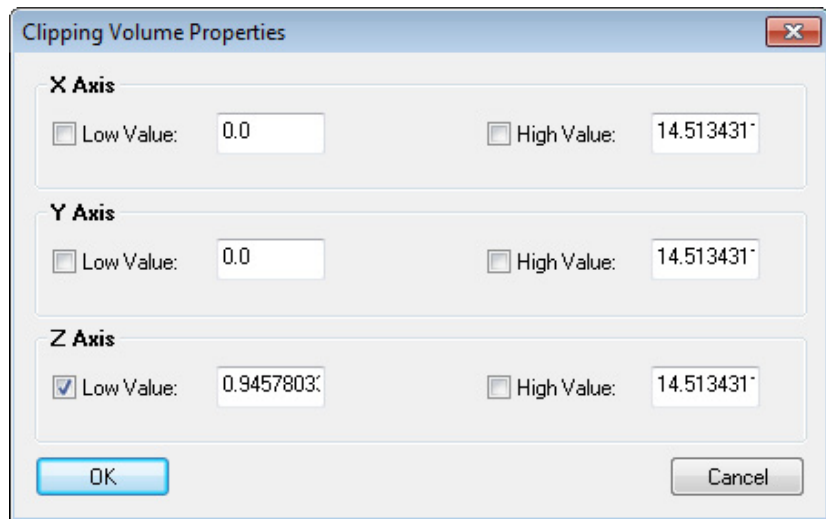
Figure 26-24. Clipping Entities and controls



The clipping volume box can be directly controlled graphically. By placing the mouse over an edge or corner the an arrow will appear graphically allowing you to drag that boulder to either increase or decrease its length. Particular clipping faces of this box can be turned

on or off by hovering over them until the side illuminates and then clicking. For more explicit control these controls can be accessed through the **edit** button (Figure 26-25).

Figure 26-25. Edit Clipping Volume Controls



To define a clipping plane for multiple clouds

1. Go to **View > View Control > Clipping Planes** (Alt + x)
2. Define a global visual clipping plane as desired
3. Press the **Set as Cloud Default** button

Clipping planes are cloud specific and set through the properties of each cloud independently. However, by setting up a clipping plane through the view control and applying it as a Cloud Default you are setting up each new cloud that is built to inherit this clipping plane definition when it is built. This allows you to easily set a single plane as the default clipping plane for a job and then measure as many clouds as you would like. Each newly built cloud will then have this defined clipping plane. In order to then turn off this feature just go back to the global control again and reset the cloud default for the job.

Cross Section Clouds

Cross Section Clouds provide an additional method to visualize the data subdivided in segments at a distance across a part. This can be very helpful in understanding trends in deviation along the length of a part and can help to simplify the display regionally.

Cross Sections Clouds in SpatialAnalyzer -

<https://youtu.be/JbawRLRHpGE>

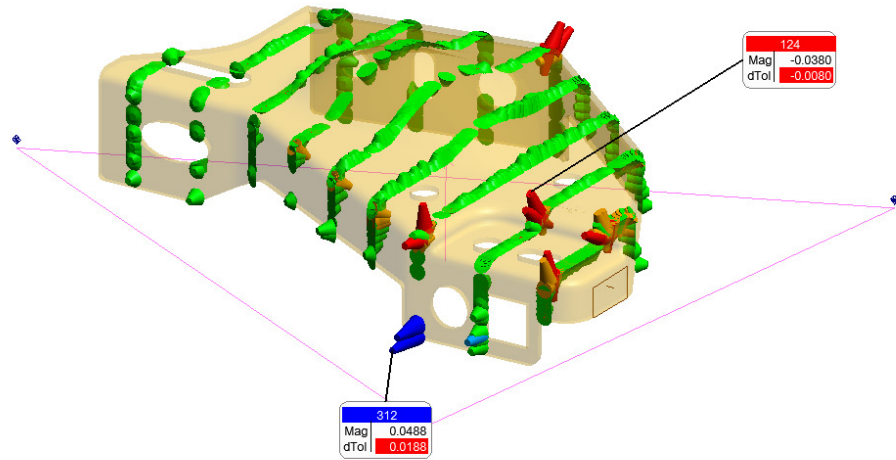


Figure 26-26. Cross Sections

Cross sections can be built from the menu command **Construct>Point Clouds>Auto-Filter to Cross-Sections**.

Cross Section Configuration

The Build Cloud Cross Sections dialog provides the necessary controls to define the filtered cloud sections. As inputs, it requires a **Cross Section Reference Object** which defines the direction axis for the cloud processing. This can be any object that defines a direction and will use the Z axis of that reference object.

When using **Planar Sections**, this reference object defines the direction the normal vector of the filter planes will lie, and for **Cylindrical Sections** it defines the central axis for the coaxial cylinders. When the direction object is a b-spline, planes will be added normal to the spline at the specified distance along its curve, using the spline direction at that point.

Alternatively, the *Use Selected* option can be used to select from pre-existing planes to use in the filter process.

Figure 26-27. Build Cross Sections Dialog

The following parameters can be set:

- **Start Distance.** Defines the distance from the origin of the reference object to the first cross section
- **Section Spacing.** The distance between the cross sections
- **Proximity Threshold.** Defines how close data must be to the cross section to be included in the resulting cloud
- **Maximum Section Count.** Defines the total number of cross sections from the start location.
- **Limit Planar Cross Section Radius.** When enabled this option restricts point selection to a radial proximity to the axis of the reference object.
- **Project Proximate Points to Reference Plane.** This check box provides an option to project selected points onto the cross section plane to ensure perfectly planar results.
- **Cloud Thinning Settings.** This button provides access to the additional cloud thinning controls (see “[Cloud Filtering and](#)

Sub-Sampling” on page 570).

Cross Section Cloud Properties

In addition to the standard cloud properties dialog, Cross Section Clouds provide a section display control panel.

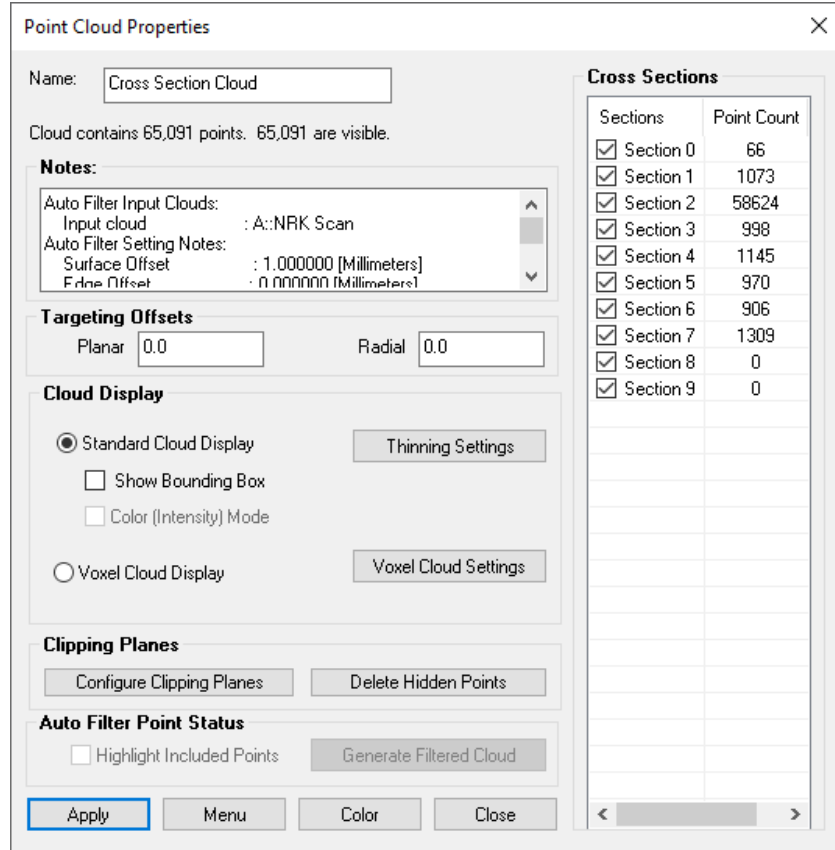


Figure 26-28. Cross Section Cloud Properties Dialog

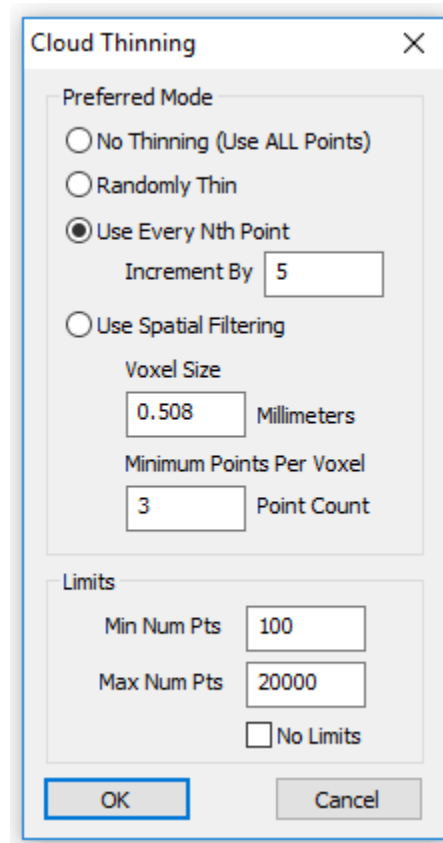
This cross section panel provides display control for each of the individual cross sections, allowing you to turn on and off individual sections as desired.

- Double click on any individual section to display only that one section.
- Highlight a selection of sections using either CTRL-click or SHFT-click, then R-click to select.

Cloud Filtering and Sub-Sampling

Many functions within SA provide access to an additional layer of thinning either as part of the association or evaluation process. This thinning of the input cloud can greatly increase processing speeds by reducing the size of the resulting cloud. The standard cloud thinning controls include the following options (Figure 26-29):

Figure 26-29. Cloud Thinning



- **No Thinning (Use All Points).** Selecting None will disable input cloud thinning.
- **Random Thin.** A random selection of points will be taken from the point input cloud until the **Desired Count** is reached. This method ensures an unbiased selection of cloud points for use. A random selection will pick different points each time it is run but ensures the same number of cloud points are returned.
- **Every N-th Point.** Provides either of two filter methods. If the **Maximum Point Count** is set to 0 it provides a simple decimation strategy where it will return 1 cloud point for every **Skip Interval**. When used together with a **Max Point Count**, the **Skip Interval** is effectively expanded to equal the Input count divided by the **Max Point Count**, ensuring no more than the **Max Point Count** will be returned.
- **Use Spatial Filtering.** This filtering mode uses a moving filter window to walk through the data evaluating points within the specified 3D volume or **Voxel Size**. It attempts to return the most representable cloud point from the available data within that region by retaining the point closest to the average. The **Minimum Points Per Voxel** control provides a means to exclude regions of low density as part of the sub-sampling process. This can be helpful in excluding outliers.

Auto Filter Point Status

Note: Auto Filtering using Spatial or Voxel thinning methods will not be saved or activate the Auto Filter Points Status window.

A Scan Strip Cloud contains the scan-line probing information used to build fine meshes. Unfortunately, many auto filter operations produce standard clouds that don't contain this information as a result of the sub-sampling process.

The *Auto Filter Points Status* section in the cloud properties window (Figure 26-4) provides a solution to this problem. When an auto filter operation is performed, the cloud points detected as part of the filter are saved in the original scan stripe cloud (Figure 26-30).

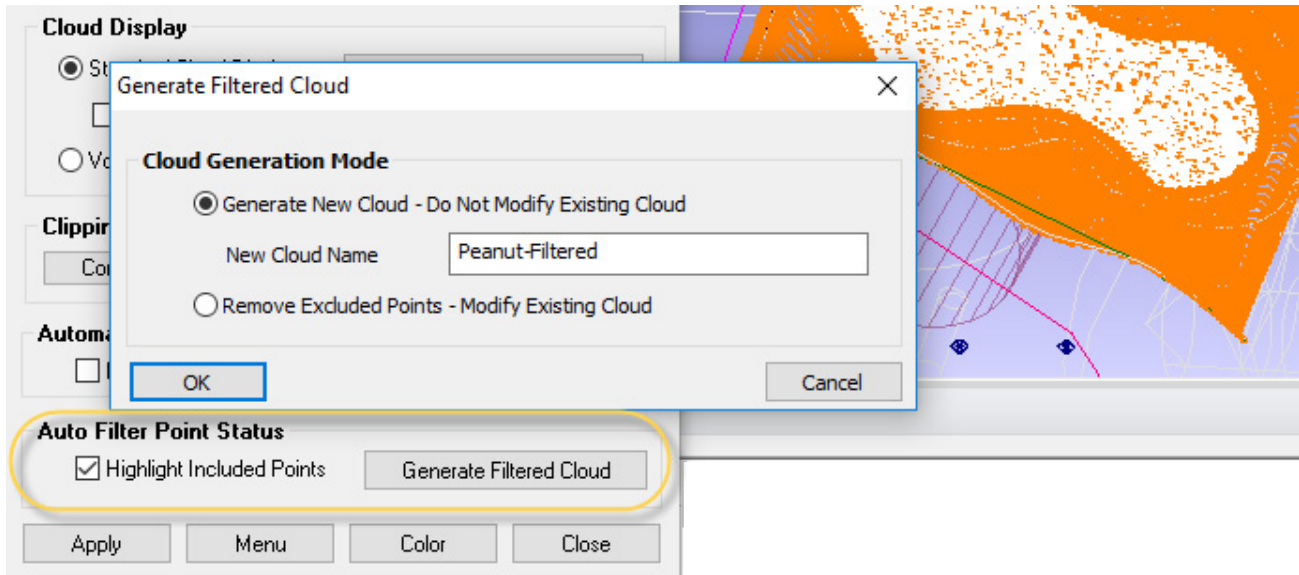


Figure 26-30. Auto Filter Point Status Controls

This provides a means to highlight the marked cloud points and build a new point cloud using that data for either meshing or reverse engineering purposes.

Point Cloud Construction Options

Create Empty Scan Stripe Cloud

Building an empty scan stripe cloud allows an operator to configure the cloud prior to measuring. For example a clipping plane can be set up in advance as well as the meshing options defined as needed prior to the scan.

From Existing Clouds

Creates a new point cloud using a copy of the cloud points contained within the selected clouds and providing a full selection of sub sam-

pling options.

From Existing Clouds Uniform Spacing

This command builds a new cloud from the selected clouds with a reduced number of cloud points. It returns an approximately uniform distribution of cloud points keeping only the closest cloud points at the desired point distance specified (see “[Cloud Filtering and Sub-Sampling](#)” on page 570).

From Existing Cloud Points

Creates a new point cloud copying the selected set of individual cloud points.

Random Data

Builds a randomly generated point cloud using the desired number of points within the specified X,Y, Z limits.

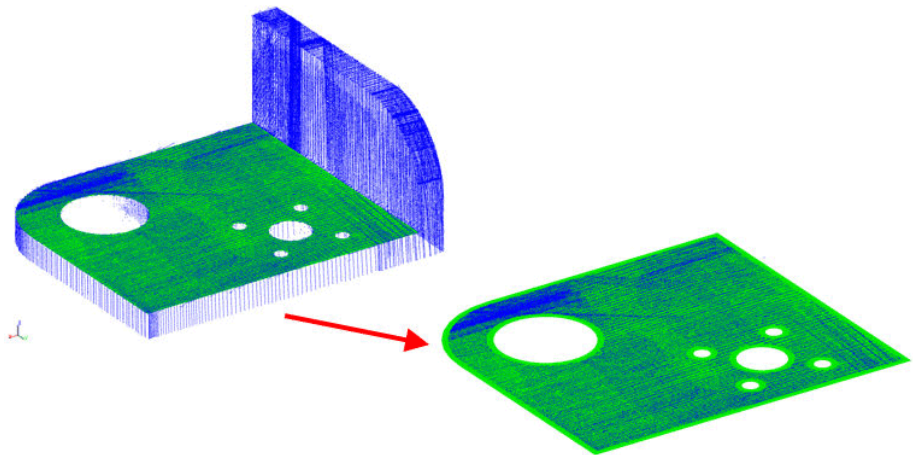
From Existing Point Group

Converts a point group into a point cloud.

Auto Filter to Faces

New point clouds can be build from other clouds based upon the proximity of cloud points to a selection of surface faces using: **Construct>Point Clouds>Auto Filter to Faces** (Figure 26-31). Choose a proximity threshold or “Surface Offset” in either the positive or negative direction or both. You can also limit the selection to points a certain distance away from the edge of the selected surface faces using the “Edge Offset” control.

Figure 26-31. A cloud created based on proximity to a surface face.



A variation on this command can also be found for feature inspection

and automatic cloud association:

1. To filter clouds only to the surfaces faces used by a GD&T Datum or Feature Check a similar command can be found through **Construct>Points>Auto Filter to Faces** and **GD&T>Feature Inspection Auto Filter**. For more information refer to “[Feature Inspection Auto Filter](#)” on page 671
2. To filter clouds only to the surfaces faces referenced by Geometry Relationships a similar command can be found through **Relationships > Geometry Comparison > Autofilter to Nominal Geometry**. This function is described in the following section.

Auto Filter to Vectors

New point clouds can be build from other clouds based upon the proximity of cloud points to a selection of vectors or vector groups using **Construct>Point Clouds>Auto Filter to Vectors**. This option provides the ability to define a cylindrical zone about a vector with a predefined radial proximity and distance from the vector origin and build new clouds for each vector that fit these requirements([Figure 26-32](#)).

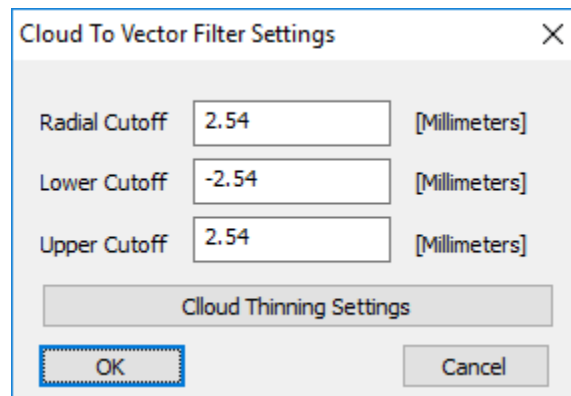


Figure 26-32. Auto Filter to Vectors Dialog Controls

Auto Filter to Lines

New point clouds can be build from other clouds based upon the proximity of cloud points to a selection of lines. Controls are the same as are provided for filtering to Vectors.

Auto Filter to circular Arc B-Splines

New point clouds can be build from other clouds based upon the proximity of cloud points to a selection of B-splines. Cloud points beyond a plane constructed normal to the spline at either end will be ignored.

On Object's Vertices

This function operates the same as the construct points command but returns a single cloud with the object vertex points included as cloud points.


Auto Filter to Cross -Sections

This option builds a cross-section cloud ((see “[Query Point Cloud to Point Cloud](#)” on page 586).

From Frame Set

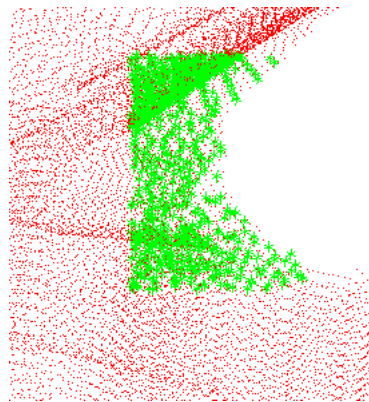
This function builds a single cloud with cloud points built from the origins of the frames in the selected Frame Set. It will ask for a starting and ending frame and build the origins for the frames in that range.

Building Points from Point Clouds

 **Note:** Creating more than about 20,000 constructed points will result in increasingly sluggish interaction with the software. When dealing with very large datasets, use cloud points.

You can convert cloud points gathered from a scan into normal constructed points as needed. These newly-constructed points will retain the offsets saved with the cloud and will appear as constructed points. This can be helpful for a number of other functions that require regular points as inputs.

Figure 26-33. Points constructed from cloud points.



To Convert Cloud Points Into Constructed Points:

1. From the menu, select **Construct>Points>From Cloud Points>Pick Cloud Points**.
2. Select the points to convert.
3. The *Make Points from Clouds* dialog will appear ([Figure 26-34](#)).


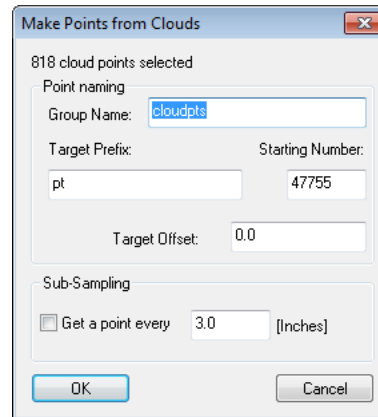
 **Note:** If the cloud is being thinned, only the thinned subset will be selected.

Figure 26-34. The *Make Points from Clouds* dialog.



4. Specify the group name, an optional target prefix and starting number, and an offset for the resulting points. If desired, select the *Get a point every xxx* option to subsample the cloud.
5. Click **OK**. If subsampling, confirm the dialog.

Converting Clouds to Points

Similar to converting cloud points to points, you can convert an entire cloud to points. The procedure is almost identical to creating points from cloud points.

To Create Points from Clouds:

1. From the menu, select **Construct>Points>From Cloud Points>Pick Clouds**.
2. Select the points to convert.
3. The *Make Points from Clouds* dialog will appear.
4. Specify the group name, an optional target prefix and starting number, and an offset for the resulting points. If desired, select the *Get a point every xxx* option to subsample the cloud.
5. Click **OK**. If subsampling, confirm the dialog.

Filtering Clouds to Points

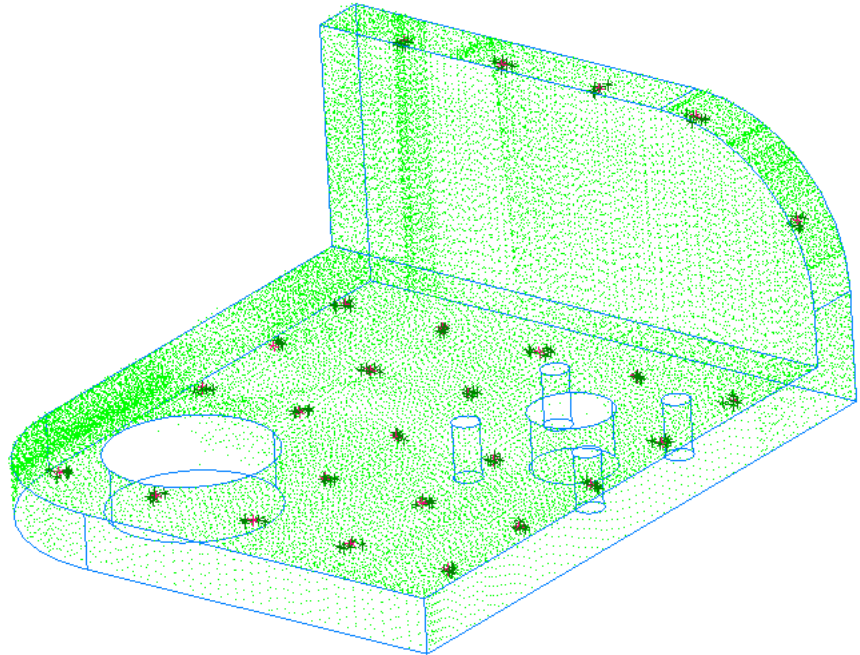
Given a set of clouds and a set of points, SA can filter the clouds to the points. You provide a desired proximity and maximum number of points, and choose the existing points to serve as filter points. SA will create a new point group containing a subsampled set of up to the prescribed number of points (within the proximity) for each filter point (Figure 26-35).

This command is useful, for instance, when you have a set of nominal points on a design surface that indicates where you'd like to measure, and you have a surface scan of the part. The command will filter the clouds to those points.

To Filter Clouds to Reference Points:

1. From the menu, select **Construct>Points>From Cloud Points>Filter Clouds to Points**.
2. Select the points to use as filter points.

Figure 26-35. Filtering a cloud to a set of reference points. Here, up to 5 points were created for each reference point.



3. Pick one or more clouds to filter.
4. At the prompt, specify the resulting group name.
5. Enter a filter proximity—the maximum distance that a resulting cloud point can be from a filter point. Entering **0** will create a point only at the closest cloud point.
6. Specify the maximum number of points to create for each filter point. Entering **0** will pick all points within the prescribed proximity.

Note: You may wish to average the resulting points after this operation.

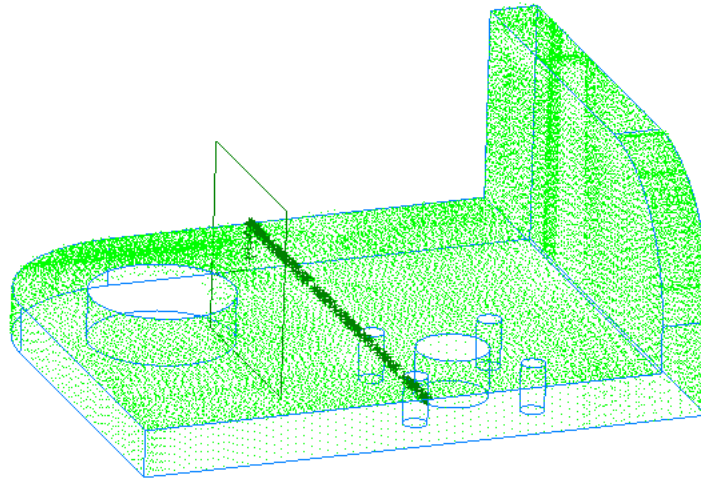
Filtering Clouds to a Plane

You can filter one or more clouds to single a plane. After specifying a proximity to the plane, SA will create a point group or cloud containing all cloud points within the prescribed proximity to the plane (Figure 26-36).

To Filter Clouds to a Plane:

1. From the menu, select **Construct>Points>From Cloud Points>Filter Clouds to Plane**.
2. Select the desired filter plane.
3. Select one or more clouds to filter.

Figure 26-36. Filtering a cloud to a plane.



4. At the prompt, enter the desired destination group name (or cloud name, if you choose to create a cloud).
5. Specify a filter proximity, which indicates how close a cloud point must be to the plane to be used. If you enter 0 , all source cloud points will be used.
6. The *Filter Clouds to Plane* dialog will appear (Figure 26-37):

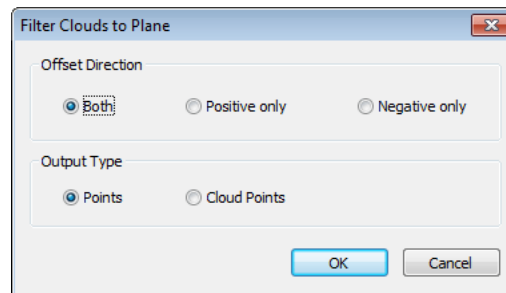


Figure 26-37. Options for filtering clouds to a plane.

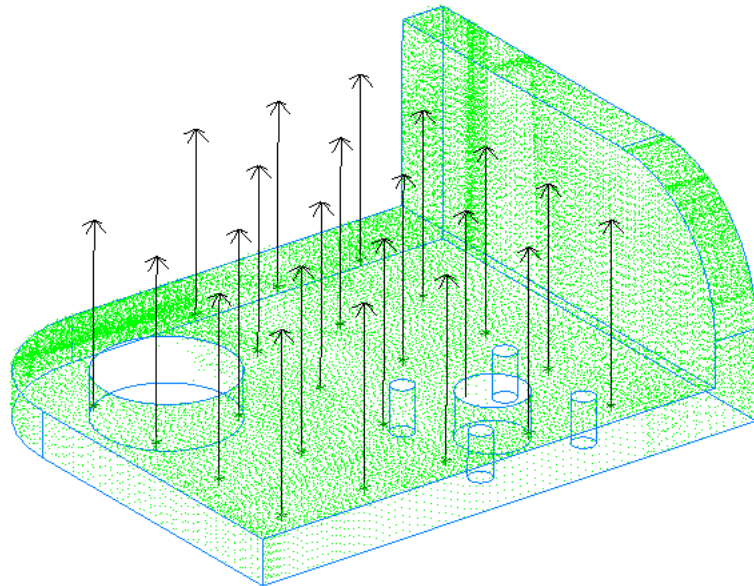
- **Offset Direction.** Determines the location of the proximity region. *Both* accepts points up to the prescribed filter proximity on both sides of the plane. *Positive only* only accepts points on the positive side of the plane, and *Negative only* only accepts points on the negative side of the plane.
 - **Output Type.** Determines the format of the resulting points—should they be *Points* or *Cloud Points*?
7. Select the desired options, then click **OK**.

Filtering Clouds to Vectors by Resolving Points

Given a set of clouds and a set of vectors, SA can filter the clouds to the vectors. You provide a desired threshold around each vector, a proximity to the vector's base, and the minimum number of required points, and SA will—for each vector—extract the points that meet the specified thresholds and average them into a single point (Figure 26-38).

This command is useful, for instance, when you have a set of nominal vectors on a design surface that indicates where you'd like to measure, and you have a surface scan of the part. The command will filter the clouds to those vectors.

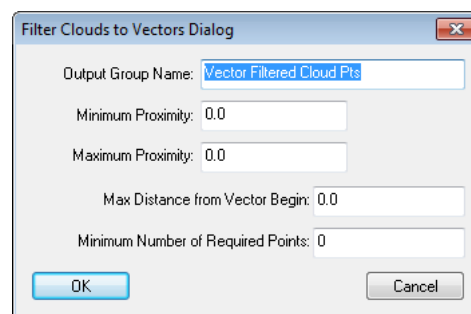
Figure 26-38. This cloud has been filtered to nominal vectors.



To Filter Clouds to Vectors:

1. From the menu, select **Construct>Points>From Cloud Points>Filter Clouds to Vectors-Resolve Points**.
2. Pick the clouds you'd like to filter.
3. Select the vector groups you'd like to filter to.
4. The **Filter Clouds to Vectors** dialog will be displayed (Figure 26-39):

Figure 26-39. The *Filter Clouds to Vectors* dialog.



- **Output Group Name.** The name for the resulting point group.
- **Minimum Proximity.** The minimum allowable distance from a point to the vector to be considered.
- **Maximum Proximity.** The maximum allowable distance from a point to the vector to be considered.
- **Max Distance from Vector Begin.** The maximum distance that a cloud point can be from the base of the vector to be

Note: *Max Distance from Vector Begin* prevents ambiguity when there are two or more coaxial vectors.

considered.

- **Minimum Number of Required Points.** The minimum number of points required for an averaged point to be created. If this minimum is not met, no averaged point is created.
5. Enter the desired options, then click **OK**.

Extracting Center Points From Scanned Spheres

SA can search a point cloud and detect registration spheres within the cloud, and build the center points for those detected spheres. To do so, from the Clouds & Surfaces tab select **Build from Clouds>Points >Auto-Detect Sphere Centers**.

Provide the command with the expected sphere diameter, acceptable RMS tolerance, and minimum number of points required, and the command will detect spherical data matching the proposed diameter from the scan data, fit spheres to them, and create their center points (Figure 26-40).

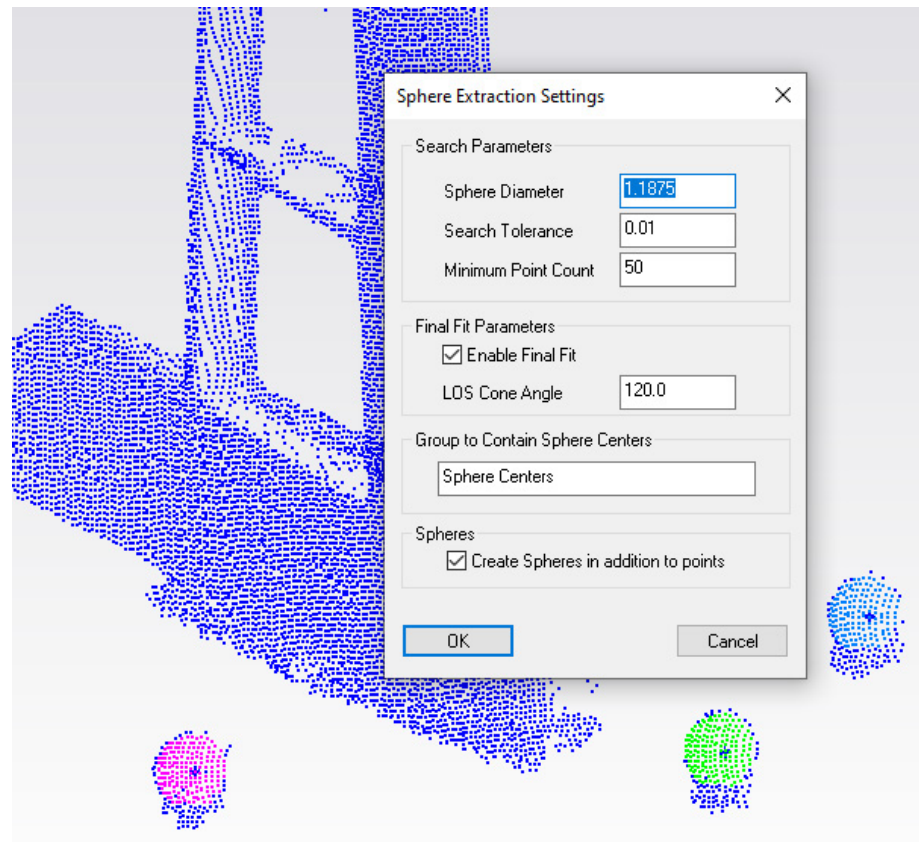
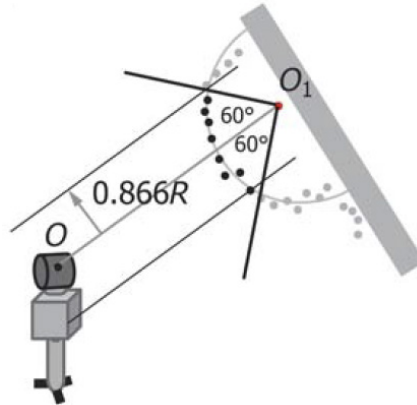


Figure 26-40. Registration sphere centers have been automatically extracted from this scan.

Create point from cloud sphere extraction now includes an option to perform a “Final Fit” option which implements *ASTM E3125-17* fitting algorithms. The point cloud data is filtered so that only a central cone opening angle of 120° is used to compute the sphere and its center point (Figure 26-41).

Figure 26-41. ASTM E3125-17 offers an outlier rejection strategy that uses the instrument's line of sight.



Note that the input cloud must be associated with an instrument to establish the line of sight vector used for cloud segmentation as per this standard which then applies a conical outlier rejection strategy to further refine the fit results.

To Auto-Detect Sphere Centers from Raster Scan Data:

1. From the menu, select **Construct>Points>From Cloud Points>Auto-Detect Sphere Centers**.
2. The *Sphere Extraction Settings* dialog appears (Figure 26-42):

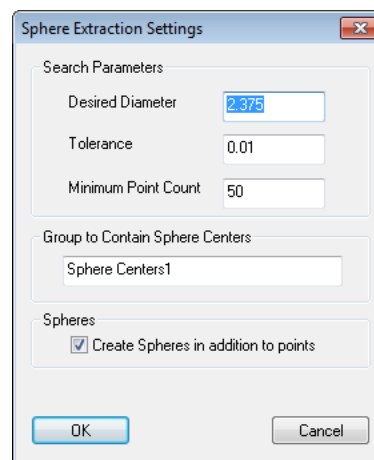


Figure 26-42. Specifying sphere extraction parameters.

- **Sphere Diameter.** The nominal or expected diameter of the registration spheres to detect from the cloud.
- **Search Tolerance.** The maximum allowable deviation for a given point from the desired diameter in order to be considered.
- **Minimum Point Count.** The minimum number of cloud points required to satisfy the tolerance to consider a sphere to be “found”.
- **Final Fit Parameters.** When enabled this option employs the ASTM E3125-17 method for cloud point rejection using the instrument’s line of sight and the **LOS Cone Angle** specified.

- **Group to Contain Sphere Centers.** The name for the resulting group to contain the sphere center points found.
 - **Create Spheres in addition to points.** When enabled, SA will create the spheres (at the desired diameter) in addition to the center points.
3. Select the desired options, then click **OK**.
 4. After calculation, SA will display the **Sphere Auto-Detection Results** dialog (Figure 26-43). This contains the RMS error, maximum error, and point count for each detected sphere.

Name	RMS	Max	Point Count
Sphere 0	0.005	0.010	197
Sphere 1	0.005	0.010	329
Sphere 2	0.005	0.011	250
Sphere 3	0.004	0.010	194
Sphere 4	0.005	0.010	381
Sphere 5	0.004	0.008	89
Sphere 6	0.005	0.009	155

Searched 738783 points using 7723 search tasks.
Search time: 3.6 sec

Figure 26-43. Sphere Auto-detection results.

5. Click **OK** to accept the results, or **Cancel** to cancel.

Cloud Based Alignment/Registration Tools

SA provides a selection of cloud to cloud, cloud to mesh, and cloud to CAD alignment tools that can be used to align clouds and meshes as needed within your job file. The following options include cloud/mesh specific alignment tools. At any time points can be built from cloud points for use in other alignment tools (see [“Building Points from Point Clouds”](#) on page 575).

Aligning Clouds to CAD / Mesh / Clouds

Align Cloud to CAD and Align Cloud to Mesh

An “intelligent” Cloud to CAD or Cloud to Mesh alignment tool is available directly through the cloud’s right-click properties. It provides a convenient way to align a cloud and associated instrument to a CAD model. This function first prompts for the operator to select the CAD surfaces to consider and then performs the alignment reporting the fit results and transform.

This function is designed to be quick and easy to use. It is intended for clouds with nearly full coverage of the CAD and that have already had any splash data or outliers removed. It performs an initial alignment based upon the perimeter of the scan and then refines that alignment. If the alignment is canceled it can be run a second time using a finer detail mesh of the CAD model which will take longer but typically returns better results.

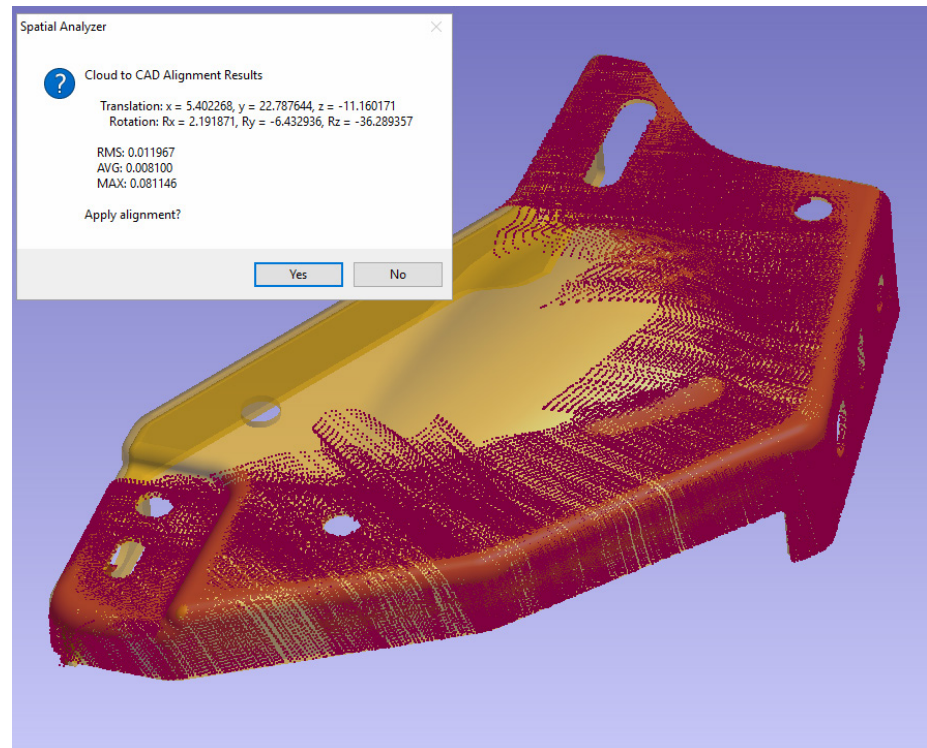


Figure 26-44. Align to CAD

Video demonstration available here-

<https://youtu.be/lzTV3lK4rVg>

Align N Points to Objects

In many cases a simple one click alignment solution is not adaptive enough and a better starting condition is needed. For this purpose we have the *Align N Points to Objects* command (also available through the right-click menu). This alignment tool provides the ability to choose starting points for the alignment which are used for a best fit prior to the final optimization.

While any combination of feature types can be aligned to each other, cloud or mesh to clouds, meshes and CAD models its typically best to build a mesh for the alignment rather than directly aligning cloud points to cloud points because the mesh provides greater flexibility in the alignment and less variability in results.

To use the Align N Points to Objects command do the following:

1. Right-click on the object you want to align (either Point Cloud

or Mesh) and select *Align N Points to Objects*, choosing Cloud, Mesh or CAD as needed. Both a properties dialog and a second graphical window will open to help you select corresponding points on both the moving feature and the feature to align to.

2. Select the **Control Points** button and then click directly on the stationary feature within SA to identify location points (at least 3 Points).
3. Select the **Rough Align Points** button and then click within the second graphics window on the moving feature to identify the corresponding points (Figure 26-45). You can delete the last selected point location with the delete key.

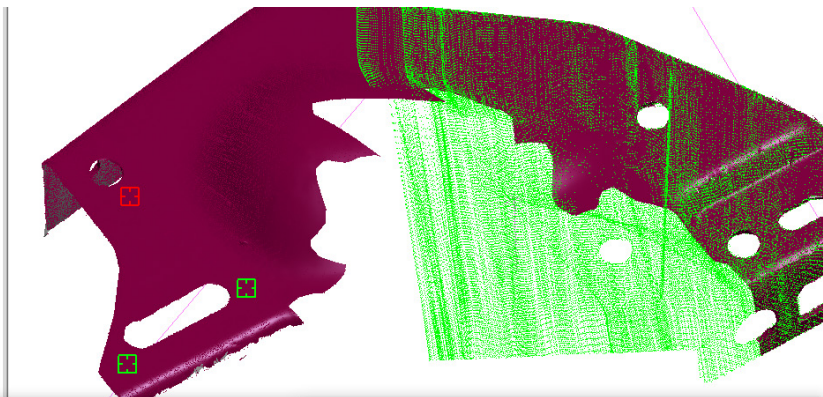
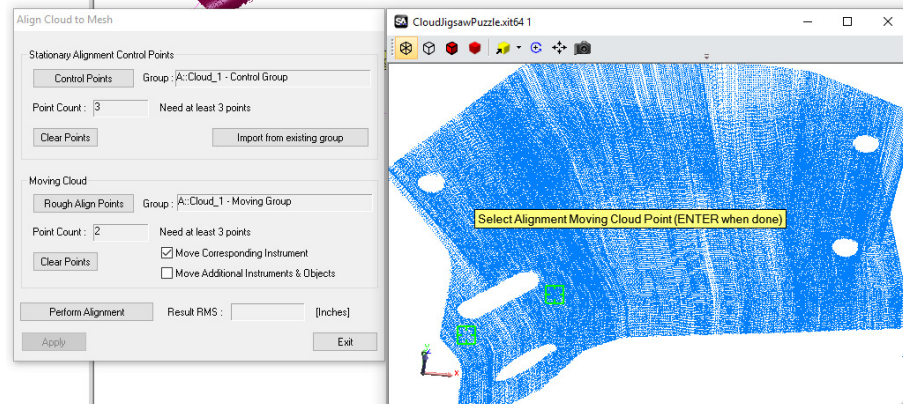


Figure 26-45. Align N Points To Objects



4. When satisfied with the selected points press the **Perform Alignment** button and verify the alignment. You can do so both graphically and by checking the resulting RMS. At any time the applied alignment can be edited until you exit the dialog.

Quick Align Clouds to CAD

Based upon the Quick Align to CAD function, this equivalent command first prompts for selection of alignment points located on the CAD model and then prompts the operator to select the corresponding cloud points from the cloud. These cloud points are then fit to the CAD model at the designated locations.

Point Extraction and Best-Fit

When using registration targets the best way to align those common targets together between scans is to extract the alignment points and then Best-Fit those points together. To do so try the following:

1. Extracting Center Points From Scanned Spheres from each cloud ensuring that the corresponding point groups have the same point names (see [“Extracting Center Points From Scanned Spheres”](#) on page 580).
2. Select **Analysis> Best-Fit Transformation> Points to Points** (Ctrl+B). For more details on the Best-Fit alignment refer to the alignments chapter (see [“Best-Fit Transformations”](#) on page 350).

Cloud to Object Relationship Optimization

Cloud to Object Relationships are very powerful tools. They allow you to define subsampling options as needed and provide a dynamic query that can be used to dial in an alignment to highest accuracy. The same degree of control is provided for cloud points in relationship fitting as is available for use with points. For more details refer to the Relationships chapter (see [“Minimizing Relationships”](#) on page 444).

Refit Scan Passes

In addition to aligning separate clouds to each other a lower level adjustment can be made to align separate scan passes. To do so, right-click on the cloud of interest and select **Refit Scan Passes**. This will build a new, adjusted point cloud aligned to the specified tolerance. The alignment transform and statistics are recorded within the new cloud’s **History**.

Each time a scan is started and stopped a new scan pass is recorded. These passes can then be aligned to each other using the features measured within the scan and adjusting the relative transforms of subsequent scans passes to the first.

This can be helpful in two ways. In cases where an instrument made a significant shift in position, such as shifting to the far side of a part but is still measuring into the same point cloud, this tool can help align the passes to each other. This could correct for a slight instrument calibration error in a particular joint for example. This is very helpful in smoothing a resulting mesh.

Second, if you combine two or more point clouds into a single cloud using a construction command these scans will be marked as separate passes within the resulting cloud. The re-fit scan passes can then

be used to refine the alignment of these separate cloud passes within the cloud with respect to each other.

Query Point Cloud to Point Cloud

The distance between two point clouds can be evaluated directly using a Cloud to Cloud Query. Rather than generating a vector group (which is not optimized for so much data) this operation build a new point cloud which is colorized to display the deviations (Figure 26-46).

Select two point clouds. The first is the reference cloud and the second is the measured object cloud which will be projected onto the reference. The Max Distance determines the furthest distance points will be considered, limiting the size of the query.

Each point is projected in the direction of the local normal of the cloud on a local plane. In other words, the “distance” is not the closest distance to a point of the target cloud, but the closest distance to a local plane of the target cloud.

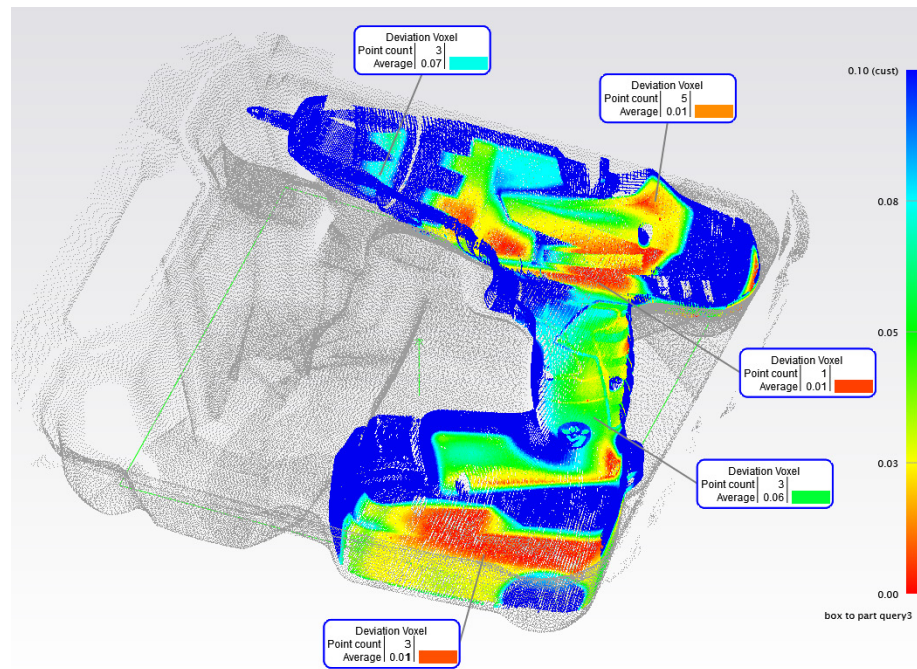


Figure 26-46. Example Point Cloud to Point Cloud Query

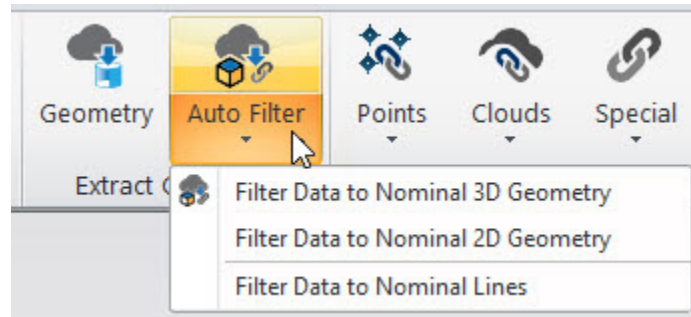
Cloud to cloud queries require a probing direction in order to determine if a distance is a positive or negative number. For this reason white clouds will only return positive magnitude deviations. With two scan stripe clouds the deviation values will return both positive and negative magnitude (starting in SA 2021.3) values as are returned when using a cloud to mesh query.

Feature Extraction from Point Clouds

Auto Filter to Nominal Geometry > 3D Extraction

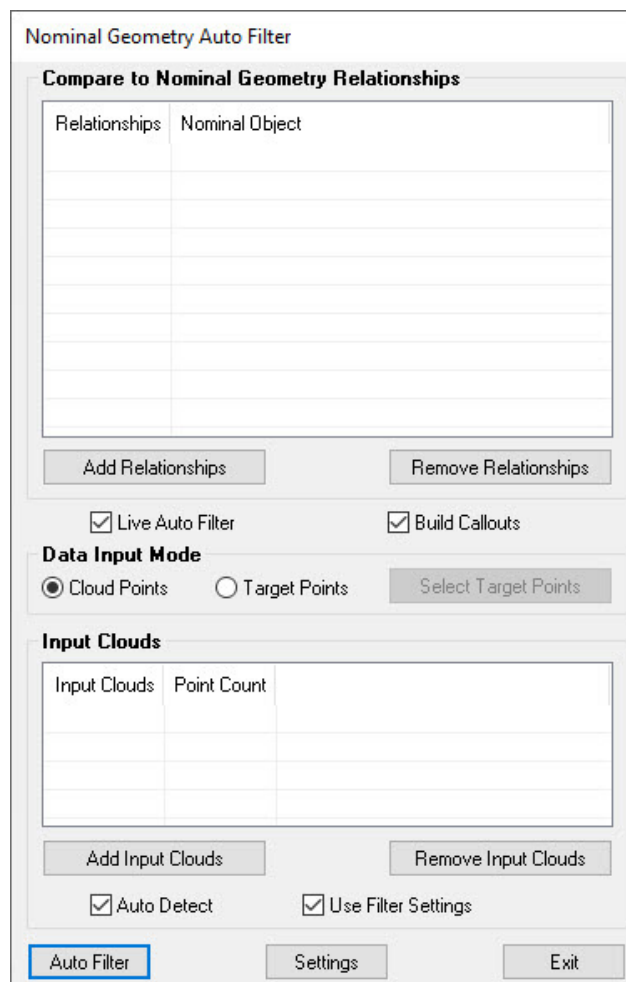
The nominal geometry within a GR-Feature can be used to perform feature extraction based upon proximity. This can be done per feature by directly associating a cloud to it or by using the batch processing option **Features>Extract Geometry> Auto Filter** (Figure 26-47).

Figure 26-47. The 3D Geometry filter is the default option if you press the Auto Filter button on the Features tab.



Selecting Filter Data to Nominal 3D Geometry will bring up the following dialog (Figure 26-48), which offers several modes of operation.

Figure 26-48. Nominal Geometry Auto Filter



Live Auto Filter for Clouds

The Nominal Geometry Auto Filter dialog offers a *Live Auto Filter* option. This allows you to simply select the **From CAD** button and click on the CAD model to build a nominal feature, and automatically filter cloud data based on proximity to the newly created feature.

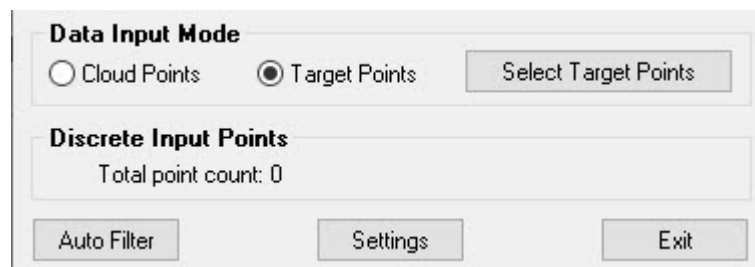
To use this option simply open the dialog and verify that the default options are still enabled.

- **Live Auto Filter.** This option needs to be enabled.
- **Data Input Mode.** This selection must be set to Cloud Points. Live Autofilter is not currently supported for points.
- **Auto Detect.** Cloud Delection option is enabled. This identifies cloud points from visible clouds in the region. If clouds are hidden or you want to limit the clouds used this option should be turned off and clouds directly associated with the extraction process.

3D Extraction with Points

The same feature extraction process can be used to detect points within the specified proximity. The points are first queried to the surfaces to correctly compensate for point offsets and then the same filter process is used. To filter points to GR-Features select the *Target Points* option. This will then enable the **Select Target Points** button (Figure 26-49) which can be used for filtering points.

Figure 26-49. By selecting the Target Points option you can then use the Select Target Points button to choose points to include in the filter.



Full Cloud Based Operations

A set of relationships must be selected for batch processing and filtering. To so use the following options:

- **Add Relationships.** Provides the ability to select a set of pre-define Geometry Relationships.
- **Remove Relationships.** Allows the user to remove the currently selected relationship from the list to consider.
- **Add Input Clouds.** Provides the ability to select a set of point clouds to use for the filter process.
- **Remove Input Clouds.** Allows the user to remove the current-

ly selected point cloud from the list to consider.

The default setting for the 3D Auto Filter operation is to use the saved settings within each feature. This is ideal when working with a pre-built template that includes custom settings for each feature's needs. To turn this off, uncheck the *Use Filter Settings* check box at the bottom of the dialog.

Auto Filter Proximity Settings

By pressing the **Settings** button you will first be prompted to choose from a cloud thinning controls and then offers filter controls.

Cloud Thinning Controls

- **Count Based - Cloud Thinning Options.** This selection mode and control provides access to standard cloud thinning controls including Nth point decimation and limit selection.
- **Spatial Based - Voxel Size.** This thinning options provides a more advanced volume processing approach to thinning. The *Spatial Filtering* size designation defines the size of the volume to be processed to return a single point, and the *Minimum Points* designation provides a means to exclude outliers and ignore volumes without a sufficient point count.

As an alternative specific proximity filter settings can be set by using the **Settings** button.

Using Generic Filter Settings

This filter process takes the selected point clouds and detects from them the points that fall within a specified proximity to the nominal feature. This filter process falls into two categories: 2D and 3D objects and the controls for each are separately designed.

For 3D objects such as Planes, Cylinders, Cones and Spheres there are two basic parameters to consider (Figure 26-50):

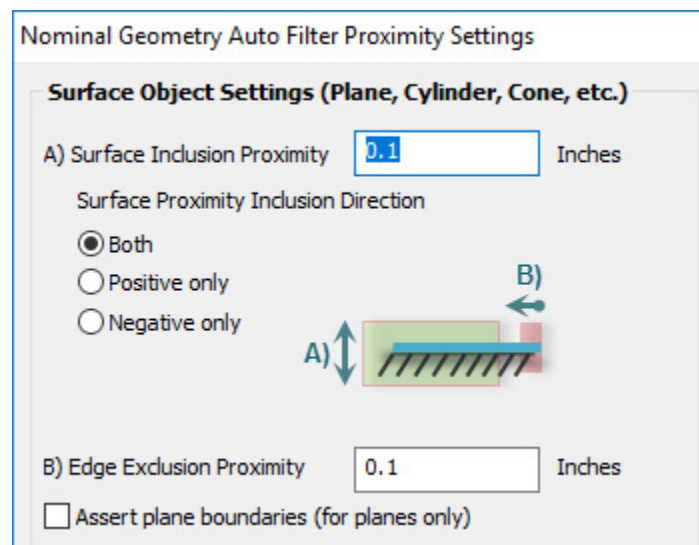


Figure 26-50. Nominal Geometry Auto Filter Proximity Settings for 3D features

- **Surface Inclusion Proximity (A).** This parameter defines how close the cloud points must be to the surface of a feature. This can be limited to only one side of the nominal feature if desired. This needs to be set large enough to account for any error in the measurements but not so big as to pick up data from other features.
- **Edge Exclusion Proximity (B).** Cloud points too close to the edge of a feature often include bad points. For this reason we provide a means to exclude those points based upon a preset distance from that edge.

2D features such as circles require an additional parameter because they are both detecting the measured plane used for projection and the filter points. Therefore there are three parameters to consider (Figure 26-51):

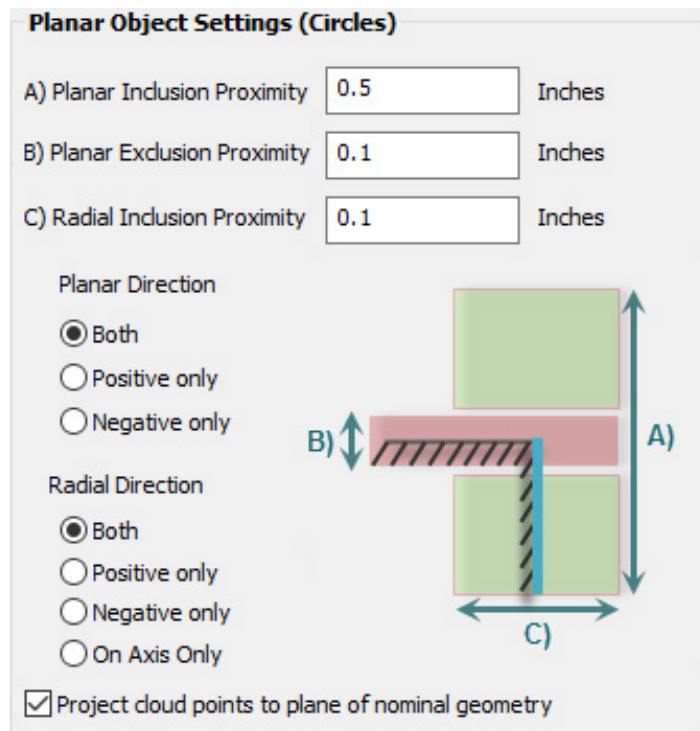


Figure 26-51. Nominal Geometry Auto Filter Proximity Settings for 2D features

- **Planar Inclusion Proximity (A).** This parameter defines the distance from the planar surface or how deeply into a hole the filter will look.
- **Planar Exclusion Proximity (B).** The filter is looking to gather points from the inner wall of the measured cylindrical surfaces not the measured plane points. To exclude these, this parameter allows you to set the extent of data ignored relative to the nominal planar surface. This can be compared to the Edge Exclusion Proximity (B) setting for 3D features.
- **Radial Inclusion Proximity (C).** Much like the Surface Inclu-

sion Proximity using for 3D Objects, this parameter provides a setting to define the extent of data included relative to the inner wall of the hole. This parameter can be set fairly large unless there are other holes or features close by. The numeric value entered defines the distance from the wall of the hole or, when the “On Axis Only” option is enabled, relative to the hole center axis.

These values defines either the total bidirectional distance, when the **Both** option is checked or a single direction value when either the **Positive only** or **Negative only** options are set.

Using Surface Faces for Edge Detection

When working with composite structures such as a plane with holes in it or a partial sphere or cylindrical cutout filtering to the nominal feature doesn't provide enough information to accurately detect the correct points. In these cases the CAD Faces can be saved with the nominal feature to more precisely restrict the filter.

As an example consider a rounded corner on a CAD part. If you build the nominal cylinder from this face the filter process will include all the points within a proximity to that cylinder. The problem with this is that the CAD is only a partial cylinder and this will throw off the resulting fit as shown here in [Figure 26-52](#).

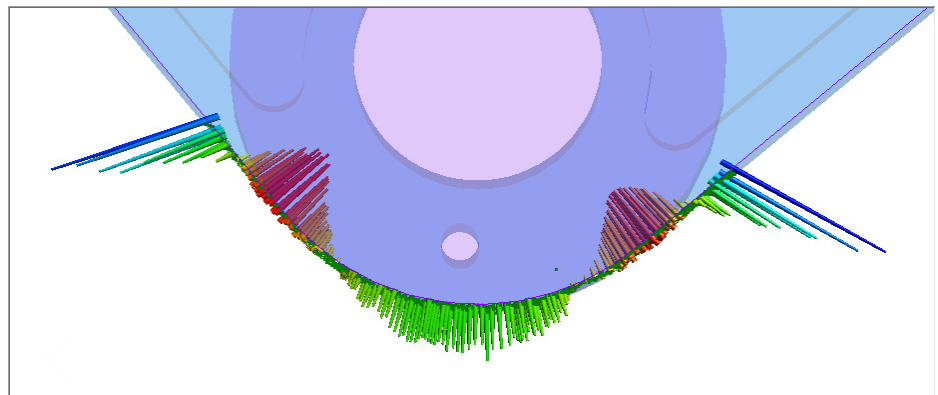
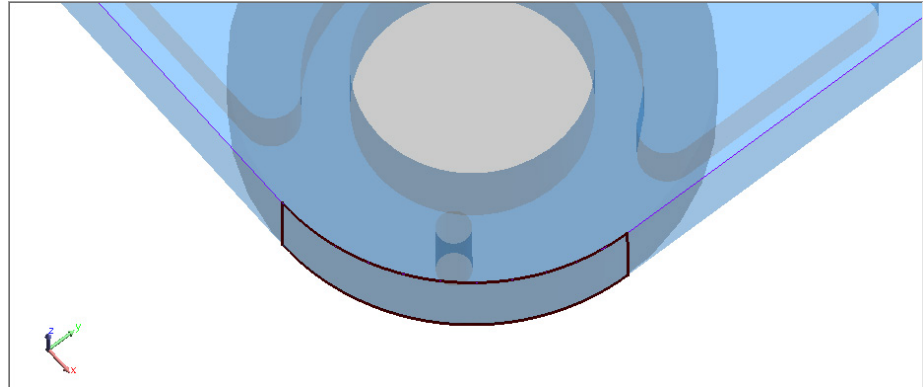


Figure 26-52. Partial cylinder proximity filter problem. Showing the need to save CAD face information.

To properly filter to this face we have added the ability to record the faces used to build the geometry with the nominal cylinder selection. This can either be done through a right-click option added to the toolkit or directly through the relationship properties.

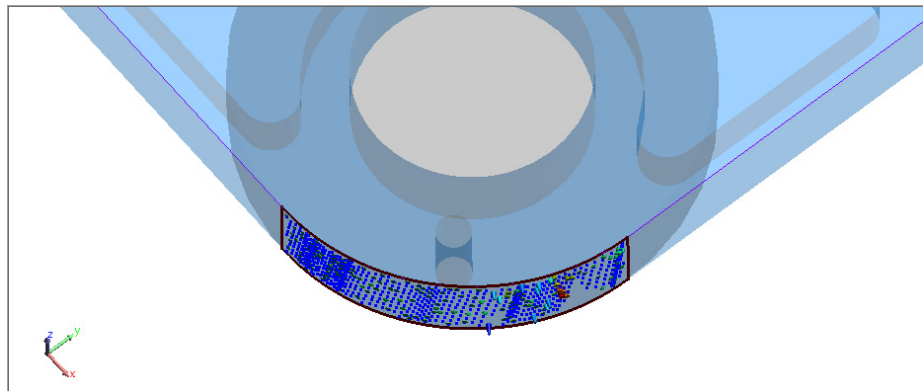
When a face is saved with the nominal geometry you will see it highlighted with the geometry when it is highlighted as shown here in [Figure 26-53](#).

Figure 26-53. CAD face saved with the nominal cylinder.



Using the CAD faces with the proximity filter allows you to ignore data beyond their extents and better isolate the data as shown here in [Figure 26-54](#).

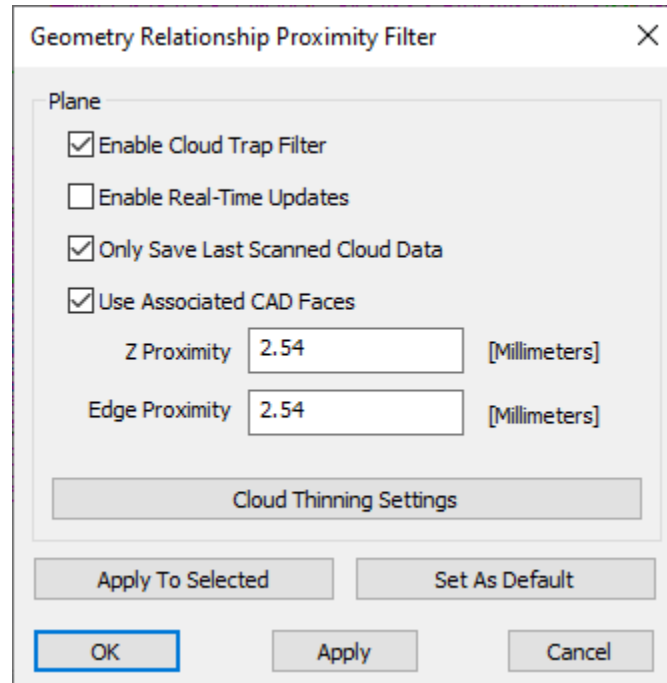
Figure 26-54. Properly filtered data using the associated CAD faces and resulting vectors.



Feature Specific Filtering

Each GR-Feature has its own auto-filter proximity settings built in. These initial settings are pulled from the **User Options>Display Tab, GR Proximity Filter** and saved with the feature. They can be accessed at any time through the Right-click **Configure Cloud Trapping Filter** option. This will open a unique feature type specific set of controls ([Figure 26-55](#)).

Figure 26-55. Feature Specific Proximity settings for a plane.

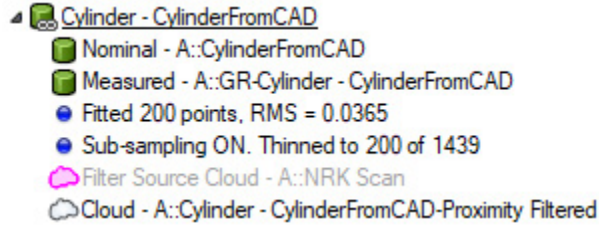


- **Enable Cloud Trap Filter.** This option allows trapping to be filtered based upon proximity to the nominal geometry.
- **Enable Real-Time Updates.** When enabled the geometry is fit dynamically to the cloud data as it is gathered. This can be very helpful in some cases and can cause slow downs depending on the data acquisition rate. With this option off, geometry is fit when trapping is completed.
- **Only Save Last Scanned Cloud Data.** With this flag set, all previously acquired cloud data is cleared as input into the geometry relationship. With flag off, whenever you are trapping cloud data to a relationship, the most recently scanned cloud is appended to the list of any previously acquired cloud data such that any previously acquired cloud data is saved.
- **Use Associated CAD Faces.** When filtering, consider the CAD faces as well as the nominal geometry. This will impact edge proximity.
- **Cloud Thinning Settings.** This is the standard cloud thinning settings used in many places within SA. It allows you to sub sample the cloud used in the fit operation.
- **Apply.** The Apply button can be used to trigger a re-sampling of the cloud data from the source cloud. And the **Apply to selected** option can be used to apply settings to a selection of features at once.

In order to maintain data integrity the source cloud is saved with a GR-Feature but not edited by it. Instead, a sub sample of the source

cloud is made as a copy and linked as well so that it can be used for the purpose of fitting and outlier rejection. This allows you to more unwanted outlier cloud points with edition the source cloud (Figure 26-56).

Figure 26-56. Example GR-Cylinder showing both a Source Cloud and the associated cloud used for the fit.



These settings are applied directly from within a particular relationship using **Associate Data > Proximity Filtered Cloud** or through trapping into the feature with a live instrument.

Feature Specific Filtering (Circles and Slot)

The 2023 version of SA added additional extraction methods for circles and slots to simplify hole extraction. These methods are based upon a standard developed for Volkswagen (VW) to facilitate automotive applications.

These extractions are available through the use of the right-click **Configure Cloud Trapping Filter** option. The reference or input cloud is saved and extractions are performed from this reference. This extraction process will build additional clouds within the GR Circle as show here (Figure 26-57):

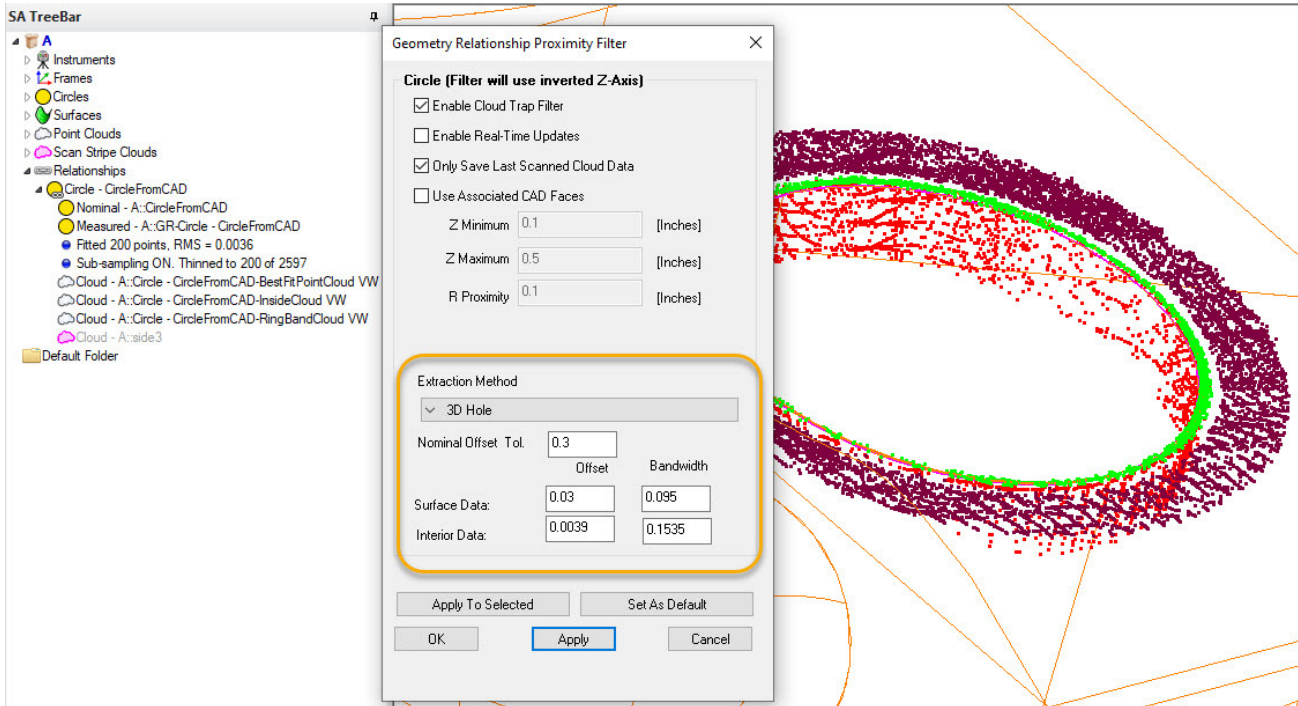


Figure 26-57. 3D Hole extraction which is using a Scan Stripe Cloud as an input and builds 3 results. The local plane, the interior data, and a projected set of data used for the circle fit.

Extraction methods include:

- **3D Interior Data.** Provides the original interior cylinder extraction data only. This method typically needs an additional projection plane reference within the GR-Circle.
- **3D Hole.** This method provides a complete 3D hole extraction. It offers the ability to extract a ring of data around a hole to define a local plane as well as extracting data from within the hole. It is assumed that the hole direction corresponds to the surface normal direction. The interior data is then projected to this local plane for the fit (Figure 26-58). The offset and width of the surface data can be set to define the local plane fit and the offset and width/height of data down inside the hole can also be set (Figure 26-59).

Figure 26-58. VW Extraction methods for automotive applications were added in 2023

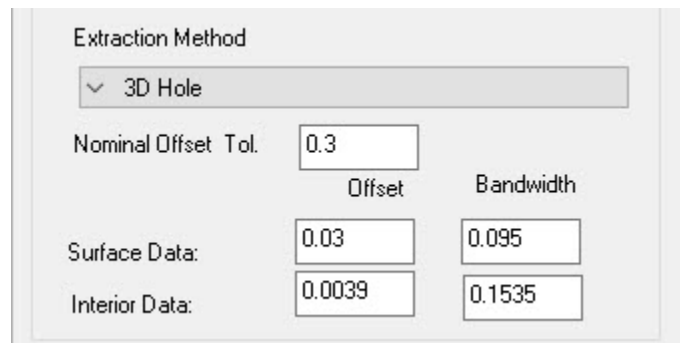
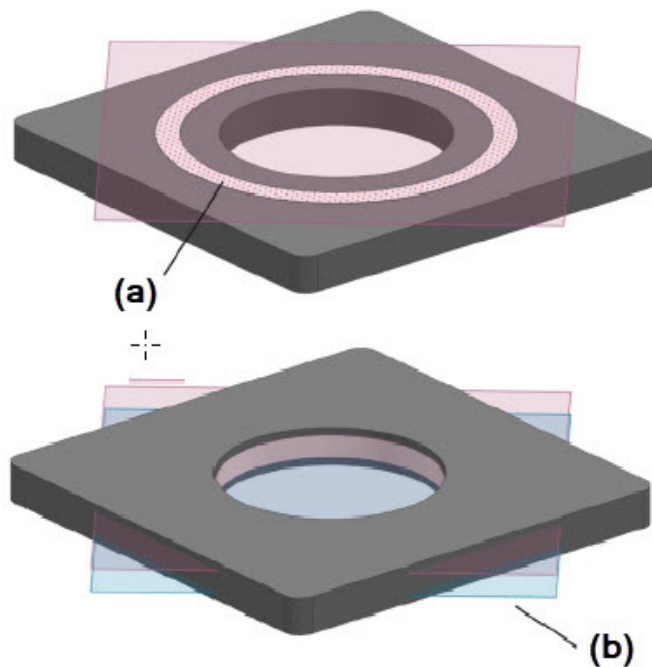


Figure 26-59. 3D Hole extraction is performed in two parts: the planar ring is extracted (a) and then data within the hole (b)



- **2D Hole.** This fit methods is for sheet metal applications or other situations where data cannot be measured within the hole. A local plane is still extracted to define the surface orientation

but only the boundary points are used to define the diameter of the hole.

- **3D Hole in a Surface.** This method provides an option to work with curved top surfaces rather than a planar surface (Figure 26-60).

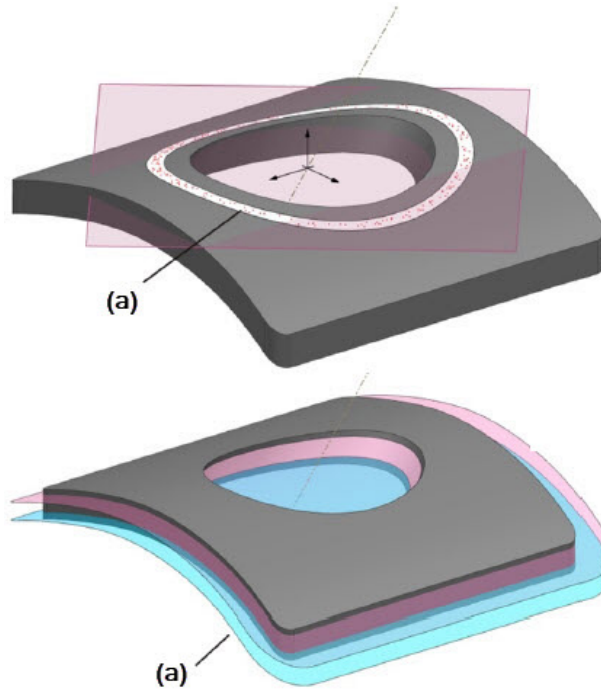


Figure 26-60. Hole drilled within a curved surface. First a local surface is established (a) and then data from within the hole is gathered (b) and projected to establish the hole position and diameter.

- **3D Plunged Hole.** Flanged holes are different from 3D holes in that the interior data is now above the projection plane and projected down onto it (Figure 26-61).

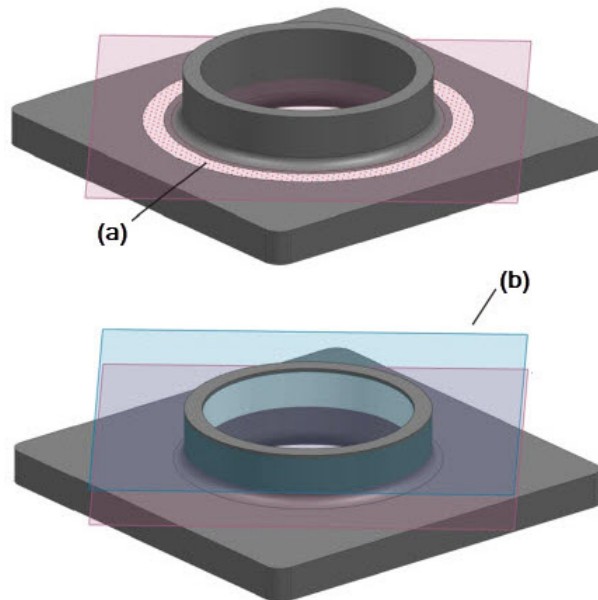



Figure 26-61. 3D Flanged Hole includes a planar extraction (a) and an interior extraction (b) but anticipates data above the projection plane which differentiates it from a 3D Hole extraction.

- **2D Plunged Hole.** This method first establishes a local set of

data for projection plane and then projects all the data to this plane and extracts using the boundary points, much the same as a 2D hole.

- **3D Plunged Hole in a Surface.** Provides an option to extract data from a flange above a curved surface and uses the data within it to define the circle diameter.
- **2D Plunged Hole in a Surface.** The same except only the boundary points are used to define the diameter.

Use the **Apply to Selected** and **Set As Default** options to apply settings between features and adjust the defaults for your application.

The **Auto Filter to Nominal Geometry >3D Geometry** option  can also be used to perform this filter process. It will link an input cloud to the geometry and apply an extraction. An option to “**Use Filter Settings**” is available to directly take advantage of the saved settings from with each feature or the **Settings** button will open options for filtering and sub-sampling.

If you need to edit the settings for any circle after that you can use the **Apply to Selected** option within the settings dialog of any individual circle. There is also an option within the Relationship category heading to **Refresh Filtered Cloud Data** that will re-execute the filter process.

Auto Filter to Nominal Geometry >2D Geometry

2D Geometry extraction is a specialized subset of the Compare to Nominal Geometry function used for holes (circles and slots). It can be used to perform feature extraction from a point clouds based upon proximity to nominal geometry, where only a surface scan is available, such as with sheet metal. This command can be accessed through **Relationships > Geometry Comparison > Autofilter to Nominal Geometry>2D Geometry** or through the toolkit.

This alternative Filter to Nominal function uses the same dialog as the 3D Geometry extraction tool with the addition of a tolerance and two processing buttons (Figure 26-62):

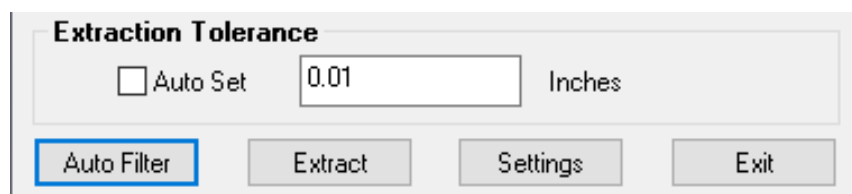


Figure 26-62. 2D Extraction specific buttons

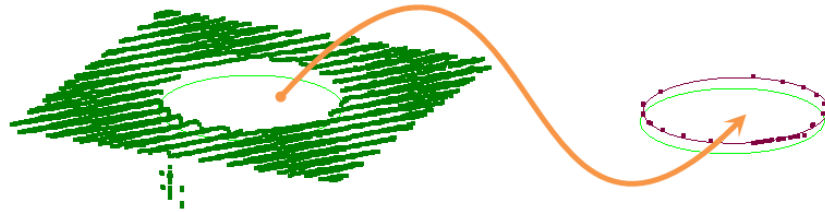
The 2D Geometry extraction process is performed in 2 steps:

1. The **Auto Filter** Button should first be use to extract mini clouds in close proximity to nominal geometry in order to isolate and identify the cloud points of interest (use the settings button to

set filtering and extraction depth to build the mini clouds).

2. The **Extract** button can then be used to build the extracted circle points and associate them automatically with the generated relationship.

Figure 26-63. 2D Extraction process in action



The extraction process will delete the referenced mini-cloud built from the auto-filter process and automatically associate the extracted points to the relationship as long as a circle fit can be achieved within the specified tolerance. If it cannot identify geometry, it will not remove the filtered cloud so that a manual inspection is possible. For additional information on the tolerance settings see “[Extraction Tolerance](#)” on page 600. The *Auto Set* option when enabled will adjust the tolerance based on the *maximum* radius over all active nominals.

All filter settings are disabled in 2D Geometry extraction mode except for the depth control (A) which is still used to determine how far above or below the nominal hole plane data is collected.

Auto Filter to Nominal Geometry>Lines

Filtering clouds to lines only considers a radial deviation and is therefore handled separately. The controls for this function include the proximity to the line (Radial Cutoff) and the distance from the ends of the line, distance from the origin (Lower Cutoff) and distance from the end of the line (Upper Cutoff). Subsampling options are also available .

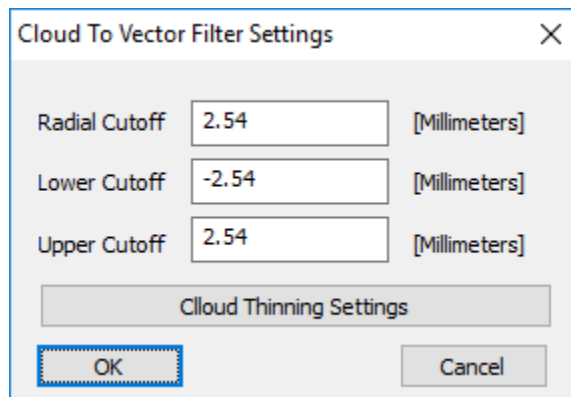


Figure 26-64. Auto Filter to Lines

Extract Geometry from Clouds

SA can be used to extract features from a point cloud without CAD or any nominal references of any kind. This is done by detecting geometry within proximity to a seed point (or click-point). This command can be accessed through **Relationships > Geometry Comparison > Extract Geometry from Cloud**.

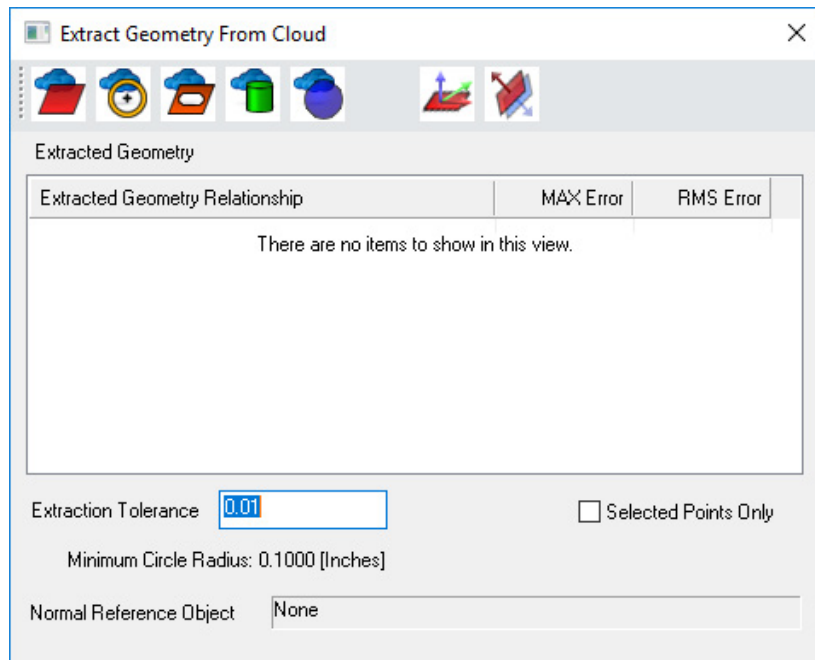


Figure 26-65. Extract Geometry from Clouds Dialog

Feature Selection and Cloud interaction

To use this extraction dialog, simply open the dialog so that it is visible, select the geometry button of interest, and then click on a cloud point to define the start of the search region.

- **Extract Plane.** Click on the cloud in a planar region. SA will then grow a plane from that starting point including cloud data as long as it remains within the specified tolerance.
- **Extract Single Circle from Plane (Left-Click).** Click on the cloud in a planar region. SA will then fit a plane to that data set and search within proximity to that click point for a single circle or hole in the planar region selected.
- **Extract Multiple Circles from Plane (Right-Click).** Click on the cloud in a planar region. SA will then fit a plane to that data set and search throughout the plane for circles or holes that are within the specified tolerance.
- **Extract Circle from Cylinder (Alt+Left-Click).** Click on the cloud in a cylindrical region. SA will then fit a cylinder to that data set and build a circle at the end of the fit cylinder closest to the click point.

- **Extract Single Slot from Plane (Left-Click).** Click on the cloud in a planar region. SA will then fit a plane to that data set and search within proximity to that click point for a single slot.
- **Extract Multiple Slots from Plane (Right-Click).** Click on the cloud in a planar region. SA will then fit a plane to that data set and search throughout the plane for slots that are within the specified tolerance.
- **Cylinder.** Click on the cloud in a cylindrical region. SA will then grow a cylinder from that starting point including cloud data as long as it remains within the specified tolerance.
- **Sphere.** Click on the cloud in a Spherical region. SA will then grow a sphere from that starting point including cloud data as long as it remains within the specified tolerance.

Extraction Tolerance

A single tolerance parameter is presented.

Relative to the seed point, the “ExtractPlane” function tries to grow a plane (or other geometry) and as it grows, it filters out points that are not within this plane tolerance. If data is fairly planar then there is a good chance that the correct set of planar cloud points will be harvested with a tight tolerance. However, if you pick a seed point in a region with adjacent points that are not clearly planar then the plane may start off in a bad orientation. Larger tolerances will increase the likelihood of getting the correct orientation but include more outlier points in the fit. Therefore an optimal balance must be found.

Acceptance criteria for a “good” circle (currently hard coded values included here only as a reference) :

- $RMS \leq 0.1 * radius$
- $MAX Pt Dev \leq 4 * tolerance$
- Number of points ≥ 10
- $Radius / (max cloud dimension) < 4$
- $Tolerance \leq 0.1 * Radius$ (not used when filtering to nominal geometry)

Selected Points Only (check box)

When enabled, this option will prompt the user to select the subset of the cloud to consider after selecting the starting point. This can be very helpful for detecting individual features and accelerating the search process by reducing the amount of data to consider.

Normal Direction and Reference

Without a nominal reference, geometry has no directional reference. Predefining a reference object allows all the resulting objects to be

correctly oriented. This is important if additional point measurements are going to be taken and the correct offsets need to be accounted for. Object direction can also be reversed individually by selecting the newly built object in the Extract Geometry from Cloud dialog, or in the fit settings of the resulting geometry relationship.

Meshing Point Clouds

A mesh is an approximated surfaces built from triangles (or polygons). These surfaces can be either imported or built directly from cloud data and they can be used for a wide range of purpose including simplified reference surfaces that can be used as a proxy for a surface's true curvature. The vertice points used to define the triangles provide the only accurate position information which the connecting lines are used for visualization purposes.

There are a couple of different mesh formats:

Polygonized Surfaces Polygonized Surfaces

Polygonized Surfaces are primarily imported CAD models of a very low resolution, although some legacy construction and editing can be done most meshing tools now use the Scan Strip Mesh.

Graphical Entities Graphical Entities

Graphical entities can be edited but include additional visual elements. For example a **Colorized Graphical Mesh** is generated from the proximity of points or point clouds to a surface the resulting feature in the tree is a Graphical Entity. This is use for visual representation only.

Scan Stripe Mesh Scan Stripe Mesh

Scan Stripe Meshes are SA's primary tool for working with and editing mesh representations of cloud data. They can be built from a specific Scan Stripe Cloud, much like auto-vectors built from a relationship. The probing transforms embedded within Scan Stripe Cloud can be used to provide the direction vectors necessary to make Real-time meshing possible. It also provides the ability to build finely detailed meshes from cloud data.

- **Real-Time Mesh.** Built as measurements are taken.
- **Coarse Mesh.** Provides a quick visualization tool a mesh of roughly a uniform grid density.
- **Fine Mesh.** Provides a detailed surface approximation for the scan data and is optimized for triangle density and distribution.

Scan Stripe Mesh Operations

Scan Strip Meshes

Scan Stripe Meshes display a graphical rendering of cloud data in either of two formats:

- **Course Mesh.** Which is a course or minimally processed mesh meaning that it uses a fixed grid size.
- **Fine Mesh.** This mesh must be built from course mesh and uses the specified meshing parameters defined in the mesh properties to built a finalized or detailed mesh.

General Mesh

Both course and fine general meshes can be built from any point cloud or combination of point clouds. It requires a instrument association but does not require a scan stripe cloud as an input. To built a general mesh either right-click on a cloud directly or use the command **Construct>Mesh from Point Clouds**. This will open the mesh settings dialog (Figure 26-66).

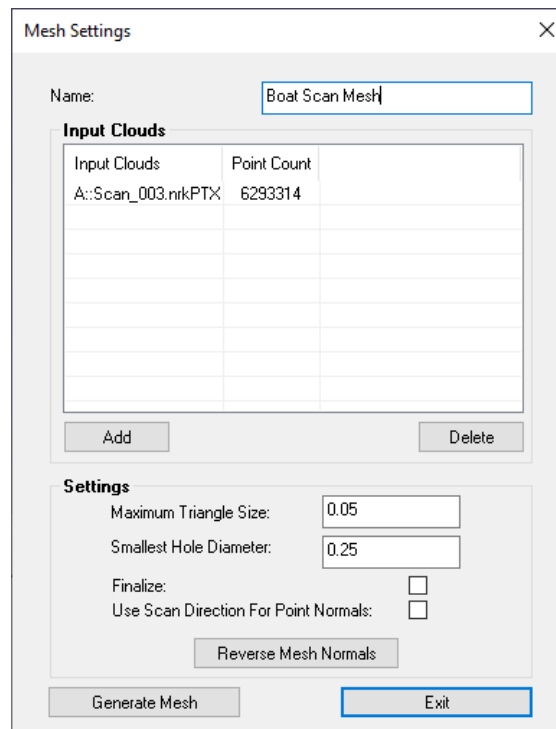


Figure 26-66. Generic Mesh Construction Dialog

General Meshes only offer 2 parameters:

- **Maximum Triangle Size.** This value defined the resolution of the resulting mesh by defining the size of the triangles used. The small the max triangle size, the finer the mesh.
- **Smallest Hole Diameter.** This value controls the size of the gaps within a mesh that are bridge or covered. With this value very small the mesh will have lots of holes. But with a large

value, holes that are actually there within the mesh could be covered over in the meshing process.

- **Finalize.** The finalize check box is used to build a finalized mesh. This takes much longer and produces a much more optimized mesh when the mesh is generated.
- **Use Scan Direction For Point Normals.** This option allows the normals saved in a Scan Stripe Cloud to be used to help orient the normals of mesh segments. However, when combing clouds, or meshing scans performed with many passes from different directions, this may not be helpful. The positive side of the mesh needs to be oriented correctly for correct probe compensation.



Figure 26-67. Fine Mesh Generation

The **Reverse Mesh Normals** button can be used to reverse the normal of a mesh for accurate probe offset compensation. However, it should be noted that breaks in the mesh can cause local segment reversals. The Extended Parameter Mesh is much more resistant to these because it uses the scan line probing vectors.

For basic meshing of point data the General Mesh can also be used. For example if you wish to build a 2D mesh built from all of the measured points as the vertex points for the mesh you can do so by entering 0 in both the mesh parameters ([Figure 26-68](#)).

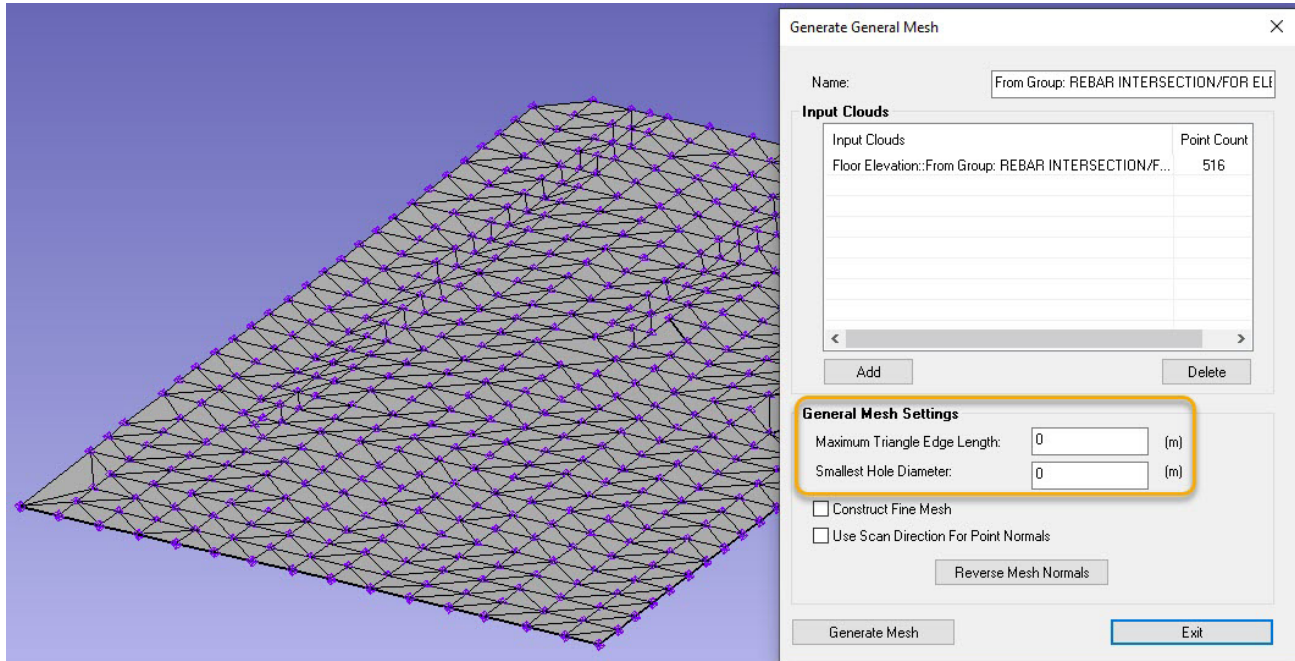


Figure 26-68. Basic 2D Mesh fit to points.

Extended Parameter Settings

In SA version 2024.1 the old Extended Parameter Mesh settings have been replaced with an updated file based meshing solution. This new solution expands the existing General Mesh configuration to offer 3 levels of control:

- Semi-Automatic using the General Mesh Settings (described in the prior section)
- Manual Controls- where additional parameters are exposed
- Fully Automatic - using a JSON file configuration for the meshing process.

Manual Controls

At the bottom of the Generate General Mesh dialog is now an option to enable Extended Parameter Meshing.

This meshing process proceeds as follows:

1. **Preprocessing:** This step consist in processing the cloud for the meshing: could be reduction, noise filtering, explode by distance to keep the main part.
2. **Initial Meshing:** This step will generate the first mesh that will be then improved in the next step. For that, a 3d mesh is created, then hole filling is applied and finally border extension is called to better fit the cloud border.
3. **Post Processing:** During this step, a global refine is done to add

details in curved area, free borders are smoothed and subdivided, general smoothing can be done. Mesh reduction, to avoid having many triangles in flat areas, is also part of this Post Process Mesh step.

Figure 26-69. Extended Parameter Settings offer access to more advanced meshing controls

Using the *Extended Parameter Settings* expands on the initial two parameters used in the general mesh by added 4 additional controls.

- **Max Triangle Edge.** Provides a control over the largest triangle size that will be produced.
- **Min Refine Edge.** Offers a control over the smallest edge length that will be produced
- **First Pass Deviation.** Controls the resolution of the Initial meshing step
- **Second Pass Deviation.** Controls the final resolution of the mesh.

There are 3 presets (*Detailed*, *Normal*, and *Fast*). If you select one of these presets the JSON configuration file can be exported and then edited.

Fully Automatic

The Import button will not populate these manual control fields but will simply initiate the meshing process using the saved settings in the file.

JSON File Configuration

The meshing process offers the following parameters:

Parameter Name	Default Value or Ratio	Link with which parameters	Information on the Parameter
Main Parameters			
MainParameters:AverageEdgeLength	2.0	Not a ratio, defined in the unit of your point cloud	The average triangle size when we start the meshing. You can adjust this value depending of the size of the object and the distance between points in the Point Cloud.
MainParameters:SmallestHoleDiameter	6.0	Not a ratio, defined in the unit of your point cloud	Define the minimal hole size that you want to detect. -1.0 means that we try to fill holes as much as possible.
MainParameters:IgnoreScanDir	false		Some clouds could contain bad scan direction data. This parameter is used to avoid using scan directions
Preprocessing			
PreProcessCloud:ExplosionDistance	20	Not a ratio, defined in the unit of your point cloud	Disconnected parts are split using this threshold distance and we keep the biggest part. You can modify the threshold for the split or remove this parameter if you want to keep everything.
PrePreocessCloud:NoiseReductionThreshold	80	Unitless. Value between 0 and 100	Can filter the cloud to reduce the noise of the cloud. 80 is a good value if the cloud is noisy.
ReductionRatio	1	Unitless. Value between 0 and 1	Reduce the cloud before meshing in order to speed up computation. 0 means that no point is kept, 1 means that all points are kept.
Initial Meshing			
InitialMeshing:BorderExtensionParameters:Deviation	*0.1	MainParameters:AverageEdgeLength	This parameter is used to extend border after the first meshing step. The idea is to better fit to the border of the cloud.
InitialMeshing:MaxTriSize	*3	MainParameters:AverageEdgeLength	Maximum triangle size
InitialMeshing: FillNonRegularHoles	0	Unitless	3 levels of Holes Filling: 0: Preserve holes in areas without acquired point cloud. 1: Try to fill some holes that are suspect due to missing data 2: Try to fill lot of holes to create watertight mesh.
InitialMeshing:MaximalHoleElongationRatio	3.0	Unitless	Ratio between largest and second largest extend of the more elongated hole to be preserved.
SmoothingDistanceFactor	2.0	MainParameters:AverageEdgeLength	Use projection on implicit surface to de-noise the data. The factor multiplying the Average Edge Length to determine the distance of the Polynomial Surface from points are sampled.
PostProcessMesh			
PostProcessMesh:BorderImprovement::MaxEdgeLength	*0.5	MainParameters:AverageEdgeLength	During border improvement, we have to subdivide the triangles around borders. This value indicate the max edge around borders.
PostProcessMesh:BorderImprovement:DistanceSelection	*2.0	MainParameters:AverageEdgeLength	Distance around borders to select triangles.
PostProcessMesh:BorderImprovement:SmoothingIntensity	2.0	Unitless	Intensity of the smoothing around borders. 0 means no smoothing, 5 means very high smoothing
PostProcessMesh:BorderImprovement:MaxDeviation	0.1	MainParameters:AverageEdgeLength	Maximal distance allowed between surfaces before/after the smoothing
PostProcessMesh:Smoothing:Intensity	2.0	Unitless	Intensity of the smoothing around borders. 0 means no smoothing, 5 means very high smoothing
PostProcessMesh:Smoothing:MaxDeviation	*0.1	MainParameters:AverageEdgeLength	Maximal distance allowed between surfaces before/after the smoothing
PostProcessMesh:Reduce:MaxDeviation	*0.0025	MainParameters:AverageEdgeLength	Reduce triangles in flat areas: this parameter defines the max deviation allowed between the original mesh and the reduce one.
PostProcessMesh:Reduce:EdgeMaxLenght	* 4	MainParameters:AverageEdgeLength	Maximal edge length allowed in the resulting mesh.

Parameter Name	Default Value or Ratio	Link with which parameters	Information on the Parameter
*PostProcessMesh:Refine:MaxDeviation	* 0.01	MainParameters:AverageEdge Length	This parameter is used to refine the mesh. This parameter is linked to the final quality of the mesh AND also link to the accuracy of the scanner/point cloud. If this parameter is too small compare to the noise thickness, it will lead to 'waves' effect in the final mesh.
PostProcessMesh:Refine:MinimalTriangleSize	* 2	PostProcessMesh:Refine:MaxDeviation	Minimum triangle size during there fine.
PostProcessMesh:Refine:DistanceWrongPoint	* 0.5	MainParameters:AverageEdge Length	Maximum distance between points and mesh to be used for the refine: points that are far (using this param) are not used.
PostProcessMesh:CurvatureSelection:SelectionRadius	* 8	MainParameters:AverageEdge Length	Radius of the neighborhood used for the curvature selector.
PostProcessMesh:CurvatureSelection:CurvedThreshold	50	Curvature radius (the reverse curvature) in document units.	Threshold for considering a triangle as curved or not (before the closing). The threshold is a minimum curvature radius.
PostProcessMesh:CurvatureSelection:WhitelistThreshold	30	Curvature radius (the reverse curvature) in document units.	Threshold to select a triangle regardless of the closing. (White List Threshold must be less than or equal to Curved Threshold.)
PostProcessMesh:CurvatureSelection:BlackListThreshold	100	Curvature radius (the reverse curvature) in document units.	Threshold to not select a triangle regardless of the closing. (Black List Threshold must be greater than or equal to Curved Threshold.)
PostProcessMesh:CurvatureSelection:ClosingRatio	* 1	PostProcessMesh:CurvatureSelection:SelectionRadius	Ratio of selection radius used for closing (dilation and erosion of the selection).

Here is an example of a .JSON configuration file:

```
{
  "Version": "0.2",
  "BaseUnit": "mm",
  "EnableLog": false,
  "MainParameters": {
    "AverageEdgeLength": 1.500,
    "SmallestHoleDiameter": 5.000,
    "IgnoreScanDir": false
  },
  "PreProcessCloud": {
    "ExplosionDistance": 20.0
  },
  "InitialMeshing": {
    "BorderExtensionParameters": {
      "Deviation": 0.1
    },
    "MaxTriSize": 3.000,
    "FillNonRegularHoles": 0,
    "MaximalHoleElongationRatio": 3.0,
    "SmoothingDistanceFactor": 1.0
  },
  "PostProcessMesh": {
    "BorderImprovement": {
      "MaxEdgeLength": 0.5,
      "DistanceSelection": 2.0,
      "SmoothingIntensity": 2.0,
      "MaxDeviation": 0.1
    }
  },
  "Smoothing": {
    "Intensity": 2.0,
    "MaxDeviation": 0.1
  }
}
```

```

},
"Refine":{
  "MaxDeviation":0.010000,
  "MinimalTriangleSize":0.001500,
  "DistanceWrongPoint":1.5
},
"Reduce":{
  "MaxDeviation": 0.0025,
  "MaxEdgeLength": 3.0
},
"CurvatureSelection":{
  "Selection Radius": 8.0,
  "CurvedThreshold": 50.0,
  "WhiteListThreshold": 30.0,
  "BlackListThreshold": 100.0,
  "ClosingRatio": 1.0
}}}

```

Scan Stripe Mesh Editing

Stitch Mesh


Two meshes can be combined using the right-click menu command **Stitch Mesh**. This command allows you to select a second mesh and blend them together within an overlap tolerance as a single result, building a new mesh.

Mesh Dissection

A mesh can be composed of a continuous set of linked polygons or a group of distinct segments. This group of segments can be split apart and recombined much like the faces of a CAD model.

Use **Mesh by Dissecting** to create additional entries in the tree for each distinct mesh segment, and **Mesh from Collection of Meshes** to build a single mesh for a selection of segments. This is different from **Stitching** in that the mesh segments are unaltered.

Hole Filling

Holes can be filled within a mesh using the command **Clouds & Surfaces>Build from Clouds>Mesh>Edit Scan Strip Mesh** . This option searches the mesh relative to a click point and fills holes that fit the selected criteria.

Limited Filling Options.

Limited filling is the safest approach. It sets restrictions on the extents of the filling that can be performed .

- **Wall Thickness.** This parameter can be used to fill horseshoe shaped holes in an edge. It allows you to specify the size of gap to bridge (a value of -1 disables this bridging).
- **Max Triangle Length.** The length of triangles used to bridge

a hole can be restricted to a specific length (a value of -1 removes any limits).

- **Tension.** Tension determines the curvature of the mesh that is used to span a hole, allowing you to follow the natural contour of the part within a hole rather than filling it with a flat cap. Tension is defined numerically.
- Values <0 reverse the curvature in the middle of hole
- A value of 0 does not add any curvature
- Values >0 add curvature. By default, curve fitting should use a value near 1.

To use this function set the values you would like to apply, press the **Fill Individual Holes** button, and then click on the mesh next to the hole you wish to fill.

The option to **Fill All Holes in Selected Mesh** can also be used to fill any holes that fit the filling criteria.

Hole filling is an art and can produce unexpected results. The **Undo All Fills For Last Selected Mesh** button can be used when a fill operation does not go as expected.

Unconditional Filling

This option will fill unconditionally even if the shape is not smooth. This option can produce sharp edges and intersections because some of the filters are disabled but provides an alternative when **Limited Filling** is not providing the desired results.

Hole Filling a Mesh in SpatialAnalyzer -

<https://youtu.be/7kfC xv8qOBc>

This Page Intentionally Left Blank.

Geometric dimensioning and tolerancing, more commonly known as GD&T, is a system for defining and implementing engineering tolerances for part and assembly geometry. In this chapter, we will discuss how SA supports the engineer's ability to read and communicate GD&T design from CAD drawings.

SpatialAnalyzer's GD&T Inspection tools provide engineers with the ability to perform standardized inspections in accordance with the guidelines of the ASME or ISO standards. It provides the tools to allow the technician to import fully annotated CAD models and utilize the annotations and tolerances provided, or build their own annotations from CAD models, SA geometry, or even directly from measured points that define geometry. It also provides the ability for a designer to build an inspection process that walks an operator through the process of measuring the datums and features of interest following predefined measurement location indicators. Once points are assigned to a feature check it is immediately validated with a pass or fail result. The check can then be added directly to dynamic reports.

SA's GD&T evaluation process can be customized to ensure compliance with one of the following supported standards:

- *ASME (1994), ASME (2009), and ASME (2018)*
- *ISO (1983), ISO (2004), ISO (2012), and ISO (2017)*

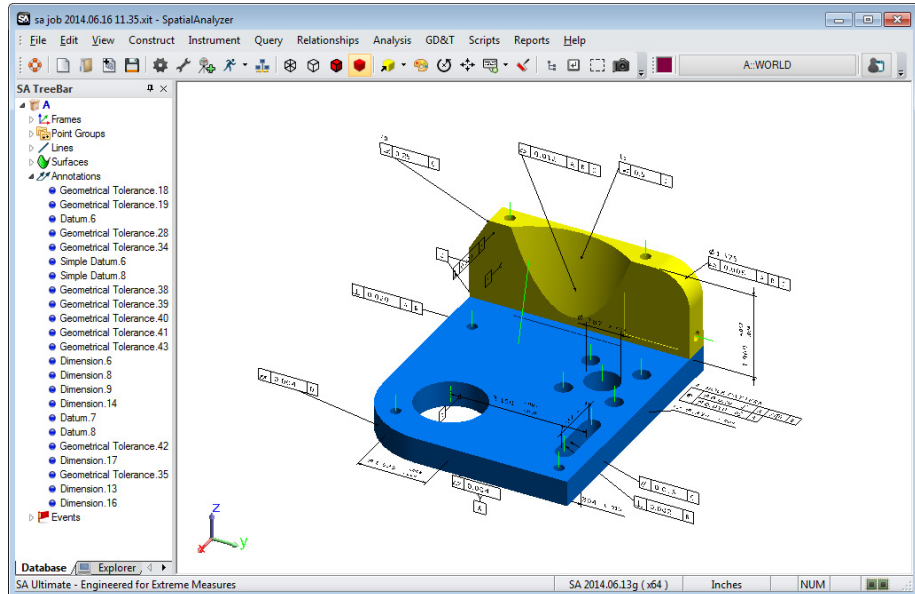
You may select the standard you wish to use by navigating to the Analysis tab in the **User Options** dialog. Select **GD&T Options** and then choose one of the options listed. For more details see "[GD&T Evaluation](#)" on page 640.

Importing CAD with GD&T Annotations

To import a CAD model with annotations select **File > Import > Direct CAD Access**. If you would like to change your import preferences, navigate to **File > Import > Direct CAD Access Settings**.

In the *Direct CAD Access Settings* dialog, check the *Annotations* option. Once imported, the annotations will appear graphically and in the tree in the Annotations category (Figure 27-1).

Figure 27-1. Imported CAD Annotations.



Annotations will be imported into the current active collection along with the CAD surfaces. An additional collection will be added containing the imported CAD Views and will be labeled "CADViews:" followed by the name of the imported CAD file. This collection will include the view orientation frames associated with the annotations as well as a set of Callout views. These callout views include the imported orientation of the model and the fill Annotation display as well as one for each group of commonly oriented annotations.

Creating GD&T Annotations

SA provides the ability to add GD&T annotations to a CAD model and provides the tools necessary to conduct a GD&T analysis on constructed SA geometry. To build your own annotations in SA use the GD&T tab in the SA toolkit (Figure 27-2), and for a basic case just follow these steps:

Defining Datums:

1. Choose between CAD Faces or SA Objects
2. Click the Datum letter button to start selection

3. Click in the graphics on the desired CAD face or SA Object to both define the attachment point for your datum and designate the feature as a datum.

Defining Checks

1. Choose between CAD Faces or SA Objects
2. Enter the Tolerance and Datums to include in the check (using the order of priority)
3. Click on the check button to define the check type
4. Click in the graphics on the desired face or object to define the features to inspect.

Notes on Selection

Here are a set of helpful hints and things to keep in mind when building annotations using the GD&T Toolkit tab:

1. The GD&T Toolkit tab is intended to be a quick means to build annotations in the graphics. For greater control or to edit the annotations simply open the annotation properties.
2. After an annotation has been created it will contain all the features associated and tolerance information. When the properties dialog is open the associated features will be highlighted and the annotation can be positioned as necessary.
3. The Feature Type selected greatly impacts the construction of GD&T annotations.

Creation Options

The GD&T Annotation tab is located within the SA Toolkit. If it is not visible, select **View > Show Toolkit Bar** from the menu. This toolbar is the starting point for building annotations and setting up GD&T evaluations when not importing annotations with a CAD model. With a logical workflow starting from the top and working toward the bottom, the toolbar begins with the *Creation Options* section (Figure 27-2).

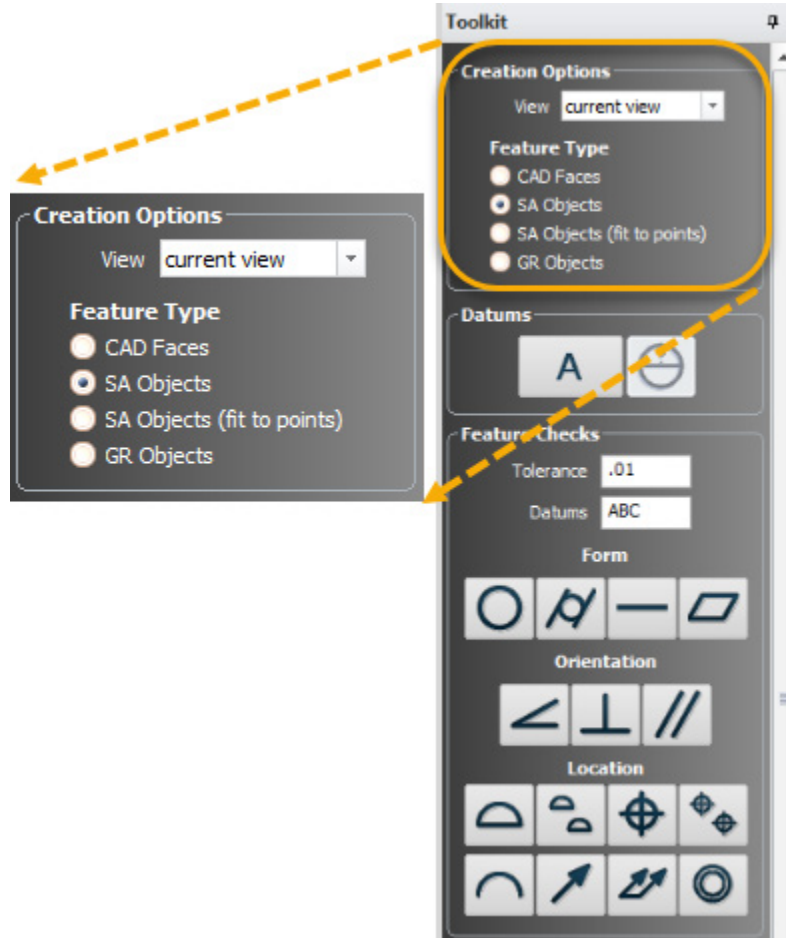


Figure 27-2. GD&T Creation Options Section.

View

This sets the orientation of the text for the annotation to be added. Selecting *Current View* sets the annotations to the current screen orientation and allows them to rotate to always face the user. The view orientations available are drawn from the **Preset Views** list which can be accessed and edited by going to **View > View Control > Preset Views**. The view selected, as well as the text size, can be edited within the annotation properties once the annotation is created.

Feature Type

Tip: In all cases GD&T Annotations must be added to the nominal (design) feature. If they don't exist they must be created.

The Feature Type settings within the Creation Options section defines and limits the type of objects to which an annotation may be added. You can choose from *CAD Faces*, *SA Objects* or *SA Objects (fit to Points)*. In all cases the GD&T annotations must be added to the *nominal (design)* features. If these nominal features do not exist in your job they must be created because they define the baseline for the evaluation. Only the *SA Objects (fit to points)* option allows you to select measurements.

- When you select *CAD Faces*, the annotation will be added to and include the specific selected CAD faces you chose from

- a particular model regardless of whether or not a model has been dissected into individual faces.
- Selecting *SA Objects* allows selection of any primitive SA geometry type as the nominal feature.
 - *SA Objects (fit to points)* is a special case which allows you to add an annotation directly to measured points. When you select this option you will be limited in your choices to building Datums, Form, and Orientation Checks. When adding annotations you will be prompted to define the geometry type you wish to fit to the points. Doing so will perform the following actions:
 1. Datums & Form Checks. The selected feature type will be built fit to the selected points, an annotation will be added to this feature, and the check will be evaluated based on the specified tolerance.
 2. Orientation Checks. Form checks require the nominal feature to be built at a nominal angle to the specified datum. Therefore, the feature fit to the selection points will not be RMS fit but will be align to the points while holding this nominal angle. Once built, an annotation will be added to the features and it will be evaluated using the specified tolerance.
 - Selecting *GR Objects* is another special case which allows selection of Compare to Nominal Geometry Relationships directly. This creates a bridge between a standard inspection and a true GD&T inspection by doing the following:
 1. The nominal SA Object defined within the GR-feature will be automatically assigned as the reference object for the GD&T Annotation.
 2. The associated points from the selected GR-Feature, used to build the measured feature, will be used by the resulting feature checks.

Datums

The two buttons in the datum section are intended for the quick addition of datums to a part. Press the **Datum letter** button and then click on the part to define the tie-in point for the leader line used by the annotation and to select the associated CAD faces. The Datum letters increments automatically to the next available letter. To enter a letter that is not in sequence, hold the letter you wish to use down on the keyboard when you press the datum button. It will then add this selected letter when you click on the part. Datum letter utilization is controlled by the selected GD&T Standard. ASME does not allow the use of the letters **I**, **O**, and **Q**, so they are excluded in the sequence (Figure 27-3).

Figure 27-3. Datums Section of the Toolkit



- Double or great datum letter designations such as **AA** or **AC** can be created and are recognized in SA. However, they cannot be built through the toolkit. You can build them using only one letter "A" for example and then open the annotation properties and change the name after building the annotation. Another alternative is to use **GD&T> Make Annotation for GD&T Datum** to build a blank datum annotation, add the Name designation desired and then associate the feature object.

Datum Targets

Datum Targets can also be added to a predefined datum feature. They are added by first clicking on the datum target button in the toolkit and then on the datum feature at the location you would like to identify as the target location. Targets acquire the name of the datum and increment numerically. Datum targets are tied to a surface face or object. However, for greater positioning accuracy, the end of the leader line can be linked directly to points with known coordinates.

Feature Checks

Like Datums, Feature checks are added simply by selecting the check type button and then the associated features. The only major difference is that special attention needs to be given to the Tolerance and Datums assignment text boxes. These settings need to be entered prior to adding the check for them to be applied (Figure 27-4).

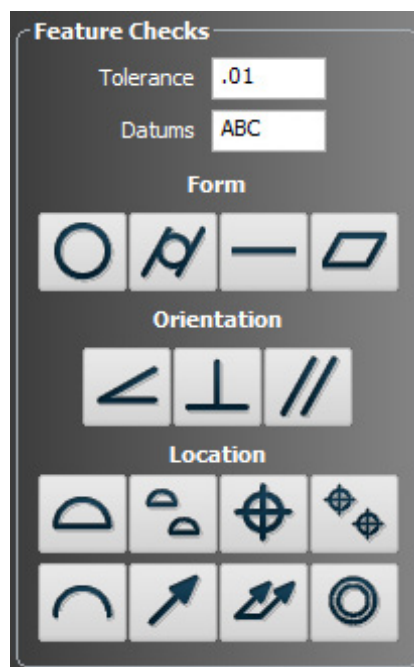
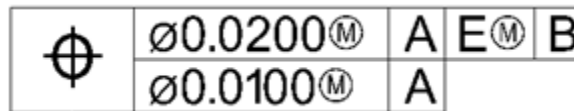


Figure 27-4. Feature Check Section of the Toolkit

can be added by first setting the selection mode clicking on the CAD face, object, or selecting points in the same was as datums are added. However, it is important to first specify the associated datum(s) and tolerance(s) used by the check. In addition to specifying any individual datums used in the check such as **ABC** you can also use “-” to indicate compound datums such as **A-B** and use **@L** or **@M** to specify the minium and maximum material conditions respectively. The composite true position check shown [Figure 27-5](#).

- **Tolerance.** $\varnothing.0200^{\text{M}}$, $\varnothing.0100^{\text{M}}$
- **Datums.** **AE@MB**, **A**

Figure 27-5. Composite True Position check.



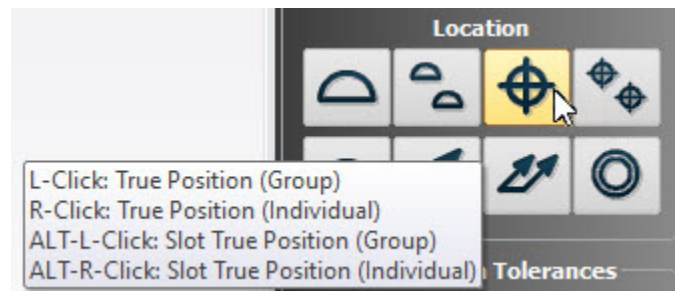
Tip: To add Feature Checks that reference double letter datums go to **GD&T >> Make Annotation for GD&T Feature Check** to build a Feature check.

Double letter datum entry in the Feature Check reference is currently not supported through the toolkit. To add Feature Checks that reference double letter datums go to **GD&T >> Make Annotation for GD&T Feature Check** and build a bank Feature check. Within its properties more complex datum structures including double letter datums can be added (such as AB-AC).

Group Verses Individual Options

Several options exist under the position button ([Figure 27-6](#)):

Figure 27-6. Position Group verses individual options are shown if you hover over the position button.



- **L-Click: True Position (Group).** The standard left-click mouse operation will create a single position annotation as a check on the selected features, including a patter of holes for example. Each of the features are evaluated, and the results can be found in the features table, but only the worst feature in the list is used for the check result.
- **R-Click: True Position (Individual).** This right-click option offers the ability to build a set of separate annotations on each of the selected features. Use this option when separate annotations are desired for each selected feature.
- **ALT-L-Click: Slot True Position (Group).** This option works as

above but pre-configures the position check as a **slot** and is required when working with open slots with MMC conditions. It can also be used to check the position of a pattern of round or square slots.

- **ALT-R-Click: Slot True Position (Individual).** This option will build a single annotation on each individual slot.

GD&T Dimensions and Tolerances

Dimension tolerances provide both a means of checking simple dimensional aspects of a part and also provide the base of reference for other checks that use a material modifier.

To add dimensional annotations you must first set the Tolerance type, whether that be **Limit** (which provides the option to enter the low limit and high limit values explicitly) or **Nominal +/-** (which provides the option to enter the nominal value and the low and high deviation values from that nominal). These settings are saved with the annotation and used in the evaluation. The tolerance definition can then be edited in the annotation properties later dialog if needed.

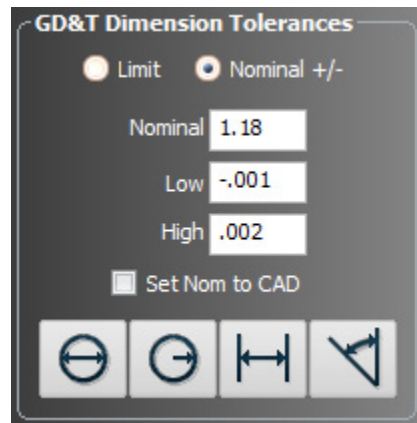


Figure 27-7. Dimension Section of the Toolkit

MMC Designations on Position

As noted above it is possible to include an MMC designation using the “@L” or “@M” designations in the Tolerance or Datum fields. When this is added, and the position annotation is created, an additional prompt will appear: “*Build Corresponding Diameter Annotations for Selected Features?*”

Pressing **Yes** will build a corresponding check using the settings pre-entered within the *GD&T Dimension Tolerances* section. Therefore, it is important to adjust these values before adding the position check in order to take advantage of this workflow.

Adjusting Size and Placement of Annotations

The placement and size of any annotation can be edited at any time. However, imported annotations will be converted when you do so. It should be noted that changing the properties of any annotation

including placement and scale will convert the annotation from the CAD format to the NRK format annotations. This may change the look and number of leader lines of these annotations to some degree.

The size and position of an annotation is saved in the annotation properties. While the properties of an annotation is open the graphical placement can be edited by dragging the annotation to a new location in the graphics window. There is also a numeric control within the properties dialog to adjust the size of the annotation text.

Global Adjustments

Global adjustments to annotation placement and text size can be made by using the utilities available through the Annotation Right-click Menu. *These functions will convert CAD imported annotations to NRK Annotations which will graphically change to some degree.*

- **Set Text Scale.** This option will allow you set the text scale for all annotations in the job file at one time.
- **Drag Annotations.** This Graphically Drag Annotations dialog allows you to edit the position and orientation of any selected annotation. Set the view, text rotate, and text size you want and then select an annotation to apply those settings.

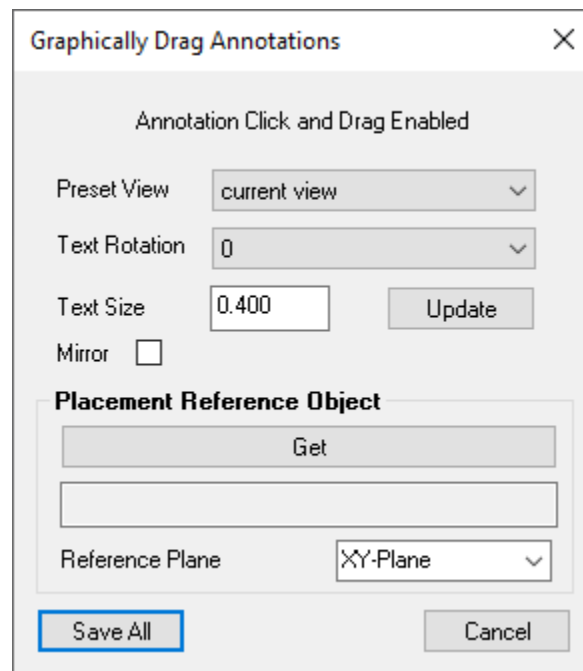


Figure 27-8. Drag Annotations Placement Control used to position multiple annotations at one time.

Using the Drag Annotations Tool.

Annotations can either use the “Current View” which dynamically rotates to face the screen at all times or can be constrained such that the annotation text is placed on a plane.

The hardest part of position an annotation in 3D space is controlling the depth. Your annotation may appear perfect but, when rotated 90 degrees, be far from the part because of this view plane issue. To address this the **Placement Reference Object** selection was added. This allows you to constrain the placement of an annotation to a plane and position it while remaining in Current View (screen oriented) display.

This is done as follows:

1. Select an object defining a direction, such as the World frame.
2. Select a reference plane such as XY-plane.
3. Position any annotations you would like in the graphics. Doing so will constrain the leader lines for the selected annotations to be parallel to the reference plane.

Alternatively, annotations can be directly constrained to a view plane by selecting a specific view. Doing so will lock the annotation text to a plane defined by the viewing direction and the annotation anchor point. This method works very well when combined with callout views.

GD&T Dimensional Annotations

Dimensional tolerances have greater constraints placed on them and only allow 2D placement with respect to the selected features. Right-clicking on a dimension annotation and selecting Adjust Leader Lines will bring up the following dialog (Figure 27-9).

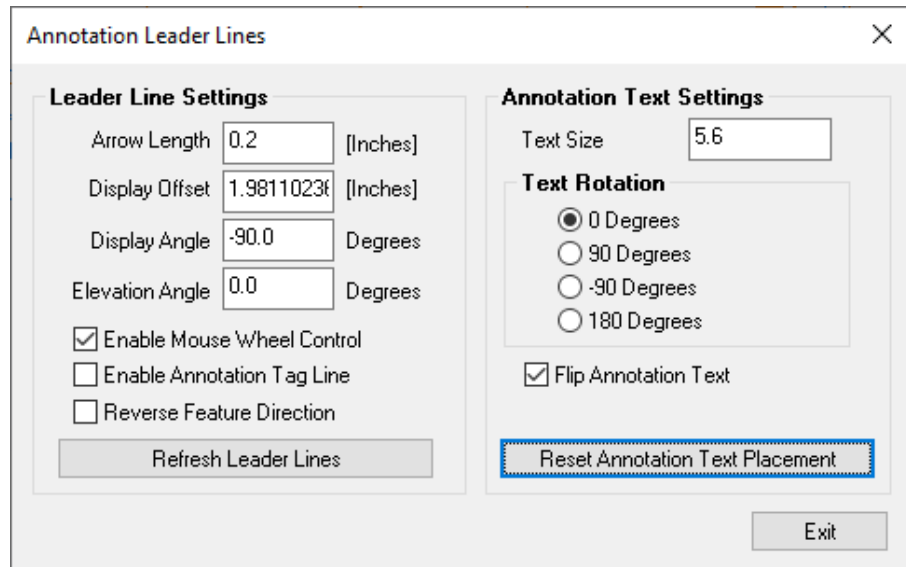


Figure 27-9. Dimension placement controls use for diameter, radial and distance between checks.

- **Display Offset.** (Ctrl + Mouse Wheel) shifts the annotation in and out from the reference features along the display plane.
- **Display Angle.** (Alt + Mouse Wheel) rotates the annotation about the selected feature
- **Elevation Angle.** (Ctrl + Alt + Mouse Wheel) tilts the angle of

- the display plane.
- **Annotation Tag Line.** Provides an option to display a line from the center of the dimension to the annotation text much like a standard annotation.
 - **Reverse Feature Direction.** Flips the display plane from one end of a feature to the other.
 - **Annotation Text Settings.** Annotation text can be graphically dragged into positioned and the radio selection boxes can be used to change the orientation of the text.

Annotation Layout Tools in SpatialAnalyzer are demonstrated here:

<https://youtu.be/k87DL-8rGo8>

GD&T Entities

If you look in the TreeBar in SA you will find that all imported and constructed annotations are contained within an *Annotations* category. Often it is useful to contain all the design CAD and Annotations within a single collection. You can then create a new collection for a particular inspection.

To inspect a feature simply right-click on the Annotations category and select *Make Feature Checks*. The annotation will be decoded and two additional categories will be built:

- **Datums.** Provide a link between the annotation and the measurements that are used to evaluate the datum. They also offer a means to control point offsets and cloud thing.
- **Feature Checks.** Like Datums, provide a link between the annotation and the measurements that are used to evaluate the check. They also offer a means to control point offsets and cloud thing.

Feature Checks and Datums are much like relationships in that they maintain the association between measured points and the features specified in the annotation. (For more information on associating points to with datums and feature checks refer to “*GD&T Workflow*” on page 664).

Collection Specific Inspection

Its important to note that Datums and Feature checks are isolated by collection. A fundamental principle in the way that GD&T inspection has been integrated within SA is to allow multiple parts to be inspected in the same SA Job file. This means that the working collection is critical when using controls offered in the Annotation heading such as *Make Feature Checks* or *Refresh Datums/Feature Checks from Anno-*

tations. These command will only update the datums and features checks within the current working collection.

This collection isolation also means that feature checks will only use the datums in the same collection as they are in. The examble below shows an example of 3 separate inspections using the same set of annotations (Figure 27-10).

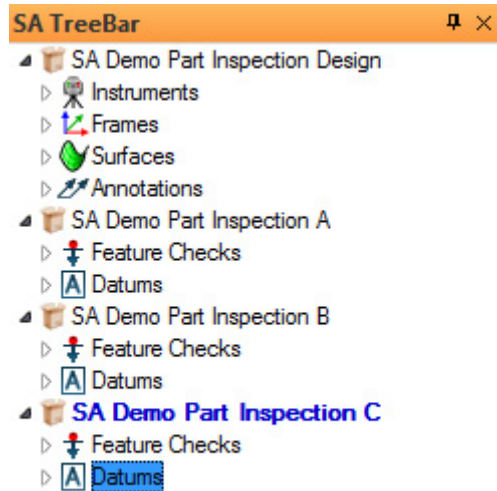


Figure 27-10. GD&T Entities in the Treebar.

The easiest way to find things in the tree is to right-click and select *Show In Tree* the feature checks links are now shown within the Annotation category as well to make things easier to find.

<https://youtu.be/62mO3GIYd7k>

GD&T Annotation Properties

Annotation in SA are used to define both the graphical placement, object associations and tolerance information used to define datums and checks. All GD&T Annotations will contain a subset of the following options regardless of the type of Datum or Check they represent.

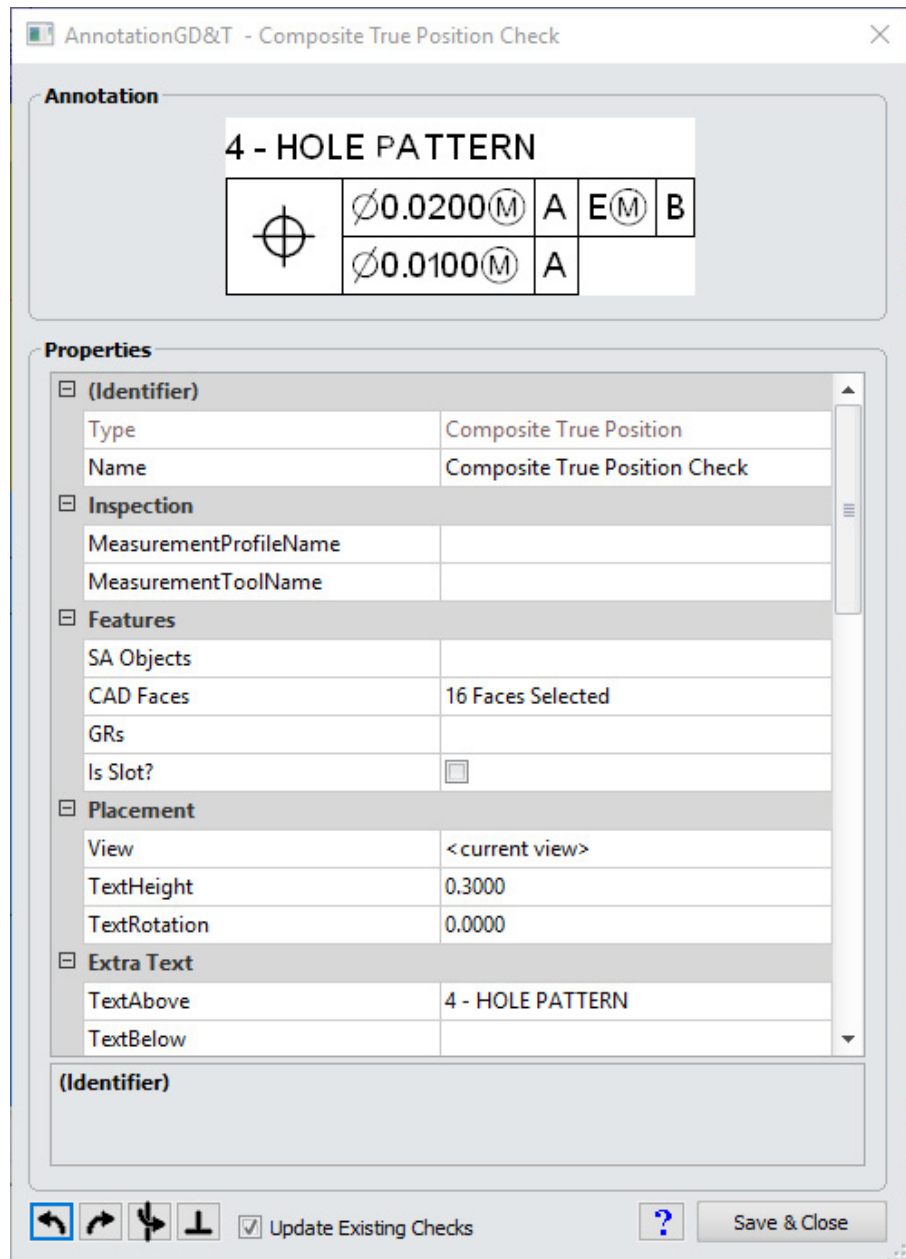


Figure 27-11. Example Annotations Properties Dialog.

The *GD&T Datum* annotation properties dialog allows you to edit the following settings:

(Identifier)

This section is used to describe the datum or check.

- **Type.** This field identifies the type of check represented by the annotation.
- **Name.** The datum letter used as an identifier in the annotation (currently support single and double letter annotations and will automatically exclude I,O,Q for ASME).


Inspection

These controls are used with the Inspection>Inspect process described in the Workflow Section (see “[GD&T Workflow](#)” on page 664).

- **MeasurementProfileName.** Controls the measurement mode set when inspection is initiated for this check.
- **MeasurementToolName.** Controls the tooling set when inspection is initiated for this check.

Features

The Feature Type section provides a list of the associated objects for the annotation. It also allows you to re-select the associated features and define additional reference objects as needed. You must select either CAD faces or SA objects.

 **Tip:** The point where the annotation leader line connects to the associated objects can be adjusted here. It used the first click point in the selection process.

- **SA Object.** The selected SA objects define both the annotation placement and the features evaluated as part of the check.
- **CAD Feature.** The selected CAD faces define the annotation placement and the features evaluated as part of the check.
- **GRs.** The selected GR-Features (Geometry Relationships) used to define the annotation placement and the features evaluated as part of the check. The Nominal geometry from the selected GR-feature will be used.
- **SA Offset Object.** SA Offset Objects provides directional reference information or alignment restrictions for surface datums.
- **Auto Detect CAD Faces.** Profile checks can have this option enabled which allows all measured faces to be evaluated under the specified tolerance. This provides a convenient means to apply a global tolerance to a part.
- **Is Slot.** This check box designates this feature as a slot which triggers a separate evaluation process specific to 2D slots.





Placement

These controls provide the ability to redefine the text size, placement and orientation in the job.

- **View.** The View defines the saved orientation of annotation text.
- **Height.** The Text Height defines the height of the annotation text. This is expressed in job units.
- **Text Rotation.** The Text Rotation defines the rotation in the current view.

With the annotation editor open the graphical annotation can be moved and placed in a position that is easy for the user to read. The

annotation leader line is defined using a click-point on the first surface face or SA Object selected when that feature is assigned in the annotation. To move the annotation simply select the annotation in the graphics by left-clicking the annotation. While holding the left mouse button the annotation can be moved in the plane of the screen. Using orthogonal views can help to precisely locate annotations in 3D. The view can be change under the View parameter in the annotation editor. Controls at the bottom of the annotation editor help in fine tuning the position and orientation of the annotation.

-  Rotates the annotation box 90 degrees counter clockwise.
-  Rotates the annotation box 90 degrees clockwise.
-  Mirrors the annotation box.
-  Straightens the leader line normal to the selected surface face.

Extra Text

- **TextAbove.** Allows a user to display text just above the reference frame for a check.
- **TextBelow.** Allows a user to display text just below the reference frame for a check.

Certain Alt + Number entries can be used to add symbols to this text line. A brief list of these symbols includes the following:

- **Alt + 0176.** Will produce the degree symbol °
- **Alt + 0177.** Will produce the plus & minus symbol ±
- **Alt + 0226.** Will produce the diameter symbol Ø

Cylinder

When a datum/check is added to a cylindrical feature the following additional controls will be provided.

- **Cylinder Mode.** Allows for selection between hole and pin cylinder evaluations. It is used to properly define probing offsets.
- **Axial Extent Mode.** Provides control of the way full cylinder evaluations are performed (see “[Cylinder Evaluation Methods.](#)” on page 631).
- **Auto Full Cyl Eval.** It also provides control of the way full cylinder evaluations are performed (see “[Cylinder Evaluation Methods.](#)” on page 631).

Initial Alignments

When working with offset objects and surface datums or pattern features it can be important to provide a good starting alignment for the evaluation to ensure the best solution is found.

- **Custom Initial Alignment.** This field provides the ability to reference a defined alignment in the tree (RPS or Sequent Alignment) that is used prior to the full GD&T evaluation.

CAD Feature

- **Force Surface Feature.** Provides a radio button which forces the selected feature to be interpreted as a general surface.

Datums

Datum designations for a check are established in this section, which consists of two parts: a summary section and the individual datum selections (Figure 27-12).

The summary section defines how many datums are referenced as primary, secondary and tertiary and outlines the construction of the lower tier with the user of check boxes.

The following details section allows you to edit each datum designation and call out any Material Modifiers used with those Datums.

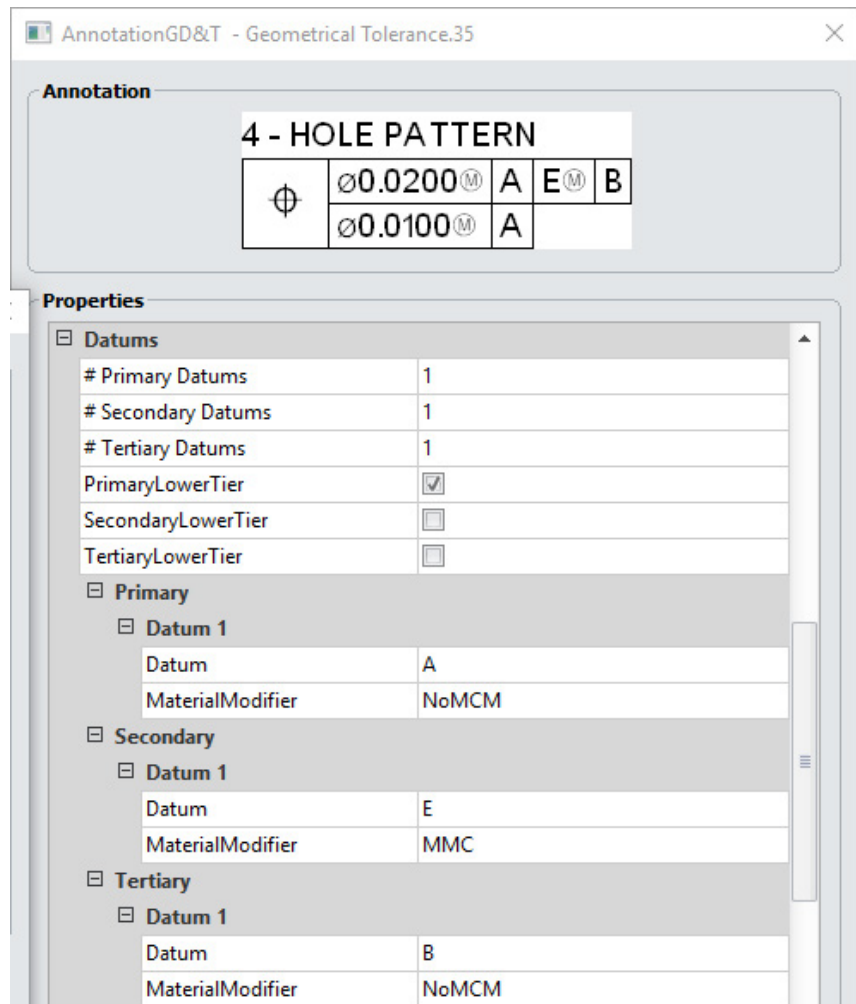


Figure 27-12. Datum Designations

Tolerance

- **Zone.** Applicable tolerance zone. See “[Feature Check Specifics](#)” on page 650 for more information on Zone options.
- **Tolerance.** Tolerance value for the feature check.
- **Material Modifiers.** Defines the material modifier for the feature check’s selected datum. Choose from *Maximum Material Condition*, *Least Material Condition* or *Regardless of Feature Size*.
- **Has Projected Zone.** If checked, the projected zone will be used.
- **Projected Zone Value.** Value used when projected zone is being used.
- **Tolerance Lower Tier.** Tolerance used in Lower Tier evaluation.
- **Outer Tolerance Lower Tier.** This designates the amount of the tolerance outside the surface of the part for the lower tier check. It allows for asymmetrical tolerance zones to be set. For symmetric tolerances this value should be set to one-half the total tolerance. Starting in the 2016.09.01 version Unilateral annotations are displayed graphically ([Figure 27-13](#)).

Figure 27-13. Unilateral Symbols



- **Material Modifier Lower Tier.** Optional material modifier designation for the lower tier tolerance.
- **Tolerance Type.** Dimension offer a tolerance type designation.

Datums and Feature Checks

The *Datums* dialog as well as the *Feature Checks* dialog ([Figure 27-14](#)) have identical settings with a few minor exceptions:

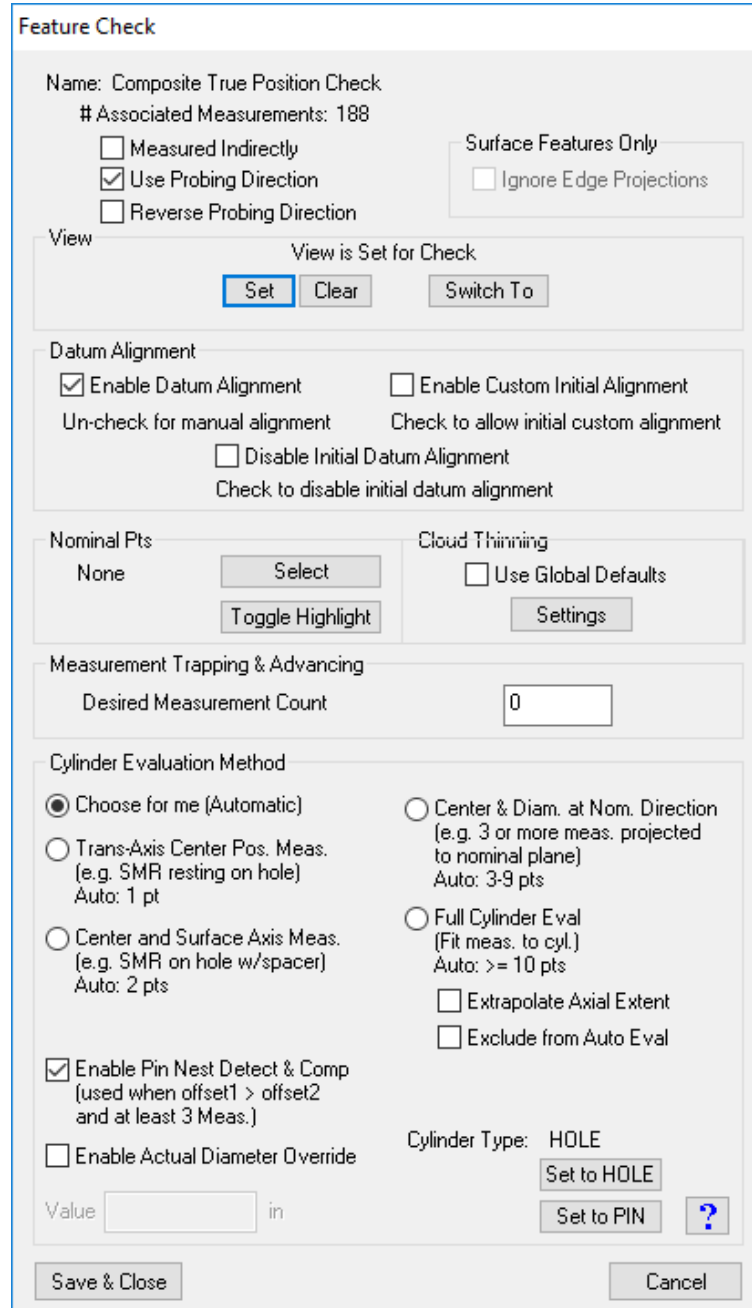


Figure 27-14. Datum and Feature Check properties dialogs.

- **Name.** Displays the name of the Datum/Check.
- **# Associated Measurements.** This displays the number of points associated with the check.

Point Controls

Under the Datum/Check name are a set of check boxes which define the way in which offset measurements are applied and allow for alternative probe offset controls.

- **Measured Indirectly.** At times, it can become necessary to measure a feature indirectly. Meaning for example, that you

measure the table top to represent the bottom surface of a part clamped to the table. Normally the probing direction defines the way in which we handle probing offsets. However in this case, you are measuring a mating surface, changing the offset direction. To account for this, we have an option in the *Feature Check* properties dialog that allows you to specify that the measurements were taken indirectly.

- **Use Probing Direction.** Probing direction is either the line of sight of the instrument (for a tracker as an example) or the direction vector recorded with a point taken by a 6D probe. Using this information typically ensures proper probe compensation.
- **Reverse Probing Direction.** There are times when a reflector is visible on the far side of a part but the measurement was taken in the opposite direction of the line of sight. This option allows probe compensation to be manually set in the opposite direction from normal use.
- **Reverse Normal (Unique to Open Slots).** The direction of an open slot or midplane is not well defined because it often is measured by the same instrument from 2 directions or by multiple instruments. Therefore a simple direction control is available.
- **Ignore Edge Projection.** Surface Profile checks are allows a little flexibility based upon the extent of the surrounding surfaces. This check box forces any points off the measured surface to be ignored.

View Controls

A saved view can be recorded with each feature check and displayed when it is either activated for measurement or evaluated for reporting purposes. Saving a preset view with a check ensures that a comfortable viewing angle is obtained whenever measurement is in progress. The GD&T **Inspection >Design, Inspection>Inspect** workflow process provides a convenient way to predefine views for a series of checks. The views from the Inspection Design interface are saved automatically and can be recalled or edited by going directly to this section in the specific datum or feature check.

- **Set.** Defines the current view perspective and zoom depth and saves it with the check.
- **Clear.** Removes the currently saved View.
- **Switch To.** Resets the graphics to the saved view.

In addition to the saved view orientations an option has been built into the right-click menu of the feature check category to enable **Show Only Active Feature Check**. When enabled, this option changes the visibil-

ity of the annotations such that only the annotations directly linked to the selected check are displayed. This included the selected check, the datums it references and any additional annotations that share the same feature associations.

Datum Alignment

A GD&T alignment performs a direct fit to the datums as an initial alignment, the measurements of the features to be checked are then divided up and the final evaluation process is conducted. These options provide tools to troubleshoot each aspect of the evaluation process.

- **Enable Datum Alignment.** Un-check for manual alignment. This performs a GD&T evaluation of the check in the current job alignment without using the GD&T control frame datums to align. When disabled this is no longer a standard GD&T evaluation and the datums will be marked out in the report. However, this option can be helpful in troubleshooting your results.
- **Disable Initial Datum Alignment.** This option provides the ability to skip the normal datum alignment process prior to point and feature processing. When pre-aligned this option can help ensure that the correct points are associated with the correct features in a pattern. However, it requires the points to be closely aligned to the part in order to evaluate correctly.
- **Enable Custom initial Alignment.** When a custom initial alignment is specified in the annotation properties this check box allows it to be applied as the first step in the evaluation process, providing the GD&T evaluate a better initial alignment.

Cloud Thinning

Controls the extent of thinning performed on point clouds associated with this check during evaluation.

- **Use Global Defaults.** With this default option enabled, the check will use the system thinning controls defined in the **Feature Check Cloud Thinning** settings on the Analysis tab of the Users Options.
- **Settings.** This button opens a dialog allowing you to specifically define the thinning used in that check, independent of the global settings.

Datum Alignment

- **Enable Datum Alignment.** When unchecked, the feature check will be evaluated in its current position and the datum alignment will not be performed. This is primarily for diagnostic purposes only (see [“Disabling Datum Alignment”](#) on page 638).

Nominal Pts

Nominal Points provide control of the selected nominal points used in the GD&T Inspection>Inspect routine. These points are used to lay out the bulls-eyes on the part used by the inspector as a guide to measure the part in the correct location. Nominal points can be added to the job through the Inspection Design interface or can be selected manually for a particular feature through this dialog. The highlight button can be used to graphically identify the nominal points.

- **Select.** Allows you to re-establish which points are associated as nominals for the check.
- **Toggle Highlight.** Allows highlighting to be toggled on and off for those associated points.

Measurement Trapping & Advancing

- **Desired Measurement Count.** This parameter is used to control the number of points captured from a live instrument during trapping. When the specified number of points are associated with the feature trapping is stopped automatically (refer to GD&T Workflow section).

Cylinder Evaluation Methods.

Cylinder can be evaluated either in 2D as circle center point locations or in 3D as an axis comparison along a length. Settings in this section control the way that the specific feature is evaluated (Figure 27-15).

Figure 27-15. Cylinder Evaluation Method section.

The position of measured cylindrical holes or pins can be established in either of 4 ways depending on the users needs. These methods are as follows:

2D Projected Evaluation Methods:

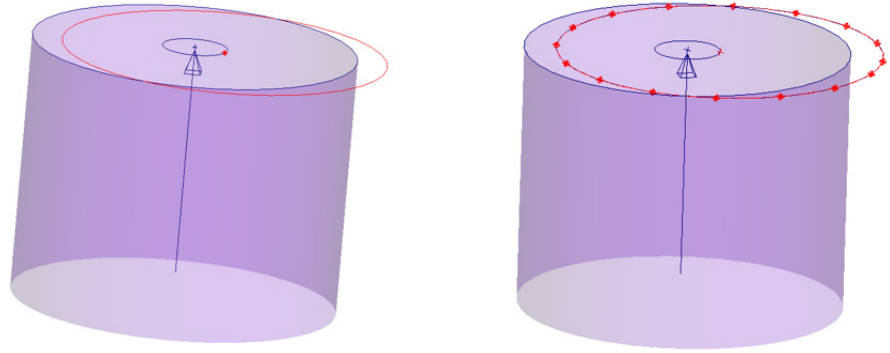


Figure 27-16. 2D Evaluation Methods



Warning: The Single Point Method should not be used for Primary or Secondary Datums because they do not control angular degrees of freedom. They are always oriented by the nominal cylinder axis and this can lead to incorrect datum alignments.

- **Trans-Axis Center Position Measurement.** (Single Point Method) This method takes either a single point or averages a set of points and compares the resulting point to the nominal cylinder axis while using the nominal direction and radius for the evaluation. An example of this type of check is resting a probe tip of a larger diameter on a hole.
- **Center & Diameter at Nominal Direction.** (Circle Center Method) This method projects the associated measurements to the end cap of the nominal cylinder. The results of the projection are then used to fit a circle. The circle then determines the center point and the diameter while the nominal cylinder is used for the direction. This is an excellent choice in situations where you have very little coverage of the cylinder on either side of a cross section plane to define the cylinder axis.

3D Axis Evaluation Methods:

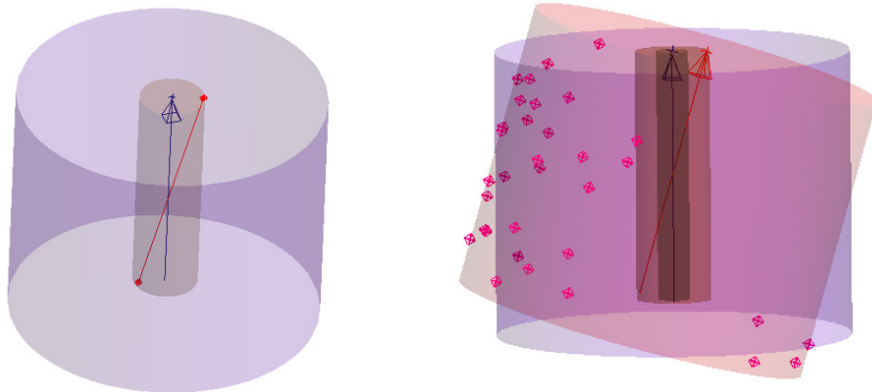


Figure 27-17. 3D Evaluation Methods

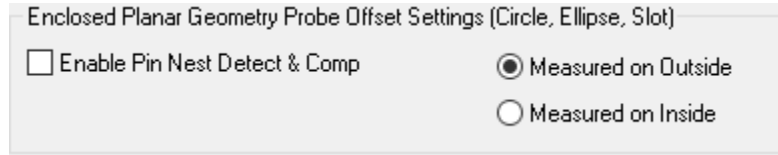
- **Center and Surface Axis Measurement.** (Axis Measurement Method) This method assumes that multiple measurements of the cylinder axis were taken at varying depths. A line is fit to the measurements and that line determines the center and direction of the cylinder while the nominal cylinder is used for the radius. For example, using a probe resting on a hole with and without a spacer could be used to define a cylinder center axis.

- **Full Cylinder Evaluation.** (Full Cylinder Fit Method) This method does a cylinder fit from the provided measurements. The points are assumed to be measurements of the cylinder walls (pin or hole), and the length of the cylinder is determined by the extent of the measurements along the cylinder axis. In this case the measurements are used to determine the central axis and radius/diameter of the cylindrical feature.
- *Extrapolate Axial Extents.* This option provides the ability to extrapolate the GD&T evaluation beyond the extents of the measured cylinder to the full extent of the nominal cylinder. This option is more true to the letter of the standard but the extrapolation can lead to false part rejection, particularly if there is sparse data coverage along the length of the pin or hole.
- *Exclude from Auto Eval.* This option allows prevents the Full Cylinder Evaluate mode from being used. This option is helpful when measuring with a limited number points and expecting a 2D result. With this option off and you measure 10 points for example we will do a full cylinder fit which can produced an unexpected result.
- **Choose for me (Automatic).** This is the default used when creating datums and checks with cylindrical features. In this mode, the evaluation method is determined automatically based on the number of measurements. When set to this option the following designations will be made: if one point is associated, *Trans-Axis Center Pos. Meas.* will be chosen; if two points are associated, then *Center and Surface Axis Meas.* will be selected; if three to nine points are associated, then *Center & Diam at Nom. Direction* will be selected; and if ten or more points are associated then *Full Cylinder Eval* will be chosen.
- **Enable Pin Nest Detect & Comp.** With greater than 3 measurements (use to define a planar direction) pin nest compensation can be used. The planar offset is applied in the direction of the plane defined by these measurements and then the radial offset is applied laterally based upon the cylinder type designation, either pin or hole.
- **Enable Actual Diameter Override.** This option allows the user to manually stipulate a diameter for a check. It was added to allow quick evaluation of other checks such as True Position checks that have a material modifier where only a single point was taken (and therefore the actual diameter is unknown).

Circles, Ellipses and Slots

While annotations can be added directly to features other than cylinders (circles, ellipses or slots), these are 2D only evaluations and offer only a limited selection of fit controls as shown here (Figure 27-18).

Figure 27-18. Circle, Ellipse and slot evaluation options

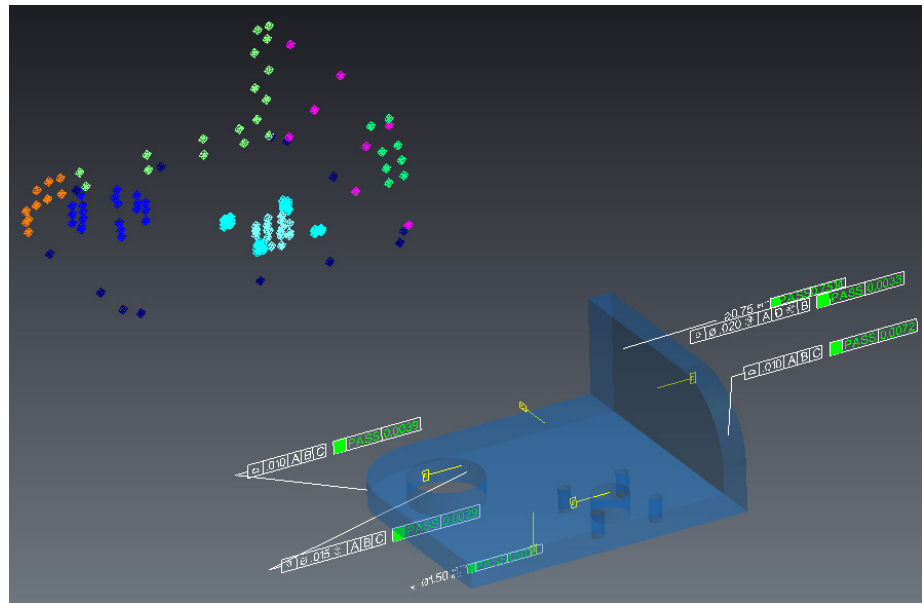


Selecting between inside and outside probe compensation is the only tool that is need to control the fit of these entities.

GD&T Alignment

When performing a GD&T inspection, alignment is performed relative to the specified datums for each individual check. This is often *not* the current instrument alignment in the job file. This can be a surprise to folks new to GD&T, but is required in order to correctly evaluate each check. You can see this is true by evaluating the GD&T checks with the instrument not aligned to the part at all and verifying they still report the correct results.

Figure 27-19. Results Evaluated without correct alignment.



The current instrument alignment is not used for analysis unless specifically specified with in its properties dialog (see [Figure 27-22](#) on page 638).

Warning: An initial alignment is needed for some evaluations that use a nominal direction reference so always pre-align prior to performing your GD&T evaluation.

Note that diameter checks and other checks that use the nominal features for reference will not evaluate correctly if you are not at least close in alignment with the nominal features. Therefore, an initial alignment is necessary prior to performing a final evaluation.

Tools

To get an idea of the alignment used by the GD&T evaluation with specific datums, right-click on the Datum Category and select “Align”.

You will be prompted to specify the datum scheme, picking primary, secondary and tertiary where applicable. This will then move the instruments and points associated with the datums to the correct orientation based on this datum scheme. Because each feature check is evaluated based upon its own datums it will be necessary to include or exclude only the datums you wish to use in the alignment to simulate the alignment for the particular check. This alignment tool is primarily for visualization and troubleshooting purposes. It will not exactly duplicate the evaluation in all cases. For this reason results should be compared the transform values reported in the feature check summary table.

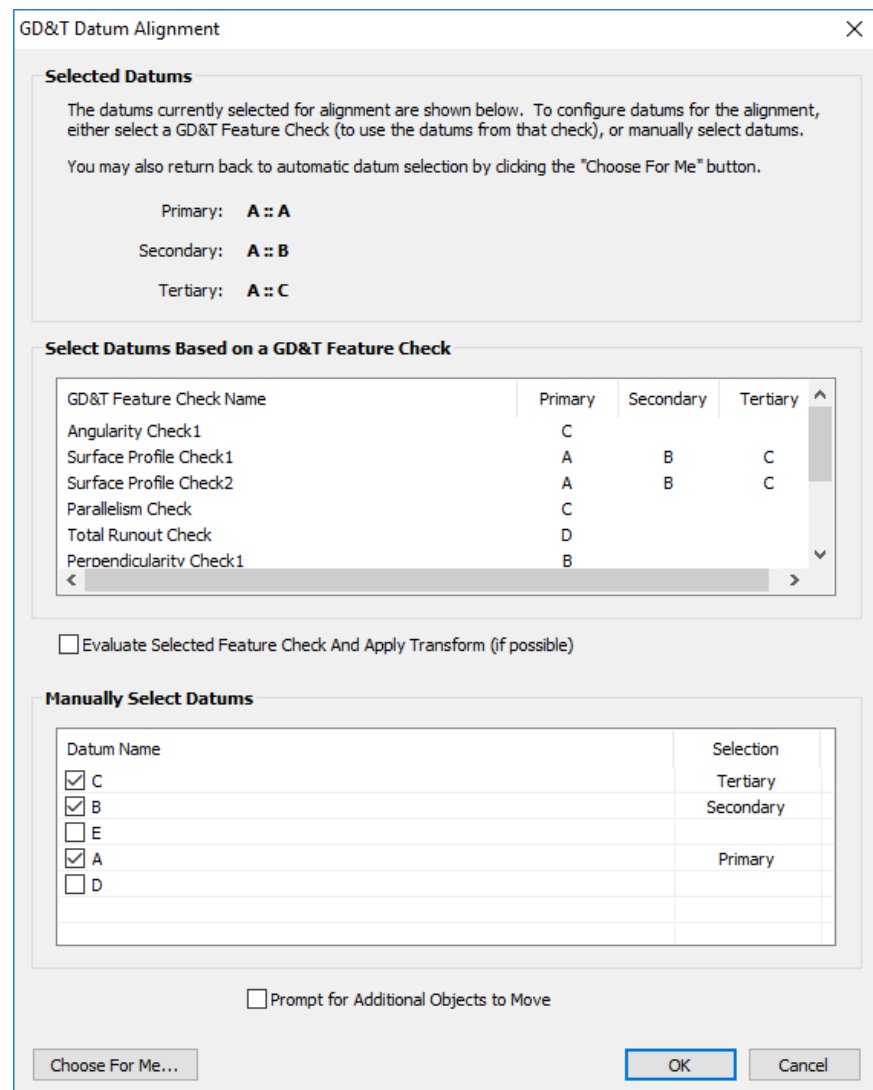


Figure 27-20. Datum Align Tool

Datum selection can be set in either of two way in the GD&T Datum Alignment dialog (Figure 27-20):

1. The middle section lists each feature check. If you wish to set the datum scheme to a match that of a particular feature check then the specific check can be clicked on in this list. Do-

ing so will set the datum selection accordingly.

Using the *Evaluate Selected Feature Check And Apply Transform (If Possible)* check box does just that. It uses the selected check and applies the resulting transform resulting returned from evaluating that particular check. If the solution is fully constrained, this will zero out the fit transform for this particular check. This can be very helpful in troubleshooting or when comparing to the results returned by a relationship.

2. The final or “Manual” section allows a datum scheme to be set that is not currently available in the job file or differs from one of the existing datum schemes. You can select and designate any particular datum as primary/secondary or tertiary as needed.

To accelerate and the calculation process and to ensure that an ideal alignment solution is found for a particular check it is highly recommended that an initial instrument alignment, using one of SA’s other alignment tools such as Quick Align to CAD or Relationship Fitting, is performed first prior to performing the check, particularly if using surface datums.

Summary Fit Transform and Alignment

The actual transform used for the check evaluation is reported in the Summary table for each feature check (see “[Reporting Options](#)” on page 674). You can directly use this information to align exactly to the reported position. To do so follow these steps:

1. Copy the Transform directly out of the report bar table using Shift + Drag Select to copy the transform to the clipboard. You may want to increase the decimal precision shown when you do this.
2. Set the World frame as the working frame and copy it.
3. In the copied frames transform properties press paste to enter the transform.
4. Use **Instrument>Locate (Transform to Part)>Frame to Frame Transformation** selecting the World frame as the Source Frame and the new frame as the Destination Frame.

Adjusting Datum Fit Directions

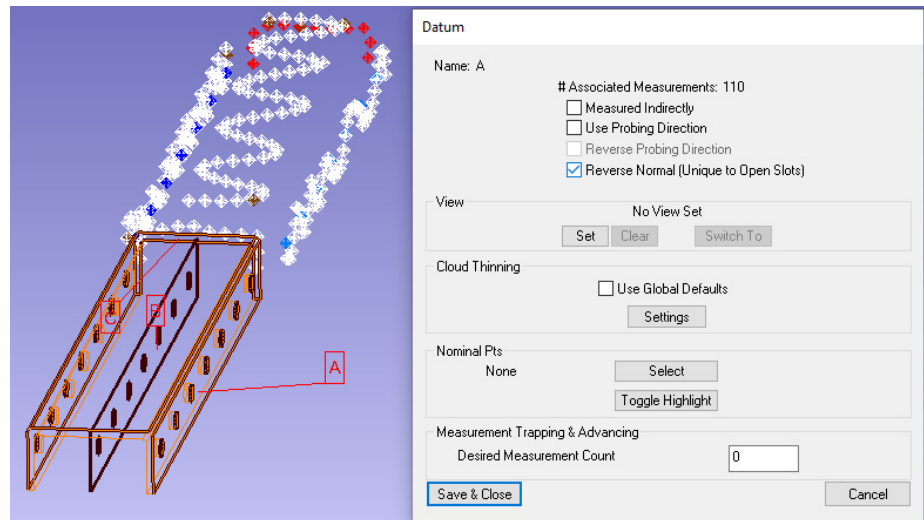
Datum alignments can produce surprising results from time to time because a datum could be flipped. This can happen in a couple of scenarios:

- A simple point measurement uses the line of sight from the instrument to the point to determine the positive side of the part. If you measure using an 1.5” reflector it is easy to mea-

sure around a corner, but in that case you need to identify that within SA.

- Open slots or mid planes have a direction that also needs to be controlled (Figure 27-21).
- Finally you can measure a part indirectly such as measuring the table top to define the bottom of the part.

Figure 27-21. Incorrect alignment resulting from a datum being flipped. This can be fixed in the Datum properties.



Datum alignment control is handled within the Datum properties (see “Point Controls” on page 628).

Pattern Datums

Multi-feature datum configurations are supported such that a pattern of holes can be used for alignment. However, all input feature must be cylinders (or cylindrically faces). Slots and circles are current not allowed as part of a pattern datum, but cylinder evaluation methods do apply such that 2D cylinder evaluations can be made if circles are desired.

Surface Alignment

When using datums that reference surfaces that are of a complex or non geometric shape (non-planar nor cylindrical) some additional requirements are involved in order to clarify which degrees of freedom each datum is intended to control. There are two approached that can be used to do so: using offset objects or providing a custom initial alignment.

Using Offset Objects

- The data should be pre-aligned to the part using other non-GD&T techniques before attempting GD&T analysis, such as Quick Align to CAD or relationship fitting.
- It is necessary to specify an **SA Offset Object** within the datum an-

notation which will be used to determine the degrees of freedom to constrain for that datum during alignment.

This offset geometry is what will “maintain contact” during alignment to subsequent datums and/or the main tolerance feature and so it needs to be chosen in such a way that it represents the control the datum is intended to exert over the solution. For example, a slightly curved plate designated as a datum can be treated as either a planar surface or a cylindrical surface. The offset object selection clarifies this.

GD&T evaluation begins with an initial alignment step performed internally and also use an iterative solution. When surface datums are involved both are disabled. This was necessary to ensure stable results with surface datums and since the data is required to be close before starting they aren’t needed anyway.

Custom Initial Alignment.

An alternative to using offset objects is to provide a full reference alignment. Custom Initial Alignments provide the ability to reference a defined alignment in the tree (either RPS or a Sequential Alignment) that is used to align the features for evaluation. To define a Custom Initial Alignment the following steps must be performed:

1. An RPS or Sequential Alignment must be developed in the tree.
2. This alignment must be referenced from within the Annotation properties.
3. The **Enable Custom Initial Alignment** option must be checked to indicate that this particular check is to use that option.

When enabled this custom initial alignment will be executed as part of the evaluate of the check.

Disabling Datum Alignment

When doing a GD&T inspection, alignment is performed relative to the specified datums for each individual check. The current instrument alignment *is not* used for analysis unless specified. However, there is an option in the Feature Check propties menu to disable datum alignment. When checked, the feature check will be evaluated in its current position and the datum annotation will be crossed out in the evaluation (see [Figure 27-22](#)).

Figure 27-22. Disabled datum alignment.



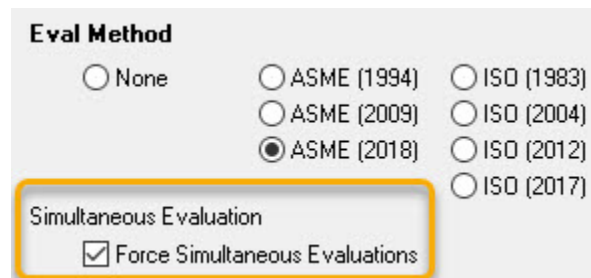
This can be a very helpful tool in understanding why a check is failing because it allows you to separate the datum alignment from the

feature evaluation process.

Simultaneous Evaluations

Position and Profile checks using a common Datum Reference Frame (DRF) can be evaluated simultaneously if so desired. This evaluation method is a specified method within the ASME standard and can be more true to real life in that it more directly replicates part assembly limitations. However, it is not the default in ISO evaluations and must be enabled, through the User Options (see “GD&T User Options” on page 640), to be performed as default within SA.

Figure 27-23. Simultaneous Evaluations can be set as default in the GD&T User Options



As of SA 2024.1 there are now two ways to perform a simultaneous evaluation. You can either go to the *User Options* and turn on the option to *Force Simultaneous Evaluations* which will ensure that any check that is evaluated will do so, or a simultaneous evaluation can be manually applied with the default turned off, as follows:

To use a simultaneous evaluation, right-click on one of the feature check in the list, select **Inspection > Simultaneous Evaluation**. The Initial transform from this selected check will then be applied (which could be a Custom Initial Alignment) and then all of the checks that reference the same DRF will be evaluated simultaneously as a unit.

When manually performed, the check results will then be locked such that each check result can be independently viewed without forcing a recomputation. They will also be marked as such in the Datum and Tolerance Summary Table (Figure 27-24).

Figure 27-24. Simultaneous Evaluation results noted in the Datum and Tolerance Summary Table.

GD&T True Position Check				
True Position 1 (Reported in A::WORLD)				
Simultaneous Evaluation - Results Locked				
CUSTOM INITIAL ALIGNMENT - A::Datum ABC				
v		Dist Out		
0.0016		0.0000		
Datum Alignment Results				
Tx	Ty	Tz	Rx	Ry

In order to re-evaluate these checks the results will need to be cleared by right-clicking on the feature check category heading in the tree and selecting **Clear All Check Results**. This is not the case when the default is enabled.

For many parts a mix of simultaneous and free evaluations may be specified. In that case an integrated MP may be the best approach to ensure the correct checks are evaluated as intended. This 2024.1 version also adds full support for simultaneous evaluations through MP.

GD&T Evaluation

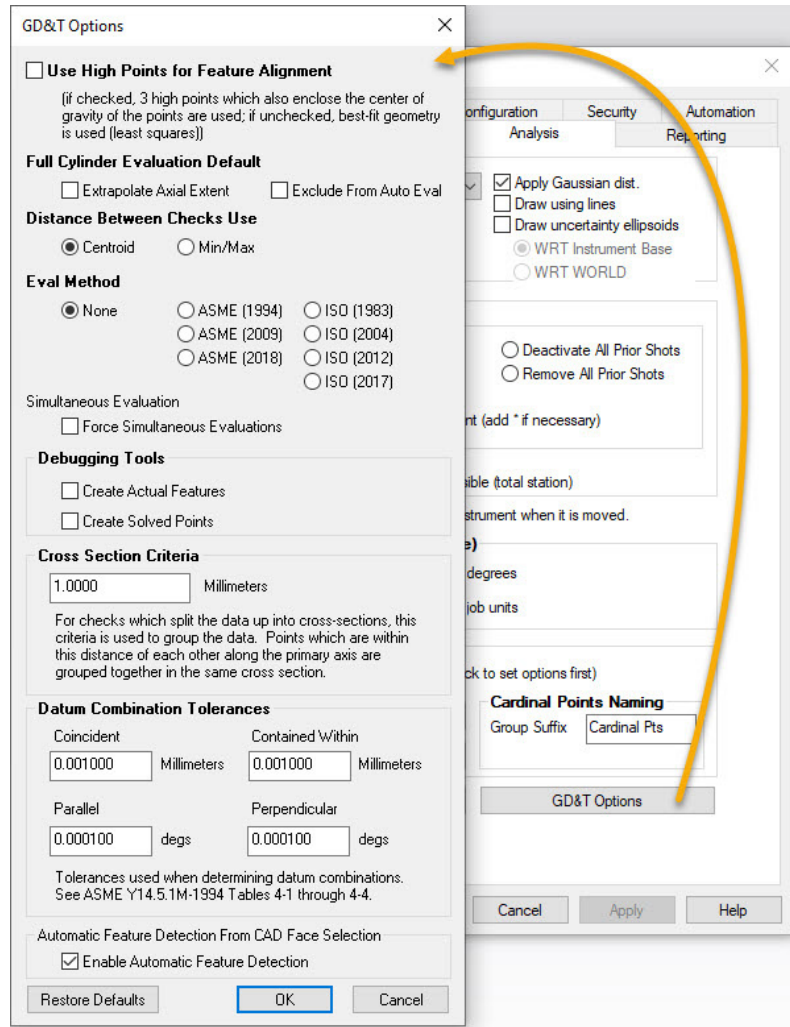
As feature checks are measured, the number of points associated with them are shown in the tree. The evaluation process, however, is iterative and manually triggered for efficiency purposes.

To evaluate a particular check you can view it in the report bar. As long as the report bar is up you can click on a check it directly in the tree and it will re-evaluate to ensure the status is current. Alternatively, an **Evaluate all Checks** option is available in the right-click menu of the feature checks category. Once evaluated, a PASS/FAIL status with then also be displayed in the tree. You can also Go to **GD&T>Evaluate All Feature Checks** to evaluate all the checks in the job file.

GD&T User Options

There are several important global evaluation settings to select from found in the GD&T Options within the Users Options Dialog on the Analysis Tab ([Figure 27-25](#)):

Figure 27-25. GD&T Options dialog within the User Options dialog.



- **High Point Utilization for Feature Alignment.** If checked, three high points which also enclose the center of gravity of the points are used while if unchecked, the best-fit geometry is used (Least Squares).

Full Cylinder Evaluation Defaults

- **Full Cylinder Evaluation Defaults.** These options control the default fit settings passed to the annotations and define the way in which cylinder/Pin measurements are evaluated (see “[Cylinder Evaluation Methods](#).” on page 631). The default settings are applied to annotations when they are built or imported but can be edited within the annotation properties directly.

Distance Between Checks Use

- **Distance Between Checks Used.** You can choose between using the Centroid or the Min/Max when reporting the distance between checks.

Cross Section Criteria

- **Cross-Section Criteria.** In order for a group of measurements to be considered a cross section, they must be within the default 1mm of each other in the cross section direction. In other words if you fit a plane to the data none of the points would be greater than .5mm from the plane. In order to take measurements in this format a geometry trigger measurement mode would be easiest. This cross section criteria can be adjusted by selecting **GD&T Options** under the *Analysis* tab in the **User Options** dialog. The number of cross sections used in the analysis is reported in the summary table of the check.

Datum Combination Tolerances

- **Datum Combination Tolerances.** These tolerances are used to determine how nominal features are to be handled. It accounts for slight variations in a nominal model and allows for grouping of nearly perfect objects as needed within these categories.

Automatic Feature Detection From CAD Face Selection

- **Enable Automatic Feature Detection.** This option can be used to enable or disable automatic face detection. When enabled, SA will search adjacent faces on the CAD model and include them in the selection. This allows all 4 faces defining the walls of a hole to be selected with a single click, as an example.

GD&T Evaluations verses RMS fits

The GD&T Analysis within SA follows the ASME standard evaluation. Therefore you will find that the results of a geometry fit or relationship fit which are based on an RMS fit will not return exactly the same results. Take the example of a flatness check. When we build a geometry relationship fit to the points we report the flatness. This value is based on comparing the point deviations to the best fit plane. In GD&T, however, flatness is evaluated as the separation between two parallel planes bounding the data. These planes are then free to rotate in space to find the closest position they can reach to each other while still containing all the measured points between them. This process will almost always return a slightly better result than an RMS fit because of this additional degree of freedom. But it is also much more influenced by outliers in the data.

Decimal Precision

It is important to understand that the displayed decimal precision imported with an annotation or set in the Users Options does not limit the underlying GD&T analysis. So, for example, if you have the deci-

mal precision set to 2 digits and have a flatness check with a tolerance of 0.02, the check can still fail if the measured deviation is 0.02001. This makes sense considering that the standard is based upon hard limits. The measurement is compared directly to the specified tolerance value and any deviation outside the limit signifies nonconformance to those limits. If you can't get the pin in the hole then it fails.

Evaluation Zone/Length

The evaluation zone/length is set by the extents of the measurements associated with the feature check. Therefore, if you have a tolerance specification for a particular length part but can only measure a portion of that part. It may be necessary to adjust the tolerance to account for the shortened evaluation zone.

Cylinder Evaluation & Coverage

Cylinders can be evaluated either in 2D or 3D depending on the settings defined in the users Datum or Feature Check properties dialog. By default the number of points will be used to determine this method. For more details on these settings see "[Cylinder Evaluation Methods](#)." on page 631.

When measuring cylindrical features and using the "full eval" mode, it is important to provide enough measurements so that the intended cylinder orientation can be determined from the measurements.

Consider the following cylinder with 8 measurements in [Figure 27-27](#). It is possible to interpret the cylinder in 3 different basic orientations.

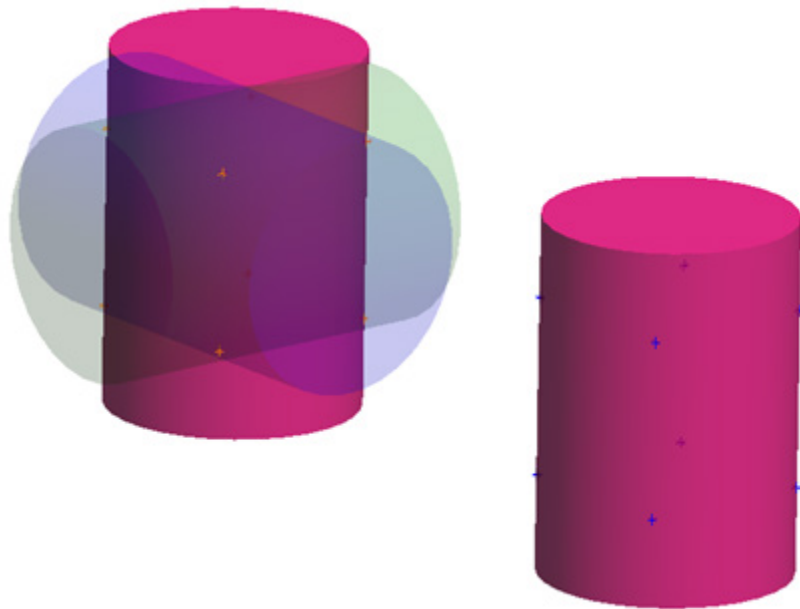


Figure 27-27. Possible cylinder fit orientations with only eight points defining a cylinder location.

To avoid this problem, cover the cylinder more completely with measurements so that it is clear which of those 3 orientations is intended.

One question which may arise is why the nominal feature is not sufficient to lock in the correct orientation. When SpatialAnalyzer evaluates a GD&T check it first performs an initial alignment on the data representing the datums. This initial fit is done without knowledge of the nominal orientation because data is often far out of alignment initially as is common with portable metrology. Once a fit is performed the fit cylinder is compared with the nominal cylinder. Using the above as an example, if the wrong cylinder fit result was chosen (not having the nominal information) then the alignment would attempt to rotate the data incorrectly. Providing more complete measurement coverage avoids this problem.

Slots and Open Slots

SA Supports the use of rounded slots, square slots, and open slots (or mid-planes) in the evaluation process.

Standard Slots (Round/Square)

In order to use slots to define datums and checks they need to be manually identified in the annotation in order to be recognized in the evaluation process (Figure 27-28).

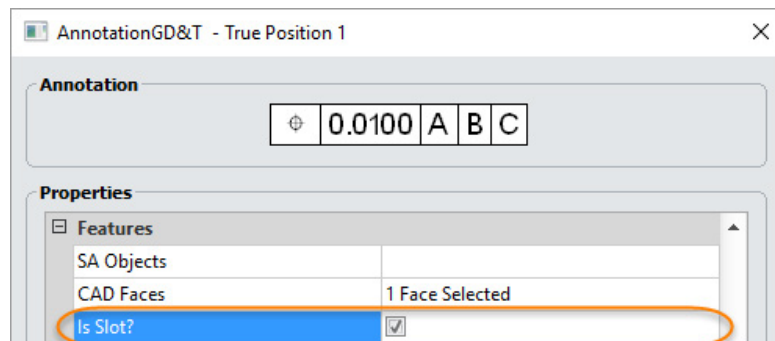


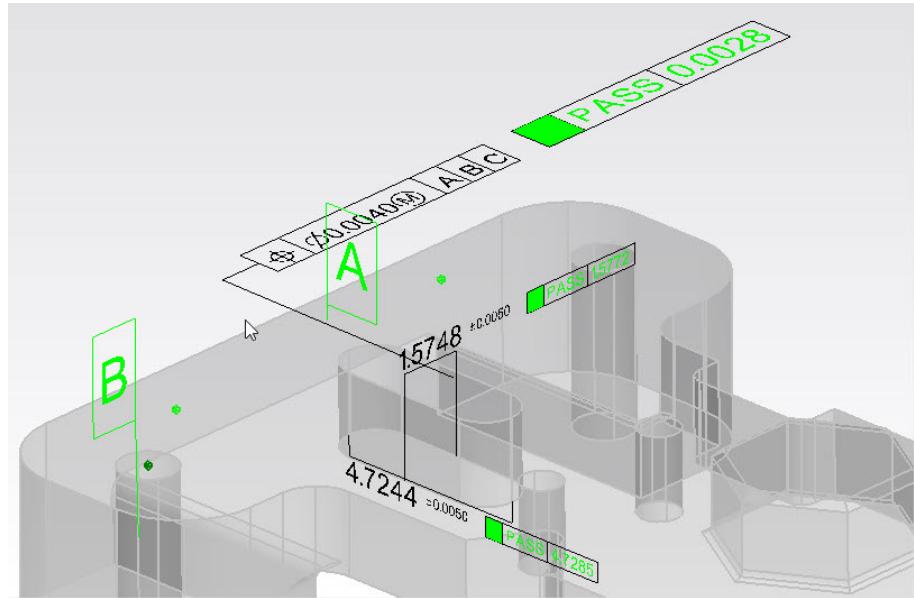
Figure 27-28. Slot Designation

Marking a check as a slot triggers SA to search the surrounding surface faces for a representative slot feature. Only a single associated CAD face is necessary and additional face selections will only slow the search process.

Slots are 2D entities. For this reason it maybe useful to either measure a slot with a pin nest or using an alternative depth control.

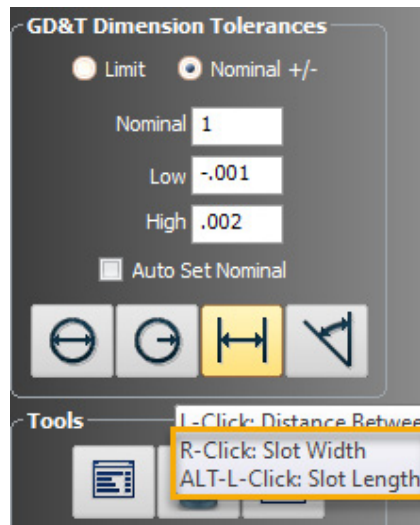
Like other holes, slots can be evaluated in position. Material Modifiers can be applied in both length and width. To use a modifier on a slot, a *Width* check must always be added. To verify the position of a slot in both Length and Width with modifiers, add both supporting checks. The MMC condition will be defined by the most restrictive of the two checks.

Figure 27-29. Position check on a round slot using MMC with both length and width checks



Width checks are a special case and are not the same as a *Distance Between* check. In order to define a Width check through the SA Toolkit a new right-click option has been added to the Distance Between check button (Figure 27-30).

Figure 27-30. Right-click Option to add a Width Check

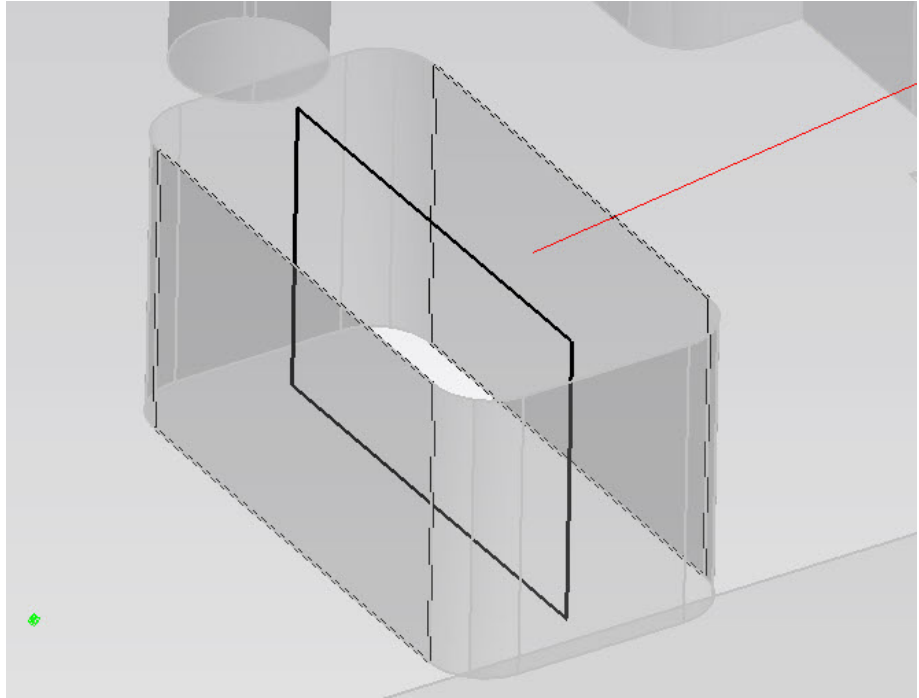


Open Slots & Width Datums

In addition to a closed slot features, SA can also evaluate the position of a midplane or open slot.

Position checks on a midplane can be enabled using the *Is Slot* check box. In addition, there is a requirement that 2 and only 2 parallel planar features are associated with the check. When a midplane is defined you will see the midplane displayed graphically when the check is highlighted (Figure 27-31).

Figure 27-31. Mid Plane Evaluation of a slot.

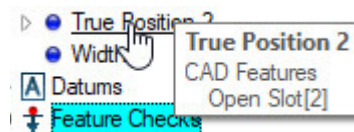


In order to evaluate a midplane as an open slot with an MMC designation there are additional steps required to get the width and position checks linked up. These are as follows:

1. Go to the *User Options > Analysis Tab > GD&T Options* and turn OFF the *Automatic Feature Detection*.
2. In the GD&T toolkit enter the tolerance followed by “@M” to designate the MMC designation and adjust the dimension settings as needed.
3. Use *Alt+Left Click* on the position check button to add a slot dimension and its corresponding width check.
4. Open the position check annotations properties and edit the Tolerance zone to be Planar and select a plane to define the direction of the check (typically a plane built from one of the two parallel surface faces).

Additional steps are required because the Automatic Feature Detection searches for underlying geometry. SA will identify two planes and will not consider the check to be an open slot unless so designed. This can be seen if you hover over the position check once created. The association needs to be set to Open Slot (Figure 27-32):

Figure 27-32. Open Slot feature association.



Once set correctly the feature check will be evaluated and take advan-

tage of the width check's bonus tolerance (Figure 27-33).

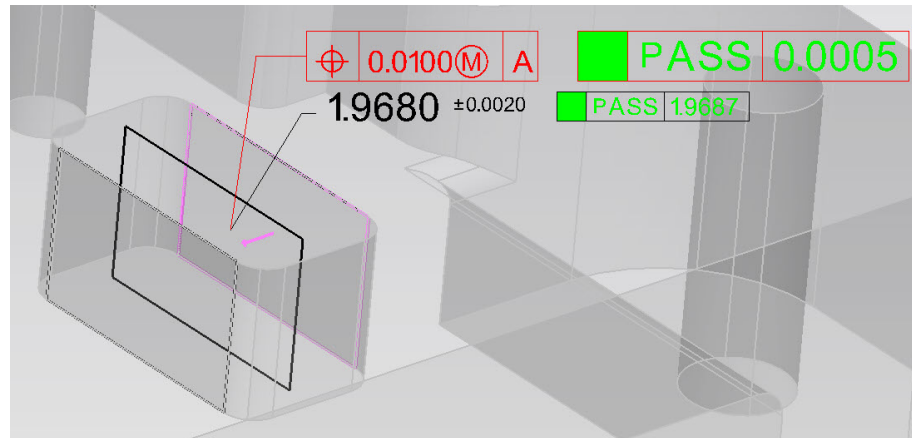


Figure 27-33. Midplane slot with MMC width check.

Point Cloud Thinning Controls

Point Cloud measurement adds an additional level of complexity to GD&T evaluations. A desire to use all available data often needs to be weighted against obtaining a quick and accurate solution. SA provides a selection of cloud thinning and subsampling options to help with this task. The global **Feature Check Cloud Thinning** control is available on the Analysis Tab in the User Options (Figure 27-34).

A description of the individual settings is available in the chapter on point clouds (see “[Cloud Filtering and Sub-Sampling](#)” on page 570).

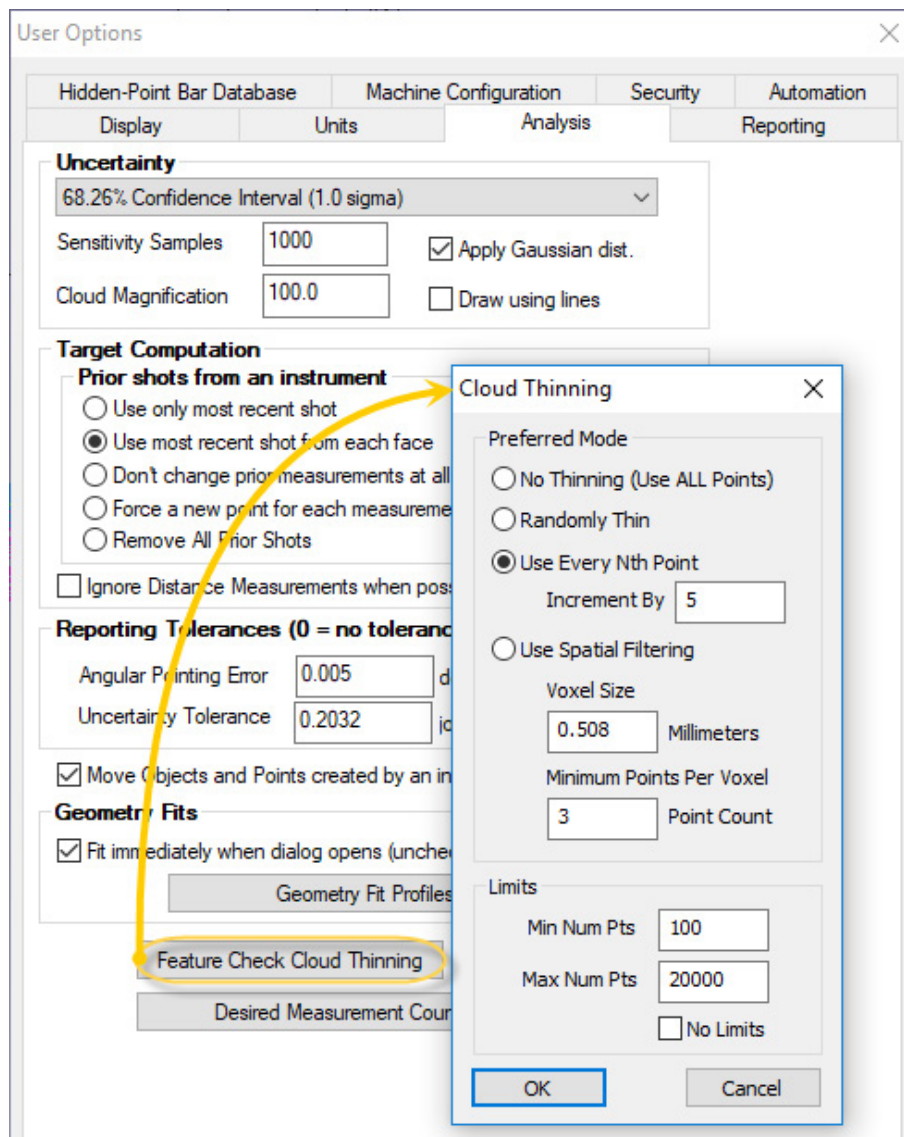
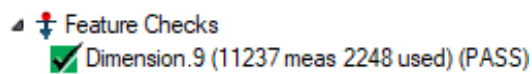


Figure 27-34. Cloud Thinning Controls

In many cases a single check may need to be sub-sampled to a different extent than other checks in the job. An example of this would be a planar region with multiple holes. The plane could be over sampled extensively in capturing data to define the hole positions. Equalizing the data coverage in all the features would in this case speed the processing time.

The individual check thinning controls are in the right-click Feature Check Properties (see Figure 27-14). The total number of points included in the check and the number used are both reported as part of the evaluation results in the tree for clarity:

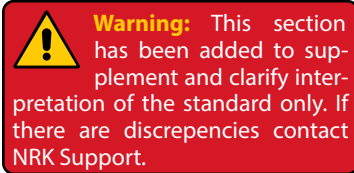
Figure 27-35. Cloud thinning and point usage reported in the tree.



The Details Table of the report will only report the points used for the

evaluation (see “[Details Table](#)” on page 677).

Feature Check Specifics



This section has been offered to help explain SA operations, as a supplement to directly using the standard, not as a replacement for it. The hope is that it will provide some insights into how checks of different types are evaluated in SA. A simplified summary of the intended use of each check is provided as well as an example evaluation process. For additional information or clarification always check with the standard and if there are discrepancies contact support@kinematics.com.

Cross-Section Checks

A number of Feature Checks require measurements to be made in cross sections to be correctly evaluated. These checks are marked with an asterisk symbol (*) and include the following checks:

- Circularity
- Straightness
- Profile of a Line
- Circular Runout
- Concentricity

Keep in mind that if measurements are not within a 1 mm band within cross sections they will be considered part of a second cross section. This can have a signification impact on the evaluation process (see “[Feature Check Specifics](#)” on page 650).

Form Checks

Circularity*

Circularity is evaluated in cross sections along the normal vector of the nominal feature (circle or cylinder). The tolerance establishes the maximum distance allowable between two concentric circles containing the measurements of that part at the cross section location ([Figure 27-36](#)).

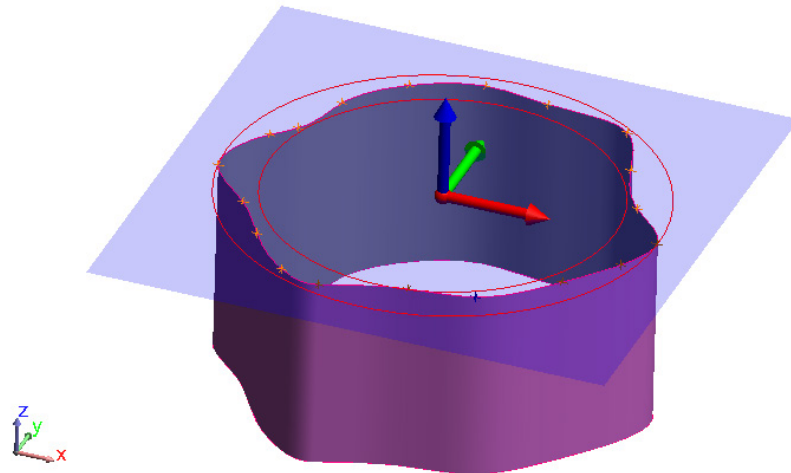
Evaluation Process within SA:

1. Cross sections are established along the normal vector of the nominal feature and the data is divided into groups based upon their position along this vector.
2. A plane is fit to the first point in each cross section group (using the actual feature direction vector) and then the points are projected onto that plane.
3. The minimum size bounding circles are then established on

the plane containing the data while holding a common center point and the distance between these is used to evaluate the cross section.

4. The check reports the worst of the cross section results.

Figure 27-36. Circularity



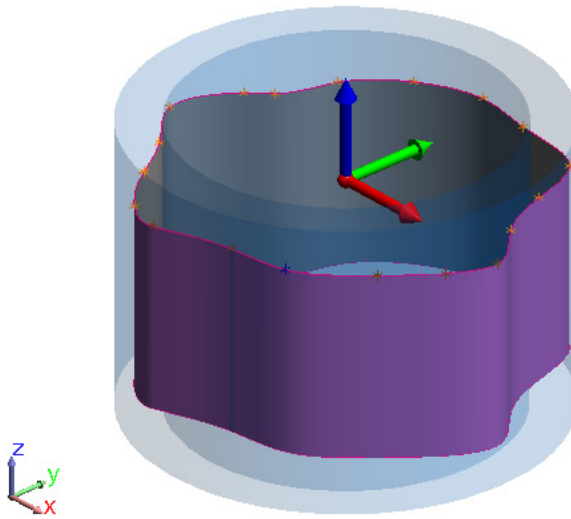
Cylindricity

Cylindricity checks ensure that all the specified points are the same distance from a common axis. The tolerance establishes the maximum allowable distance between two concentric cylinders containing the measurements of that part (Figure 27-37).

Evaluation Process within SA:

1. Two coaxial cylinders are built bounding the extents of the data.
2. The cylinders are allowed to freely rotate to establish the minimum distance between them while still containing the data.
3. The final distance between these cylinders is compared to the tolerance for evaluation.

Figure 27-37. Cylindricity



Straightness*

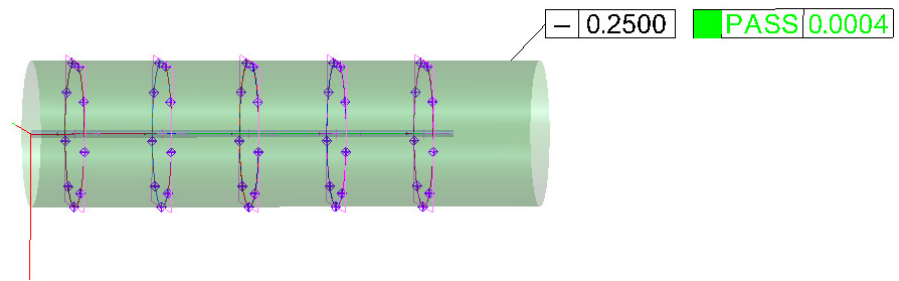
Straightness is evaluated in cross sections. It ensure that all the measured points are within a specified proximity to a straight line. This can be evaluated for lines directly or cylinders (axis), which define a zone where the median line must lie (Figure 27-38).

Evaluation Process within SA:

1. The direction vector is established for the measured feature (line or cylinder)
2. Cross sections are established along this direction vector and the data is divided into groups based upon their position along this vector.
3. A plane is fit to the first point in each cross section group (using the actual feature direction vector) and then the points are projected onto that plane
4. Median points are established for each cross section, either directly (for lines) or by fitting circles to the data (cylinder)
5. These median points are then compared to the nominal axis line and the check reports the worst of the cross section results.

Per Unit Length Straightness

Straightness can also be evaluated on a per unit length basis. This setting ensures that the specified straightness is held within an acceptable distance but allows a gradual curve to occur on a large part. When selected the straightness tolerance will be evaluated only within the specified length and then a new segment will be evaluated in sequence down the part. The *Per Unit Length Step Over Percentage* setting controls how much overlap is used in adjacent evaluations.

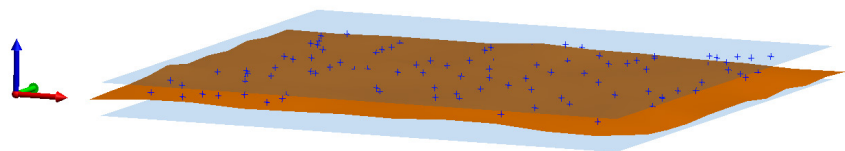
Figure 27-38. Straightness

Straightness can also be evaluated using a per unit length setting. Typically all the measurements are used within a common tolerance, but with this setting enabled a tolerance can be applied within a specified length. To enable **Per Unit Length** open the annotation properties and Check the option. You will need to specify the **Per Unit Length Distance**, which defines the length used for the evaluation and the **Per Unit Length Step Over Percent** which determines the extent of overlap in the analysis process.

Flatness

Flatness checks ensure that all the specified points lie approximately on a common plane. The tolerance establishes the maximum allowable distance between two parallel planes bounding the associated measurements of the part (Figure 27-39).

1. Two parallel planes are built bounding the extents of the data.
2. These planes are allowed to freely rotate to establish the minimum distance between them while still containing the data and the resulting distance is compared to the tolerance.

Figure 27-39. Flatness

Per Unit Area Flatness

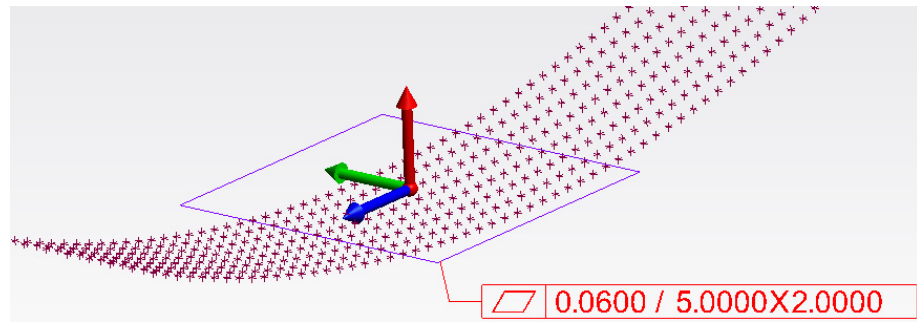
Flatness can also be evaluated on a per unit area basis. This setting ensures that the specified flatness is held within an acceptable distance but allows a gradual curve to occur on a large part. To enable this mode, check the **Per Unit Area** option under the tolerance and additional options will appear (Figure 27-40).

Figure 27-40. Per Unit Area Flatness Tolerance Settings.

Tolerance	
Tolerance	0.0600
Per Unit Area	<input checked="" type="checkbox"/>
Area Type	Rectangular Area
Per Unit Area Length Distance	5
Per Unit Area Length Step Over Percent	50.0000
SA Object	1 Object Selected ...
Per Unit Area Width Distance	2
Per Unit Area Width Step Over Percent	50.0000

A separate length and width area designation can be used to apply the tolerance in a larger or smaller area along the two directions. The length direction is defined using the Z-axis of the selected SA Object.

Figure 27-41. The Z-axis of the selected SA object is used to define the Length direction in a per unit area flatness check.



The above example applies the tolerance in a 5 inch area along the length and a 2 inch area along the width assuming the working frame was selected as the direction object.

In the summary data, the origin for each of the zones is NOT the planar zone origin, it is the centroid of the points that were used for testing in the given zone.

Orientation Checks

Angularity, Perpendicularity and Parallelism

Orientation checks require at least a single datum for evaluation. The measurements of the datum are first aligned to the datum features and then the measurements of the check feature are evaluated relative to that alignment.

Perpendicularity and Parallelism are special cases of an angularity check where the angle is either 90 or 0 degrees respectively and are therefore evaluated in the same way. Angularity checks ensure that all the specified points lie at approximately a specified angle relative to the datum features. The tolerance zone can be defined as planar region or as the deviation between two axes.

Evaluation Process within SA:

- **Planar Evaluation.** The minimum distance between two parallel planes at the specified angle containing the measurements of the feature (Figure 27-42).

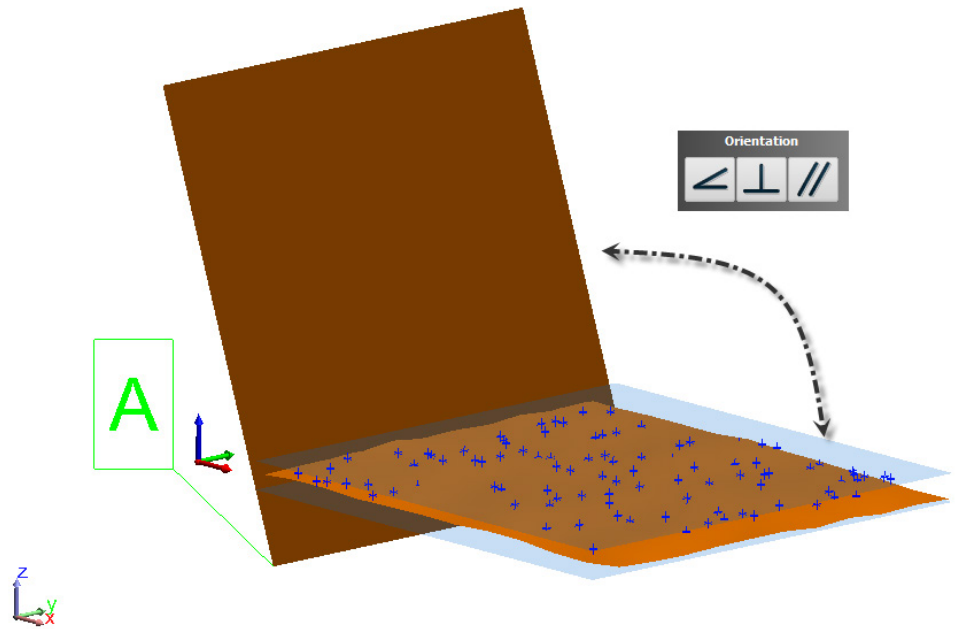


Figure 27-42. Planar Evaluation

- **Axial Evaluation.** The minimum diameter of a cylindrical zone at the specified angle containing the measurements of the feature (Figure 27-43).

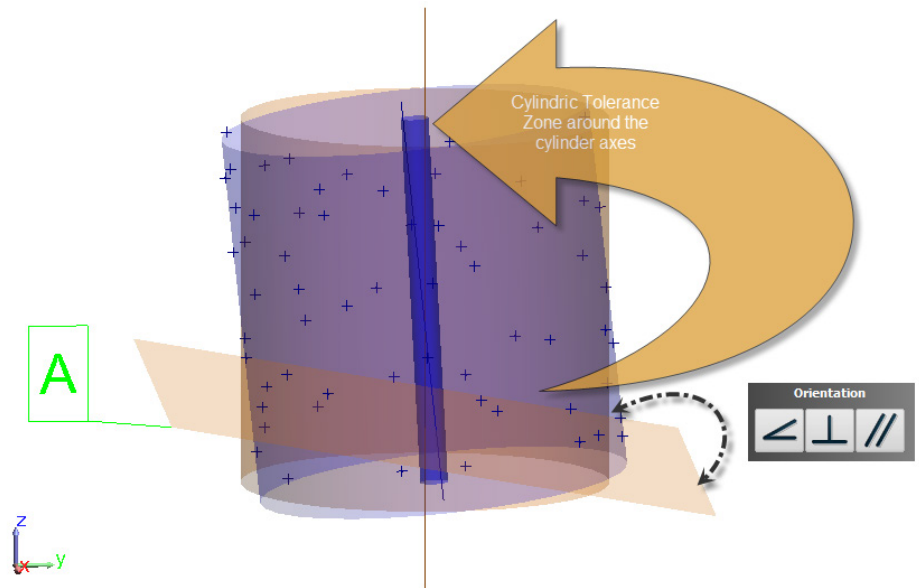


Figure 27-43. Axial Evaluation

Location Checks

Location checks are the heart of GD&T analysis and provide the greatest flexibility and most complexity. They can be evaluated excluding

datums entirely or by including as many datums as are needed to fully constrain the model. Only a simple example of how the evaluation process could be performed is presented.

Profile of a Surface

Profile checks are quite versatile. They provide the ability to evaluate size, form, orientation and location and the tolerance boundary can either be symmetrical or asymmetrical as needed, using any number of datum references. Measurements are fit to the features to minimize the point deviations as allowed by the any degrees of freedom not locked down by the datums and the max deviations are then compared to the tolerance boundary to evaluate the check (Figure 27-44).

Evaluation Process within SA:

1. Feature measurements are allowed to rotate and shift, depending on the datum constraints to minimize the point deviations relative to the specified surface.
2. Evaluation is made based on the greatest point deviation with respect to nominal surface and tolerance boundary.

Reporting

Surface profile, by default, establishes a tolerance zone which is equally disposed inside and outside the surface. The check is a measurement of the thickness required to enclose the data while still being equally disposed. Which is why its double the largest error.

Measured Deviation is calculated by first finding the point furthest away from the boundary wall (the boundary being the extents of of the tolerance zone which could be asymmetric with respect to the surface), then computing how far to the boundary as a positive number [A].

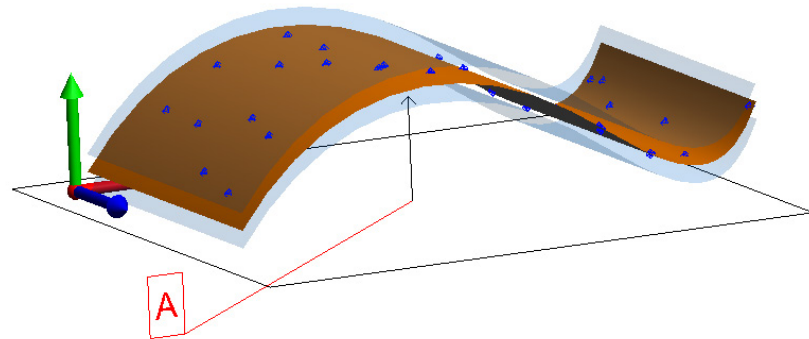
When a check passes, the Measured Deviation is reported as:

$$[\text{Total Tolerance Value}] - [A * 2].$$

When a check Fails, the Measured Deviation is reported as:

$$[\text{Total Tolerance Value}] + [A * 2].$$

Distance Out of Tolerance is reported as: $[A * 2]$

Figure 27-44. Profile of a Surface

In order to specify an asymmetric tolerance zone for a profile check, open the annotation properties and edit the two tolerance fields.

- **Tolerance.** Defines the total tolerance band used by the check from the inner extent to the outer.
- **OuterTolerance.** Defines the location of the outer tolerance on the positive or outer surface. This value controls the relative placement of the full tolerance with respect to the surface. For a symmetric tolerance this should be exactly 1/2 the tolerance value, and for a tolerance exclusively on the outside of the surface this value should equal the total tolerance.

Composite Surface Profile

Composite Surface Profile Checks provide a means to evaluate the overall position of a set of features and combine that with a tighter evaluation of the relative position of those features with respect to a less restricted set of datum constraints all within a single check.

Evaluation Process within SA:

What makes Composite Surface Profile checks unique from two separate position checks is that composite checks provide a greater degree of freedom in the lower segment check than would be allowed through a secondary check. Take the following check for example (Figure 27-45):

Figure 27-45. Example Composite Surface Profile

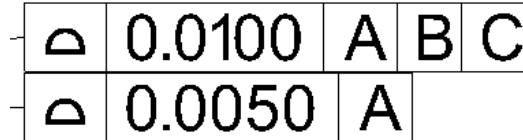
	0.0100	A	B	C
	0.0050	A		

1. The part is first fit to Datum A, B, and C, which fully locks down all degrees of rotation and translation and the upper segment of the control frame is evaluated.
2. The pattern is then completely released to translate freely and

rotate within the degrees of freedom not locked down by datum A to fit the nominal pattern. The key here is that translation is completely unconstrained.

This is very different from two independent checks. In this case the upper segment is evaluated, and then in the second check is evaluated separately. In this case the difference is slight but the part is limited in its ability to move toward and away from datum A, where in a composite check that would not be the case (Figure 27-46).

Figure 27-46. Example Compound Check



The difference is more apparent with two datums in the lower segment. The part is both constrained in position to datum A and B. This restriction in translation and only allows the part to only slide in a straight line perpendicular to the datums to fit the data, while a composite check would only hold rotation for the lower tier evaluation.

Profile of a Line

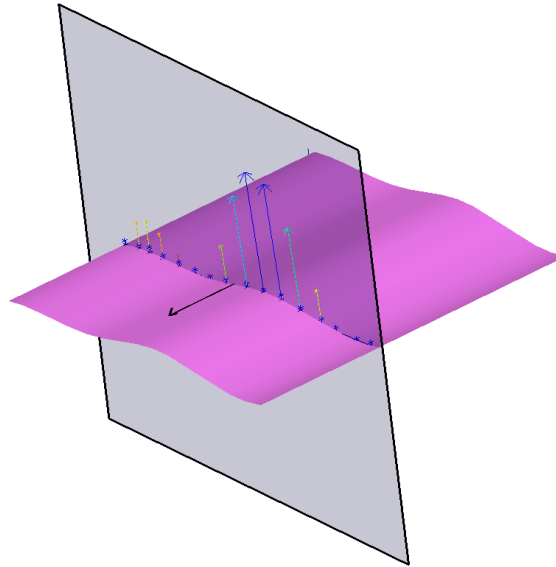
Line profile is much like surface profile except that it is a 2D check. The tolerance zone is normal to the surface profile of the feature at a cross-section. This can be a useful if you want to evaluate a single portion of a surface at a location or a set of parallel location evaluated in cross sections (Figure 27-47).

- You must associate a direction object (SA object) with the line profile annotation. This direction object defines the direction along which the data is divided into cross-sections for analysis.

Evaluation Process within SA:

1. The direction vector for analysis is established from the direction object chosen.
2. Cross sections are established along this direction vector and the data is divided into groups.
3. A plane is fit to the first point in each cross section group (using the actual feature direction vector) and then the points are projected onto that plane.
4. Measurements are allowed to rotate based on the datum constraints to minimize the point deviations.
5. The extent of the max point deviation doubled to describe the fit deviation zone and compared to the tolerance to evaluate the check.

Figure 27-47. Profile of a Line



Position (True Position)

Position checks provide the ability to evaluate size, orientation and location. The number of datums specified, the number of measurements and the type of feature determines the way in which the evaluation is conducted. Here we will just consider a basic hole position check. For cylinder evaluation options see “[Datums and Feature Checks](#)” on page 627.

- The degrees of freedom controlled by the datums determine if it checks a point location or an axis. In evaluating the True Position of a feature relative to a single datum axis you are essentially checking the coaxiality of that feature relative to that datum.
- Note that when specifying a MMC or LMC position tolerance a second annotation must also be included on the same feature to specify the nominal dimension. If you do not an error “No size tolerances defined for feature” will be seen. This second annotation will be automatically recognized and included in the evaluation ([Figure 27-48](#)).

Evaluation Process within SA:

1. The feature measurements are used to establish the center point or axis of the measured feature.
2. The maximum point or axial deviation is minimized after aligning to the datums and using any degrees of freedom not locked down by the datums.
3. The maximum deviation is doubled to define the measured deviation, any bonus tolerances are applied, and compared to the tolerance zone for evaluation.

Reporting

Measured Deviation is calculated by first finding the targets center or axis point deviation from the nominal feature and doubling it to define the tolerance band required for the part to pass.

Distance Out of Tolerance is reported as the radial distance from the maximum deviation to the tolerance boundary. Or the distance a drill bit would have to move for the part to pass when checking the placement of a hole.

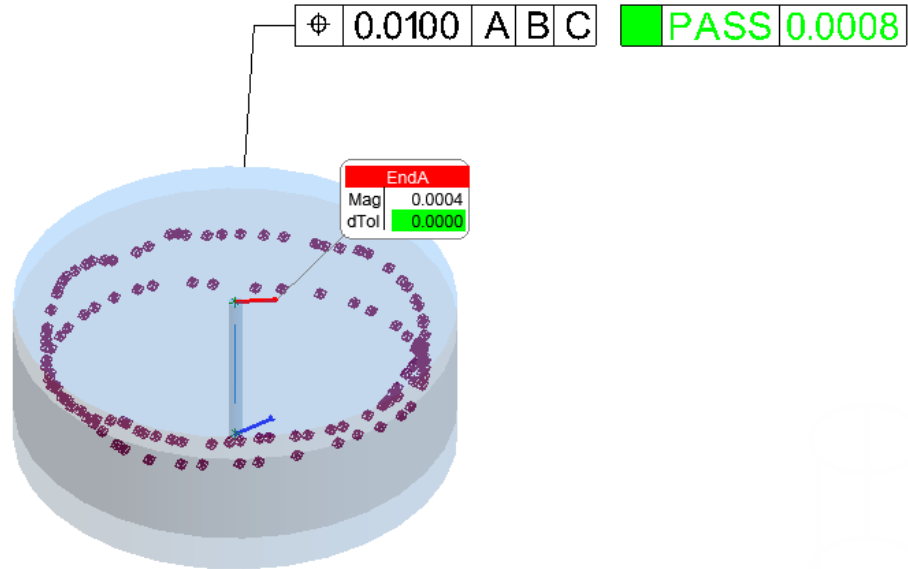


Figure 27-48. Basic Position Check

Tolerance Definitions Zones

Several Tolerance Zones options can be set for position evaluations on different features(Figure 27-49).

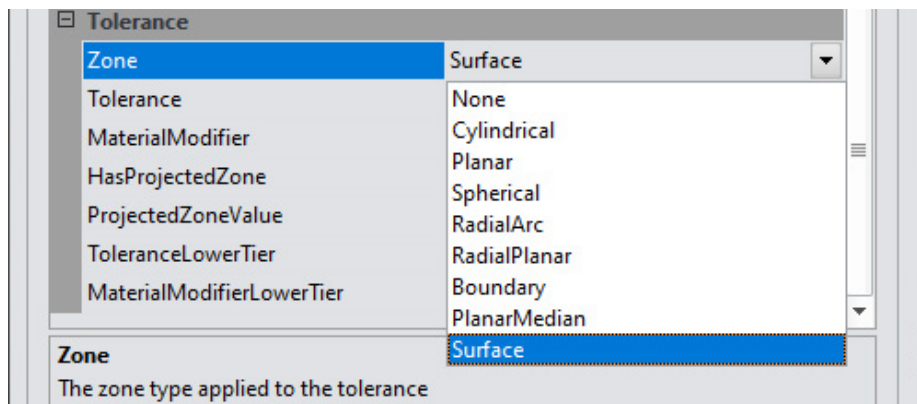


Figure 27-49. Tolerance Zone Options for a Position Annotation

- **None.** Which is a default setting that will apply either a cylindrical, planar or spherical tolerance zone based on the selected geometry type.
- **Cylindrical.** Tolerance zone is a cylinder sized by the tolerance value centered around the nominal cylinder axis. The axis of

Note: “Planar” Tolerance Zones require a reference plane selection. Adding a position check using “None” as the tolerance zone will use the selected Planar face by default without additional selection.

the actual cylinder must fall inside this tolerance zone.

- **Planar.** Two parallel planes spaced apart by the tolerance value centered on the nominal cylinder axis are used to define the tolerance zone. This allows the actual to slide back and forth between the planes. Note that this setting requires a fully constrained condition or it will rotate to a perfect result.
- **Spherical.** Currently only supported for spheres.

The geometry type has a direct influence on how a position check is evaluated. Cylinders can be evaluated as lines, circles, or full cylinders based on a user select (see “[Cylinder Evaluation Methods.](#)” on page 631). The extent of the evaluation is established by the extent of the measurements unless otherwise specified.

The offsets on points used in a position check on a line are ignored. The expectation is that measurements are on the axis.

Composite True Position

Composite Position Checks provide a means to evaluate the position of a pattern of features and combine that with a tighter evaluation of the position of each feature relative to each other feature within a single check. So for example, this allows you to check if a bolt hole pattern matches the correct location on the part at a lower tolerance while also verifying that the pattern is correct at a higher tolerance.

Evaluation Process within SA:

1. The upper tier is evaluated following the position check guidelines above.
2. The feature is then released to translate freely and rotate within the degrees of freedom not locked down by datums in the second tier and evaluated again.
3. The check is passed only if both evaluations pass.

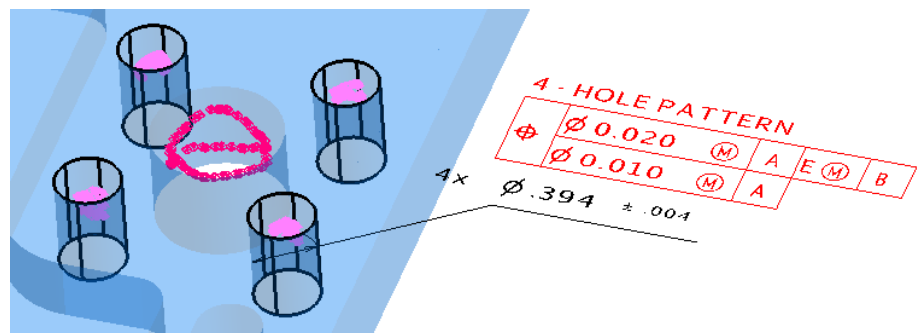


Figure 27-50. Composite True Position Check

Circular Runout*

Circular runout is a cross-section check that is used to evaluate features relative to a datum axis. With surfaces constructed around a datum axis it establishes a width variation normal to that surface evaluating both circularity and coaxiality around the datum axis.

Evaluation Process within SA:

1. The actual feature type is first established (circle or cylinder) based on the nominal type (feature to which the annotation is attached), and this feature type is fit to the entire data set to establish the direction vector
2. Cross sections are established along this direction vector and the data is divided into groups based upon their position along this vector.
3. Measurements are allowed to rotate based on the datum constraints to minimize the point deviations for all cross sections.
4. The extent of the max point deviation relative to the surface normal is doubled to describe the fit deviation zone for each cross-section.
5. The check returns the worst of the cross section results.

Total Runout

Total runout is used to evaluate surfaces constructed around a datum axis it establishes a width variation normal to that surface. The evaluation zone is continuous providing a cumulative evaluation of deviations in circularity, straightness, coaxiality, taper, and profile relative to this axis.

Evaluation Process within SA:

1. Measurements of the datum are fit to the datum feature to establish a nominal axis.
2. These features are translated along the nominal axis as allowed by any additional datum constraints to minimize the maximum and minimum point deviations.
3. The extent of the maximum point deviation relative to the surface normal is then doubled to define the diametric measured deviation and compared to the tolerance zone to report the result.

Concentricity*

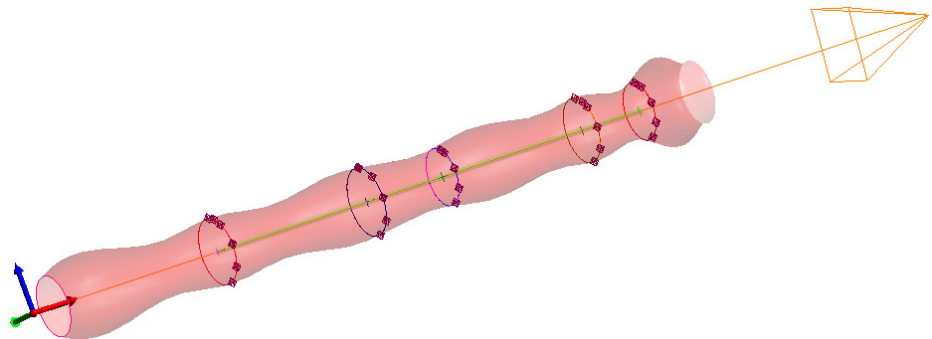
Concentricity requires a single datum to be defined. The measurements of the datum are first aligned to this datum and then the measurements of the check feature are evaluated relative to this datum by establishing a median point for each cross section and comparing

its position relative to the axis of the datum feature.

Evaluation Process within SA:

1. The measurements of the datum are aligned to the datum feature which orients the nominal feature and the measurements of it.
2. The actual feature type is established (circle or cylinder) based on the nominal type (feature to which the annotation is attached) and fit to the entire data set to establish the direction vector for this measured feature.
3. Cross sections are established along this direction vector and the data is divided into groups based upon the distance of each point along this direction vector and the cross section width definition.
4. The data in each cross section is evaluated independently. A circle is fit to the data in each group using the datum direction axis and the center or “medial” point established.
5. The distance of each median point from the datum axis is used for the evaluation.

Figure 27-51. Evaluating Concentricity



GD&T Dimensions

Dimensions operate independent of Datum alignment and report the dimensional properties of an object, such as diameter, or the distances or angle between two objects.

Tolerance Type

Tolerances on dimensions can be set in using the following types:

- **Exact.** Setting an Exact tolerance does not define a tolerance zone. Unless perfect, the results will always be out of range but can be used to clearly identify the distance out.
- **Limit.** Limit dimensions define the exact extents of the tolerance without any reference to expected value. A Limit dimension could be set as 9mm to 10mm for example.

- **Plus Minus.** A plus minus tolerance defines the upper and lower extents of a tolerance zone. This is with respect to the nominal feature's properties. No nominal value is entered because the nominal value is directly pulled from the nominal object to which the annotation has been attached. A tolerance such as +/-0.5mm is an example.
- **Plus Minus With Nominal.** Plus Minus With Nominal provides greater user control. A user defined nominal value can be entered such as 40 +/- 0.5mm.

GD&T Workflow

GD&T inspection within SA is broken down into two distinct processes:

- **Design.** Which is the upfront process of importing or construction annotations, defining tolerances and setting up datums and feature checks to scripting an inspection process.
- **Inspection.** Which is basically the process of associating the correct points with the correct datums and feature checks for analysis. Reporting is then automatic as part of this inspect.

SA provides 3 levels of inspection design:

1. **Guided Inspection.** Which is a fully featured option for laying out guide points on features, showing where to measure, controlling the view, and even instrument tooling and measurement profile.
2. **Inspection Bar Trapping and Direct Association.** In addition to GR-Features GD&T Datums and Feature Checks are included in the Inspection bar which can be used to quickly script the inspection sequence.
3. **Feature Inspection Auto Filter.** SA provides an intelligent proximity filter tool that can be used both with cloud data and points (with offsets) that will search the data and associate the correct measurements with the selected datums and feature checks.

Guided Inspection

SA's Guided Inspection provides an option for a GD&T inspection designer to layout exactly how and where to measure a part. It also provides a rehearse function that allows the designer to walk through the script in simulation to check the workflow. Once complete, the resulting script can then graphically walk a measurement technician, with little to no metrology knowledge, through the measurement

process. This guided inspection is broken into 3 pieces: Design, Rehearse and Inspect.

Building a Guided Inspection - Design

To build a guided GD&T Inspection an inspection designer should begin with a fully annotated model. This annotated model can either be imported from a CAD package or built within SA. SA objects should be converted to bounded surfaces before beginning the inspection design to allow nominal guild points to be placed on the nominal features. The design process should be conducted as follows:

1. Begin by right-clicking on the annotations in the tree and building feature checks. The majority of the inspection design including view orientations and nominal points are saved with the datums and feature checks.
2. Right-clicking in the tree and select **Inspection > Design**. There are three location where you can find this option. You can right-click on the collection containing the GD&T datums and feature checks to work through the design of all datum and feature checks at once, or you select the datums or feature checks only if you only want to work on designing the portion of the GD&T inspection.
3. Once started the GD&T inspection designer walks through designing the inspection for each GD&T feature in order just as a user would measure them. For each feature the following designations can be made:
 - **Save View for Inspection.** If checked the current view will be saved with the datum or feature check and the view will be set to this view when the **Inspection > Rehearse** or **Inspection > Inspect** routines are run.
 - **Create Nominal Points.** Nominal point measurement locations can be layed out on the model by graphically clicking on the model in the graphic view. The click selection order will be duplicated in the guided inspection. These point locations are saved as a point group and are saved with the Group Name and Point Name specified in the dialog. If you wish to remove the associated point locations you can use the "Clear NomPts" button at the bottom of the dialog.
 - **Measurement Profile Name.** The measurement profile for the instrument being used for the inspection can be set through the inspection designer as well. When the inspector is prompted to measure this feature the measurement profile specified here will be sent to the active instrument.
 - **Enable Inspection Automatic Measurement.** When enabled

this option will automatically begin the specified measurement mode on the active instrument.

- **Inspection Design - Item # of #.** This set of toggle buttons provides a convenient way of progression the included inspection features. The total number of features included in the item list depend on the selection method, whether you selected to design only the datums, feature checks, or included the entire inspection in the design by selecting **Inspection > Design** for the entire collection. The settings for each inspection feature are saved automatically when you exit the current feature.

Guided Inspection - Rehearse

Once a guided GD&T inspection has been built the designer can then use the **Inspection>Rehearse** routine to simulate the measurement process. This option allows the designer to walk through the inspection process in simulation using mouse clicks for measurements to simulate the complete GD&T inspection routine built through the **Inspection>Design** process.

1. To begin the inspection right-click on either the collection, the datums or the feature check categories and select **Inspection > Rehearse** depending on the number of features you wish to include in the rehearsal. When you do so the following dialog will be displayed with the following options (Figure 27-52):

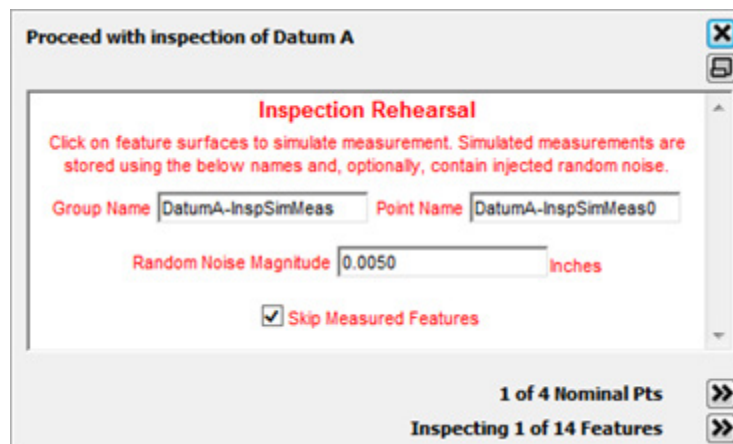



Figure 27-52. Inspection > Rehearse dialog.

- **Group Name/Point Name.** Specifies the group and target names for the click points (simulated measurements) to be saved.
- **Random Noise Magnitude.** This entry allows you to add random noise to the data measurements to simulate a more realistic result. The recorded point location will be randomly generated within the specified radius of the point click point on the surface.
- **Skip Measured Features.** When checked this option allows

the feature progression buttons to skip features that already have measurements associated and progress to features that do not. This is helpful when all but a few skipped features have been measured and you want to automatically progress to those particular features.

- **Number of Nominal Points.** These progress buttons provide the number of design measurements expected for the features and allows the user to skip through the measurement points to record measurements for only the points desired. It also allows you to backtrack and add additional measurements at a particular guide location.
 - **Inspection Design - Item # of #.** This set of toggle buttons provides a convenient way of progression the included inspection features. The total number of features included in the item list depend on the selection method, whether you selected to design only the datums, feature checks, or included the entire inspection in the design by selection Inspection>Design for the entire collection. The settings for each inspection feature are saved automatically when you exit the current feature.
2. Simply click on the surface where the bulls eyes is displayed to simulate a measurement of that location. The bulls eyes will automatically increment through the saved measurement locations. When all measurements are taken for a feature the dialog will automatically increment to the next feature in the list just as the inspection was designed.

Guided Inspection - Inspect

 **Tip:** Be sure to delete any measurements created in the Rehearse process before beginning the Inspect process.

Once a guided GD&T inspection has been built through the Inspection>Design process, a measurement technician can then right-click on the collection and select Inspection>Inspect to conduct the inspection of a part. The guided inspection will walk the operator through the inspection laid out by the designer and allow them to measure the part as specified. As measurements are taken they will be associated with the appropriate feature in the tree and an analysis will be carried out automatically. For an operator it can be as easy as measuring the point on the part designated by the bulls-eyes laid out on the nominal CAD part. The process flow is as follows:

1. To begin the inspection right-click on either the collection, the datums or the feature check categories and select Inspection> Inspect depending on the number of features you wish to include in the inspection. When you do so the inspection will begin and the following will occur: the view will be set to the saved view for the feature; the measurement profile will be set on the instrument specified; or the first inspection point will be displayed on the nominal part.

2. A dialog box (Figure 27-53) will be displayed with the following options and active trapping will begin for that feature:

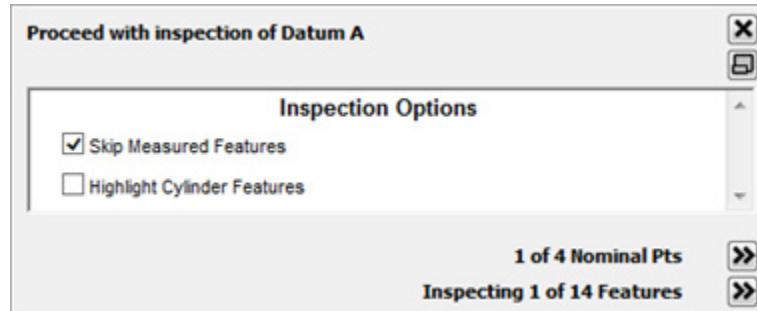


Figure 27-53. Inspection > Inspect dialog.

- **Skip Measured Features.** When checked, this option allows the feature progression buttons to skip features that already have measurements associated and progress to features that do not. This is helpful when all but a few skipped features have been measured and you want to automatically progress to those particular features.
 - **Highlight Cylinder Features.** This check box triggers an additional cylinder to be graphically drawn on the CAD model to clearly depict which cylindrical surface to measure in addition to highlighting the specific CAD faces.
 - **# of # Nominal Points.** These progress buttons are displayed only when saved measurement locations are included in the inspection. They provide the number of design measurements expected for the features and allows the user to skip through the measurement points to record measurements for only the points desired. It also allows you to backtrack and add additional measurements at a particular guide location.
 - **Inspecting # of # Features.** This set of toggle buttons provides a convenient way of progression the included inspection features. The total number of features included in the item list depend on the selection method, whether you selected to design only the datums, feature checks, or included the entire inspection in the design by selection inspection>design for the entire collection. The settings for each inspection feature are saved automatically when you exit the current feature.
3. Simply measure the actual part where indicated by the bulls eye is displayed on the nominal part within SA. The bulls eyes will automatically increment through the saved measurement locations. When no nominal measurement locations are save with the feature it is assumed that the measurement technician will measure the appropriate number of on the designated surfaces.
 4. When all measurements are taken for a feature the dialog will

automatically increment to the next feature in the list just as the inspection was designed. If nominal measurement locations were not included in the inspection design then the feature should be incremented manually when sufficient measurements are taken.

Inspection Bar Trapping and Direct Association

In addition to classical Design and Inspection routines intended to separate the GD&T inspection design phase from the inspection or operation phase, SA provides a number of tools available to more advanced GD&T users performing the inspection of a part.

Trapping Measurements from a Live Instrument

Measurements can be captured or “Trapped” directly from a live instrument while measuring without either building or utilizing the guided inspection process. For an advanced user, directly measuring points on a part and having those measurements automatically associated to a particular feature can be the fast way to conduct an analysis. Trapping can be started for any datum or feature check in either of two ways:

Trapping through the Instrument

1. Connect to your instrument.
2. Right-click on the particular Datum or Feature Check you wish to trap.
3. When finished measuring, right-click on the feature a second time and select **Stop Trapping Measurements**.

Trapping through the Inspection Tab in the Toolkit

1. Connect to your instrument.
2. Double-click on the particular Datum or Feature Check you wish to trap measurements to in the *Inspection* tab.
3. When finished measuring, press the stop button (INSERT ICON) or press the progress button (INSERT ICON) to move to the next or previous item in the *Inspection* tab list.

Either of these operations will begin trapping to the feature. If you have more than one instrument you will be prompted to choose which instrument you wish to trap measurements from. While trapping is active the feature will become highlighted in both the Inspection tab and in the tree as well as being displayed with the point count in the Heads Up Display and will remain highlighted until trapping is stopped.

Directly Associating Data with a Feature

As soon as Annotations are available, Feature Checks and Datums can be built and data can be associated with those features. A user could potentially measure all the features on the part independently of the GD&T analysis process and then come back after the fact and associate the appropriate measurements with the corresponding Datums and Feature Checks. To associate data with a datum or a feature check just right-click on the feature in the tree and select Associate Points or Associate Clouds. This will clear the list of currently associated points and or point clouds associated with the feature, if any are currently associated, and associate the selected points or clouds with the feature. Once features have associated data they can be evaluated all at once by right-clicking on the feature checks and selecting Evaluate all Checks.

Show Only Active Feature Check or Datum

A right-click option exists in the Feature Checks and Datums category's to enable *Show Only Active Feature check* or *Show Only Active Datum*. During a measurement process this option can be quite helpful in that it limits the displayed annotations to only those features directly linked to the feature in question (Figure 27-54).

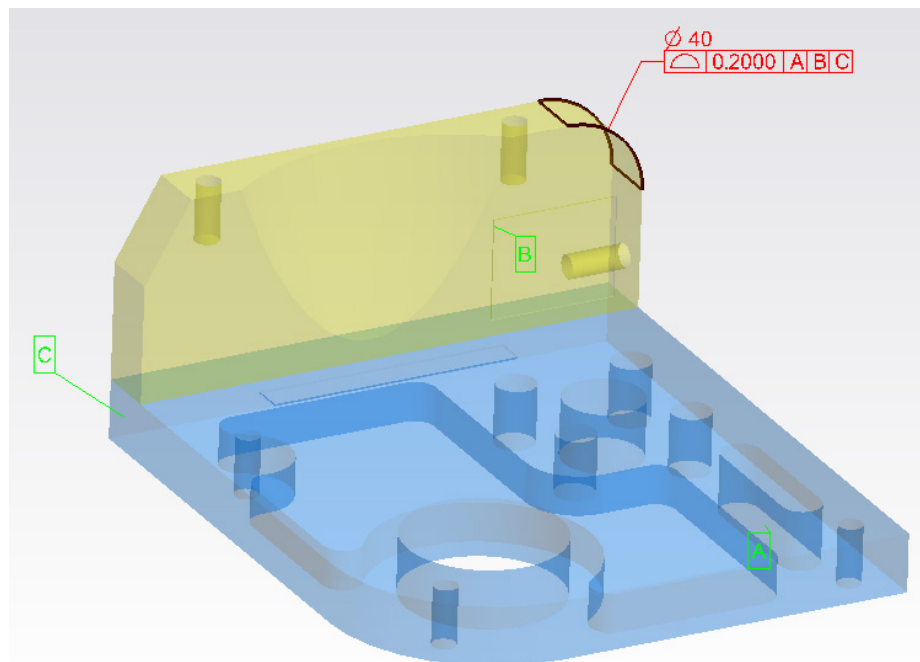


Figure 27-54. Show Only Active Feature Check Option Enabled

Remeasure Failed Checks Only

Many users asked for an easy way to go back and remeasure only the checks that failed on a prior inspection. This option, available through the **Feature Checks** category right-click menu provides the ability to do

just that. Selecting this option does the following:

1. Users the “Lock Out Trapping” option to hide passing feature checks from the Inspection panel.
2. Clears the data from the existing feature checks
3. Starts actively trapping to the first check in the list.

Feature Inspection Auto Filter

If you have measured your part independently of any type of Guided Inspection or Measurement Trapping but you have *pre-aligned* to a CAD model, it is still quite simple to associate points to the necessary checks. To do so, use the autofilter function to assign points to the correct checks. The steps in this process are as follows:

1. Double check that datums feature checks are built from your annotations and ready to receive points. If not right-click on the annotation category and select **Make Feature Checks**.
2. Navigate to **GD&T > Feature Inspection Auto Filter All Checks**. Select individual points, point groups, or even point clouds, and choose a proximity threshold used to include points with particular checks. Then simply press OK. For more details (see “Auto Filter to Faces” on page 250).

If you need to pick up additional point measurement taken along the axis of a cylinder, these can be obtained with a second run using the *Additional True Position Point Detection* check box (Figure 27-55). Also if you want to populate only a selection of checks you can use the function **GD&T > Feature Inspection Auto Filter Selected Checks**. This function works the same only it will ask you to choose which checks to consider.

3. Once this is done then navigate to **GD&T > Evaluate all Feature Checks** to complete the GD&T analysis.

This is a particularly useful approach to GD&T analysis when using a scanner. If you initially locate a scanner and then fully scan the part you are evaluating you can then filter the scan data to the CAD Faces associate the individual scans to the appropriate Features and in two clicks of a mouse button go from a single unfiltered cloud to a completed GD&T analysis ready for reporting.

Additional True Position Point Detection.

The GD&T Feature Inspection Auto Filter works great for most applications except for True Position Checks using alternate cylinder evaluation modes. Additional Hole Measurement Modes may include:

- Measure a single point in the hole (axis)
- Measure 2 points on the axis

- Place a dowel pin in the hole and measure it, indirectly measuring the hole (either as a circle or a scan)
- Measure the hole scanning with a pin nest

This check box was added specifically to fill this need. This check box option converts the operation from a surface filter to a comparison to cylinders defined by those surfaces.

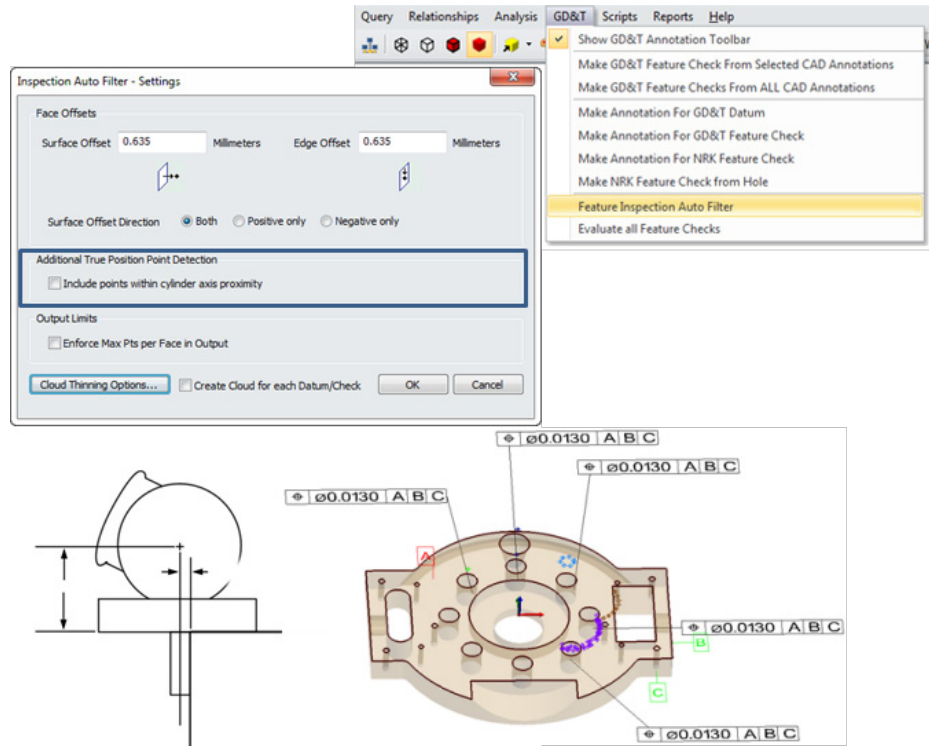
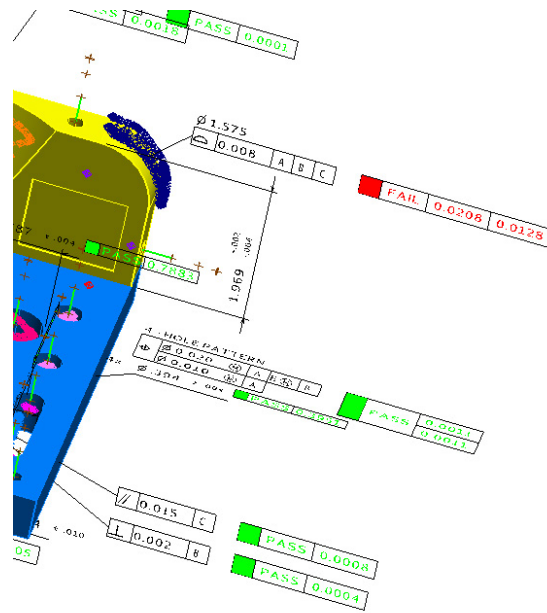


Figure 27-55. True Position Point Detection

Points are therefore compared to both the axis of the cylinder defining the pin or hole and the surface of that cylinder such that it is no longer bounded by the ends of the hole but can extend upward along its axis.

Figure 27-56. Evaluated feature checks.



GD&T Reporting

The report bar will provide the results of a GD&T Check and provides a preview of the resulting table that would be added to a report using the current reporting settings. Viewing a feature check in the report bar or adding them directly to a report also triggers the evaluation process, updating the results if needed.

By default a minimum amount of information is presented showing the Pass/Fail result as shown in [Figure 27-57](#).

Figure 27-57. Basic Check Result

\varnothing 0.0150 $\text{\textcircled{M}}$ ABC CHECK PASSED 0.0029

However, much more detailed information is available should it be needed by editing the Reporting Options for a check through the Right-click menu in the tree. These options include ([Figure 27-58](#)):

Reporting Options

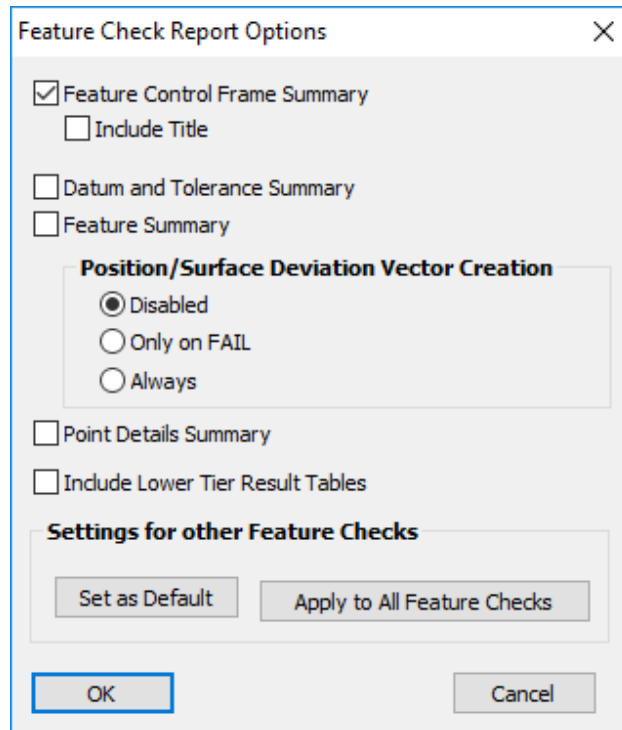


Figure 27-58. Report Options

- **Graphical Summary.** The default pass/fail information also included in the graphic view with the annotation.
- **Include Title in Graphical Summary.** Provides an option to include the check name along with the Graphical Summary (Figure 27-59).

Figure 27-59. Basic Check Title



- **Datum and Tolerance Summary Table.** Reports the essential datum fit results, tolerance and fit transform used as part of the check evaluation.
- **Feature Summary.** This table provides additional details on the specific features evaluated in the check. This includes specifics from the details table and bonus tolerance information.
- **Position/Surface Deviation Vector Creation.** With this option enabled a vector group will be built or updated each time the check is evaluated or as specified. This vector group inherits the check tolerances and graphically depicts the check results.
- **Details Table.** This table provides a full list of the point names, positions and offsets used in the valuation as well as the solved feature details for both the nominal and actual features.
- **Include Lower Tier Result Tables.** This option provides the ability to generate a second additional report tables describing

the results specific to the lower tier evaluation in a composite check.

- **Default Settings.** We provide the ability to set the desired level of detail for a particular check and apply those settings to other checks through these buttons.

Reporting Frame

A specific reporting frame can be set for any check. This is done per check through the right-click menu in the tree. Setting a reporting frame will not impact the check results but it will control the X,Y, Z values reported in the report tables. As with the Reporting options this frame designation can then be assigned to any other number of checks at once.

Datum and Tolerance Summary Table

The Summary Table provides the essential information necessary to understand the way a check was evaluated. This information includes three things: the evaluation result, the transform and the datum fit.

- **Measured Deviation.** Represents the true GD&T computed result used to determine if the check passes or fails. It represents the tolerance zone required for the part to pass and is compared to the tolerance designated in the annotation.
- **Distance Out of Tolerance.** This value is shown to help clarify the extent of deviation. It represents the distance you would have to move for the check to pass (or the radial distance out).
- **Fit Transform.** This transform defines the deviation between the current point locations in the job and the final datum alignment used to evaluate the check relative to the World Frame (see “GD&T Alignment” on page 634).
- **Datums Fit Results.** The Max, Average and Standard Deviation for the residual deviations are reported for each datum.

Feature Summary Table

The Feature Summary Table provides a slimmed down version of the feature information that has always been available in the Details Table, without all the details on the individual points. These tables will display the following additional information ([Figure 27-61](#)):

Feature Summary						
Meas Dev		0.0009		Dist Out		0.0000
Cylinder						
Eval: Full Cylinder Eval - As-Measured Extent (Auto)						
Nominal						
Pt	X	Y	Z			
Dir	-0.0000	-0.0000	-1.0000			
Radius				1.5748		
Length				0.5906		
Actual						
Pt	X	Y	Z			
Dir	-0.0002	0.0011	-1.0000			
Radius				1.8100		

Figure 27-60. Feature Summary

- Measured Deviation.** This is the Measured Deviation for the single feature. If only one feature is being inspected then it will match the check results. The check results will report the results for the worst feature in the list if more than one exists. The feature summary allows you to identify the results for each feature independently.
- Distance Out of Tolerance.** This is the Distance Out of Tolerance for the single feature. If only one feature is being inspected then it will match the check results. The check results will report the results for the worst feature in the list if more than one exists. The feature summary allows you to identify the results for each feature independently.
- Nominal.** This section includes the position and orientation and physical statistics of the nominal feature.
- Actual.** Displays the position and orientation and physical statistics of the measure feature.

For True Position Checks:

Position checks provide even greater detail including the following:

Feature Summary											
Tolerance Zone		0.0244		Meas Dev		0.0011		Dist Out			0.0000
Cylinder											
#Points	Nominal Diameter	Actual Diameter	Delta	Nominal Length	Actual Length	Delta					
78	0.3937	0.3941	0.0004	0.5906	0.1168	-0.4737					
Nominal			Actual			Deviation					
	X	Y	Z	X	Y	Z	Dx	Dy	Dz	Mag	
Solved Pt 0,0	1.9685	3.7402	0.5586	1.9680	3.7398	0.5586	-0.0005	-0.0003	0.0000	0.0006	
Solved Pt 0,1	1.9685	3.7402	0.6754	1.9683	3.7404	0.6754	-0.0003	0.0002	0.0000	0.0003	

Figure 27-61. Feature Summary Table

- Tolerance Zone.** This is the tolerance zone used to evaluate the specific feature. This is equal to the check tolerance plus the bonus tolerance used if any.
- # Points.** The number of points associated with the specific feature.
- Diametric Deviation.** If evaluating a cylindrical feature the

nominal, actual and delta diameter will be displayed.

- **Length Deviation.** If evaluating a cylindrical feature the nominal, actual and delta length will be displayed.
- **Solved Points.** For a cylinder the solved points represent the measured cylinder extents used for the evaluation.
- **Solved Point Deviations.** This section displays the delta distance from the Actual features solved points to the Nominal feature or axis. These values can be displayed graphically by creating a vector group to represent the solved point deviations using the selection in the Reporting Options (Figure 27-62).

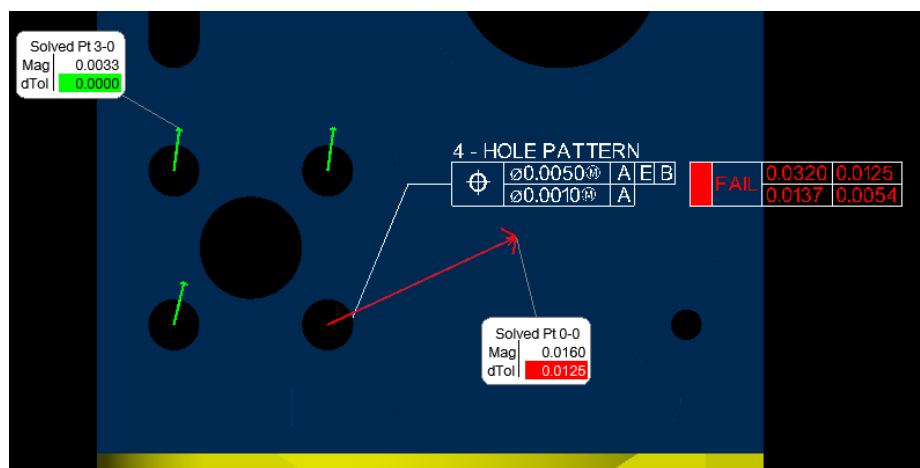


Figure 27-62. True Position Deviation Vectors

Details Table

In addition to the solved feature reporting the details table includes a specific list of each of the points included in the check and their properties.

- **Actual Data.** This list of points shows which points were used for the evaluation. It includes name, location and offsets as well as deviation information for surface evaluations.

Trouble shooting a Failed Check

ø1.575
 Δ 0.0080ABC CHECK FAILED 0.02080.0128

GD&T Surface Profile Check						
Geometrical Tolerance .34 (Reported in A:WORLD)						
Measured Deviation	0.0208	Distance Out of Tolerance	0.0128			
Datum Alignment Results						
Fit Transform	Tx	Ty	Tz	Rx	Ry	Rz
	-0.0248	0.1076	0.1225	1.3767	-0.3069	0.5387
Datums						
	Tolerance	Primary	Secondary	Tertiary	Outer Tolerance	
	0.008000	A	B	C	0.004000	
Fit Residuals Max		0.0003	0.0003	0.0007		
Fit Residuals Avg		0.0002	0.0002	0.0003		
Fit Residuals StdDev		0.0002	0.0002	0.0005		

Figure 27-63. Failed Check Results Summary Table.

When a check fails, it will report two values for troubleshooting purposes: the *Measured Deviation* and *Distance Out of Tolerance* (see Figure 27-63). The first number (0.0208) is the *Measured Deviation* or the size of the tolerance zone required for the part to pass. The second number (0.0128), is the *Distance out of Tolerance* or the necessary expansion of the tolerance zone in order for the part to pass. For the above profile check, the sum of the check tolerance plus the *Distance Out of Tolerance* will equal the *Measured Deviation* ($0.008+0.0128=0.0208$).

By convention, Position checks are reported differently. The *Distance out of Tolerance* is reported as the distance the drill would have to move for the check to pass, the sum of the check tolerance plus two times the *Distance out of Tolerance* will equal the *Measured Deviation* ($0.008+2*0.0039=0.0158$) (Figure 27-64).

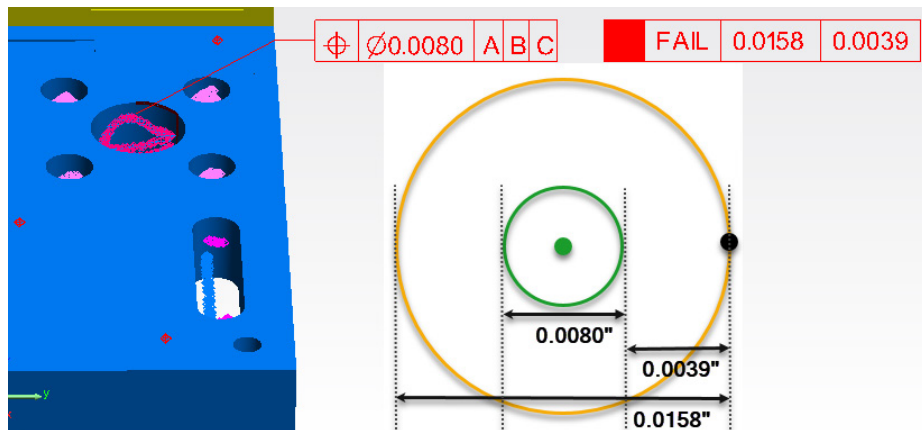


Figure 27-64. Reported values for a position check showing the Distance Out of Tolerance value.

If a check fails check the following:

1. First consider to what extent the check failed. If its large, check the datum alignment and transform. If the error is small check your fit residuals.
2. Within the Summary table is a "Fit Transform" section. Check to see if the transform is close to zero. If not, check the datum

settings to verify that the alignment is set up correctly. Its often helpful to align to the evaluated position to see what's causing the issue. For more information see “[GD&T Alignment](#)” on page 634.

3. The Summary table also tells you how well you data fit the datums. By checking to see if there was a bad fit to one of the datums you will see if the datums are contributing significantly to the error.

Adding Checks to a Report

Any individual feature check results can easily be added to an SA Report using the following methods:

- **Individual Drag / Drop.** Checks can be added through a drag and drop process like any other reportable object in the tree.
- **Collection Drag / Drop.** All the checks in a collection can be added to a report at one time by selecting the Feature Checks category in the tree and dragging it into a report.
- **Composite Quick Report.** Each individual check has a right-click menu option to be either included or excluded from a Composite Quick Report. To generate a quick report for all the feature checks not excluded at once, right-click the Feature Check category and selecting **Composite Quick Report > Generate**. This report will also include a screen capture if a view is defined for the check.

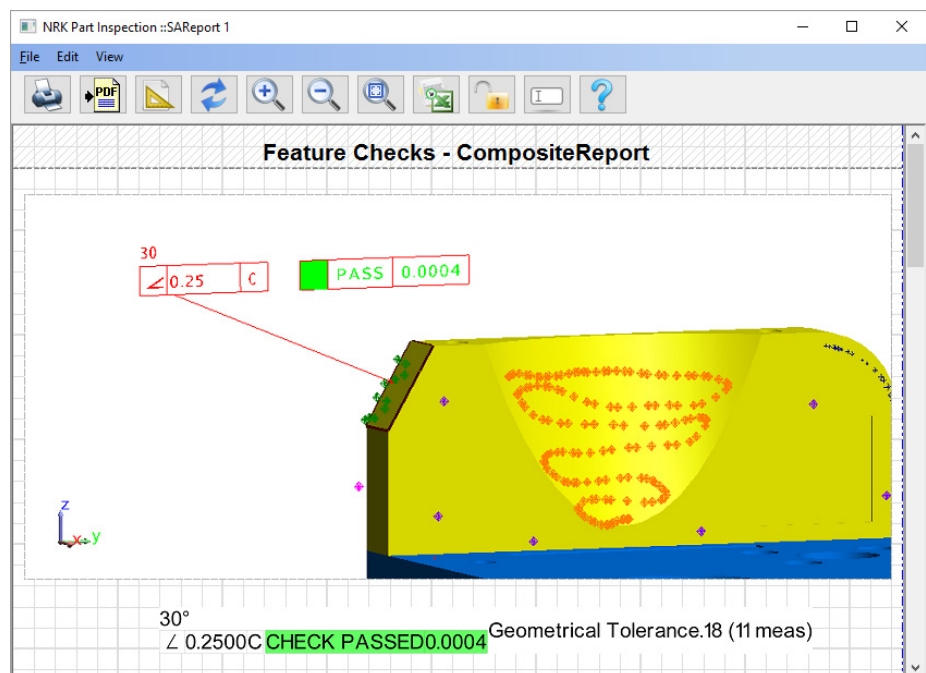


Figure 27-65. Composite Quick Report Example

Inspection Summary Table

In addition to individual check reporting a custom summary table can now be built including a selection of particular checks. To do so Right-click on the Feature Check category and select **Create Inspection Summary Table**. This option allows feature checks to be grouped as needed through an F2 selection prompt building a custom table with the selected checks.

The Custom Report Tables are designed to display only the necessary information on a single line (Figure 27-66).

GD&T Multi-gage Report									
	Check Name:	Tolerance	Special Characters	High Tol	Low Tol	Datums	Results	Measured Deviation	Distance Out of Tolerance
NRK Part Inspection									
	Geometrical Tolerance.18	0.2500				C	PASSED	0.0004	0.0000
	Geometrical Tolerance.19	0.5000				B	PASSED	0.0001	0.0000
	Geometrical Tolerance.28	0.0120				ABC	PASSED	0.0018	0.0000
	Geometrical Tolerance.34	0.0080				ABC	FAILED	0.0208	0.0128
	Geometrical Tolerance.38	0.0040					PASSED	0.0005	0.0000
	Geometrical Tolerance.39	0.0150				C	PASSED	0.0008	0.0000
	Geometrical Tolerance.40	0.0040				D	PASSED	0.0009	0.0000
	Geometrical Tolerance.41	0.0020				B	PASSED	0.0004	0.0000
	Geometrical Tolerance.43	0.0200				AB	PASSED	0.0010	0.0000
	Dimension.6	0.7870	∅	+0.0040	-0.0040		PASSED	0.7883	0.0000
	Dimension.8	0.3940		+0.0100	-0.0100				
	Dimension.9	1.9690		+0.0020	-0.0040				
	Dimension.14	3.1500		+0.0050	-0.0100				
	Geometrical Tolerance.42	0.0200				A	PASSED	0.0005	0.0000
	Dimension.17	1.1810		+0.0100	-0.0100		PASSED	1.1810	0.0000
	Geometrical Tolerance.35	0.0200	∅M			AE MB A	PASSED	0.0011	0.0000
		0.0100	∅M					0.0011	0.0000
	Dimension.13	1.5750	∅	+0.0080	-0.0040		PASSED	1.5754	0.0000
	Dimension.16	0.3940	4x∅	+0.0040	-0.0040		PASSED	0.3951	0.0000

Figure 27-66. Summary Table

When a custom table is added to a report it will become semi-static in that changes to the table will require a refresh of the report.

How to Create Easy GD&T Report Tables -

<https://www.youtube.com/watch?v=Y4tiD7CgZOI>

Dynamic Table Construction

GD&T checks can also be included in custom table added directly with in an SA report. These custom table can be configured to reference a particular check's attributes. This provides a means to build a fully customized table with results in any order desired. For more information (see "Tables" on page 698).

This chapter covers basic SA reporting capabilities, report generation and user option controls. It also covers graphical reporting tools such as Dimensions, Callouts and Charting. Additional details on configuring reporting of any particular object type or feature can be found in the related chapter on that object type. For example Relationships or GD&T each have their own reporting sections.

Building New Reports

Reporting begins with the items you wish to report. By selecting something in the SA TreeBar a preview of the report table for that item will be displayed in the “[Report Bar](#)” on page 682.

Reports can be generated in SA in several different ways depending on the complexity of the report you need. The primary methods are:

- **Quick Reports.** Quick reports are designed to be a one-click access point to the information you need for a collection, object category or selection of objects.
- **Direct SA Report Construction.** The “SA Reports” are dynamic, flexible and interactive parts of an SA job file. They allow a user to “drag and drop” items from the tree into the Report Editor where items can be arranged, formatted and edited to convey the user intent. If values/information change in the tree, the SA report will reflect those changes automatically.
- **SA Report Templates.** Report templates can be built in advance and formatted with custom headers/footers, custom tables and entry fields and fully populated for a specific reporting task. They are then updated as measurements are taken. More complexity can be added with interactive MP/SDK scripts to or-


ganize a report in advanced and its content prior to generating a report.

Reports can then be exported directly to PDF, an Excel sheet or added to a Word document as needed.

Report Bar

The Report Bar serves two primary purposes: it can serve as a quick reference to view greater details relating to the selected object either in the tree or in the graphics; and it provides a preview of the report table available to be added to a report.

Displaying the Report Bar

To display the Report Bar use the menu option **Report>Report Bar Visible** or press the Report bar button on the Reporting page of the ribbon menu 

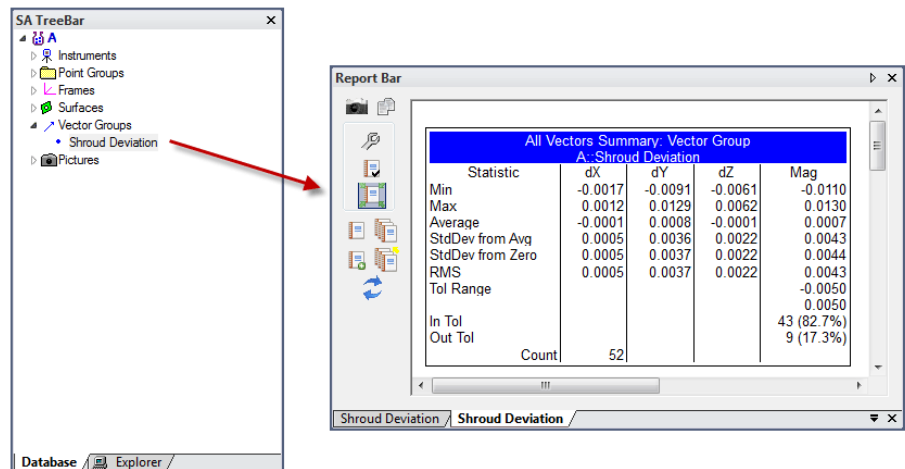
On the **Reporting** page of the **Users Options** there are additional custom configuration controls:

- **Show Report Bar Controls.** This check box controls the display of the Report Bar Toolbar.
- **Show Report Bar Grid.** This check box provides the ability to hide the grid background display from the report back.

Adding an Item

To add an item in the Report Bar simply select an item in the tree by single-clicking the item. Once selected it will display in the Report Bar. The currently selected item in the tree will always display in the current Report Bar Tab (**Figure 28-1**).

Figure 28-1. The item currently selected in the tree is displayed in the Report Bar.



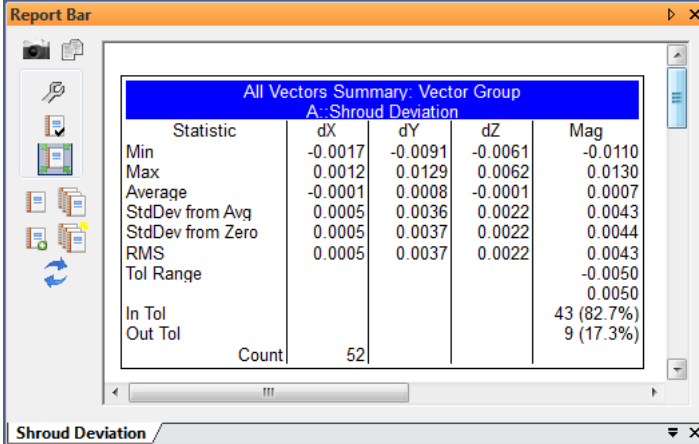
Building Reports Using the Report Bar

Report Bar is a great way to view object properties and values quickly and can be used to generate a simple report. Multiple items can be added to the Report Bar for continuous reviewing. Once finished, the Report Bar can be cleared or sent directly to an SA Report for final output.

Report Toolbar Controls

The report bar controls can be turned On/Off from the Reporting tab of the **Users Options** in the Report Bar Settings section.


Figure 28-2. A vector group in the Report Bar.




The screenshot shows a window titled 'Report Bar' with a toolbar on the left and a data table in the center. The table is titled 'All Vectors Summary: Vector Group' and 'A::Shroud Deviation'. It lists various statistics for dx, dY, dZ, and Mag, along with tolerance counts.


Statistic	dx	dY	dZ	Mag
Min	-0.0017	-0.0091	-0.0061	-0.0110
Max	0.0012	0.0129	0.0062	0.0130
Average	-0.0001	0.0008	-0.0001	0.0007
StdDev from Avg	0.0005	0.0036	0.0022	0.0043
StdDev from Zero	0.0005	0.0037	0.0022	0.0044
RMS	0.0005	0.0037	0.0022	0.0043
Tol Range				-0.0050 0.0050
In Tol				43 (82.7%)
Out Tol				9 (17.3%)
Count	52			

- **Capture Current View.** Creates a picture item in the tree of the current graphical view. The picture will display in the report bar under the current tab.
- **Clipboard Help.** Displays help information regarding methods of copying Report Bar data to the clipboard.
- **Object Properties.** Displays the current tab object properties.
- **Current Object Report Options.** Displays the report options for the current tab.
- **Include Current Object in Composite Report.** Sets the current object to be used in a Composite Report.
- **Quick Report.** Generates a Quick Report from the current tab.
- **Generate Quick Report from Tab Order.** Generates a Quick Report from all open Report Bar tabs in order from left to right.
- **New Report Tab.** Creates a new tab in the Report Bar.
- **Close All Tabs except Current.** Closes all Report Bar tabs except for the currently selected tab.

-  **Refresh Report.** Refreshes the Report Bar with current information.

Adding multiple items to the Report Bar requires multiple tabs. Add a new tab by using the  icon. The new tab will appear with the currently selected tree item. Select the next item from the tree to view.

Removing an Item

To remove a Report Bar item, click the “X” icon in the bottom right-hand corner of the Report Bar ([Figure 28-3](#)). All tabs, except the current tab, can be cleared by using the  icon.

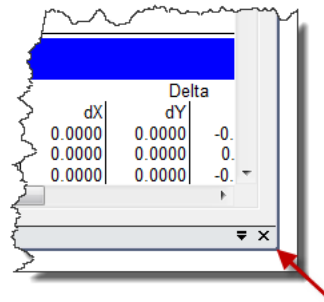




Figure 28-3. Removing a Report Bar item.

Generating a Report

If needed, an SA Report can be generated from one or all of the Report Bar tabs. Use  to generate a report of the current tab or  to generate a report from all Report Bar tabs.

Generating Reports

Quick Reports

A *Quick Report* is a fast and easy way to create an SA Report of either a single object or multiple objects, when using Composite Quick Report.

Creating a Quick Report

To create a Quick Report of a single item, right-click the item of interest in the tree and select **Generate Quick Report**. An SA Report of the object will be created and saved in the tree.

Creating a Composite Quick Report

A Composite Quick Report provides a fast way to create an SA Report of several items. A Composite Report can be generated for an entire job file from the menu **Reports>Composite Report - All Collections**, created by right-clicking on an individual collection in the tree or by selecting a single category in the tree. In each case you can select **Composite**

Quick Report to generate a report including the items in the selected category.

To exclude or include individual item from a Composite Quick Report, simply right-click the respective item in the tree and toggle the **Include in Composite Quick Report** check box.

SA Report Builder Utility

An interactive report designer utility is also available through **Reports > Report Builder Utility**. This tool makes it easier to build a report with the information needed in the order you desire. It provides a means to select items in a list format, either from the tree or by using an F2 selection method, and then organize those features as desired using drag and drop (Figure 28-4).

When the report is organize as desired the Insert Reported Items button can be used to generate a new report or add the items to the currently open SA report (replacing the existing content).

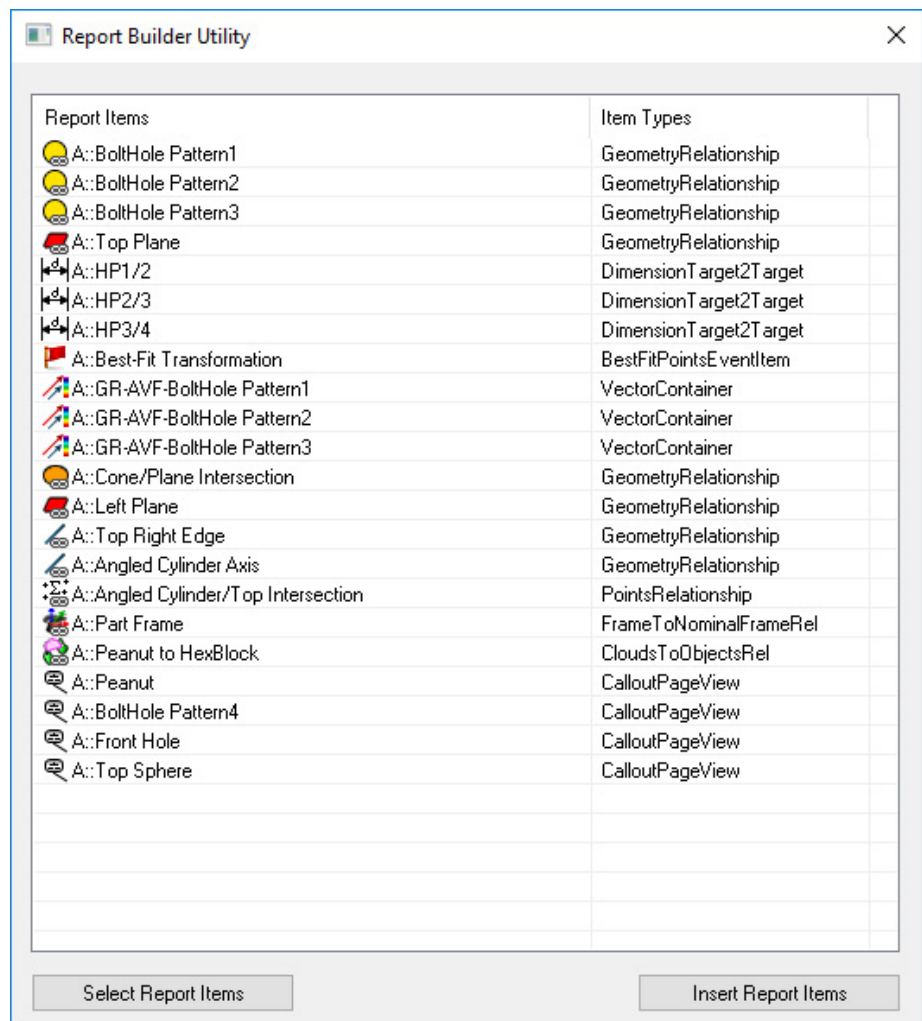


Figure 28-4. Report Builder Utility

Reporting User Settings

There are a number of User Option settings that can be applied to control the way in which reports are generated. These are found on the Reporting Tab (Figure 28-5).

The screenshot shows the 'User Options' dialog box with the 'Reporting' tab selected. The 'Reporting' sub-tab is active, displaying various configuration options:

- Logo Filename:** A text input field with a 'Select' button.
- Company Byline:** A text input field.
- Default Report Template:** A section with the text 'No Default Report Template Specified' and three buttons: 'New', 'Select', and 'Clear'.
- Quick Report Output:** A section with radio buttons for 'Excel Workbook', 'SA Report' (selected), and 'SA Doc'. To the right are three checkboxes: 'Quick Reports Include Current View Image' (checked), 'Quick Reports Include Report Name in Header' (checked), and 'Quick Reports Include Object Notes' (unchecked).
- Maximum Number of Rows Per Table:** A text input field containing '1000' and a 'Default Page Settings' button.
- Event Reporting:** A section with a checked checkbox 'Allow Automatic Event Generation', two text input fields for 'Maximum Logged Data Points' (100) and 'USMN Max Instruments Per Point' (10), and an unchecked checkbox 'Include Points Discarded from Fit in Geometry Fit Results'.
- Report Bar Settings:** A section with two checked checkboxes: 'Show Report Bar Controls' and 'Show Report Bar Grid'.
- Web Report:** A section with an unchecked checkbox 'MonoChrome HTML Reports'.

At the bottom of the dialog are buttons for 'OK', 'Cancel', 'Apply', and 'Help'.

Figure 28-5. User Option Settings for Reporting

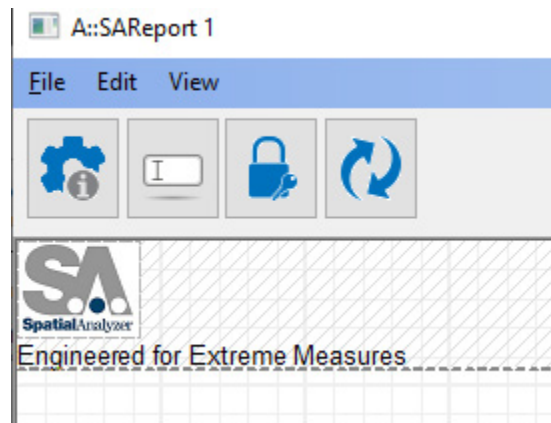
Automatic Header Addition

The report header can be pre-configured as follows:

- **Logo File.** The path to a company logo or any image you would like to add to the header of a newly generated report.
- **Company Byline.** A text entry that can be added on the top

left of the report header (Figure 28-6).

Figure 28-6. Example Default Template and Company Byline



Default Report Template

Any SA Report in a job file can be set as a report template. When this is done, the content from the template is copied as a beginning point for the new report. This can also be done by simply right-clicking on an SA Report and selecting **Default Template**.

Quick Report Output

Quick Reports can be generated as either Excel files, SA Reports or SA documents. You can also choose from the following options:

- **Quick Reports include Current View Image.** A screen capture of the graphics will be automatically added with each reported object.
- **Quick Reports Include Report Name in Header.** The optional setting adds the report name to the header of the newly created report. Disabling this option excludes a report title.
- **Quick Reports include Object Notes.** When enabled an additional table is created for each object in the report with any Object notes.

Default Page Settings

The settings for each SA report generated will include the page settings defined in this default section.

Event Reporting

The length and complexity of events can be controlled here.

Report bar Settings

Options to show or hide the report bar controls and grid are available.

Web Reports

Choose to limit web reports to Mono Chrome display.

SA Reports

Creating an SA Report

An empty SA Report can be created by going to **Reporting > Add SA Report**. This will add an empty report to the SA tree and automatically open the editor in preparation for items to be added to the report (Figure 28-7).

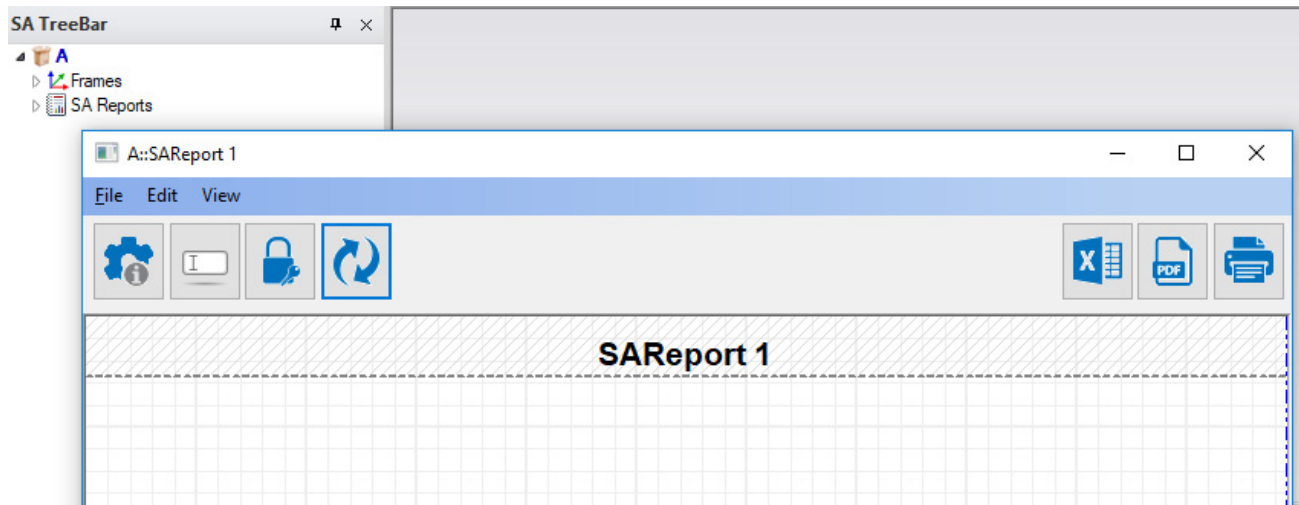


Figure 28-7. Empty SA Report

Adding Items to an SA Report

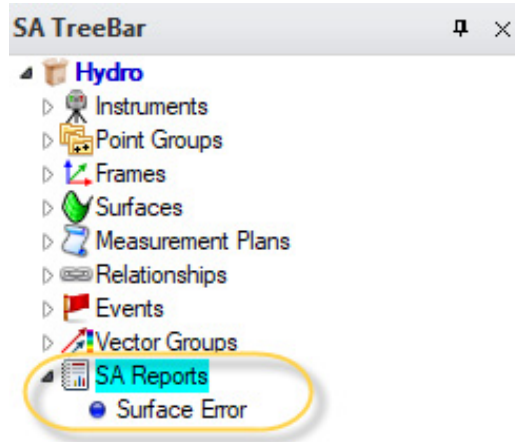
Simply L-Click and hold on an item in the tree and drag it over the report editor and release it to add that item to the SA Report in that location. Once inside the report, item report tables can be moved and positioned within the report by dragging them as well.

Building an SA Report

In addition to adding items to an SA Report one at a time, a report can be generated from a selection of the items you want to include in a report. There are several ways to do that which are discussed in detail later in the chapter:

- Composite Quick Report (see “[Creating a Composite Quick Report](#)” on page 684)
- Using the Report Builder Utility
- Creating a Report from Report Bar Tabs (see “[Building Reports Using the Report Bar](#)” on page 683)
- Using a Report Template ([Figure 28-8](#))

Figure 28-8. An SA report as it appears in the tree.



To open/edit an existing SA Report, expand the SA Reports category in the tree and right-click and select **View**. You can also double-click the respective SA Report to view it (Figure 28-8).





The Report Designer

The Report Designer is the canvas on which the user will create their report. The Report Designer supports standard functionality such as click and drag, re-sizing and Left/Right Clicking with context menus. The Report Designer also contains a toolbar (Figure 28-9) with useful reporting tools and options.



Report Toolbar

Figure 28-9. The Report Designer toolbar.



-  **Global Reporting Options/Page Setup.** Displays SA Report options such as Page Layout, Default Fonts and other options.
-  **Jump to Next.** This button provides a convenient way to navigate between entry fields for operators filling out reports.
-  **Lock Report.** When clicked the SA Report becomes static and will no longer receive updates from the tree.
-  **Lock Item Positions.** When enabled the items within an

SA report cannot be moved, preserving their location on the page. This is different from locking a single item in a report in that the tables continue to update.

- 
Report Refresh. Refreshes the SA Report with the current values from the tree. This only needed for special calculations such as report tags and callout images that are not fully dynamic or to refresh a section of a report that was locked and could not dynamically refresh.
- 
Connect to Excel. This button offers a direct connection to an new or existing Excel sheet and provides the ability to export specific tables or fields to specific locations within that file without the need for an MP.

Page Layout and Display

The report appearance can be formatted as use the page Global Reporting Options/Page Setup  button (Figure 28-10).

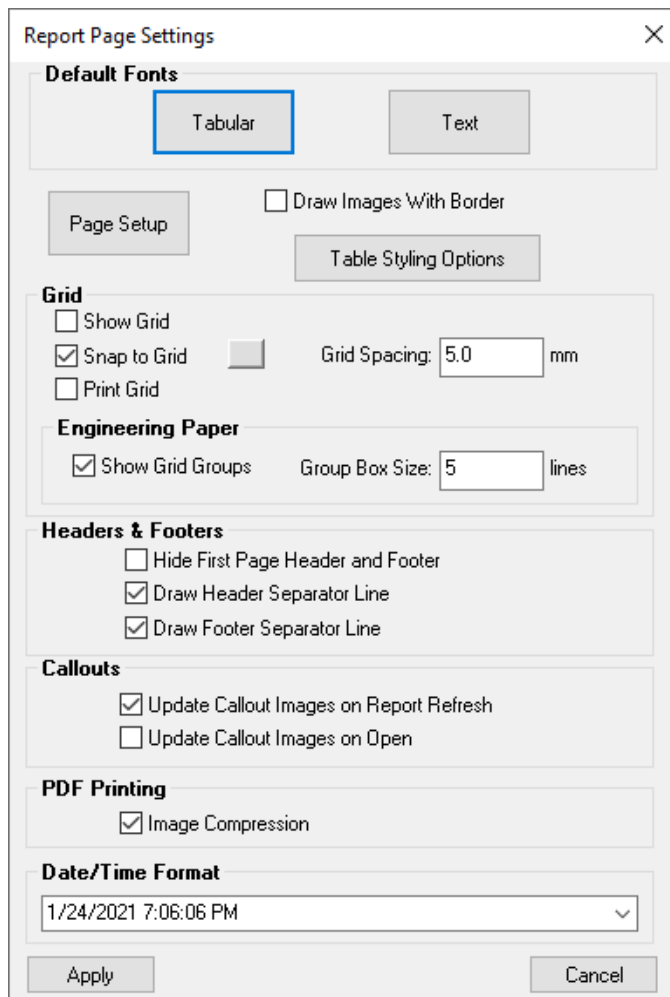


Figure 28-10. Report Page Settings

Default Fonts. The default fonts used in different entries can be set independently. These come in 3 separate categories:

- Tabular entries include object tables, fields and custom tables.
- Text entries which include basic text blocks added in a report.
- GD&T tables which use a custom GD&T font only.

Page Setup and Table Styling Options.

The **Page Setup** button provides control over Portrait or Landscape display as well as page size and margin controls.

The **Table Styling Option** button offers table header color control and grid display options for all object tables in the report.

Page Grid.

The engineering grid displayed by default on an SA Report is used for graphical layout purposes and will not print unless you enable the *Print Grid* option.

Report Header & Footers

The contents of the report header can be formatted to include a logo image and a company byline (Figure 28-11). To include this information navigate to **User Options** on the *Reporting* tab.

To edit the report header or footer content, double click within the crosshatched area to activated it. The size of the region can be controlled by selecting the separator line and dragging it up or down on the page. The content within the header or footer is duplicated on all pages within the SA Report.

The display of the separator lines and an option to hide the header and footer on the cover page is available.

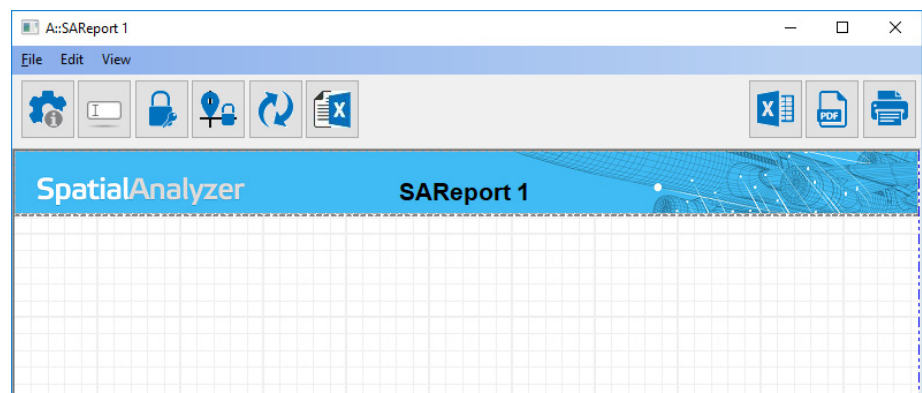


Figure 28-11. A report header in the Report Designer.

Export Options

The export options are available on the top right side of the SA report header for easy access.

- X

Export to Excel. Exports the current SA Report to Microsoft Excel.
- PDF
Save As PDF. Allows the current SA Report to be saved as a PDF document using the built in Spatial Analyzer PDF Printer. If for some reason this printer has not been installed correctly or cannot be accessed any standard PDF printer such as Adobe or Microsoft can also be accessed from the printer choice in the Print Current Report Options.
- Print
Print Current Report. Displays print dialog for the current SA Report.

Directly linking a report to an Excel Report

In addition to exporting an entire report to an excel file, individual item tables can be directly linked to Excel (*.xls). This provides the ability to specify the workbook address for report items, using either absolute sheet!col/row or named cell references (Figure 28-12).

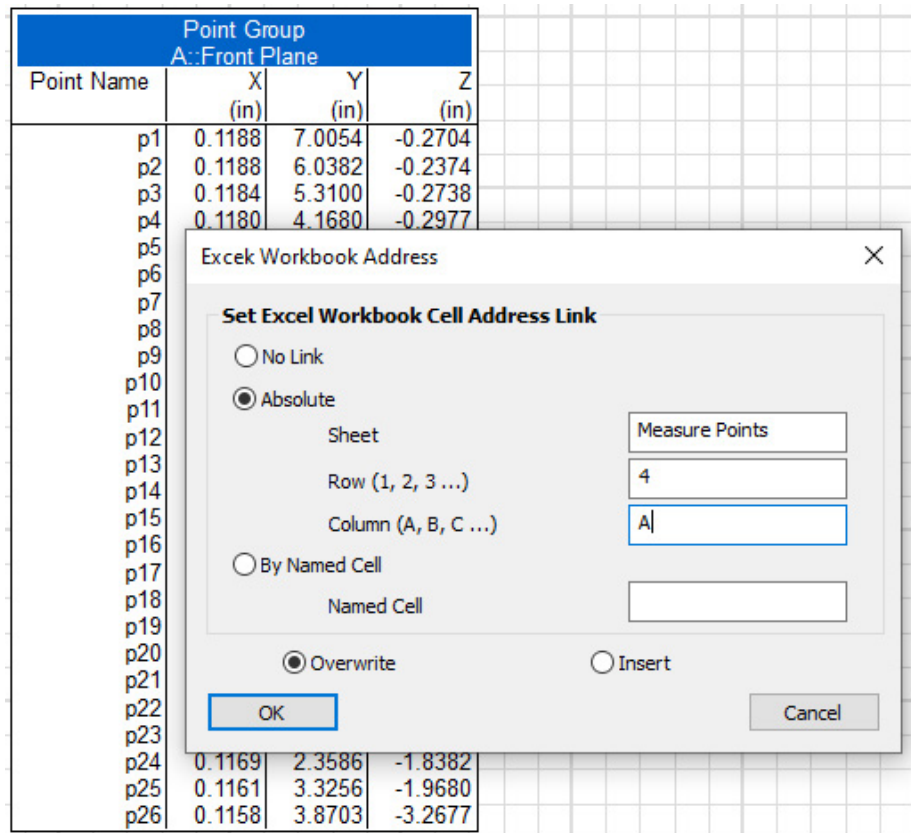
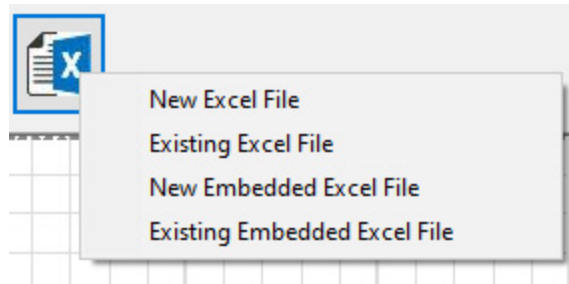


Figure 28-12. Controls for specifying the export location for a table linked to an Excel file.

To use this directed export capability an Excel file must first be selected using the **Connect to Excel** button (Figure 28-13).

Figure 28-13. Connect to Excel button used to link a report to an excel file.



Once a connection to an Excel file has been established a **Synchronize** button will be come visible. This global “Synchronize” operation will output all report items to the connected spreadsheet for which a valid workbook address has been established.

The connected Excel file remains open as a background process while a report is connected to it. The changes made will therefore, not update the associated Excel file until the connection has been terminated using the **Disconnect from Excel** button and the “Save” option performed.

Adding Elements to the body of a Report

Adding basic text blocks

Click any open space and start typing to add text. Or paste text from clipboard into any open space (Figure 28-14).

Figure 28-14. A report annotated with some text.

Vector Group A: Shroud Deviation				
Name	dx	dY	dZ	Mag
9	0.0000	0.0000	-0.0007	-0.0007
10	0.0000	0.0000	0.0011	0.0011
11	0.0000	0.0000	-0.0016	-0.0016
12	0.0000	0.0000	-0.0021	-0.0021
13	0.0000	0.0000	0.0004	0.0004
14	0.0000	-0.0001	-0.0005	-0.0005
15	0.0003	-0.0026	-0.0034	-0.0043
16	0.0012	-0.0091	-0.0061	-0.0110
17	-0.0002	0.0018	0.0005	0.0019
18	-0.0004	0.0030	0.0000	0.0030
19	0.0000	0.0000	-0.0000	-0.0000
20	0.0000	0.0000	-0.0010	-0.0010
21	0.0000	-0.0000	-0.0006	-0.0006
22	0.0000	0.0000	-0.0038	-0.0038
23	0.0000	0.0000	0.0016	0.0016

This was a bad point

Additional Object shapes can be added to the report as well to link tables or to highlight particular sections of a table. These objects include:

- **Box.** Which can be used to define any rectangular shape.
- **Ellipse.** Which can be used to define circular objects of any shape or size.

- **Line.** Which also can be set to display as an arrow.

Report Fields

Fields are text boxes that accept input via the keyboard or can be set to report dynamically computed information or by a Measurement Plan. The types of fields are as follows:

- **Input Fields.** Fields intended for input from the user. They display a yellow background (red until filled in, if a required field). The navigation tool at the top of an SA report guides a user

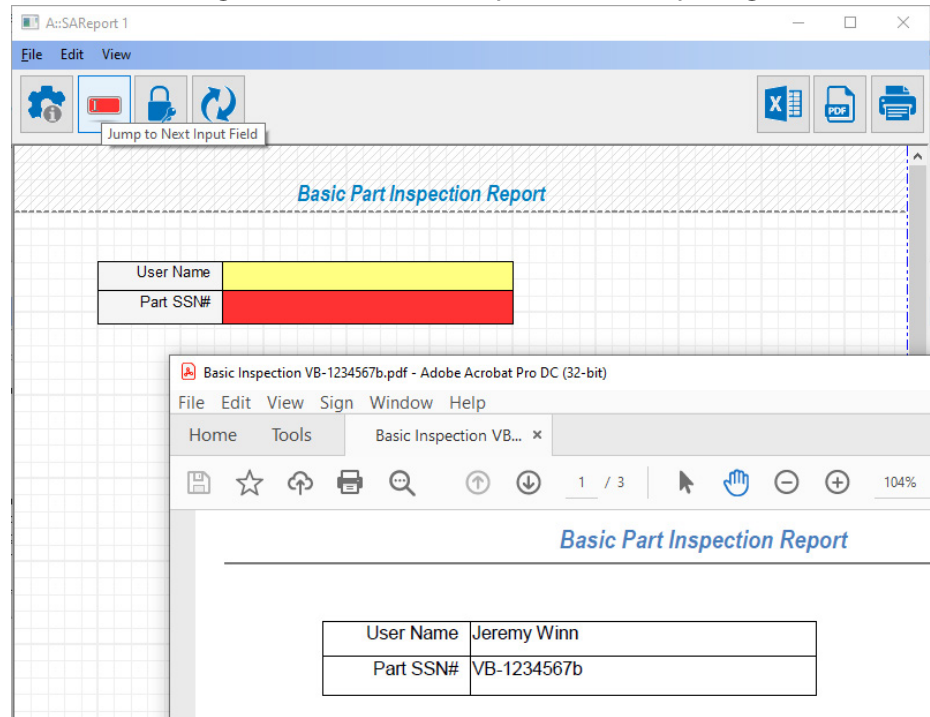


Figure 28-15. Example Input fields as part of a small table, showing input and required input file

- **Output Fields.** Fields intended to display output, either using the selected field value directly or using a custom defined Tag value set through a MP script. These fields have gray backgrounds.
- **Dynamic Fields.** Dynamic fields are defined by a user selected **Attribute**. Attributes include GR-feature values, GD&T results, vector groups statistics, or dimensional values.
- **Equation Fields.** Equations provide a means to compute values directly within a report using variables that can be manually entered or based upon attribute references.

To add a field to a report, right-click on the canvas and select **Add New Field** (Figure 28-16).

Figure 28-16. Newly Added Report Field.

To add an input field, choose the **Input** radio button. If it is a required field, check the **Required** option. A required field will have a red background until it is filled in, and any attempts to print the report or send it to PDF will display a warning dialog.

To add an output field, choose the **Output** radio button. The value displayed in output fields can be set from a specific value, or from a tag (see below).

Using Tags

Every field (both input and output fields) can be assigned a tag. There are two types of tags: *system tags* and *user-defined tags*.

- System tags include << **Date/Time** >> and << **Filename** >> and automatically populate the field with the current date/time or filename (additional system tags will be added in the future). System tags can *only* be assigned to output fields. To create a system tag, select one of the system tags from the dropdown list in the *Tag* field.
- User-defined tags are tags that you have created yourself (such as “MyTag”) that allow you to reference the field from measurement plans so that a script can fill out the field automatically. To create a user-defined tag, just type a unique tag name into the *Tag* field. You can also assign an existing tag to a field in order to populate multiple fields with the same value.

Dynamic Attributes in Fields

Dynamic Attributes can be selected using GR-feature criteria and reporting values, GD&T settings and results, vector groups statistics, or

dimensional values. These can be selected in advance and will dynamically update as measurements are made and values are updated.

An attribute is defined using a string format such as the following which will appear in the Field View line:

*"A::BoltHole Pattern4::Relationship\$Magnitude XYZ#6*1.000"*

- **A::BoltHole Pattern4::Relationship.** The relationship to reference.
- **\$Magnitude XYZ.** Defines the criteria to reference (in this case the XYZ deviation between the nominal and measured)
- **#6.** Defines the numeric resolution of the result
- ***1.000.** Provides a means to scale the attribute by a fixed multiplier. This can be helpful in reporting diametric deviation as an example.

Equation Fields

When more complex computational results are needed an equation field can be added. Equations are self-contained computational elements that use a set of variables as inputs for an equation. Variables are defined as manually entered values, attributes or references to other cells in a table (Figure 28-17).

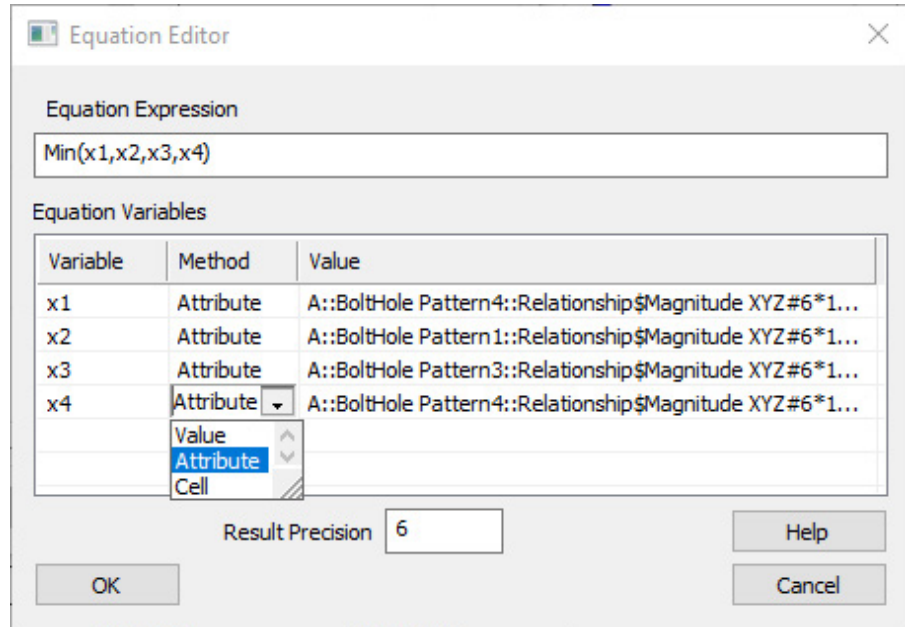


Figure 28-17. Basic Equation computing the minimum value in a list of attributes

These expressions may contain:

Variables such as X, Y, Z, TEMP, STRESS, STRAIN etc.

Functions such as SIN(X), MIN(A,B), MYFUNC(D)

Constants such as 2, 3, 50

Scientific notation like: 3E+10

Arithmetic operators: +,-,*,/,%(modulus),^(power)

Logical operators : =,<,>,<>,>=,<=,! (not),&(and),|(or)

Table cell locations like: cell(2, 1) or cell(0, 3)

Example expressions are:

$X+Y/2$

$SIN(X)+AVG(A,B,C)$

$SUM(X,Y,Z,K,L)/5$

IF(1, 2, 3) is 2,

IF (condition, val1, val2) means that if “condition” is not zero, then result is “val1”, otherwise the result is “val2”.

Additional predefined functions are available under the **Help** button in the Equation Editor.

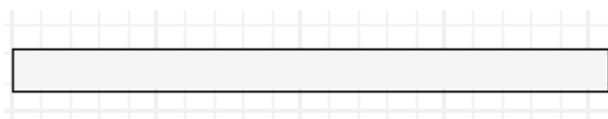
Field Value

A field value is used to populate the field. Field values are always displayed in the field itself, and are changed when a script or a user modifies the field.

Field Display options

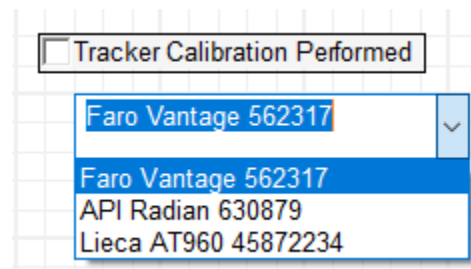
A basic field will appear in the report as a rectangular box in the report. It can be moved and sized as needed by click dragging either on the center of the field or on the middle of an edge respectively. Typically the length of the field should be set large enough to ensure that any text expected will fit within the field (Figure 28-18).

Figure 28-18. Basic Field's appearance in an SA Report



Fields can also be displayed as a **Check Box** or as a **Combo Box** which allows selection from a pre-populated list (Figure 28-19).

Figure 28-19. Check Box and Combo box examples in use in an SA Report.



Tables

Tables provide a convenient means to align and group a set of report fields together. To add a table to a report, right-click the canvas and select **Add New Table**. You can choose from:

- **Add New Empty Table.** Specify a name for the table, and the desired number of rows and columns.
- **Add New Relationship Attributes Table.** The number of rows and columns are defined by the number of GR-Features selected and the attributes to include. The Attribute selection order is used to define the table columns (Figure 28-20).

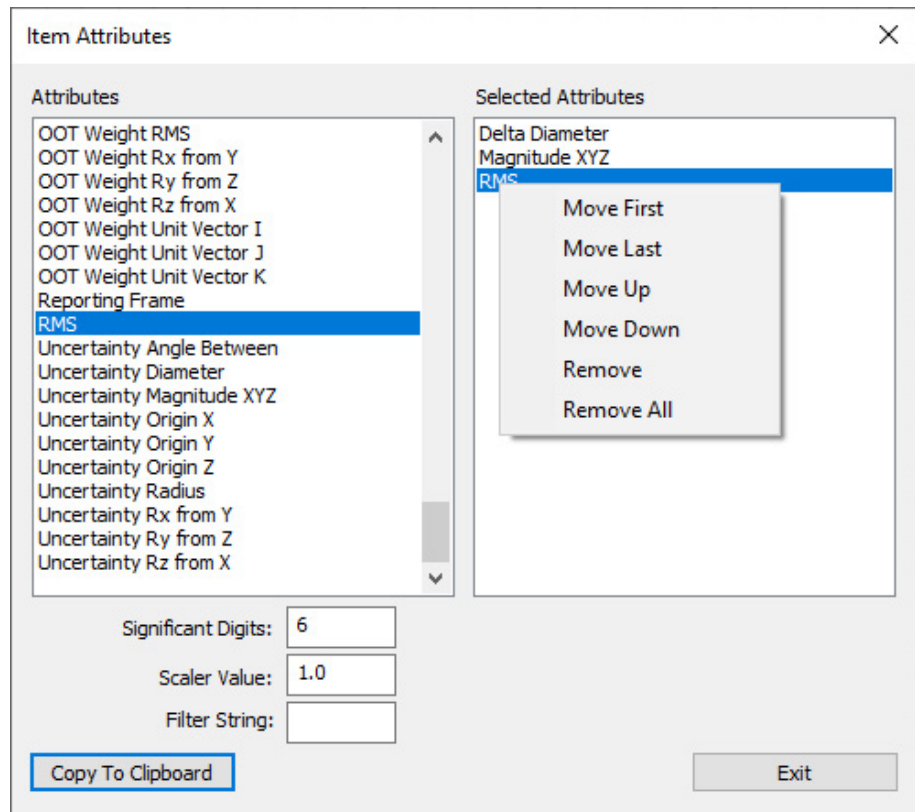


Figure 28-20. Attribute Selection as part of building a new relationship attributes table.

- **Add New Feature Check Attributes Table.** Also builds a table using the selected attributes for a list of feature checks.
- **Add new Dimensions Attribute Table.** Also builds a table using the selected attributes for a list of dimensions.

Tables can be resized by dragging their corners, or the left/right/bottom sides. Interior cells can be resized by dragging any internal horizontal or vertical line. Drag the interior of any cell to move the entire table.

The properties of each cell can also be adjusted. Add an input or output field to a cell by right-clicking a cell and selecting **Cell Options**. You can also convert a cell into a **Check box** or a **Combo box** as needed.

Table Properties.

To modify the number of rows or columns in a table, right-click the table and select **Table Properties** (Figure 28-21).

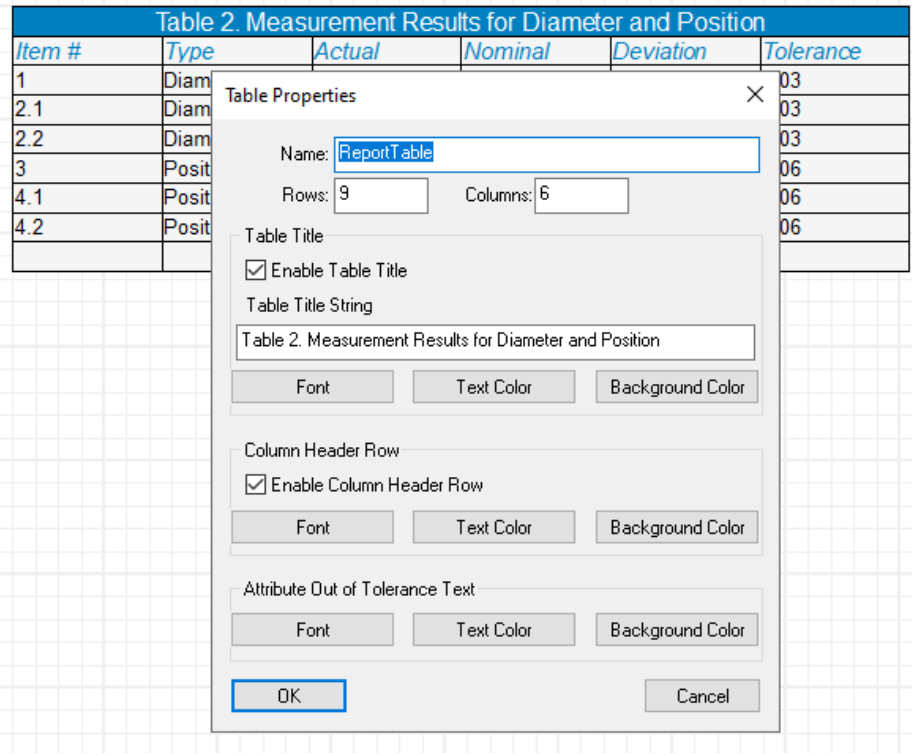


Figure 28-21. Table Properties control provides title and sizing.

The table properties includes control over the display of an optional title row and colorization of out of tolerance dynamic attributes.

To insert or delete rows or columns or to change the alignment of a cell, right-click a cell and then choose **Cell Options**.

Modifying Items in the report

Items in the report can be modified by right-clicking the report item and selecting **Report Options**. Depending on the item, a context sensitive menu will appear. All items have the following common options: **Report Options**, **Remove from Report**, **Bring to Front**, **Send to Back**, and **Properties**.

Object Placement in a Report

SA's approach to reporting has been to provide complete flexibility in placing objects on a page for reporting. Rather than being restricted to a standard sequential table display, each page can be arranged as needed much like a Microsoft PowerPoint slide. Additional auto-arrange functions were added to simplify reporting with large reports.

Page Positioning

Tables within a dynamic report can be positioned dynamically as needed. They can also be placed precisely using right-click page alignment option including **Center**, **Left** and **Right** justified. Object layering is also supported with **Bring to Front** and **Send to Back** controls as well as a **Remove Object Overlap** option.

Set Auto Arrange

Large sections of a report can be selected and formatted as needed using the right-click Set Auto Arrange functions. These include:

- **Set Auto Arrange Start.** Establishes a starting point for the selection and auto arrange process. Selection will continue to an Auto Arrange Stop point or the end of the report, whichever comes first.
- **Set Auto Arrange Stop.** Establishes a stopping point for selection and auto arrange.
- **Select Auto Arrange Block.** Selects all the items within the Auto arrange block from start to end.
- **Execute Auto Arrange.** Evenly distributes the tables within the auto arrange block.

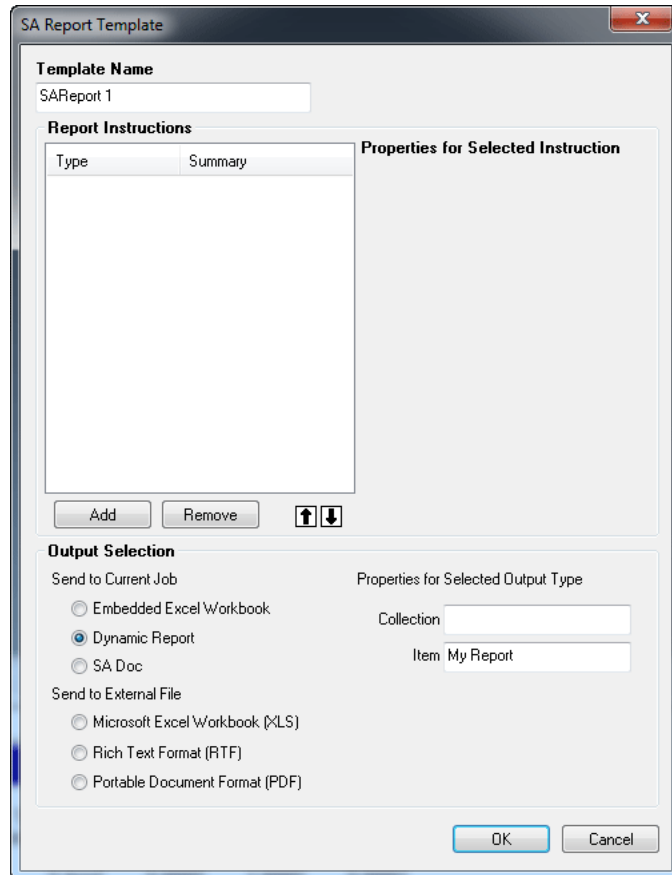
Report Templates

SA Report Templates offer a repeatable method for creating reports of items from the tree. Once created they can be used repeatedly.

Creating an SA Report Template

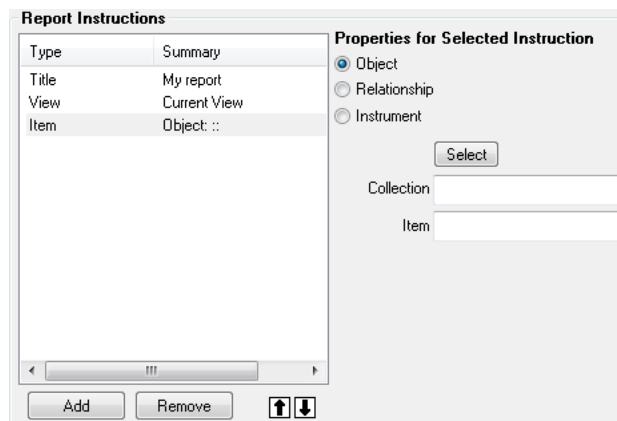
Select **Reports>Add New SA Report Template** from the menu and a report template will be created in the tree ([Figure 28-22](#)).

Figure 28-22. The SA Report Template dialog.



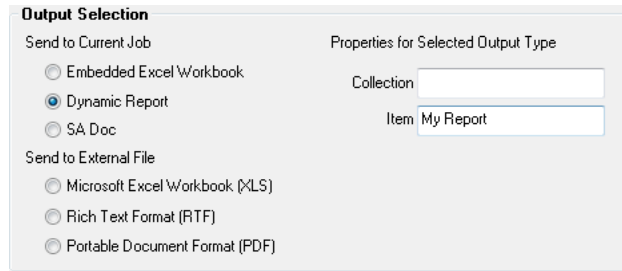
Inside the SA Report Template the inputs and outputs are controlled in the *Report Instructions* box (Figure 28-23). Simply use the **Add** or **Remove** buttons to control items availability in the report.

Figure 28-23. The Report Instructions box.



The *Output Selection* defines the output format and location of the report when it is generated (Figure 28-24). Choose from embedded locations or external locations.

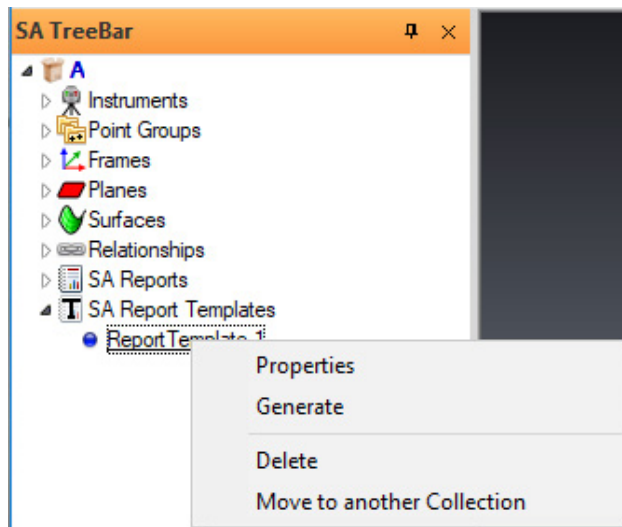
Figure 28-24. The Output Selection box.



Generating the Report

Once the Report Template is designed and saved, a report can be generated. Simply right-click the respective SA Report Template and choose **Generate** (Figure 28-25). The report will be created and saved in the prescribed output format.

Figure 28-25. Generating a report from a template.



Object Report Options

Items in the tree have report options that can be accessed and configured per the users needs. You can access these options by right-clicking the object and selecting **Report Options** either from within the report directly or from selecting the object in the SA Treebar menu.

- For Object reporting options refer to the sections starting with “[Objects vs. Geometry Relationships](#)” on page 272.
- For relationship reporting options refer to: “[General Relationship Properties & Controls](#)” on page 420.
- For GD&T reporting options refer to: “[Reporting Options](#)” on page 674.

Events

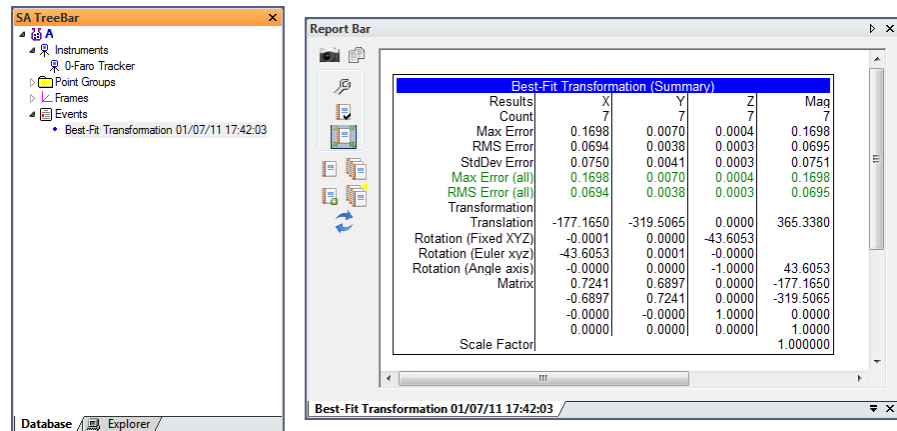
SA *Events* provide detailed information for major events that occur in SA. These events can be viewed in the Report Bar or be added to any Report (Figure 28-26). Events are created automatically when certain functions are performed and are stored in the tree.

SA Events are generated for many actions which include operations such as :

- Best-Fit Transformations
- Best-Fit Geometry
- Move Objects by Minimizing Relationships
- Unified Spatial Metrology Networks (USMN)
- Quick Align to CAD
- Queries
- CAD Imports

and more to come!

Figure 28-26. A best fit transformation event in the tree.



Queries

A **Query**, much like building a dimension, is used to determine the distance or angle between objects or a set of points. It is intended to be a one time operation that returns many values for analysis. It is typically returned as a vector group containing both a distance and direction to indicate the magnitude of displacement between the selected objects. Queries are not dynamic, and will not update as the position of objects are changed, providing a means to record relative distances at any time during a project. For dynamic queries, you can also enable **Auto-vectors** as part of a relationship (see “[General Relationship Properties & Controls](#)” on page 420).

The **Queries** section of the Reporting Tab contains numerous methods for comparing points and objects. A bulk of your analysis needs can be found under this menu item. All query commands automatically account for target offset unless otherwise noted.

The output of most query commands is either a **Query Results** dialog (Figure 28-3) or a vector group. The result dialog will contain query specific results and can be used to create a relationship by selecting **Create Relationship**.

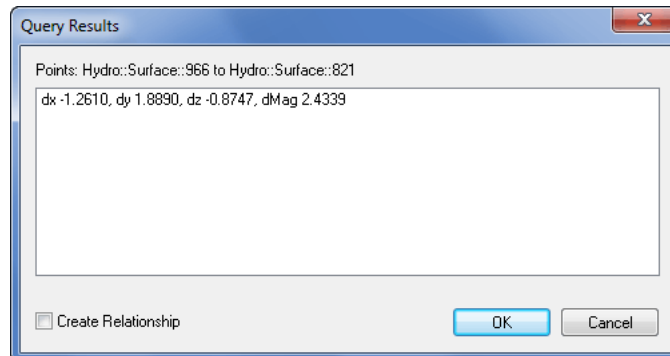


Figure 28-27. The Query Results dialog.

A vector group is a graphical depiction of the magnitude between points and objects (Figure 28-22). It is often displayed with arrows (whiskers) showing the deviation from the object to the point.

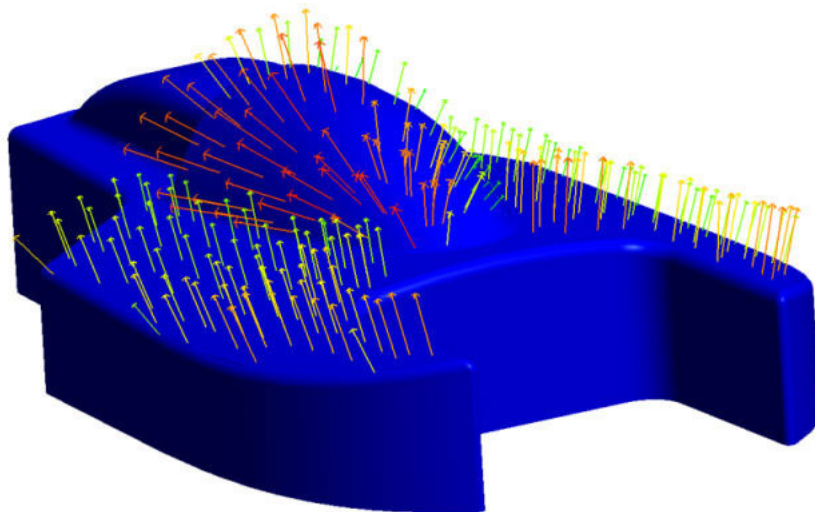


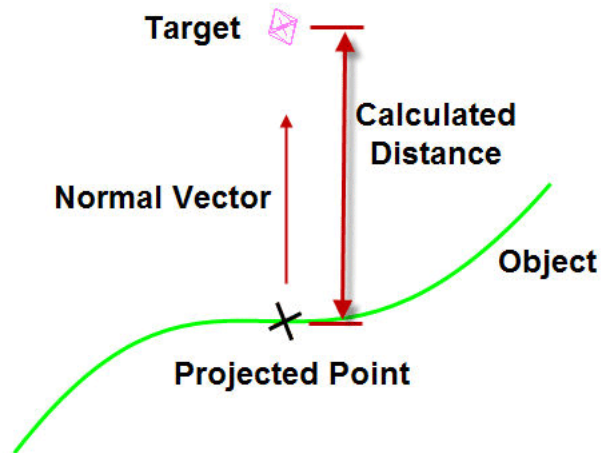
Figure 28-28. A vector group depicting error magnitude and direction.

Query Basics

When a point is compared to an object it is projected to the object using a local normal and then the difference is calculated (Figure 28-23). The below example is the result of a point to object query. The dx, dy and dz values represent the difference between the point and its projected point in working coordinates. dMag represents the magnitude. Conventionally magnitude does not carry a sign but since objects have a positive and negative side the magnitude sign can be used to determine which side of the object the point lies. Also, the

objects positive side is used for determining target offset direction. The target offset is simply subtracted from the calculated distance.

Figure 28-29. The anatomy of a query.



Query Point To Object

Displays the deviation of a point to an object. Results displayed in Working Coordinates.

Object (Ignore Offsets)

Displays the deviation of a point to an object without compensating for target offset. Results displayed in Working Coordinates.

Point

Displays deviation between two points in Working Coordinates.

Circle (ignore offsets)

Displays the deviation of a point to a circle without compensating for target offset.

Query Points To Single Point

Creates a vector group contain deviation between selected points and single point.

Points

Points to points query results in a distance map between each combination of points (Figure 28-24).

Figure 28-30. An Inter-Point Distance query finds distances between all combinations of a set of points.

	A::Surface::881	A::Surface::882	A::Surface::883	A::Surface::884
A::Surface::881		0.2865	0.6372	0.9071
A::Surface::882	0.2865		0.3511	0.6211
A::Surface::883	0.6372	0.3511		0.2728
A::Surface::884	0.9071	0.6211	0.2728	
A::Surface::885	1.1447	0.8626	0.5277	0.2665
A::Surface::886	1.0514	0.7918	0.5259	0.3687
A::Surface::887	0.8030	0.5720	0.4181	0.4307
A::Surface::888	0.4764	0.3723	0.5164	0.7041
A::Surface::889	0.3475	0.5152	0.8227	1.0620
A::Surface::890	0.5319	0.7825	1.1173	1.3688
A::Surface::891	0.7725	1.0362	1.3749	1.6278
A::Surface::892	0.6617	0.8818	1.1966	1.4331

Curves/Objects

Generates a vector group with deviations from the selected points and the closest object (if more than one object is selected).

Circle (Radial and Planar)

Generates two vector groups, one for radial deviation and one for planar deviation.

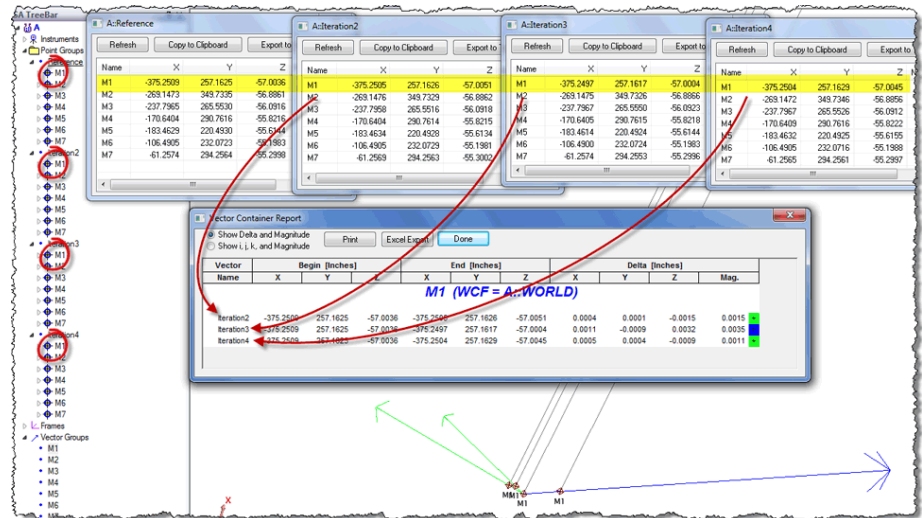
Query Group to > Multiple Groups > Organize By Groups

Use this command to instantly compare several groups back to a reference group. This command will create multiple vector groups: One comparing the reference group to group A, another comparing the reference group to group B, and so on. This is useful for comparing drift checks or doing repeatability studies, enabling you to view deviations of multiple measured groups over time.

Multiple Groups > Organize By Points (Gage R&R)

Similar to the “Organize by Groups” command described above, except the vector groups contain the same point as seen over time. This is particularly useful for viewing the variation of a specific point over time (Figure 28-25).

Figure 28-31. A multiple-group query organized by points.



Query Point Clouds to Objects

Generates vector group containing deviation of individual points in a point cloud to object(s).

Query Line to Line Angle

Computes angle and mutual perpendicular between two lines.

Plane Angle

Computes angle and mutual perpendicular between line and the normal vector of the plane.

Query Plane to Plane

Computes angle and mutual perpendicular between the two plane normals.

Query Frame to Frame

Computes position and rotational difference between two frames expressed in the working coordinates.

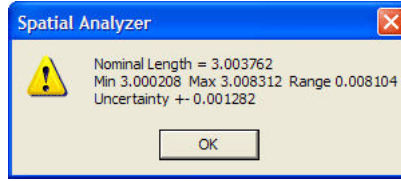
Query Object to Object Direction

Computes angle and mutual perpendicular between the two object normals.

Query Sensitivity Cloud to Sensitivity Cloud

Using the SpatialAnalyzer coordinate uncertainty computation capabilities, you can create a cloud of points around a measured point. These points represent the “cloud of uncertainty” of the coordinate. If you have clouds created for at least two points, you can determine the uncertainty of the line between them. The results are presented in [Figure 28-32](#).

Figure 28-32. Querying one sensitivity cloud to another.



Query Auxiliary Data

Generates a vector group using auxiliary data (Figure 28-91). This allows the user to view and chart auxiliary data as you would with 3D data.

Figure 28-33. The Auxiliary Data Vector Group dialog.



Cloud to Cloud Query

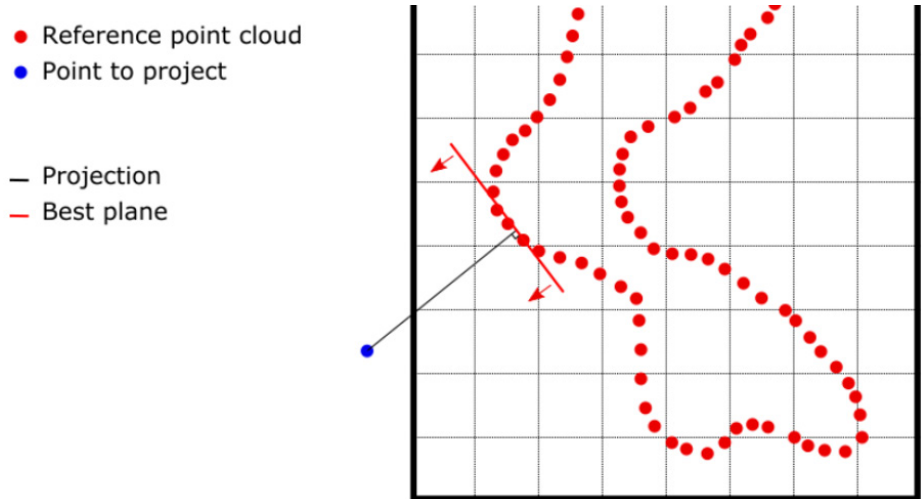
Cloud to cloud query is unique in that it generates a colorized cloud from the relative distances between cloud points. The cloud colorization and voxel callouts can then be used to determine the precise distances between the clouds at particular locations. This analysis approach is designed for larger data sets than cannot typically be analyzed using vector groups, because of performance and graphic limitations.

This command calculates a color map to show the deviations between two point clouds.

1. The first selected object is the reference object.
2. The second selected object is the measured object that will be projected on the reference. This cloud will be copied and colorized according to the deviations.

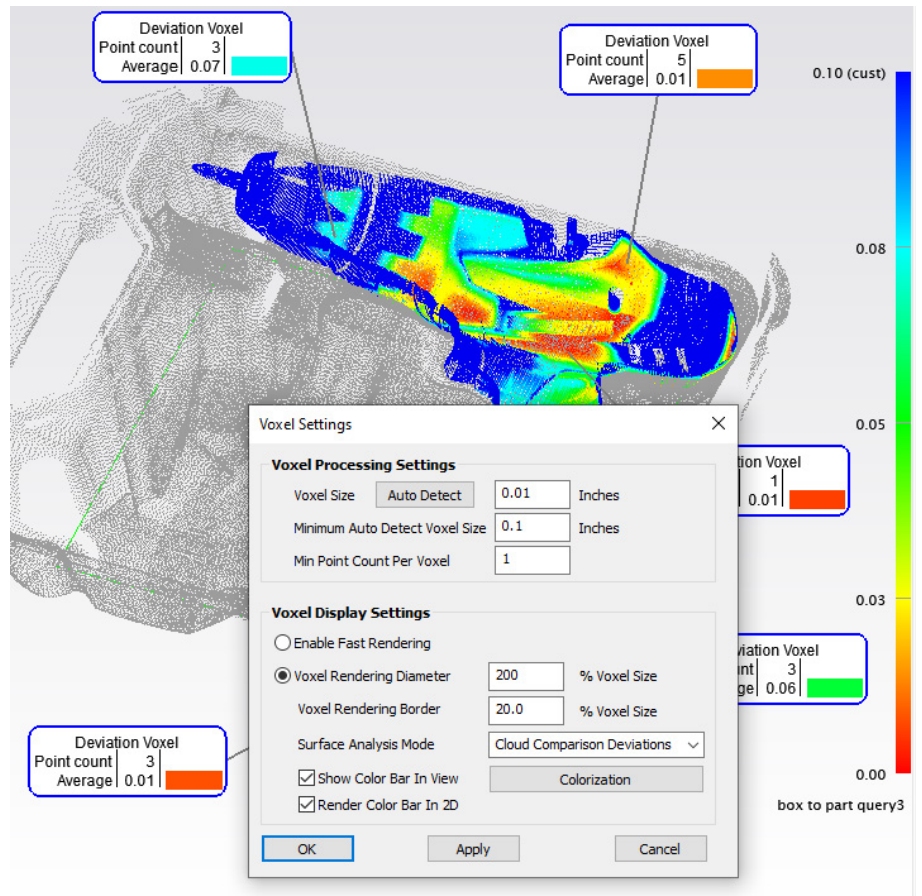
Each point is projected in the direction of the local normal of the cloud on a local plane. In other words, the “distance” is not the closest distance to a point of the target cloud, but the closest distance to a local plane of the target cloud (Figure 28-97).

Figure 28-34. Example of how a distance value is computed in a cloud to cloud query using a local plane.



Each point to plane distance value is saved in the properties of the cloud point and can be displayed using the voxel colorization controls when the *Surface Analysis Mode* is set to **Cloud Comparison Deviations** (Figure 28-16).

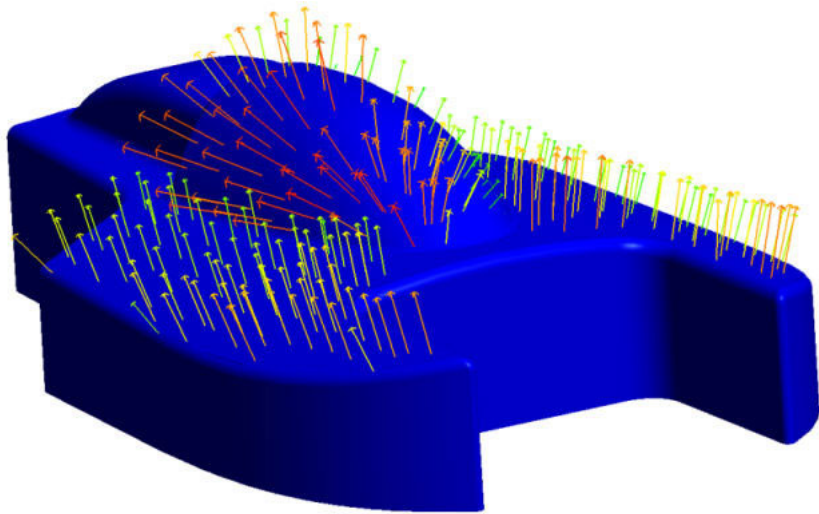
Figure 28-35. Voxel cloud display settings used to show the cloud to cloud query colorization.



Vector Groups

A vector group is a graphical depiction of the magnitude between points and objects. It is often displayed with arrows (whiskers) showing the deviation and direction from the object to the point (Figure 28-93). It can also be magnified to more clearly display relative deviations in very small numbers.

Figure 28-36. A vector group.



Vector groups are built in the following ways...

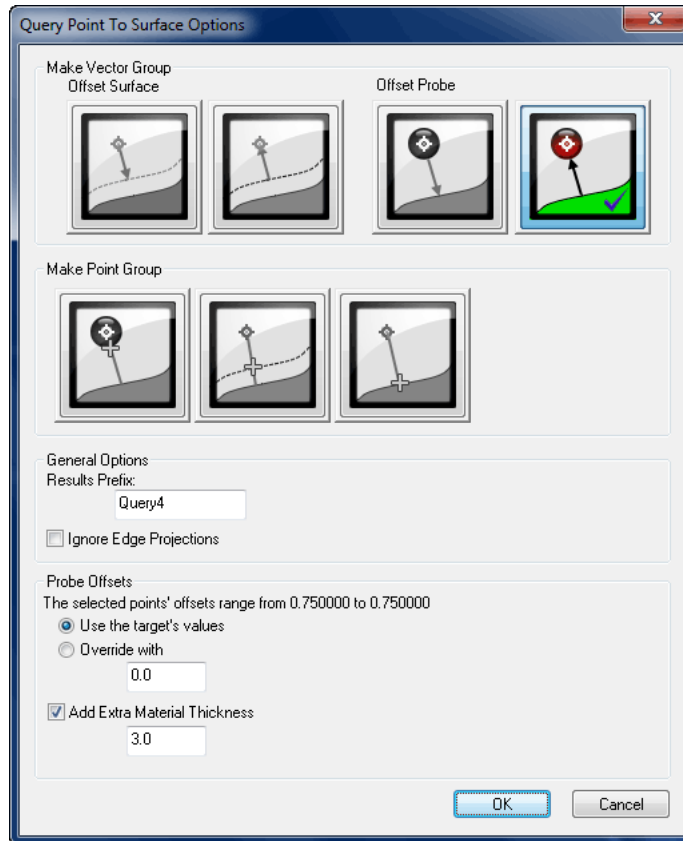
- **By using Query.** This is the most common method and creates a vector group that represents a snapshot of the deviations between points and objects.
- **Auto-Vectors created with a relationship.** Relationships can update a vector group so that rather than being a single measurement event the vector group is always updated to display the current deviations.
- **By construction.** At any time a snapshot of an autovector can be captured or there are additional construction commands to build or combine vector groups together.

Vector Group Properties

Projection Options

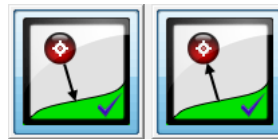
The projection options dialog (Figure 28-90) can be seen in several places in SA but is mainly used in Query Points to Objects and relationships. The projection options allow the user to control vector group direction, add material offsets, create offset point groups, and change probe offset compensation.

Figure 28-37. The Query Point To Surface Options dialog.



Offset Probe

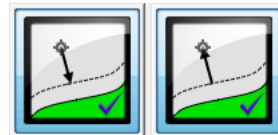
Figure 28-38. The Offset Probe vector group options.



These two options (Figure 28-89) create vector groups depicting deviation of the surface from the probe (left) and of the probe from the surface (right, the default option). Both of these options create vectors with the exact same magnitude but exactly opposite in direction. The left option therefore depicts a typical “build” perspective (how far do I need to go to get to nominal?), whereas the right option depicts a typical “inspect” perspective (how far am I from nominal?).

Offset Surface

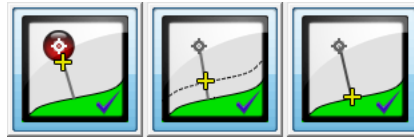
Figure 28-39. The Offset Surface vector group options.



These options (Figure 28-40) create vector groups with the head or tail on the center of the target and all offsets applied at the object. This is sometimes used with thin parts to prevent the vector from disappearing behind the nominal surfaces.

Make Point Group

Figure 28-40. Projection options for making point groups.

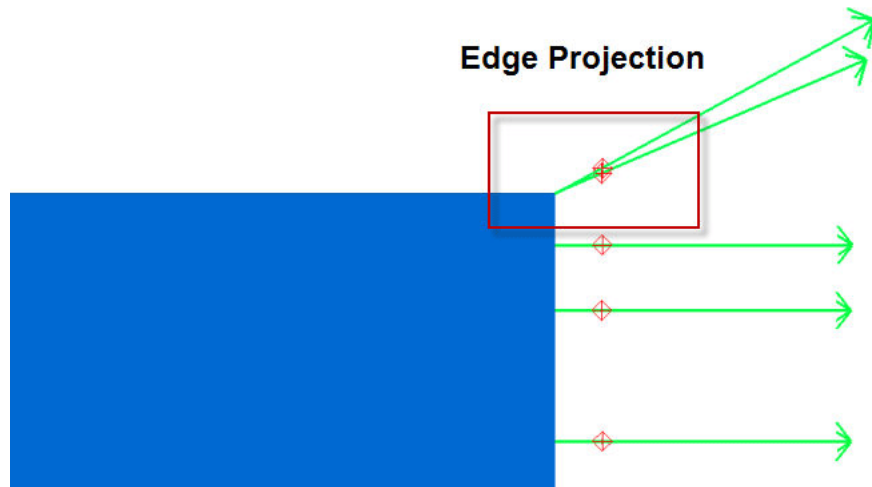


These options (Figure 28-88) create point groups instead of vectors. The first option applies any offset at the probe center. Unless additional material thickness is added, this represents the probe's contact point on the surface. The second option applies all offsets at the surface, and the third option creates projected points on the object to which the query is being performed.

Ignore Edge Projections

The **Ignore edge projections** option is used when you would like to ignore points whose projections onto objects lie on a surface edge (Figure 28-87).

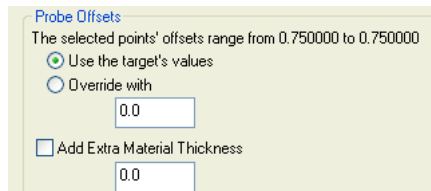
Figure 28-41. Two points which are edge projections.



Probe Offsets

The geometry offset allows the user to use the point specific offset or override the offset (Figure 28-83). The "Use the target's values" will display the offset of the selected points.

Figure 28-42. Probe offset options.



Vector Group Properties

The vector group properties allow the user to view vector group statistics as well as change the graphical appearance (Figure 28-79).

Figure 28-43. The properties of a Vector Group.

Components Included in Vector

The X, Y, Z check boxes provide the ability for the deviations to be reported in 2D or along a single axis as needed and can be turned on or off in any combination. These component controls are relative to the working frame unless a reporting frame for the vector group has been set.

Statistics for Mag (XYZ)

The properties dialog also reports summary of the statistics for the vector group. It reports the following values:

- **Max.** The largest vector deviation
- **Avg.** The average deviation for the entire vector group
- **Min.** The smallest vector deviation or largest negative deviation in the group.
- **Vector Count.** The total number of vectors in the group
- **StdDev from Zero.** Reports the standard deviation relative to zero rather than to the average.

- **StdDev from Avg.** Reports the standard deviation from the average.
- **RMS.** The square root of the average of the deviations.

Display Settings

Vectors can be displayed in several display formats. These are differences in the way in which the same information is presented graphically.

Draw As and Scale Controls

- **Arrows, Blotches and Tubes.** Vectors can be displayed in several ways. Traditional display includes Blotches (dots that are oriented relative to the vector direction but do not show the length of the vector, Arrows, and Tubes (which use both the blotch width and taper outward to reflect the full arrow length) (Figure 28-77).

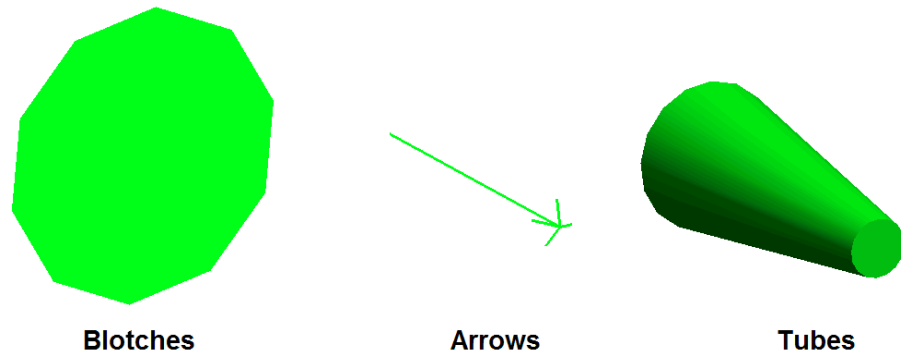


Figure 28-44. Different methods for depicting error.

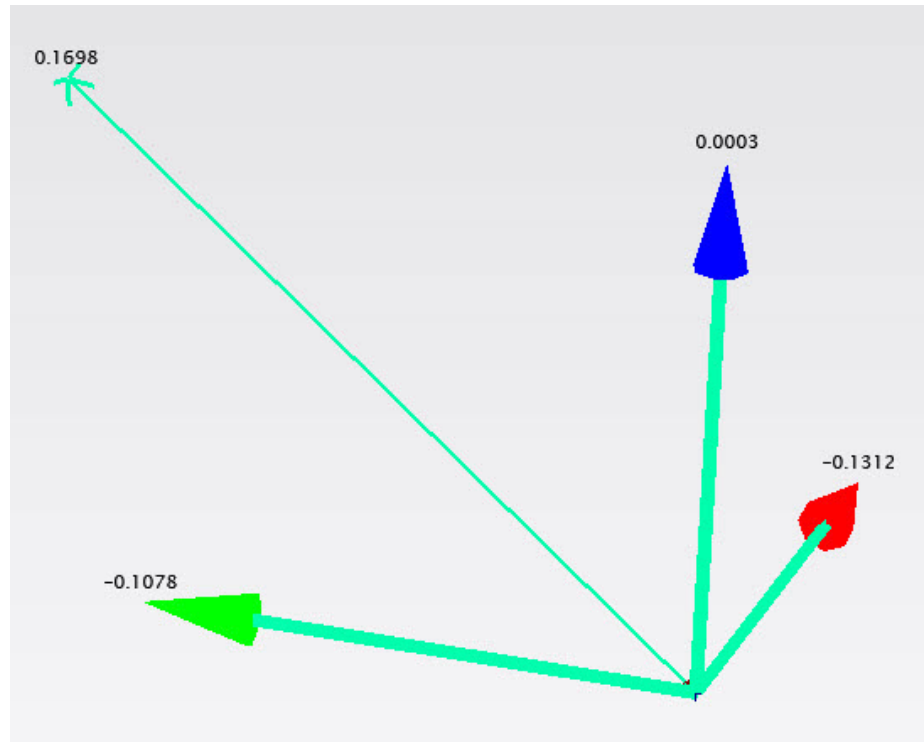
- **Magnification.** Vectors are scaled by default. This scaling helps a user report errors that would otherwise be extremely small. Enter a scale value of 1 to see the actual length of the deviation being reported.
- **Line Width and Blotch size.** Line width controls with width of the graphical line used to depict a vector while the blotch width is used to control the size of a blotch. Tubes use the Blotch size for a base diameter.

Trifoils

Trifoils offer an alternative to traditional vector display option. They use a frame representation for each vector indicating the direction of each component of the deviation in the working frame. Trifoils are not Magnified like vectors and therefore use their own magnification value which applies to all vectors evenly. This can be very helpful when depicting errors on a large part or with widely varied results.

Figure 28-45 shows an example of a single vector shown both as an arrow and as a trifoil. The legs of the trifoil indicate the direction of error and the vector labels indicate the amount for each component.

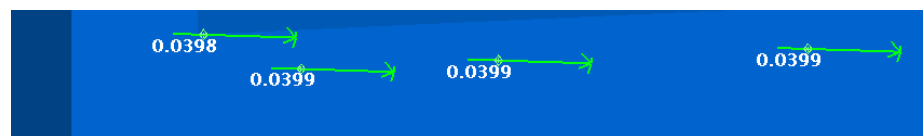
Figure 28-45. Trifoil vector display



Label and Tolerance Controls:

- **Label Vectors With Values.** Display each vector's magnitude (Figure 28-78).

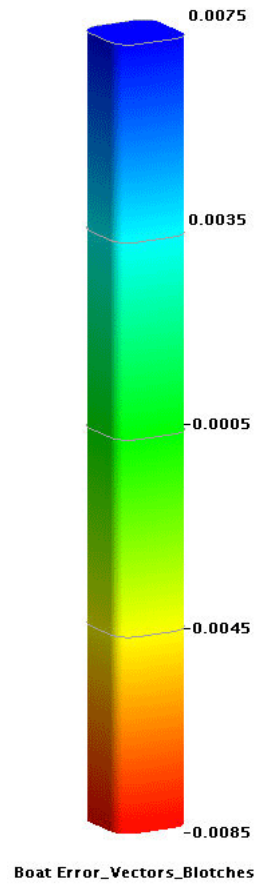
Figure 28-46. A set of vectors that have been labeled with values.



- **Vector Magnification.** Magnifies the graphical length of the vector. Improves viewing when deviation is small.
- **Line Width.** Increase the graphical width of the vector for improved viewing.
- **Show Out of Tol Only.** Only displays vectors that are out of tolerance.
- **Show in View.** Displays the color bar for the vector group (Figure 28-75).

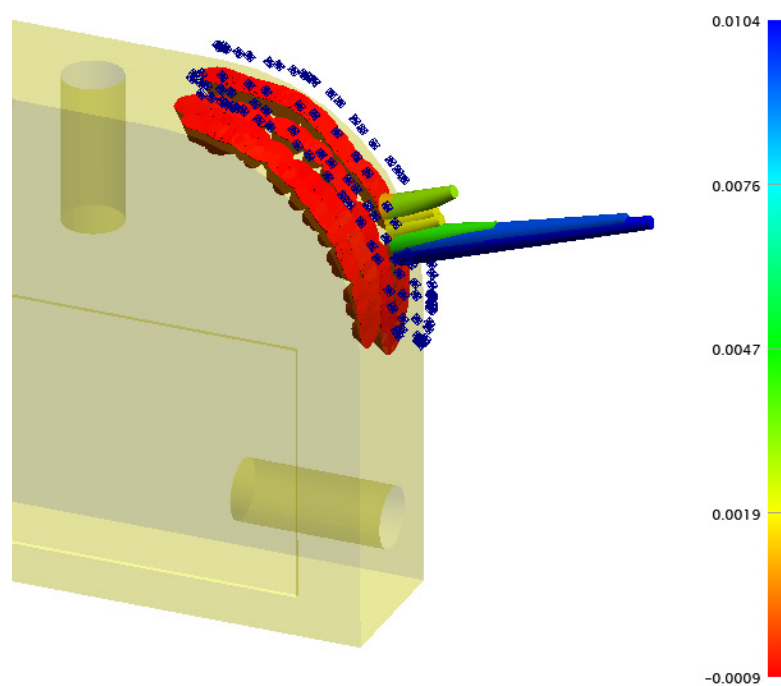
Color Bar

Figure 28-47. A vector group color bar.



- **Redern in 2D.** Converts the Color Bar display to thin 2D version (Figure 28-48).

Figure 28-48. 2D Color bar



- **Show Percentages.** Adds additional labels indicating that percentage of points in each segment of the color bar
- **Show Fractions.** Adds additional labels indicating the fraction of the total number of points in each segment of the color bar.

Edit Colorization

The **Colorization** button provides access to the Color Profile selection dialog. Individual profiles can be set to more clearly display the deviation results relative to predefined tolerances and limits.

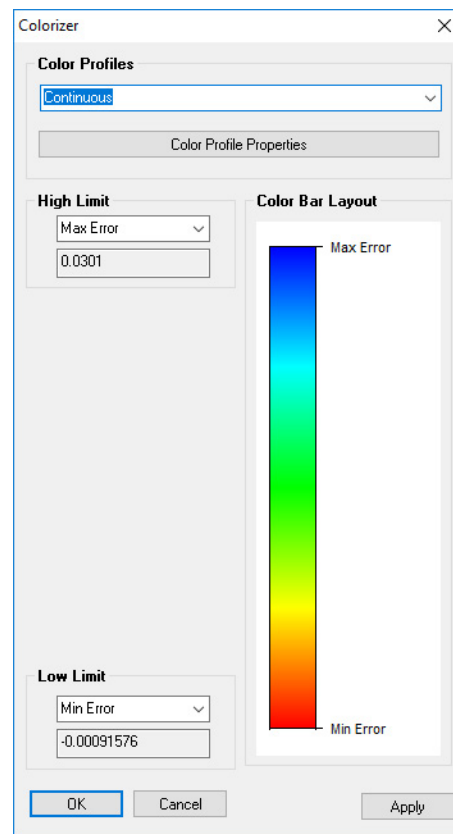


Figure 28-49. Vector group Color Profile selection dialog.

Color Profiles currently offered include the following:

- **Single Color.** Which sets the vectors in that group all to one color.
- **Continuous.** Which provides a continuous transition of colors set between the High and low limits.
- **Tolerance (Continuous).** Which displays a continual transition in color between the extent of the tolerance zone and the high/low limits, but all data within tolerance have a single color.
- **Tolerance (Go / No-Go).** This profile only displays 3 colors defined by the tolerance range, where data above tolerance, within tolerance, and below tolerance each are designated sin-

gle color.

- **Tolerance (Go / No-Go With Warning).** This profile is the same as Tolerance (Go / No-Go) with the addition of 2 warning colors which indicate if values are close to being out of tolerance.
- **Discrete Colors.** This mode offers an option to divide deviations within a selectable number of discrete color bins.

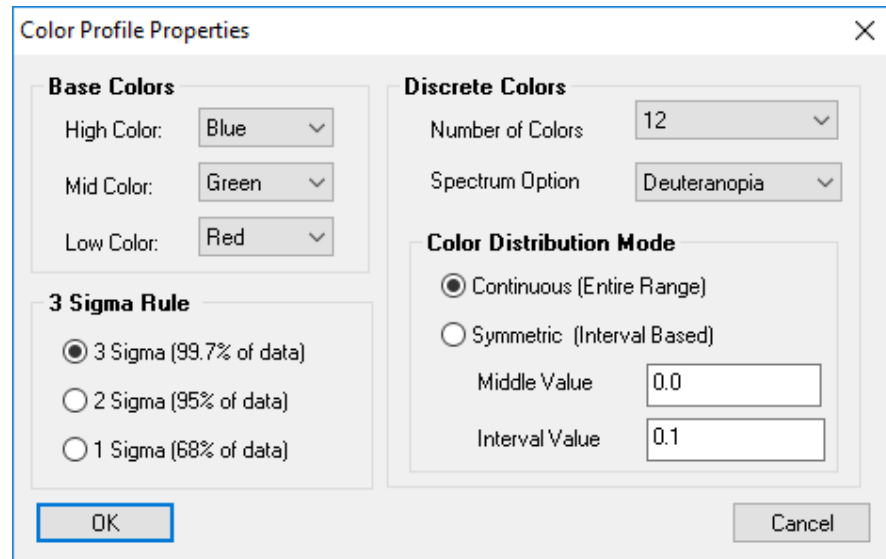


Figure 28-50. Color Profile Properties

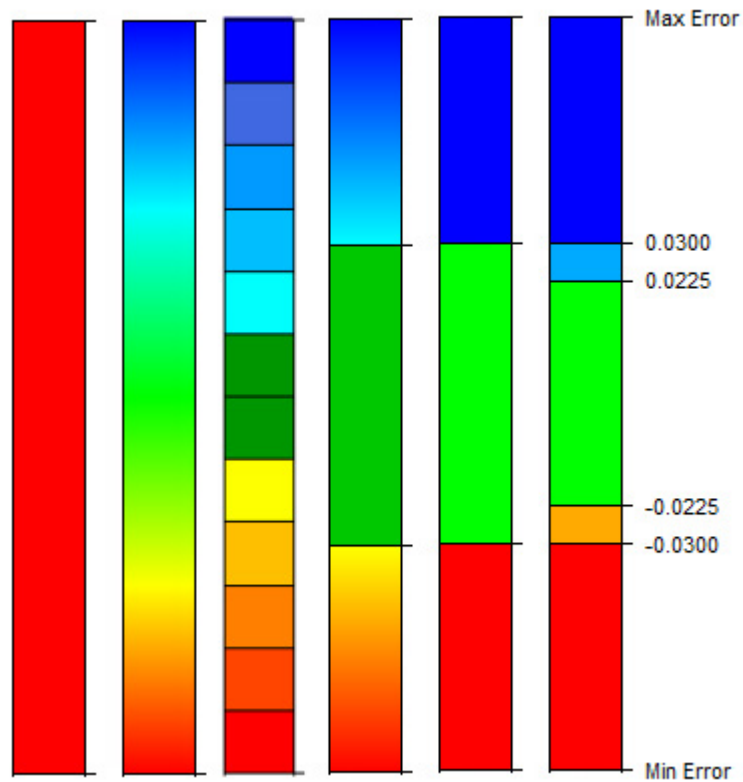
The Color Profile Properties button provide access to customization options for used by the different color profiles (Figure 28-68).

- **Base Colors.** The base colors are used to determined the basic colorization used for the color bars, including High, Mid, and Low color selections.
- **Sigma Selection Rule.** In addition to setting the colorization based upon the limits of the data or a custom value a user may choose to set the color distribution at a proportion of the data extent. This can be helpful in ignoring outliers.

Discrete Color selection and Color Distribution Mode options are only used for the Discrete Color Profile.

- **Discrete Colors.** A selection can be made between the number of different colors to be used (4, 8, 12) and if a special color display option to help with color blindness should be applied. Both a gray scale option and a set of colors designed for people with deuteranopia type color blindness is available.
- **Color Distribution Mode.** The discrete colors can be distributed either evenly within the extents of the dat by selecting the Continuous option or symmetrically by distance about a Middle Value within fixed distances or Interval Values.

Figure 28-51. Color Profiles



The Color Profile selection will reveal a different number of controls depending on the number of colors displayed. These controls define the placement of the colors available in the profile. These controls include the following:

- **High/Low Limit.** The upper/lower extent of the defined color displayed.
- **High/Low Tolerance.** The upper/lower extent of the tolerance zone.
- **High/Low Warning.** The value which defines the beginning of the warning zone, which will extend to the High/Lower Tolerance value.

The High and Low limits can be set based upon several parameters:

- **Max / Min Error.** This setting will adjust the color display such that the maximum and minimum values in the data define the extent of the color.
- **+/- 3 Sigma.** In noisy data sets outliers can cause all the colors to be compressed. When 3 Sigma is used the extent of the colorization is defined by 3 Sigma or 99% of the data.
- **Custom.** A custom value can also be entered.

Currently the only Color Profile that provides full control over every

possible setting is the Toleranced (Go / No-Go With Warning). If the limits of the color are set to a custom value or 3 Sigma then the data values could extend beyond the defined colorization. In that case we add darker color above or below the defined region.

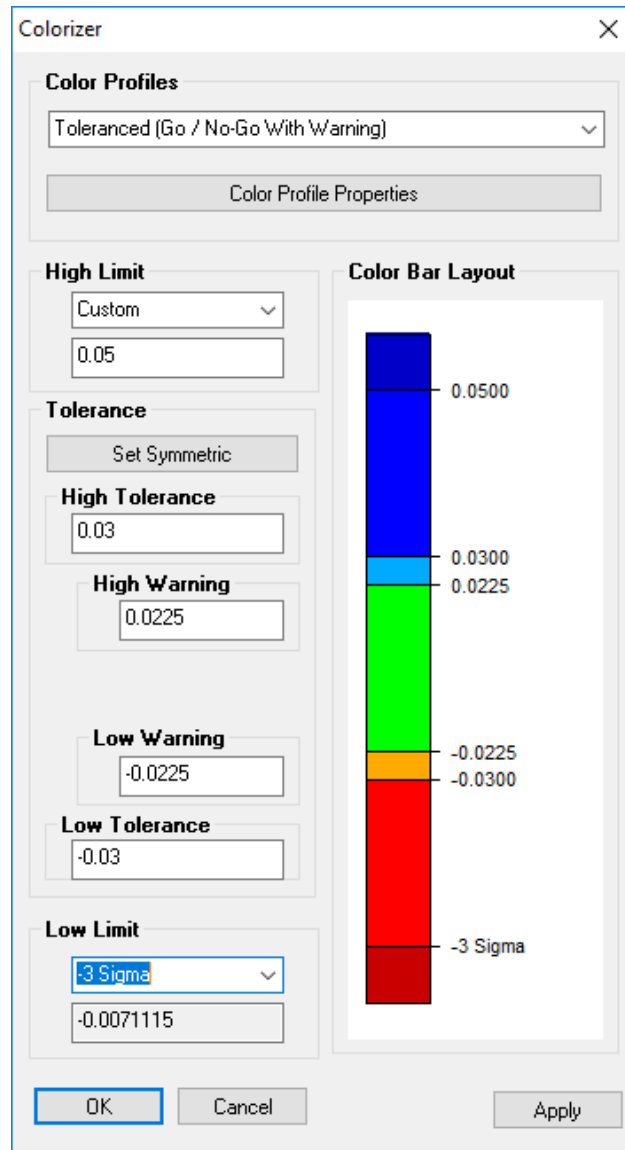


Figure 28-52. Tolerance (Go / No-Go With Warning) with custom limits

Report

The **Report** button will generate a report showing each vectors begin, end and delta coordinates (Figure 28-74). It also displays magnitude with the respective colorization. This report can be printed or exported from this dialog. This report can also be generated in other formats such as Quick and Composite reports.

Each column can be sort ascending or descending when name value is double left-clicked.

Vector Name	Begin [Inches]			End [Inches]			Delta [Inches]			
	X	Y	Z	X	Y	Z	X	Y	Z	Mag.
924	-0.8293	4.1807	1.8840	-0.8287	4.1807	1.8859	0.0005	-0.0000	0.0020	0.0020
925	-0.8128	4.4542	1.8802	-0.8121	4.4542	1.8825	0.0006	-0.0000	0.0023	0.0024
926	-0.5541	4.4910	1.8010	-0.5532	4.4912	1.8027	0.0009	0.0002	0.0017	0.0019
927	-0.9186	4.5213	1.9081	-0.9180	4.5213	1.9105	0.0006	-0.0000	0.0025	0.0025
928	-1.0208	4.7594	1.9335	-1.0202	4.7594	1.9359	0.0006	-0.0000	0.0024	0.0025
929	-1.8369	-0.8633	2.3727	-1.8319	-0.8634	2.3752	0.0050	-0.0001	0.0025	0.0056
930	-1.9593	-0.8481	2.5895	-1.9553	-0.8482	2.5921	0.0041	-0.0002	0.0026	0.0048
931	-2.1385	-0.8561	2.8159	-2.1357	-0.8563	2.8187	0.0028	-0.0002	0.0028	0.0040
932	-2.3225	-0.8469	2.9694	-2.3213	-0.8471	2.9713	0.0012	-0.0002	0.0019	0.0023
933	-2.5231	-0.8251	3.0716	-2.5226	-0.8252	3.0729	0.0005	-0.0001	0.0013	0.0014
934	-2.7429	-0.8349	3.1249	-2.7428	-0.8349	3.1254	0.0001	-0.0000	0.0005	0.0005
935	-2.9903	-0.8657	3.1330	-2.9903	-0.8657	3.1334	-0.0000	-0.0000	0.0004	0.0004
936	-3.2167	-0.7487	3.1097	-3.2164	-0.7486	3.1086	0.0003	0.0001	-0.0011	-0.0012
937	-3.4473	-0.8198	3.0178	-3.4459	-0.8196	3.0151	0.0014	0.0002	-0.0027	-0.0030
938	-3.6592	-0.8156	2.8681	-3.6563	-0.8154	2.8650	0.0029	0.0002	-0.0032	-0.0043
939	-3.8454	-0.8305	2.6566	-3.8409	-0.8305	2.6533	0.0045	0.0001	-0.0033	-0.0056
940	-3.9703	-0.7880	2.4582	-3.9650	-0.7880	2.4553	0.0052	-0.0000	-0.0029	-0.0060
941	-4.0766	-0.7862	2.2385	-4.0698	-0.7864	2.2356	0.0069	-0.0001	-0.0029	-0.0075
942	-4.1150	-0.4816	2.1181	-4.1132	-0.4817	2.1171	0.0018	-0.0001	-0.0010	-0.0020
943	-4.0038	-0.4912	2.3627	-4.0002	-0.4914	2.3810	0.0036	-0.0001	-0.0017	-0.0040
944	-3.8617	-0.5202	2.6340	-3.8590	-0.5202	2.6322	0.0027	-0.0000	-0.0018	-0.0033
945	-3.6994	-0.5450	2.8367	-3.6975	-0.5450	2.8349	0.0019	0.0000	-0.0018	-0.0026
946	-3.4825	-0.5316	3.0108	-3.4817	-0.5315	3.0095	0.0008	0.0000	-0.0013	-0.0015

Figure 28-53. Viewing a vector group report.

Chart

The chart button will generate one of three predefined charts (Figure 28-73).

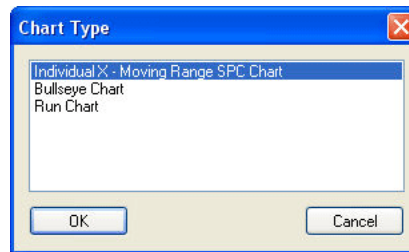


Figure 28-54. Chart Type.

The user can select the inputs for the chart type (Figure 28-72):

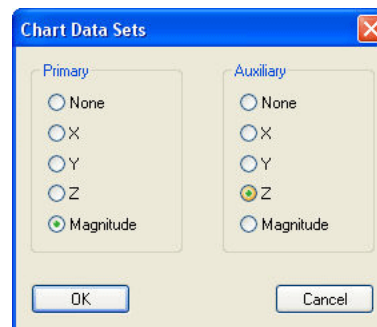


Figure 28-55. Axis settings for the charts.

Once the chart is generated (Figure 28-71) it is saved in the SA Treebar.

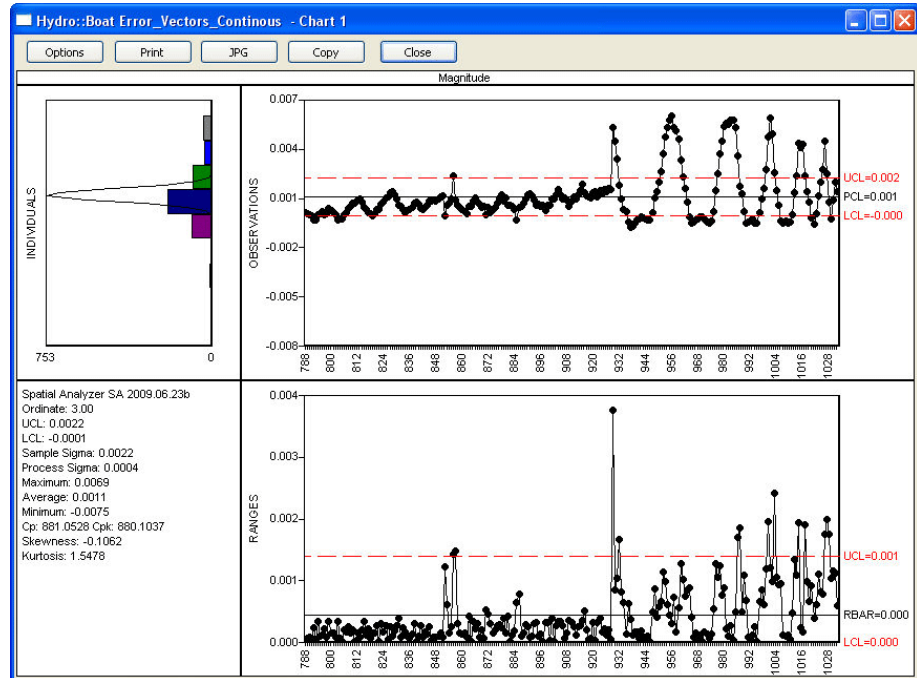


Figure 28-56. The resulting generated chart.

Chart Options

For more information on Statistical Process Control (SPC) and charting options refer to “[Charting](#)” on page 750.

Vector Construction Options

By Clicking Surfaces

Creates a vector group from clicking on surface face. These vectors are of a fixed magnitude but will use the surface normal direction.

From Existing Vectors

Creates a new vector group from selection of individual vectors.

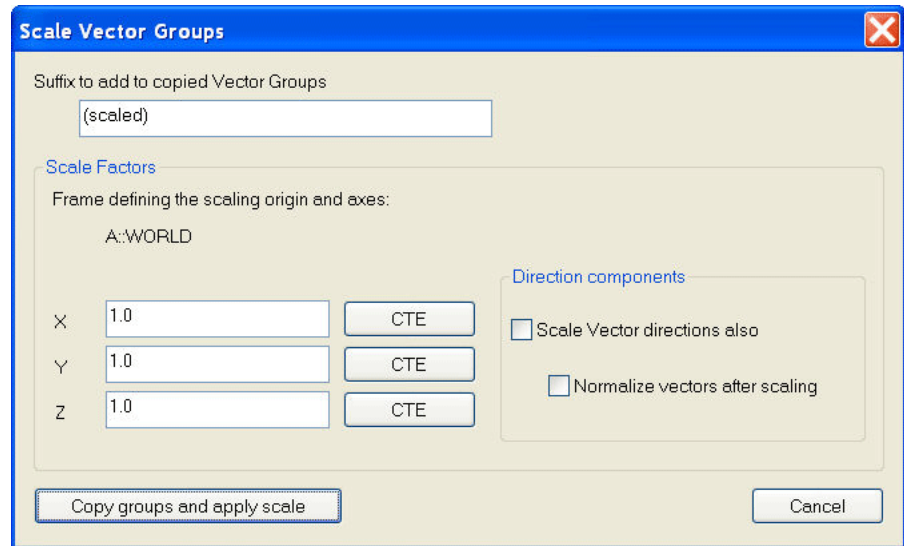
From Existing Vector Groups

Creates a new vector group from selection of existing vector groups.

Scale Existing Vector Groups

Creates a new vector group scaled about the selected frame. User may select a scale factor for each component of the vectors ([Figure 28-92](#)).

Figure 28-57. The Scale Vector Groups dialog.



Copy Vector Groups Excluding Obscured Vectors

Creates new vector group only coping visible vectors. Vectors blocked from current point of view are not copied.

Area Profile Check

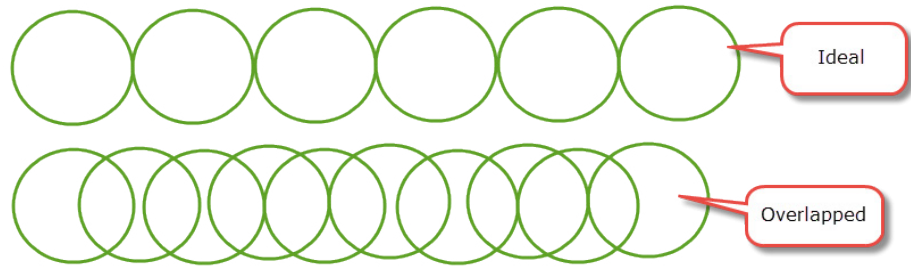
Area Profile check builds a new vector group to display the relative difference between the evaluated vector group(s) and the reference vectors with a specified area (**Construct>Vector Group>Area Profile Check**).

The Area Profile check has several inputs:

- **Reference Vectors.** This can be a grid of vectors across the model which designate the centroid of an evaluation region. These reference vectors are used to define each evaluation region and they also define the base magnitude for comparison within each region.
- **Vector Group(s).** The vector group or groups to be evaluated.
- **Radius to check.** Defines the distance from the reference vector used for each evaluation in selecting surrounding vectors to evaluate.
- **Area Tolerance.** The tolerance to use for the evaluation (which will be applied bi-laterally) and is used for vector colorization (go/no go).

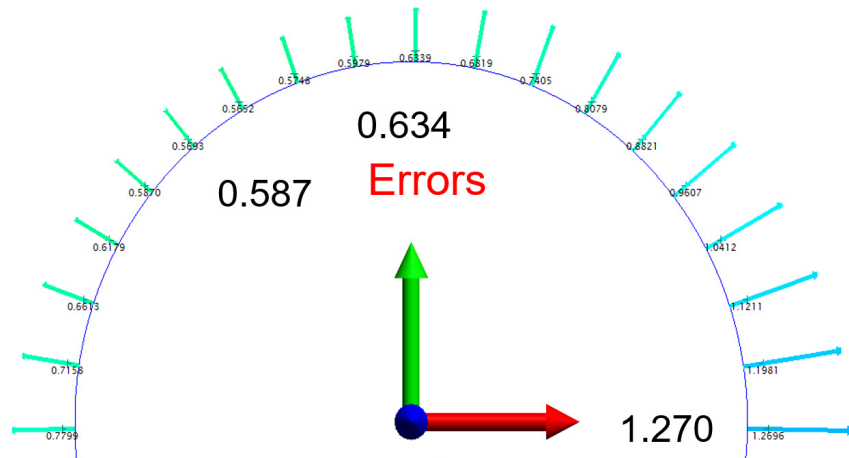
Reference Vectors can be simply sub-sampled or built yourself. The spacing of these reference vectors define the evaluation coverage. With more gaps, regions will be missed, however, with more overlap you end up with redundant data and a slower evaluation (**Figure 28-58**).

Figure 28-58. Reference Vector Distribution and Evaluation Overlap



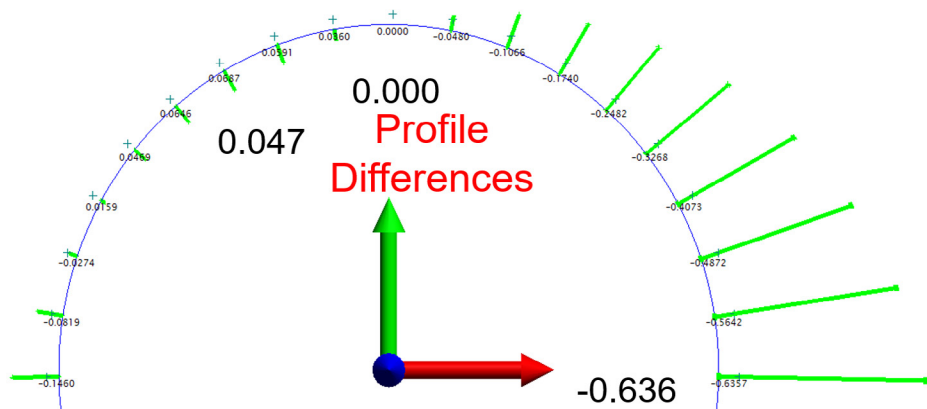
As an example, if you wish to evaluate the area profile at 0.005" in every 5" segment using the following Reference Vectors (Figure 28-70):

Figure 28-59. Reference Vectors



This Area Profile Check would produce the following result (Figure 28-69):

Figure 28-60. Area profile Vector Group Results



The evaluation process is as follows:

1. For each reference vector a list of vectors is created that fall within the evaluation radius.
2. For each vector in the list a new vector is created with the same position and orientation. The magnitude of this new vector is equal to vector magnitude minus the reference vector magnitude.

3. The color (Red/blue out of tolerance, or Green in tolerance) is set based upon your specified tolerance.

The resulting vector group shows the difference between the reference vector magnitude and the returned vector for each evaluated region and can be useful in evaluating the relative change in magnitude within a region.

From Point Measurement Probing Directions

Creates a vector group representing the probing direction stored for each selected point.

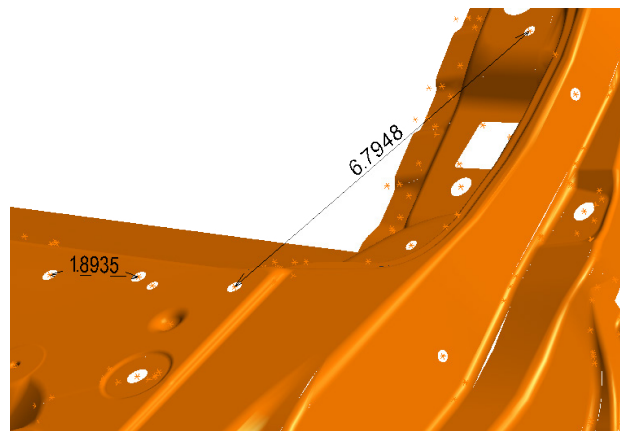
From Polygonized Surface Normals

Creates a vector group with a vector normal to each face of the polygonized surface.

Dimensions

Dimensions provide a simple method to annotate the graphical view with both linear and angular dimensions between any combination of points and objects (Figure 28-61). They are dynamic entities in that both the numeric values and the leader lines update in real time as the points and objects move in the job file. This allows you to set up dimensions in advance and see changes graphically as fits or alignments are updated with each new measurement.

Figure 28-61. Dimensions between hole centers.



Standardized dimensional analysis is a separate process performed using GD&T checks. For more information on dimensional checks see “Creating GD&T Annotations” on page 612.

Types of Dimensions

There are several different types of dimensions that you can create:

- **Point to Point.** Shows the linear distance between two points.
- **Point to Object.** Shows the linear distance between a point

and any object.

- **Object to Object.** Shows the linear distance or angle between any two objects.
- **Diameter/Radius.** Shows the diameter or radius of a circle or cylinder.

Both Point to Point and Point to Object dimensions can also be extracted directly from a point cloud.

Working With Dimensions

To Create a Dimension:

Dimensions can be built in one of the following ways:

- From the **Query>Make Dimension** menu, select the desired dimension type. Select the appropriate points or objects to dimension. A new dimension will appear in the graphical view.
- When the **Query Results** dialog is displayed as the result of an appropriate query, check the **Create Dimension** checkbox (Figure 28-62):

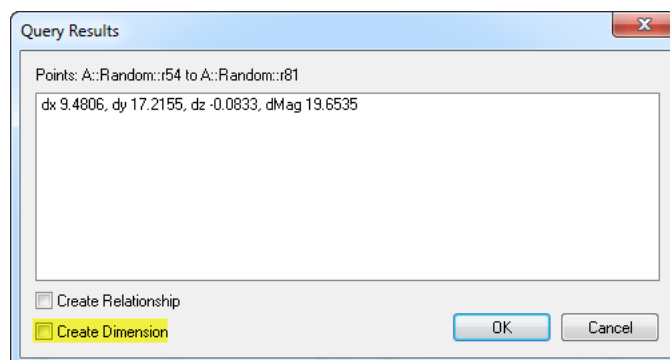


Figure 28-62. The *Create Dimension* checkbox in the *Query Results* dialog.

- Select a dimension configuration from the drop-down on the Reporting tab of the ribbon bar. The icons provide the ability to pre-configure dimension to display as desired depending on the icon selection.

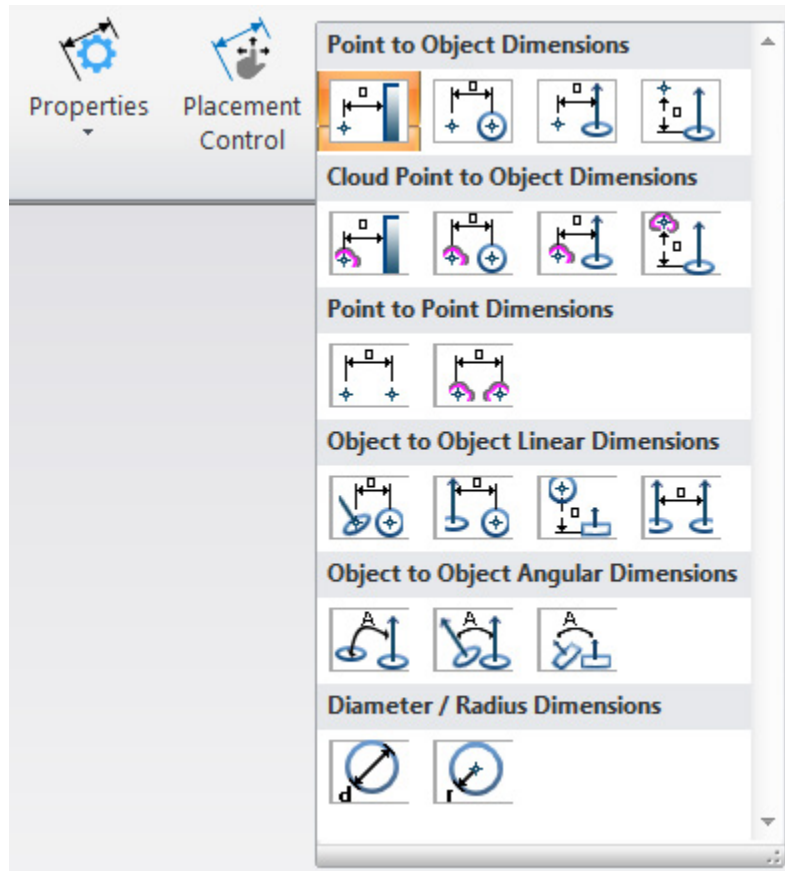


Figure 28-63. Icons for Pre-configured Dimension Construction

Properties Common to all Dimensions

Default Visual Display Settings

A set of default **Dimension Settings** are available through the **Users Options** on the display tab. These default settings will be used any time a new dimension is created and include the following properties:

Dimension Default Colors

- Dimension Default Color.** Dimension color can be set independently as needed for the text, leader and arrow head. The default colors are defined within the Users Options (Figure 28-64).

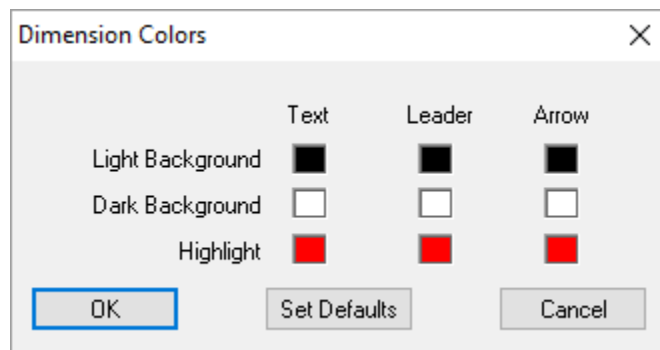


Figure 28-64. User Dimension Color Options

Text Settings

Dimensions can be oriented based on their leader lines, or can be set to always face the screen:

- **Text Size.** Controls the height of the displayed dimension.
- **Screen.** The dimension always faces the screen.
- **Along.** The dimension is oriented along the leader line.
- **Lateral Offset.** This value allows the dimension to be shifted laterally relative to its axis on the XY plane of the reference frame.
- **Offset Along Dimension Line.** This value allows the user to shift the placement of the text along the dimension line and can be entered as either a positive or negative number.

To Interactively Change the Graphical Placement of Multiple Dimensions at Once:

- Right-click the *Dimensions* category in the tree and select **Dimension Cursor Operations** to open the dialog. While this dialog is open dimensions can be positioned graphically using the keyboard shortcuts listed (Figure 28-65).

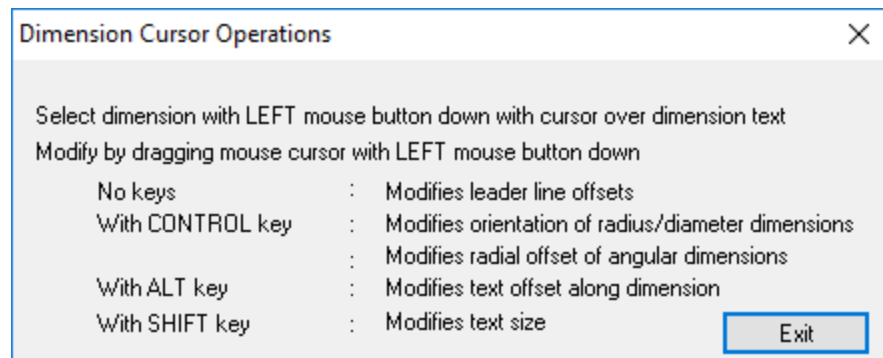


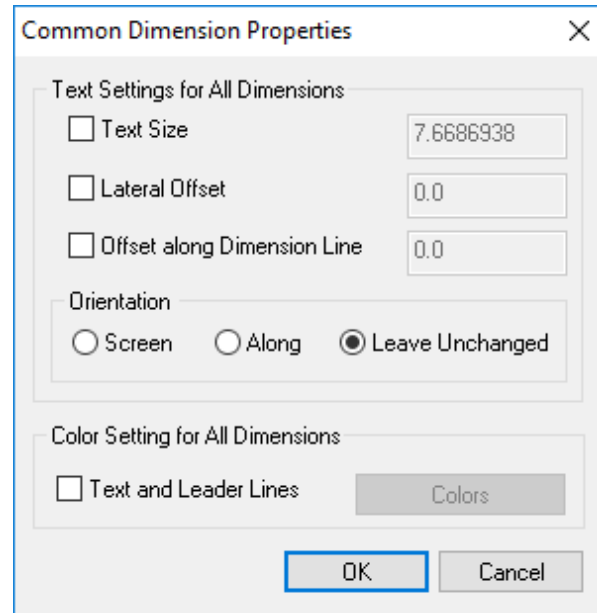
Figure 28-65. Dimension Cursor Operations

To Change Settings for All Dimensions in a single Collection:

All dimensions in a collection have a set of common properties which can be adjusted at one time using the option within the dimension category right-click menu. These controls are ideal for editing text size and placement and include the following:

- Right-click the *Dimensions* category in the tree > **Dimensions** and select **Set Common Properties** to open the *Common Dimension Properties* dialog (Figure 28-66).

Figure 28-66. Set Common Properties Dialog



To Change a Single Dimension's Properties:

- Double-click the dimension in the tree, or
- Right-click the dimension in the tree or graphical view and select **Properties** (see “Type Specific Dimension Properties” on page 730).

To Apply settings from one Dimension to a selection of others:

An **Apply to Selected Dimension** button is available in each dimension properties dialog. When selected you will first be prompted to select the dimensions you wish to consider. This will show a filtered list with only applicable dimensions of the same type.

You will then see an options dialog which allows you to choose and preview the dimension properties that you are applying to the selected dimensions .

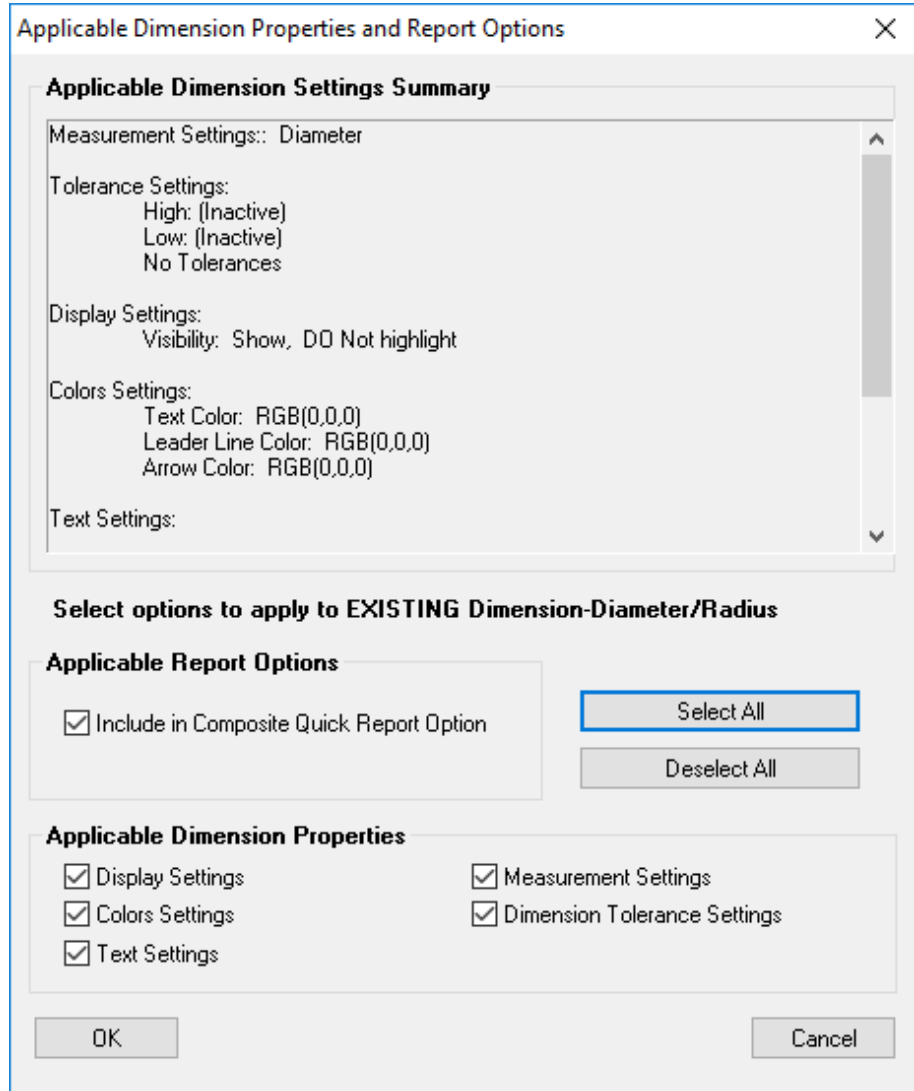


Figure 28-67. Apply to Selected Dimensions Optional Controls

Cloud Point Mode

Dimensions can be created directly from point clouds by checking the optional **Cloud Point Mode** check box. This enables Point to Point and Point to Objects dimensions to be built directly from point cloud data. Because of the inherent noise in cloud data, the user click point selection is used to define a volume or Voxel and the most representative point in that volume is then converted into a point and used for the dimension placement. This voxel size is also user controllable.

Type Specific Dimension Properties

The dimension type specific properties are listed below.

Point to Point Dimensions

These dimension types depict the distance between two points ([Figure 28-68](#)).

Figure 28-68. The properties of a Point to Point dimension.

In the *Dimension Measurement* section, each point has an offset setting:

- **No Offset.** Dimension from the target or point's center.
- **Add Target Offset.** Adds the target's offset to the dimension. This simulates an inside caliper (as if you'd measured the inside of a box).
- **Subtract Target Offset.** Subtracts the target's offset from the dimension. This simulates an outside caliper (as if you'd measured the outside of a box).
- **Tolerance.** A single tolerance value can be set per dimension which can be displayed either as a nominal +/- value or as a range. This setting will influence the display and the report table.
- **Dimension Notes.** Notes can be added to the Dimension as trailing text.

The other major advantage to point to point dimensions is the ability

to isolate reported to a single axis component or 2D component:

- **Dimension Components.** Dimensions can be set to report either the direct line deviation (Magnitude) or can report deviations along a single or set of axes (X,Y,Z) relative to the Dimension Reference Frame. Simply check the components you wish the dimension to consider.

You can also control the display orientation for the dimension, as described in “[Properties Common to all Dimensions](#)” on page 727.

Point to Object Dimensions

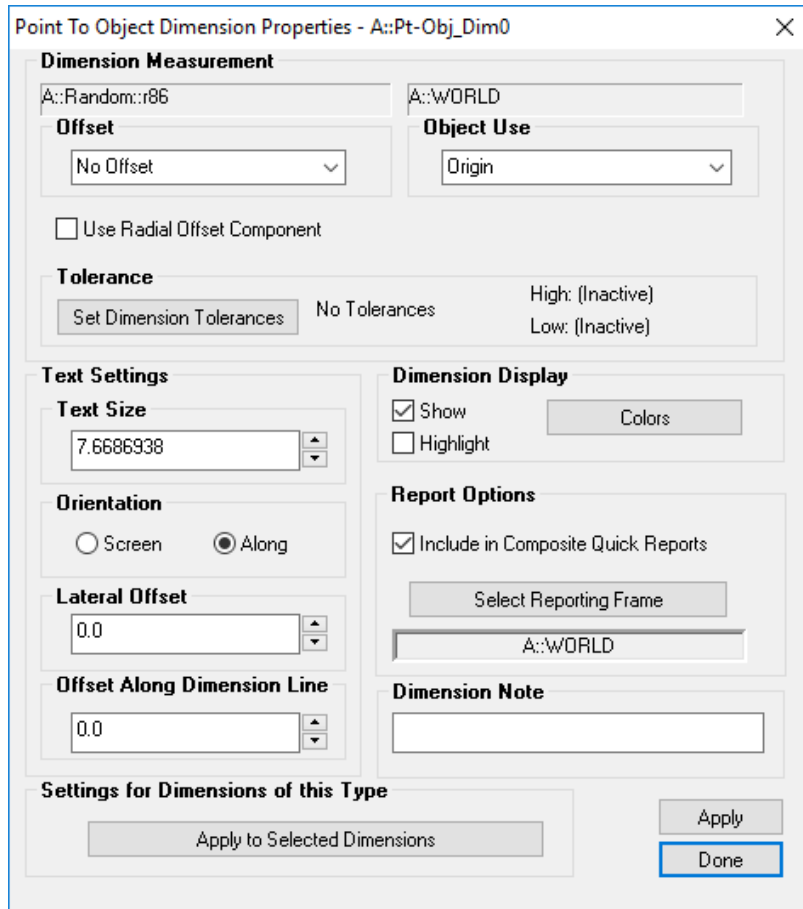
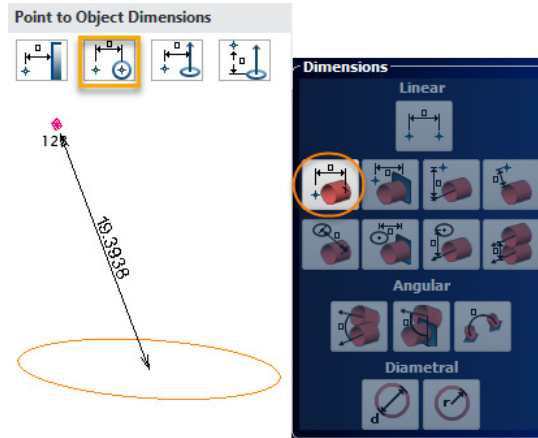


Figure 28-69. Properties of a Point to Object dimension.

In the *Dimension Measurement* section, the point offset controls are the same as a point to point dimension with addition of an option to *Use Radial Offset Component*. If enabled, the radial component of the measurement offset will be used for compensation instead of the planar component. This is useful, for instance, when you are measuring the distance from the closest point in a hole to a surface edge and are acquiring the hole measurement with a pin nest. On the other end of the dimension, the object can have one of four settings:

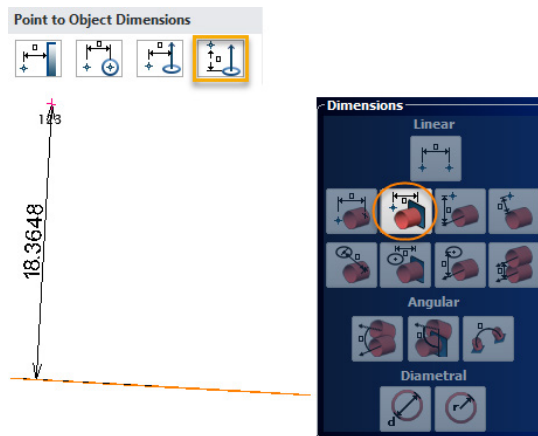
- **Origin.** Distance to the object’s internal origin (Figure 28-70). For instance, the center of a circle or plane.

Figure 28-70. Point to Origin.



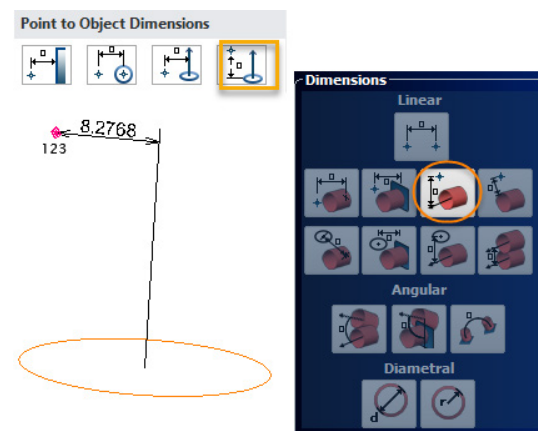
- **Plane.** Distance to the plane of the object, if applicable (Figure 28-9).

Figure 28-71. Point to XY Plane.



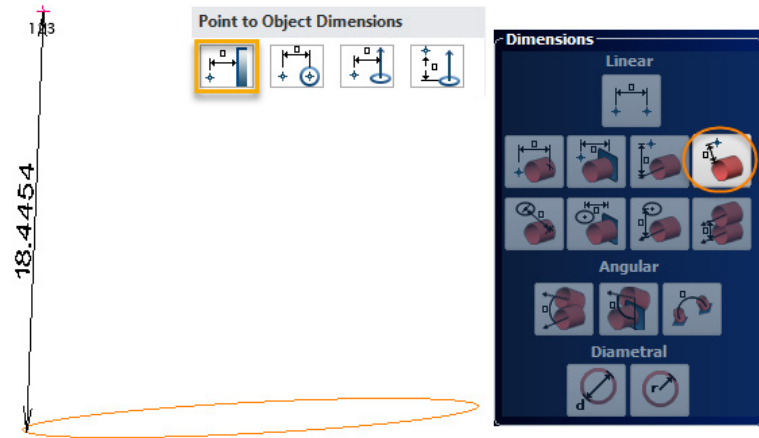
- **Axis.** Distance to the object's projected (extended) axis (Figure 28-72).

Figure 28-72. Point to Axis.



- **Surface.** Show the distance to the closest point on the object's surface/line/curve (Figure 28-73).

Figure 28-73. Point to Surface.



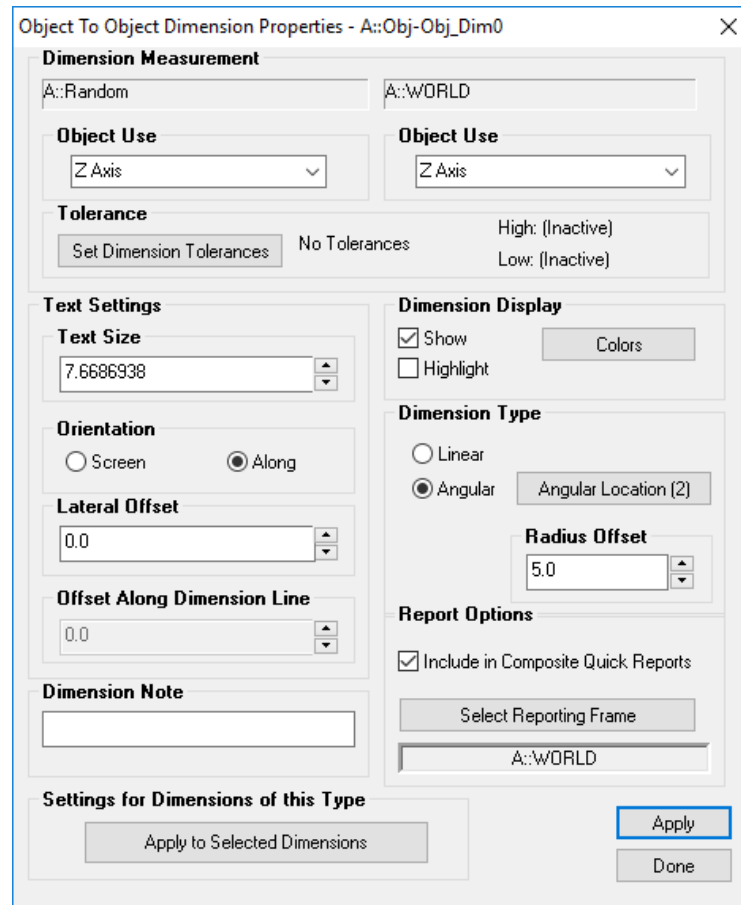
You can control the display orientation of the dimension as described in [“Properties Common to all Dimensions”](#) on page 727.

2D Point to Object Angular Dimensions

Point to Object dimensions report the minimum distance between a point and an object unless a reporting frame is asserted. In that case the dimension is projected to the XY plane of the reporting frame. This provides a means to build a 2D dimension between a point and an objects.


Object to Object Dimensions

Figure 28-74. Object to Object Dimension properties.



The dimension can be tied to one of three features of each object:

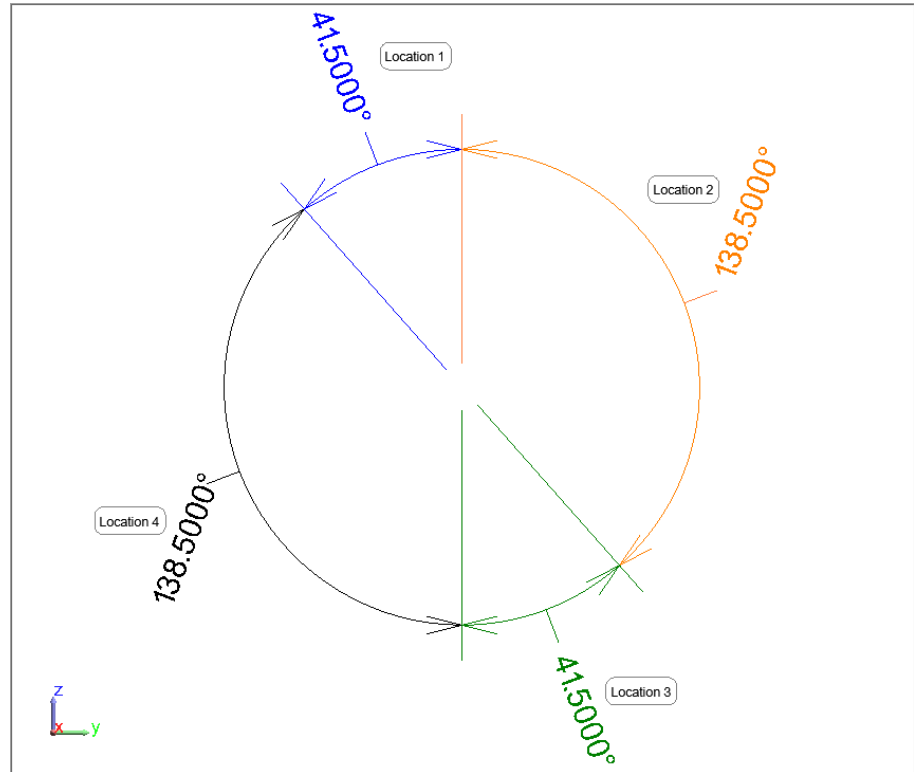
- **Origin.** The distance from the object's internal origin. This feature setting for the dimension does not allow angular dimensioning, therefore the **Angular** option becomes grayed out.
- **Z Axis.** The distance to the object's projected Z axis (if **Linear** is selected), or the angle to the object's internal Z axis (if **Angular** is selected).
- **XY Plane.** The distance to the object's XY plane (if **Linear** is selected), or the angle to the object's XY plane (if **Angular** is selected).

 **Note:** The internal Z axis of an object is typically the object's normal (for planes, circles, etc.) the Z axis (for frames) or the line direction (for lines).

Two settings particular to angular dimensioning dictate where the angular dimension appears:

- **Dimension Radius.** The distance of the angular dimension from the mutual intersection of the two objects.
- **Angular Location.** This button cycles through the four quadrant representations of the given angle, as illustrated in [Figure 28-26](#).

Figure 28-75. Angular quadrants



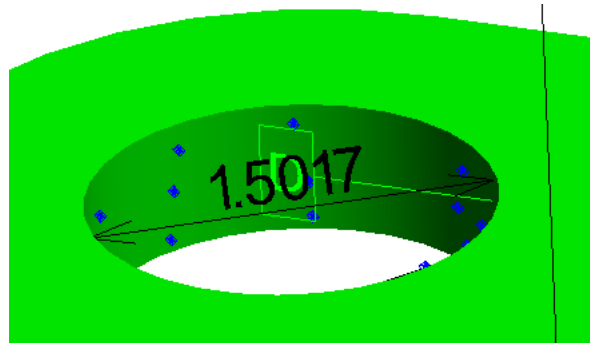
You can control the display orientation of the dimension as described in “[Properties Common to all Dimensions](#)” on page 727.

2D Object to Object Angular Dimensions

Angular dimensions between two objects by default report the minimum angle between the two include objects unless a reporting frame is asserted. In that case the dimension is projected to the XY plane of the reporting frame. This provides a means to build a 2D dimension between two objects.

Diameter / Radius Dimensions

Figure 28-76. A diametrical dimension.



Diameter/Radius dimensions are ideal when the diameter or radius of a circle, cylinder, or sphere needs to be dimensioned ([Figure 28-76](#)).

The diameter/radius dimensions are shown in [Figure 28-77](#):

Figure 28-77. Properties of a Diameter/Radius dimension.

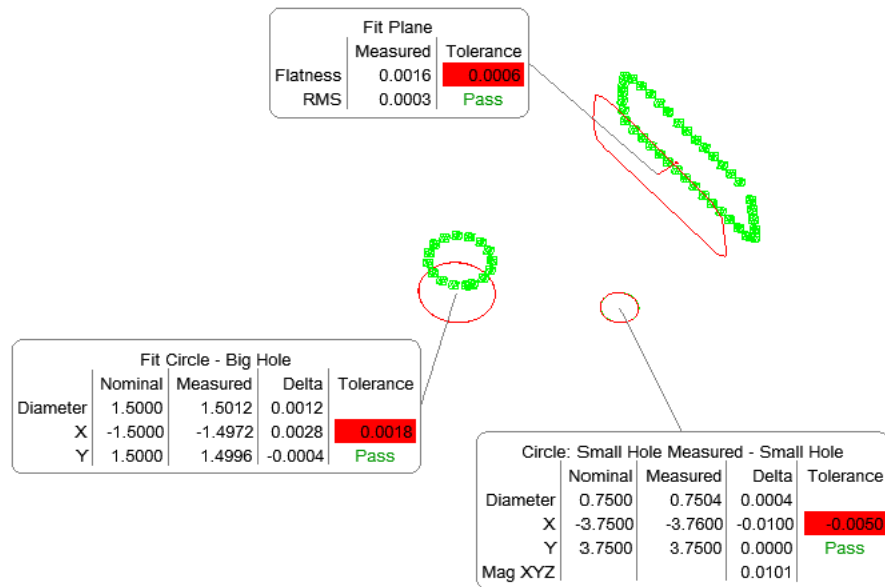
- **Display Angle.** Specifies the angle around the circle to which the dimension is oriented.
- **Dimension Radial Offset.** For angular dimensions this value controls the vertical shift of the text inward or outward from the apex of the dimension.

You can control the display orientation of the dimension as described in [“Properties Common to all Dimensions”](#) on page 727.

Callout Views

Callouts offer a quick and easy way to annotate the graphical 3D view in SpatialAnalyzer with screen oriented labels that are dynamic and easy to read. Callout views offer a wide range of control and flexibility to define a perspective on a job file and isolate the display of the objects of interest. These can be used interactively for inspection and define a reportable image that can be added directly to a report. A simple example callout is shown in [Figure 28-78](#).

Figure 28-78. Geometry relationship callouts in the graphical view.



Creating a Callout View

Callouts are typically composed of two functional parts:

- **Callout View.** Which defines the perspective used to display the callout and controls any additional visibility settings.
- **Callouts.** These are the actual annotations added to the graphics that display information about the particular objects of interest. Multiple callouts can be added to the same Callout View.

Creating a Callout View


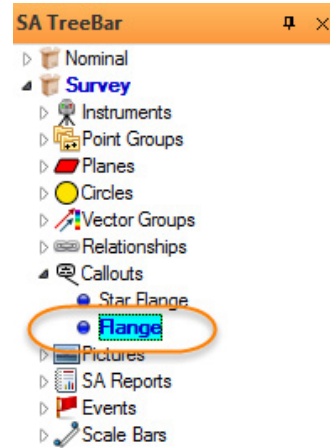
To create a new callout view simply press the  button in the Main Toolbar or use the add icon on the Reporting tab of the ribbon. This will create and activate a callout view in the tree. Double-Clicking a callout view in the tree will activate or deactivate (turn on and off) this callout.

Figure 28-79. An activated callout as it appears in the tree.



Callout View Properties

Enabling Callout Display

Single click in the tree updates the callout view in the report bar but not in the view, while double click in the tree should set the callout as active for the job file, asserts the saved view perspective and visual controls defined in the view.

Callouts can also be tied directly to GR-Features as part of an inspection process such that the desired perspective and labels are displayed when trapping begins on the feature (see “[Visual Controls](#)” on page 420).

Default Callout Properties

The default configuration for newly created callout views is defined within the **Users Options**. On the **Display** Tab in the **Default Settings** section is a **Callouts** button which opens a default callout view configuration. Edits to this control will modify the newly created callout view properties for that job. This configuration can also be saved as part of a template file.

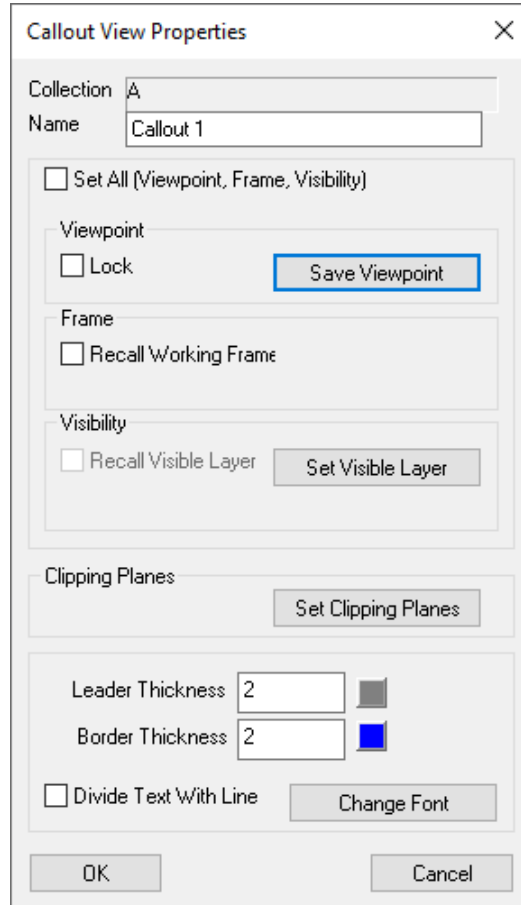


Figure 28-80. Callout view properties.

Much like composing a photograph, the callout view defines the perspective on the job. In its simplest form, it sets the view orientation and zoom depth on the 3D objects shown in the graphics.

Save Viewpoint

The **Save Viewpoint** button provides the operator the ability to redefine the saved view orientation and zoom depth used to display the features of interest. The initial view is saved when the callout is created and can be redefined at any time.

Lock Viewpoint

The *Lock* check box allows a user to lock the current view of the callout so that the graphics cannot be rotated. This option also is helpful in more precisely defining the location of callouts relative to 3D objects. If you are working on monitors with different aspect ratios this option will ensure callouts are placed precisely on the viewing plane regardless of the proportions of the graphics. All other modes maintain the placement of callouts with respect to the placement on the screen.

Recall Working Frame

The *Recall Working Frame* checkbox allows the callout to set the work-

ing frame to a specified frame when the callout is activated. This can be helpful when reporting a section of the job relative to a particular coordinate frame. However, changing working frame will require a job recompute which can slow the display process.

Recall Visible Layers

The goal of the *Recall Visible Layer* option is to hide everything but the specific items you want to display when the callout is enabled. This visibility is defined as a layer which is named and saved in Layer Manager (see “[The Layer Manager](#)” on page 742).

The idea is to have many callouts each with an independent layer or list of displayed items. So generally you will have 1 layer per callout, but you can also re-use an existing layer so the visibility does not need to be edited for all items in the job (with hundreds of objects, this can be handy) before adding a callout visibility layer. The idea is to select existing layers by name.

The *Recall visible layers* check box is used to turn on or off this mode.

- With the check box off... the visible items are not changed
- With the check box on... the visible items in the referenced layer will be displayed.

The *Add a new Layer* button should build a new layer if a layer does not exist yet for the callout, saving the list of currently displayed items in the graphics when the layer is built.

The *Update Current Layer* button should be used to update the currently selected layer with any changes in visibility for the objects.

The Layer name field is used to identify the name of a layer to be recalled. This allows an existing layer to be referenced. But if a layer of that name does not exist it should not be applied until it is defined first.

Annotation Display Properties

The final section of the callout view allows the default graphic display options to be set for the callout view. This will set the default properties for each new callout added to that particular callout view.

Enable Scalable Callouts

An option exists within the right-click menu of the Callout category in the tree to **Enable Scalable Callouts**. This option makes it possible to adjust the overall display size for all the callouts within a callout view. To do so right-click on the callout category in the tree and choose **Enable Zoomable Callouts**. Then within an active callout view, hover over one of the callouts and hold Ctr+Shift and use the mouse wheel to adjust the callout size as needed.

The Layer Manager

Often, when working with complex files, you may find yourself showing and hiding sets of related objects in the tree. For instance, if you have a CAD model and measurements of several sub-systems on an assembly, you may wish to view only a given sub-system in the graphical view.

For this purpose, you can create *layers*. You can assign objects to certain layers, then show or hide layers. This serves as a very quick way to show and hide multiple objects at once.

Every file contains a **Default** layer that cannot be deleted.

To Show the Layer Manager:

- From the menu, select **View>Layer Manager**. The *Layer Manager* dialog appears (Figure 28-81).

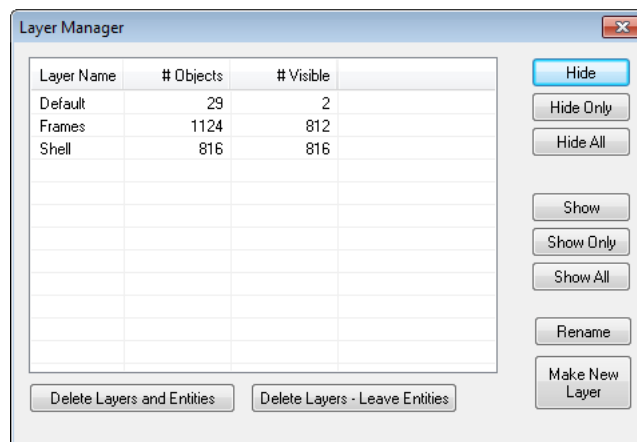


Figure 28-81. The Layer Manager.

To Define a New Layer:

- In the SA file, show only those objects you'd like to appear on the layer.
- In the *Layer Manager*, click the **Make New Layer** button.
- Type a new name for the layer, then click **OK**.
- The *Layer Manager* will show the new layer and indicate the number of objects on the layer, and the number of objects currently visible.

To Show or Hide Layers:

- In the *Layer Manager*, select one or more layers to hide or show.
- Click one of the following buttons:
 - Hide**. Hides the selected layer. Objects already hidden remain hidden.
 - Hide Only**. Hides *only* the selected layer. Objects on other lay-

ers are shown.

- **Hide All.** Hides *all* layers. The only remaining visible objects will be those that don't belong on a layer.
- **Show.** Shows the selected layer in addition to the objects already visible.
- **Show Only.** Shows *only* the contents of the layer, hiding all other objects.
- **Show All.** Shows *all* layers. The only remaining hidden objects will be those that don't belong on a layer.

To Rename a Layer:

1. In the *Layer Manager*, select the layer to rename, then click the **Rename** button.
2. Type a new name for the layer, then click **OK**.

To Delete a Layer:

1. In the *Layer Manager*, select one or more layers to delete.
2. Click one of the following buttons:
 - **Delete Layers and Entities.** Deletes the layers and any objects on those layers.
 - **Delete Layers - Leave Entities.** Deletes the layers but any objects on those layers remain.

To Add or Remove an Object From a Layer:

1. Right-click on the object in the tree or graphical view and select **Layer Info** from the context menu.
2. The *Layer List* dialog appears, showing the layers that currently contain the object.

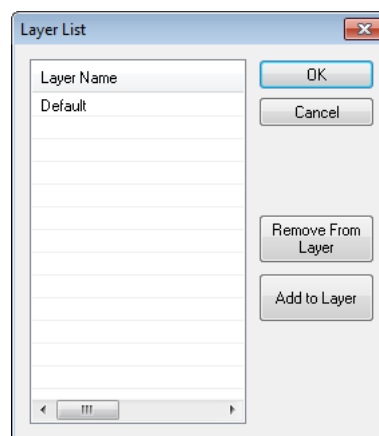


Figure 28-82. The *Layer List* dialog. This object is only on the default layer.

3. Do one of the following:
 - To remove the object from a layer, select the layer and click **Remove From Layer**.

- To add the object to one or more layers, click the **Add to Layer** button. In the **Add to Layers** dialog, select the layers you'd like to add the object to, then click **OK**.

Adding Callouts to a Callout View

Callouts can be added to just about any object displayed in the graphics in SA. To add a callout first define a callout view or set an existing callout as active and then select a callout type from the drop down list on the Reporting tab of the ribbon menu. It can also be found in the Main Toolbar or from the icons in the toolkit in the classic menus. Add callouts to points or objects by clicking on them in either the graphics or in the tree bar (Figure 28-83).

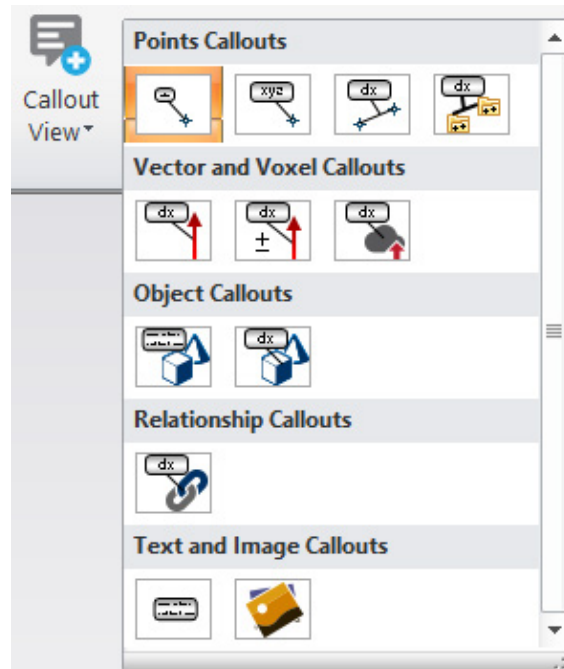


Figure 28-83. Adding Callouts

Callout Styling

Default styling settings for each newly created callout are defined in the callout view. However each callout can be modified independently. These settings include:

- **Leader Thickness and Color.** Which controls the way a leader line is drawn graphically
- **Border Thickness and Color.** Which defines the thickness and color of the border around the callout. This is not an option for vector callouts when the vector color is used.
- **Font.** The font used for the callout
- **Divide Text with Lines.** This option can be used to separate

the text of the callout more clearly.

Callout Anchor Point and Placement

The position of a callout can be edited by clicking on the callout and moving it while holding down the Left-mouse button. The leader line anchor point can also be adjusted on a reference object through the right-click menu option.

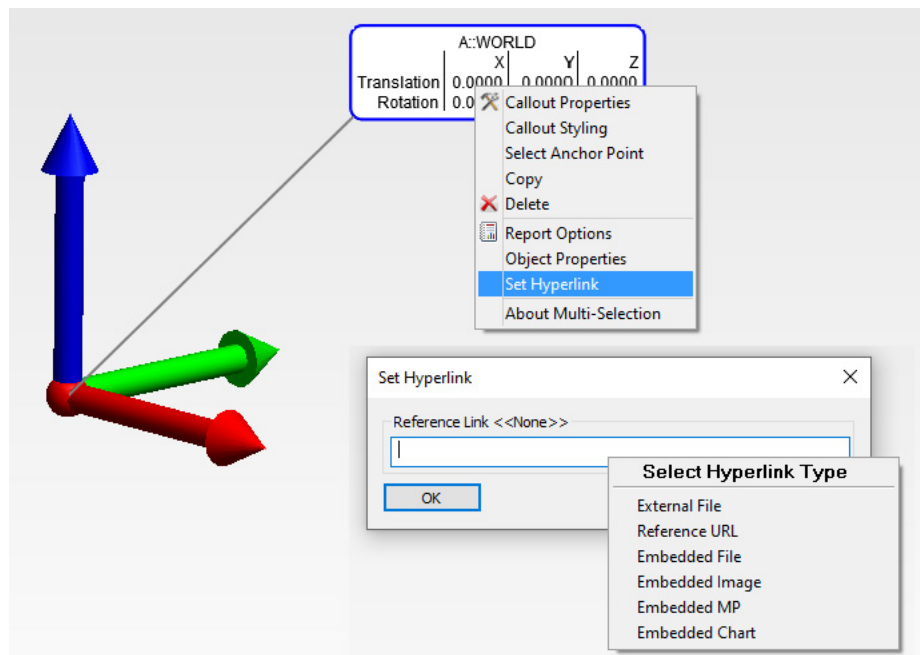
Callouts as Hyperlinks

Any callout can be set as a hyperlink. This provides the ability to add a graphic reference with a link to a pdf document for example. Or an image with the ability to run an MP.

To add a hyperlink to a callout:

1. Right-click on the Callout and select *Set Hyperlink*
2. Right-click on the reference line to *Select Hyperlink Type*. This defines the type of hyperlink to use and the reference to file of interest.

Figure 28-84. Setting a Hyperlink for a callout.



Once a hyperlink is set for a callout, the presence of a hyper link can be detected by hovering over the callout which will result in a tool tip display of an active link. To activate the callout's hyperlink a user can double-click on the callout in the graphics.

Note that any reference files will need to have an extension that is associated with an application that can be used to display the file contents. For URLs, it will be necessary for MS Edge to have been installed.

Types of Callouts

Text

A text callout provides the ability to add a simple text box to the graphic view. This can be helpful both in annotation reporting images and in guided inspections. This can be expanded by Right-click within the text box to add dynamic reporting attributes in-line with user defined text.

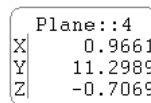
Point Labels

Point Label callouts provides a quick way to add a name label to a specific point or set of points. These callouts can be converted directly to Point Coordinate callouts if desired at any time.

Point Coordinates

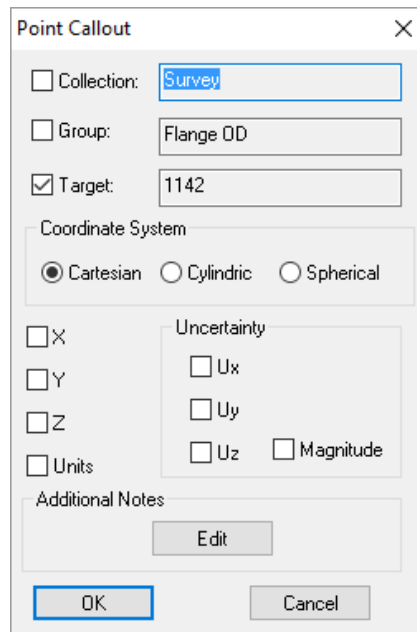
Point Coordinate callouts are pre-configured to provide basic point name and position information (Figure 28-85).

Figure 28-85. A point coordinate callout.



Right-Click the Point Coordinate or Point Label callout to access the properties and display configuration for that callout type (Figure 28-86).

Figure 28-86. Point callout properties.



Point Comparison

A Point Comparison callout can be used like a dimension to display the difference in position between two selected points. Their deltas are computed by subtracting the first selected from the second selected. Right-Click the callout to access properties (Figure 28-87).

Figure 28-87. A point comparison callout.

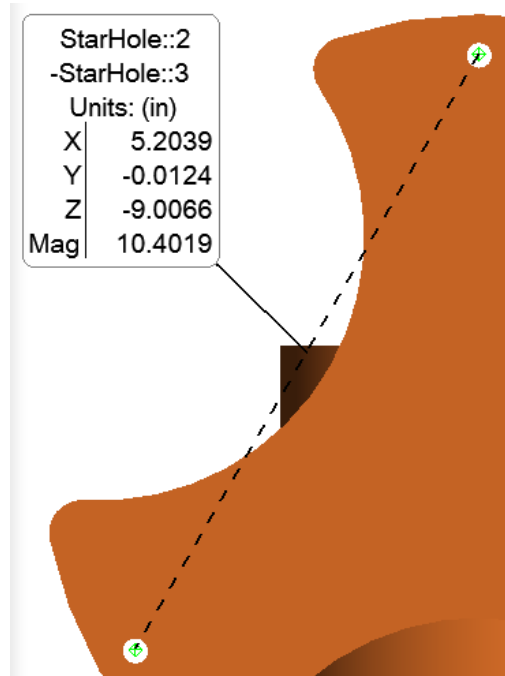


Figure 28-88. Properties for a point comparison callout.

Point Compare Callout

First Point

Collection: Survey

Group: StarHole

Target: 2

Show Coordinates

Second Point

Collection: Survey

Group: StarHole

Target: 3

Show Coordinates

Additional Field Comments

X

Y

Z

Mag

Units

Additional Notes

Edit

OK Cancel

Group Comparison

This callout option provides an efficient way to compare the corresponding point pairs in two selected groups. It adds individual point comparison callouts to each of the matched points in each group. Their deltas are computed by subtracting the first selected from the second selected. Right-Click the callout to access properties.

Vectors

Vector callouts annotate selected vectors with a vector group. They can be configured to display both the magnitude and tolerance information as needed (Figure 28-89), which can be configured through the right-click properties (Figure 28-90).

Min/Max Vector Group

In many cases users are only really interested in annotating the largest or smallest deviations displayed by a vector group. The Min/Max Vector Group callout provides a means to specify how many callout to add and have them added automatically to the vector group by picking any single vector in that group.

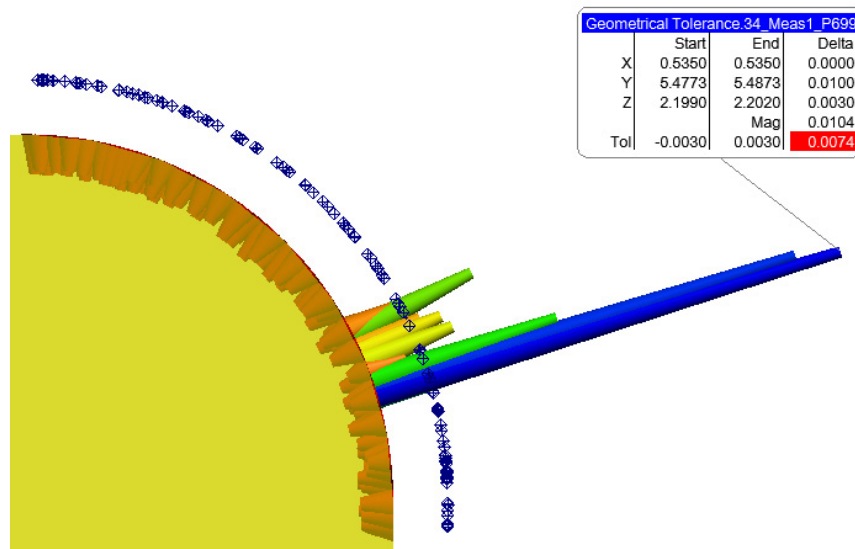


Figure 28-89. A vector callout.

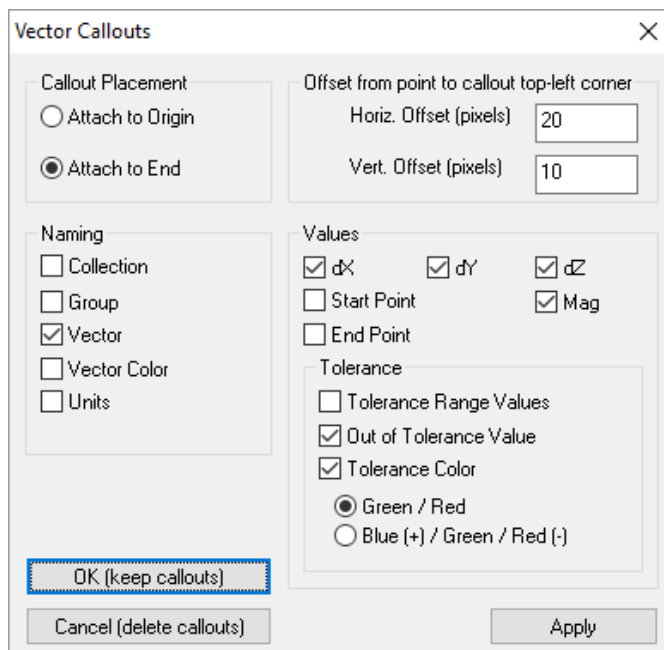


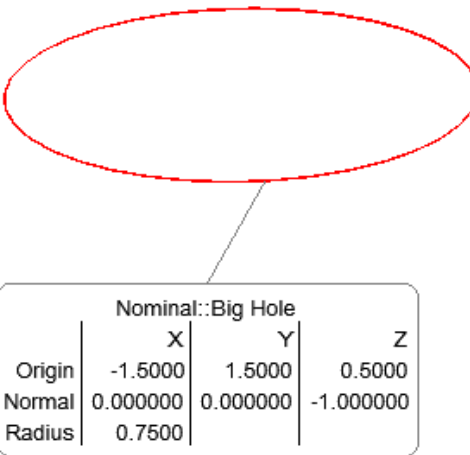
Figure 28-90. Vector callout properties.

Voxels

Voxels provide a means to identify the deviation displayed in Clouds to Objects relationship when using a voxel display. For more information refer to “[Voxel Cloud Display](#)” on page 558.

Object and Object Label

These annotations can be added to any object. Just like point labels the Object labels only list a name for the object but can be converted to full object callouts at any time. Annotates selected object properties ([Figure 28-91](#)). Right-Click the callout to access properties.

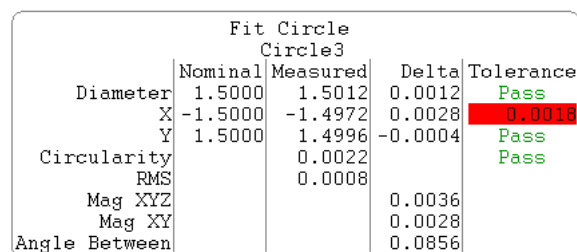


Nominal::Big Hole			
	X	Y	Z
Origin	-1.5000	1.5000	0.5000
Normal	0.000000	0.000000	-1.000000
Radius	0.7500		

Figure 28-91. An object callout.

Relationship

While relationships are not actually graphical entities, they can still be used in displaying key information about the objects or points within the job file. Callouts can be added to any relationship by selecting the relationship either through the tree or by using the F2 selection menu. An example Geometry relationship callout is shown in ([Figure 28-92](#)). Right-Click the callout to access the extensive relationship callout properties ([Figure 28-93](#)).



Fit Circle Circle3				
	Nominal	Measured	Delta	Tolerance
Diameter	1.5000	1.5012	0.0012	Pass
X	-1.5000	-1.4972	0.0028	0.0018
Y	1.5000	1.4996	-0.0004	Pass
Circularity		0.0022		Pass
RMS		0.0008		
Mag XYZ			0.0036	
Mag XY			0.0028	
Angle Between			0.0856	

Figure 28-92. A relationship callout.

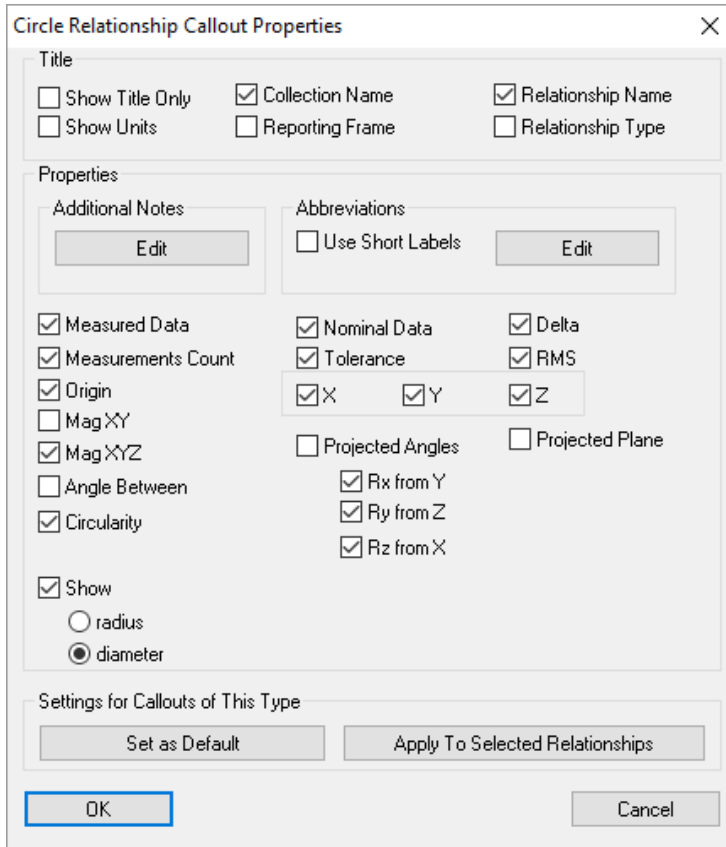


Figure 28-93. Relationship Callout properties

Charting

Charting is the most effective way to display relative differences between data points, particularly as they relate to sequential changes. SA offers basic chart generating capabilities and export options for more advanced charting applications.

Q-Das export is also available for more advanced analysis and processing capabilities. Q-DAS export has been implemented for appropriate relationships, GD&T checks, dimensions, and vector groups.

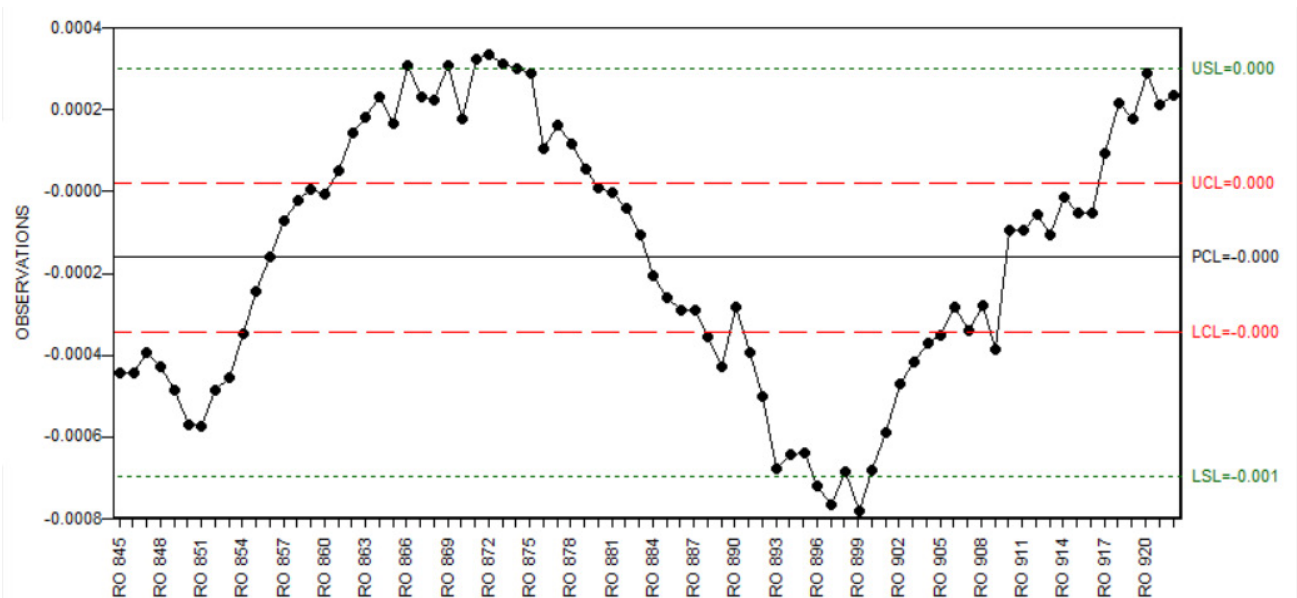


Figure 28-94. Example Chart

Available Charts

Vector Group Charting

The following chart types are available for vector groups (Figure 28-95):

- **Moving Range SPC Chart.** Moving Range charts are used to monitor individual values over time with respect to control limits for the individual values.
- **Bullseye Chart.** Bullseye Charts are used to plot two parameters relative to a specific goal.
- **Run Chart.** A Run Chart or run-sequence plot is a graph that displays observed data in a time sequence, providing a basic plot of a Primary and Auxiliary parameter.

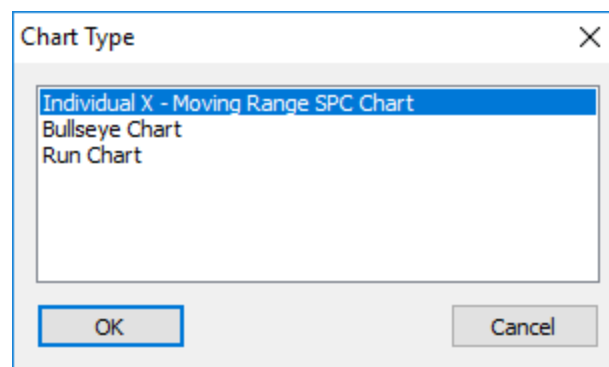


Figure 28-95. Chart Type Options

Feature SPC Charting

Moving Range SPC Charts can be added for the following features

(Figure 28-96):

- **Relationship Criteria.** Charting is available for Average, Max, Min, RMS and StdDev values.
- **Geometry Relationship (GR) Feature Criteria.** The options will depend on the geometry type and reflect the available criteria.
- **GD&T Feature Check Results.** Charts can be used to plotted changes in features checks results when multiple checks are made based upon a single annotation.
- **Dimension Magnitudes.** Charts can be used to plotted changes in dimension values

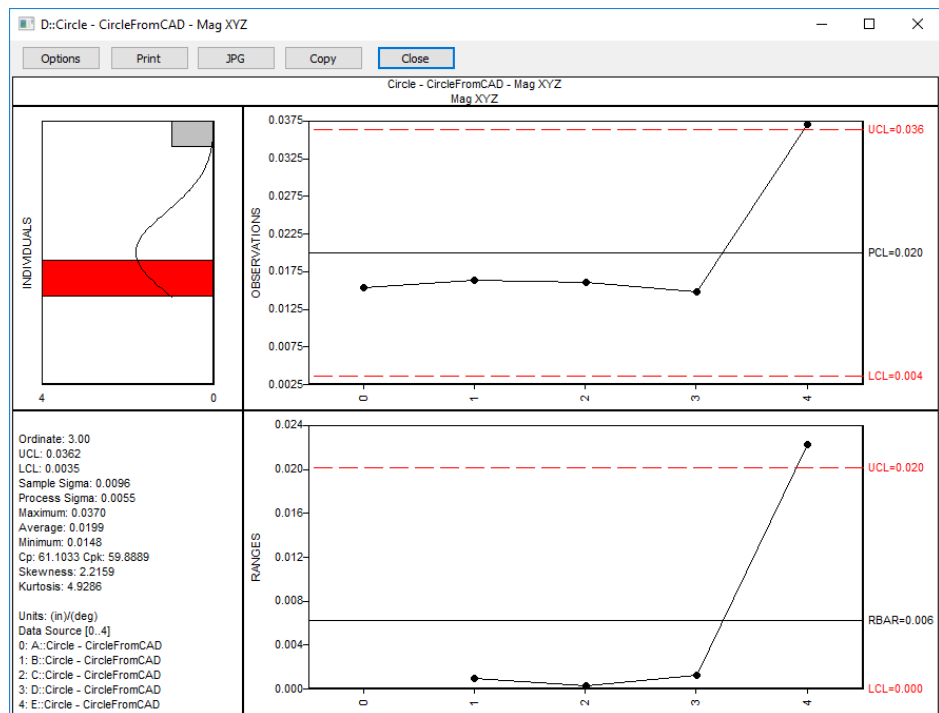


Figure 28-96. Example circle position SPC chart

Chart Plot Region Controls:

The following basic chart configurations can be made (Figure 28-97):

- **Titles.** Chart titles can be edited as needed.
- **Rainbow Coloring.** A red and green colorization can be applied to the chart back ground with respect to the control limits.
- **Control Limits.** A dotted line can be drawn on the chart to display the upper and lower control limits. This can be turned on and of and the Ordinate value adjusted (see *Ordinate* below).
- **Specification Limits.** The chart plotted area can be adjusted to display Observation values within the specified High and

Low limits.

- **Chart Range.** The plotted value index can be adjusted to display data within a specific range of values. Both a start and stop control index can be specified.

Chart Buttons

- **Print.** The Print button will export the current chart to your default printer.
- **JPG.** The JPG button will export the current chart as an image file saved in the selected directory.
- **Copy.** The Copy button will save an image of the current chart to the windows clipboard and can then be pasted in another document.

Right-Click Export to Excel

The right-click menu for a chart offers an Export to Excel option which will export a *.csv file containing the raw observation data used for the chart.

-

Figure 28-97. Run Chart Options

Chart Statistical Results

Ordinate

The Ordinate is a term which defines the y-coordinate or vertical of a point in a two-dimensional system. In SA charts this value is used to drive the placement of the upper and lower control limits and defines represents the Sigma value or standard deviation used.

Upper and Lower Control Limits (UCL & LCL)

These are the lines drawn above and below the process control line (PLC average or RBAR running average) in Range and Observation charts. They are computed from the available data and placed equidistant from the PCL line. It is formulated using the average deviation times the positive and negative standard deviation, which in this case is the Ordinate. Points plotted above or below the control limits are considered out of control.

Process Sigma

Process Sigma is a measure of the variation in a process relative to customer requirements. We measure defects on a scale of defects per million opportunities (DPMO). Any instance of failing to meet customer requirements is a defect, so a high number for DPMO is undesirable. This is implemented

Process sigma, as implemented within the context of SPC, is developed from the moving R average value -- it is NOT directly calculated from the data. The moving R values are the absolute value of the delta between adjacent observations. The average of these R values is R_{BAR} . Estimated process standard deviation = $R_{\text{BAR}} / 1.128$.

Upper and Lower Specification Limits (USL & LSL)

A value that represents the highest and lowest range in a variable typically these limits are set by a customer requirement.

Cp and Cpk

- **Cp – Process capability.** A simple and straightforward indicator of process capability
- **Cpk – Process capability index.** Adjustment of Cp for the effect of non-centered distribution. (measures how close you are to your target and how consistent you are to your average performance).

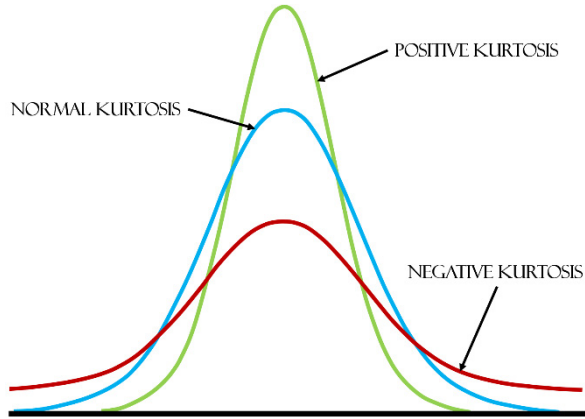
Use an analogy of hitting a target to help understand. If you hit the target in the same spot forming a good group this is a high Cp (PRECISION). Now imagine that this tight grouping had actually landed on the bullseye, you now have a high Cpk (ACCURACY).

Kurtosis

Kurtosis is a measure of the sharpness of the peak of a frequency-distribution curve. It is a descriptor of the shape of a probability distribution, which is used along with skewness. Kurtosis is sometimes confused with a measure of the peakedness of a distribution. However, kurtosis is a measure that describes the shape of a distribution's tails in relation to its overall shape.

Kurtosis quantifies whether the shape of the data distribution matches the Gaussian distribution. Gaussian distribution has a kurtosis of 0. A flatter distribution has a negative kurtosis. A distribution more peaked than a gaussian distribution has a positive kurtosis.

Figure 28-98. Kurtosis

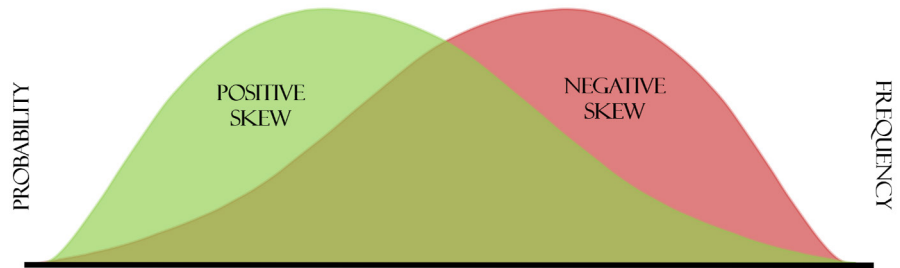


Skewness

Skewness is a term in statistics used to describe asymmetry from the normal distribution in a set of statistical data. Skewness can come in the form of negative skewness or positive skewness, depending on whether data points are skewed to the left and negative, or to the right and positive of the data average.


When data is skewed to the right, the mean and the median of the set are both greater than the mode. (Mean is the average, Median is the data center, and Mode is the number which occurs the most). Further, the mean is greater than the median in most cases. Conversely, when data is skewed to the left, the mean and the median are both less than the mode. In addition, as a rule, the mean is less than the median.

Figure 28-99. Skewness



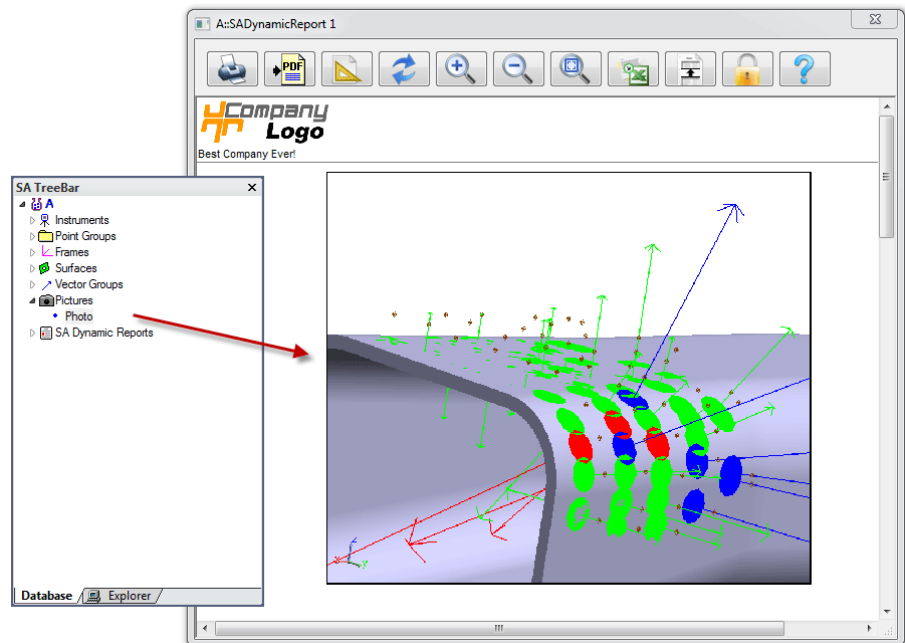
Photos

Pictures can also be added to SA and directly to SA reports. Drag-Drop photos into the graphics to save them in the tree for later use or directly into a report. They can also be imported by using the **File>Import** options.

Screen shots can be captured using the Capture button on the Reporting page .

The picture can be placed into the report by dragging, as with other items (Figure 28-100).

Figure 28-100. Dragging a picture into a report.

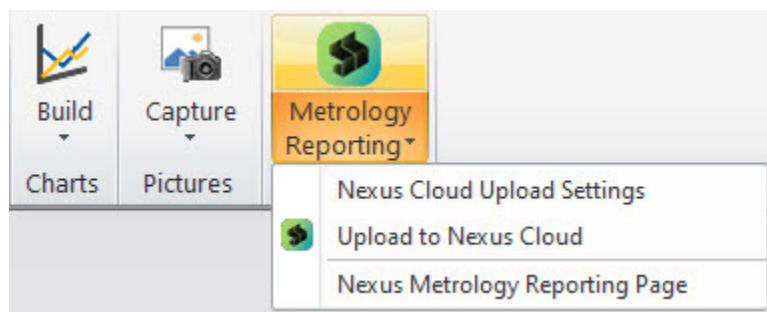


Once the picture is in the report it can be re-sized by dragging a corner of the image.

Metrology Reporting

A comprehensive cloud based solution to reporting is now available through the *Nexus > Metrology Reporting* options on the Reporting tab of the ribbon (Figure 28-101).

Figure 28-101. Metrology Reporting Controls used to send reports to Nexus.

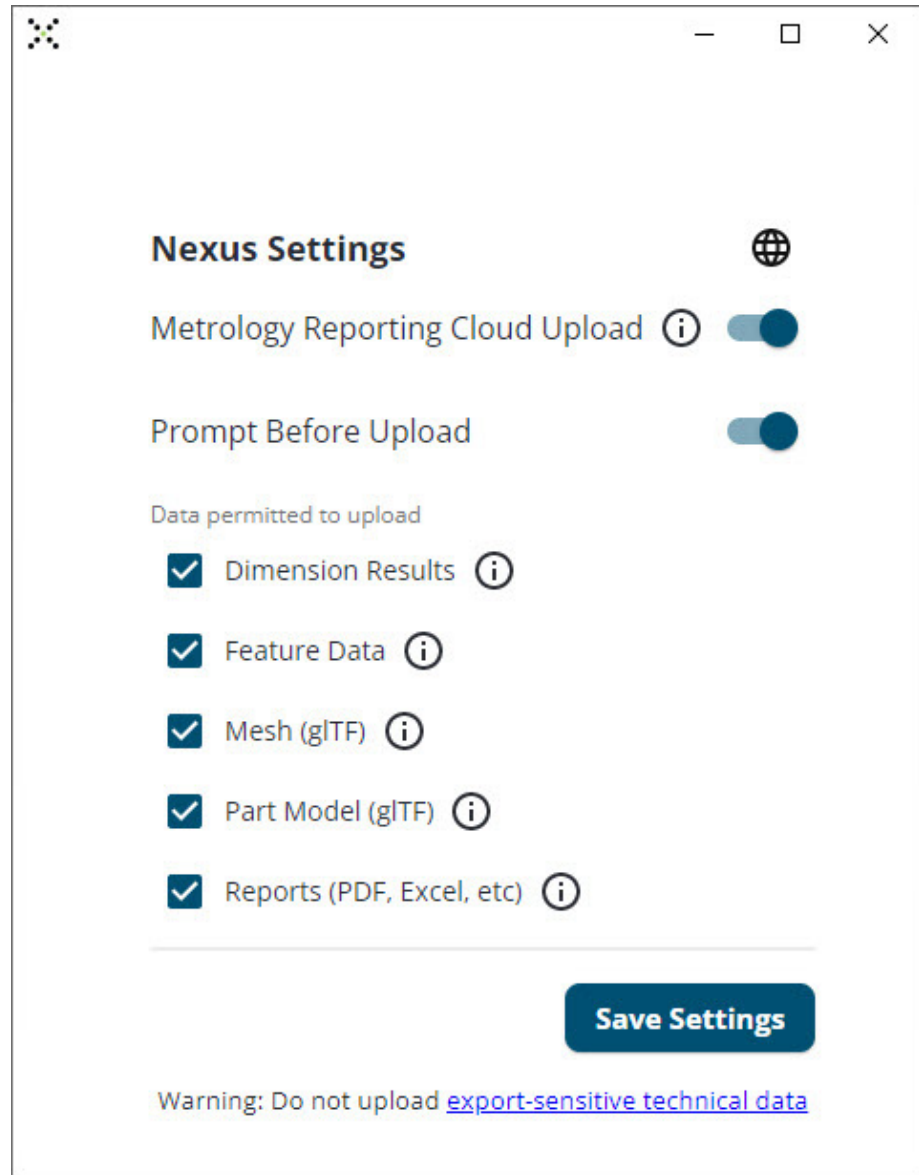


The SA 2024.1 version requires the drivers for this process to be downloaded and installed when it is first run. These drivers can be downloaded from our ftp site here:

<https://www.kinematics.com/ftp/SA/Install/Driver%20Downloads/Nexus%20Metrology%20Reporting/>

Once installed a valid email and user ID is required to connect to Nexus. The following reporting upload options will be available (Figure 28-102):

Figure 28-102. Nexus Cloud Upload settings



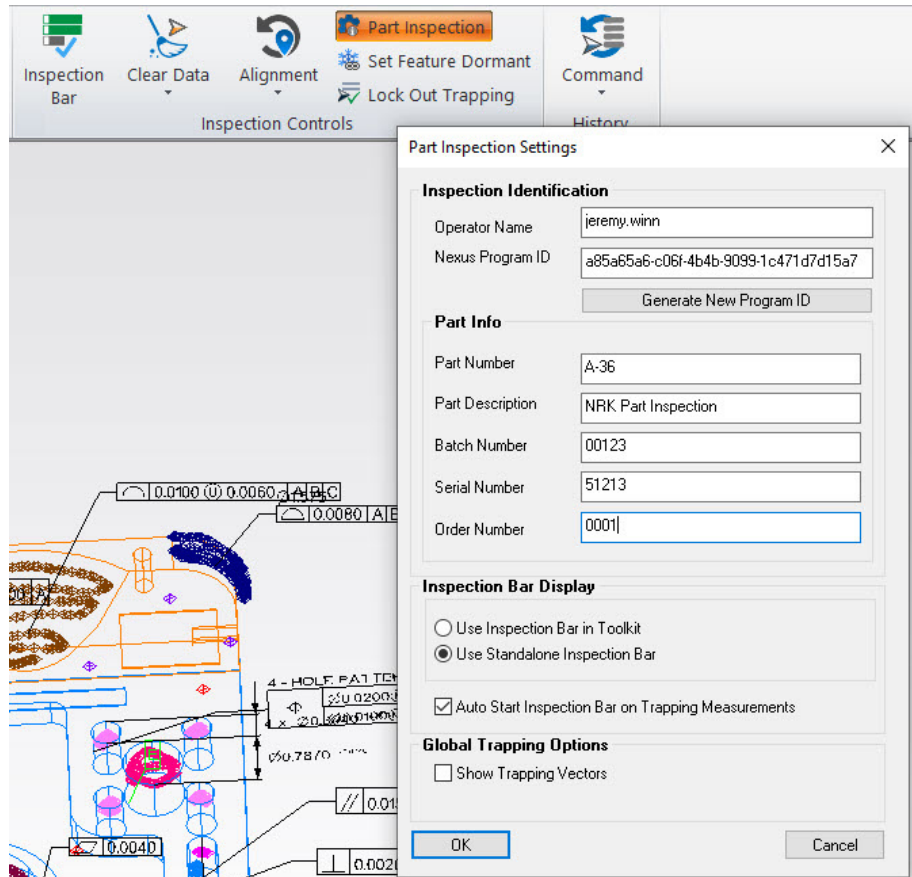
For SPC analysis parts need to have the same unique identifier. To accomplish this a template file must be used. A set of controls has been added to the *Part Inspection* options on the *Home* Tab of the ribbon.

The Nexus Program ID is a unique hash tag for the file that identifies it. If you wish to modify the existing template the *Generate New Program ID* button can be used to create a new ID. Basic part information can also be entered which is included as meta data with the part (Figure 28-103).

These same controls can be used to directly add part information into the Q-Das export as well if desired. More information on Q-Das export is available here:

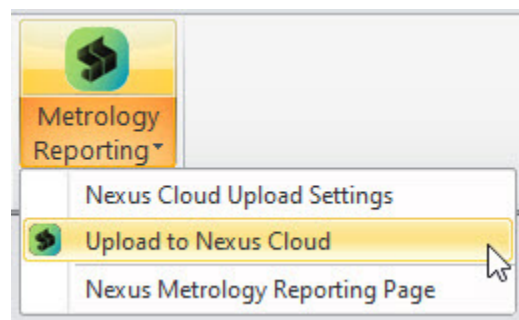
<https://kinematics.my.site.com/SA/s/article/Integrating-Q-DAS-export-with-SA-for-process-control>

Figure 28-103. Part Inspection Controls



Once configured a part file can be uploaded to Nexus at any time using the buttons on the reporting tab (Figure 28-104):

Figure 28-104. Upload to Nexus Cloud button



More information on Metrology Reporting can be found here:

<https://nexus.hexagon.com/documentationcenter/>

This Page Intentionally Left Blank.

Measurement Uncertainty

CHAPTER

29

“A measurement result is complete only when accompanied by a quantitative statement of its uncertainty.” –National Institute of Standards and Technology (NIST). This chapter introduces measurement uncertainty, discusses how uncertainty is calculated in SA, and provides some guidance on refining measurement uncertainty estimation.

What is Measurement Uncertainty?

Put simply measurement uncertainty is a statement of how sure you are that the objects true position is where you say it is.

Being able to estimate uncertainty is important. Many organizations have standards requiring that measurements include a statement of the measurement’s uncertainty. For instance, the International Organization for Standardization (ISO) “Global Product Specification” requires that part measurements have both measured values and uncertainties, and the National Institute of Standards and Technology (NIST) specifies that “a measurement result is complete only when accompanied by a quantitative statement of its uncertainty.”

It is important to realize that uncertainty calculations are always going to be a process of estimation. Instrument manufacturers have put a lot of effort into isolating and quantifying the accuracies of their measurement systems and reporting an accuracy spec with their product. However, in a real measurement application there are many more factors that can influence uncertainty. Examples include:

- Instrument uncertainty and accuracy at a given distance
- Added targeting or tooling uncertainty
- Environmental influences, such as wind, vibration, etc.
- Operator influences, particularly with instruments that rely on sighting.

- Geometry contributions
- Fixturing influences
- Scale or temperature uncertainty
- Networking instruments builds interrelated measurement systems whose combined characteristics are dramatically different than the individual source measurements.

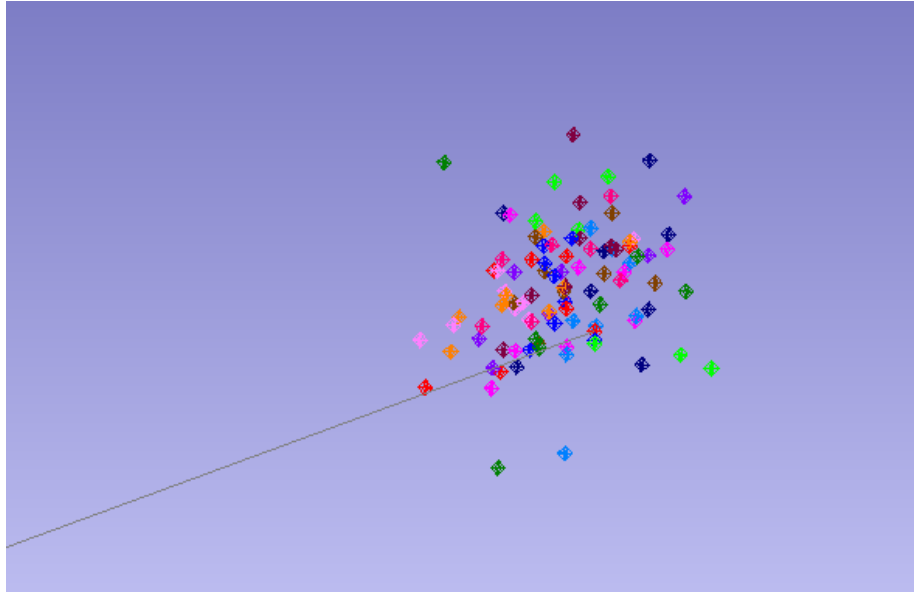
In addition, measurement uncertainty statements typically report a spherical uncertainty and do not address the more complex geometric nature of a measurement in determining its accuracy. Surprisingly the simplification of reporting a +/- value may lead you to overestimate the true uncertainty if you were to look at the position in 3 dimensional space.

SA takes a different approach to uncertainty estimation by striving for objectivity. It provides a clearly defined set of variables to actively model the uncertainty in measurements from the primary contributing variables, it provides a means to build and visualize these uncertainties in 3D and then evaluate the influence of instrument position and point distribution based upon these defined variables. It provides a tool that enables metrologists, engineers, customers, and even managers to communicate effectively.

Measurement Uncertainty and Visualization in SA

The best way to determine the uncertainty in the measurements from your instrument in your environment is to measure a point many times under those conditions. For example, if you were to simply measure a point 100 times you would end up with a group of points all with slight deviations from one another. There would be a greater number of points near the center of the group and fewer as you looked outward from that center location to the few outliers along the periphery (Figure 29-1). This variation can be described in statistical terms as a Gaussian distribution.

Figure 29-1. Multiple measurements of the same target



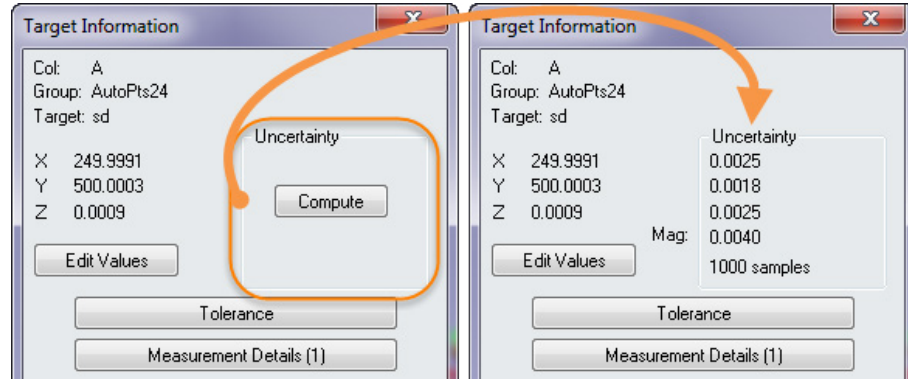
SpatialAnalyzer takes this fundamental principle and expands on it to provide the ability to compute and report unique measurement uncertainty estimates for each and every supported instrument. Rather than providing a long list of computed uncertainty estimates, SA presents uncertainty data visually as a point cloud (around the measurement) that represents the target's region of uncertainty. This point cloud is called a sensitivity cloud and represents what you could expect to see if you were to take many measurements of that same point location.

Mathematically SA utilizes a Gaussian random number generator (utilizing a Box-Muller algorithm) to build this sensitivity cloud. Considering that real-world, natural processes tend to have a Gaussian distribution this is a logical choice to model the expected distribution. It builds uncertainty clouds in three dimensions based upon the uncertainty values of the instrument and a user specified number of samples. The amount and type of error that is inserted is based on the precision of the measuring device.

Calculating Uncertainty of a Point in SA

To calculate the uncertainty of a point in SA you can simply go to the properties menu of a specific point and press the Compute button (Figure 29-2). This will build an uncertainty cloud for that point. You can also build a set of uncertainty clouds for selected points using the command **Analysis > Coordinate Uncertainty > Create Point Uncertainty Fields**

Figure 29-2. Computing Uncertainty for a Point



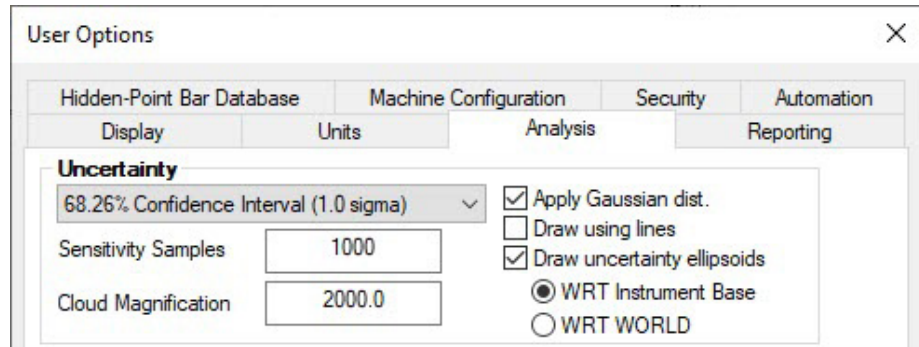
The magnification and confidence interval for uncertainty cloud display properties can be changed under the User Options. These computed values and resulting uncertainty cloud are based upon several setting:

1. The User Options settings for the confidence interval, the number of samples to compute, and the magnification value used to help visualize the cloud in the graphics.
2. Instrument Uncertainty settings for the instrument that took the measurement.

Confidence Interval and User Option Settings

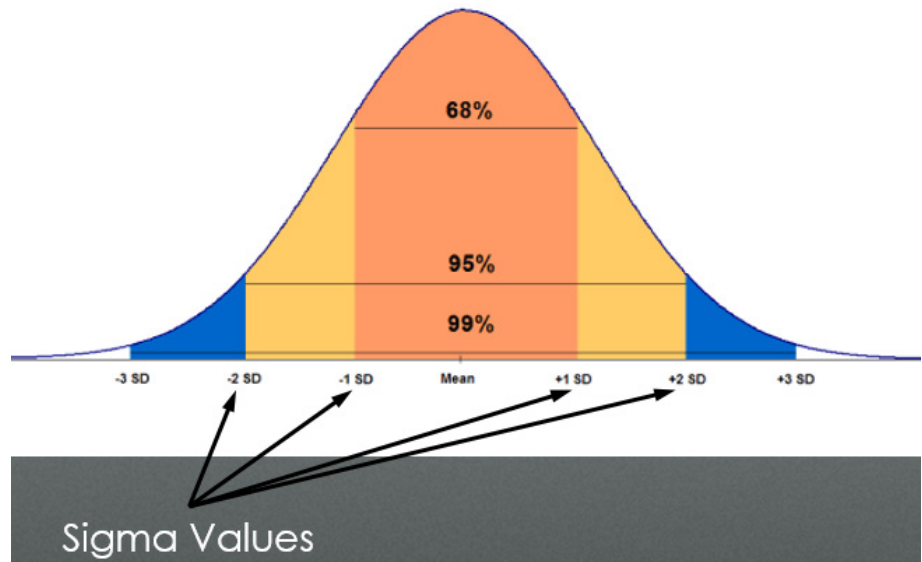
The Users Options provides a set of controls for both uncertainty computation and reporting purposes on the **Analysis** tab (Figure 29-3).

Figure 29-3. Uncertainty Computation Controls



The Confidence Interval directly determines the reported uncertainty magnitude values reported for a point. For example, if you use the default 1.0 sigma Confidence Interval then you are making a statement in your report that you have a 68.26% confidence that the true measurement lies within the uncertainty parameters. If you want higher confidence, you can set the uncertainty to 3.0 sigma for example, which will greatly increase the uncertainty magnitudes reported but allow you to state a 99.74% confidence (Figure 29-4).

Figure 29-4. Normal or Gaussian Distribution



The number of samples used in this calculation can also be adjusted (using the *Sensitivity Samples*). The default is 1000. Increasing the number of samples give you a better estimate but will also take more time to compute. Reducing the number will accelerate the process but cause outliers to have a greater influence on the results.

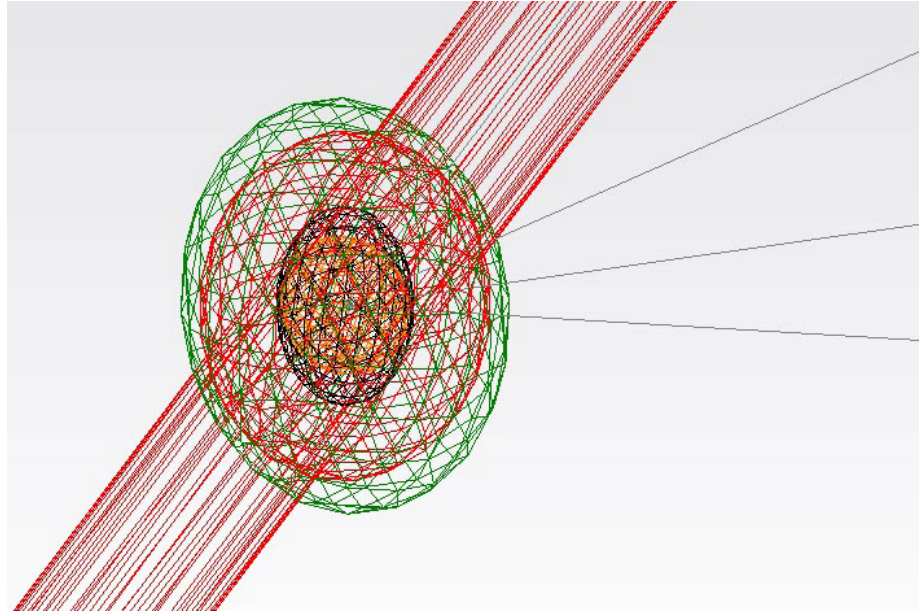
Sensitivity clouds are handled differently than other point clouds. For this reason we have a separate magnification value which helps visualize these clouds graphically. Increasing the magnification value will scale the cloud about the measurement to make it easier to see. The deviations can also be depicted as lines drawn from the measured point.

Point Uncertainty Display

The uncertainty in the position of a point in space can be displayed in either of two ways. Traditionally SA used a Point Cloud or “Sensitivity Cloud” to illustrate the potential range of variability.

Beginning in the 2022.3 version this uncertainty can be displayed as an ellipsoid that encompasses the uncertainty region. Analysis performed using CoVariance does not generate a point cloud at all and depends on this ellipsoid visualization option (Figure 29-5).

Figure 29-5. Ellipsoid Display of Uncertainty Values

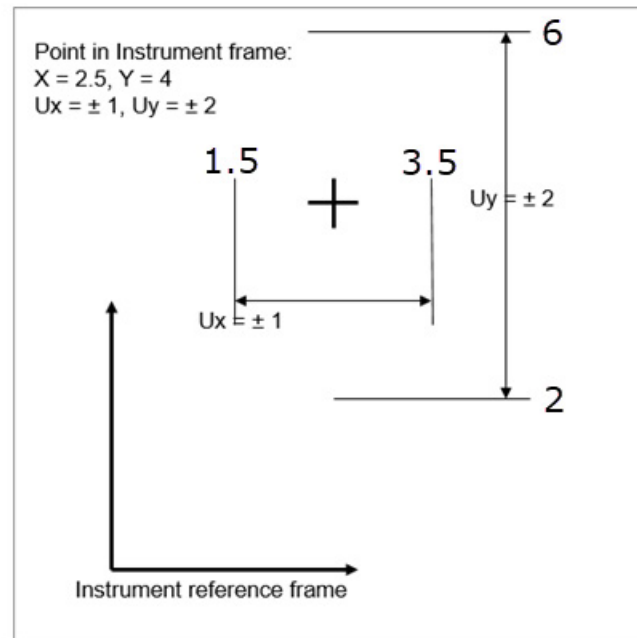


An option to toggle between full and wireframe displays is available under **Alignment>Network Alignments>Coordinate Uncertainty>Toggle Wire Frame Uncertainty Ellipse Ctr+Alt+W** (or the View menu in the classic display).

Is that +/- or the total uncertainty?

The uncertainty that we report is related to the standard deviation of the data. Picture the normally distributed bell curve with a mean at the peak, and vertical lines at standard deviation multiples on either side of the mean. The uncertainty we report is the distance from the mean to the standard deviation of the user's choice. So the expected value of the measurement is the coordinate +/- our reported uncertainty.

Figure 29-6. Graphic depicting SA's reporting of uncertainty for a point in 2D



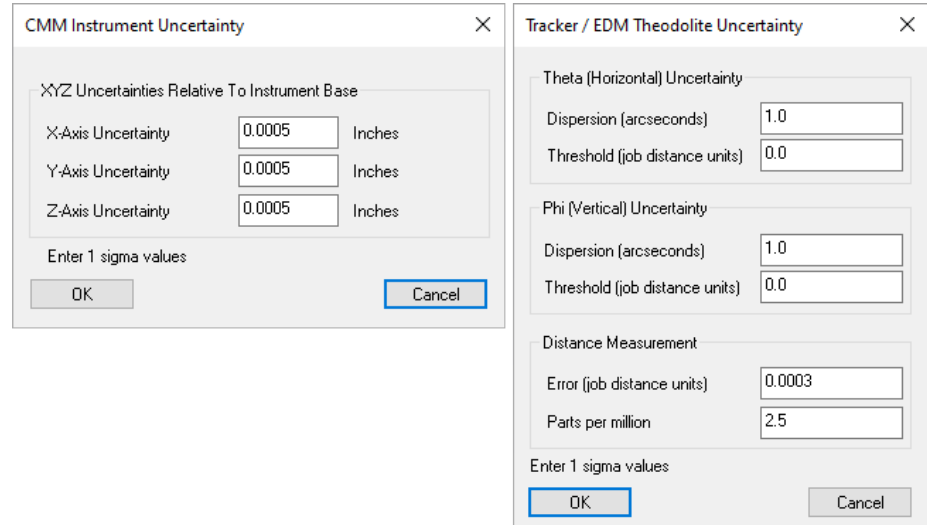
In the image above (Figure 29-6), a point is located at $X = 2.5$ and $Y = 4$. The reported uncertainty of 1 sigma in X and 2 in Y translates to an uncertainty range of 1.5 to 3.5 in X and 2 to 6 in y .

If a USMN composite point has an X coordinate of $1.0000''$ and an uncertainty in X of $0.0006''$ at a 2 sigma confidence interval, we can say with 95.46% confidence that the true X value is between $0.9994''$ and $1.0006''$.

Instrument Uncertainty Values

The point location uncertainty of a measurement is computed based upon the uncertainty parameters of the instrument that recorded the measurement. The Uncertainty Variables can be accessed through an instrument's property menu and will differ depending on the type of instrument (Figure 29-7).

Figure 29-7. Instrument Uncertainty Variables



CMM Instrument Uncertainty

PCMM arms and other robotic instruments have a series of joints that each contribute to the measurement error. Therefore an arm with 7 degrees of freedom will have 7 separate joint uncertainty values. In older versions each uncertainty parameter could be edited for each joint. However, it is very difficult to identify the uncertainty of each joint, making it difficult to enter meaningful values.

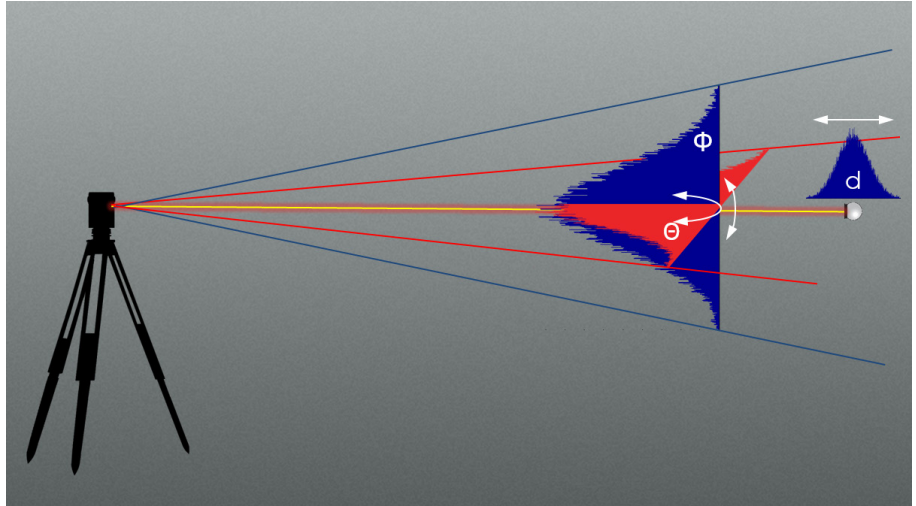
Starting with 2022.2 version of SA these controls have been greatly simplified to include only and XYZ uncertainty with respect to the instrument base. Estimated values can be drawn from the manufacturers spec sheets and this uncertainty can be computed based upon performing a USMN network alignment if desired.

Trackers / EDM Theodolite and other polar Instruments

Looking more closely at the uncertainty settings for a polar instrument such as a Tracker or Total Station you will see there are 3 core parameters. Much like a robot these parameters directly correlate to the mechanical controls or encoders used to determine the location of a measurement, and are depicted graphically in (Figure 29-8):

- **Theta [Horizontal] Angular Uncertainty.** The angular accuracy of the instrument in determining vertical position of a target with respect to the shot line.
- **Phi [Vertical] Angular Uncertainty.** The angular accuracy of the instrument in determining horizontal position of a target with respect to the shot line.
- **Distance Measurement Error.** The fundamental accuracy of the instrument in determining distance at any range.

Figure 29-8. Component Instrument Uncertainty for a Tracker



In 2019 a new **Aperture Error** parameters was added which provided an alternative means of defining angular uncertainty for a polar instrument. Aperture Error could be used either as an alternative to or in conjunction with Theta and Phi angle measurement uncertainties.

However, starting in 2022.2 these methods have been combined to provide a more complete and user friendly solution (Figure 29-9).

Figure 29-9. Example Aperture uncertainty settings. These settings could be used as a starting reference for uncertainty modeling applications.

Tracker / EDM Theodolite Uncertainty	
Theta (Horizontal) Uncertainty	
Dispersion (arcseconds)	1.0
Threshold (job distance units)	0.0
Phi (Vertical) Uncertainty	
Dispersion (arcseconds)	1.0
Threshold (job distance units)	0.0
Distance Measurement	
Error (job distance units)	0.0003
Parts per million	2.5
Enter 1 sigma values	
OK	Cancel

Angular uncertainty has been refined from the original encoder uncertainty to eliminate singular behavior of uncertainty assessment when the measurement approaches polar points. Uncertainty is now modeled much like sighting through a telescopic sight directed towards the measurement (Figure 29-10).



Figure 29-10. Error Dispersion
Model for EDM Uncertainty

Theta and Phi Uncertainties

- **Dispersion (arcseconds).** The angular dispersion value defines the angular divergence possible from the line of sight with respect to either the vertical or horizontal plane.
- **Threshold (in job linear units).** Defines the base error threshold which could model the minimum uncertainty in the location of a reflector in the nest.

The legacy approach of using elevation and azimuth as angular errors scaled by distance can produce surprising results near-polar orientations. Think of a turret gunner who can only move the gun up and down (PHI) and on a rotating platform (THETA). If he points the gun straight up, no amount of rotation will move his point of aim, i.e., uncertainty in THETA has no effect but he had better get PHI right if he wants to hit a target on the polar axis. Although this is mathematically valid, customers have observed that this did not explain field results that did not appear to exhibit this sort of singular uncertainty profile.

This problem is addressed this with the addition of a radial uncertainty parameter perpendicular to the LOS. This parameter has both a fixed minimum *Threshold* and an angular *Dispersion* component which expands proportionally based on the distance to a target, and is not influenced by polar direction. This allows the user complete control over what components of uncertainty they want to model and in what direction they are asserted.

For a very simple model, the following uncertainties are in play:

- The instrument points to the target (with some uncertainty in both PHI and THETA).
- The target reflection forms a beam spot on an image sensor within the laser tracker. Based on the activated image sensor elements, the instrument SERVO loop drives the beam to center the beam spot on the image sensor. The beam spot on the image sensor will cover more than one sensor element and its shape will probably not be a perfect circle so the accuracy with which the center of the beam spot is determined will produce some amount of dispersion error in Theta and Phi.

- The target reflector center uncertainty affects where the target is perceived to be (Threshold uncertainty).

This example is probably as simple as things can get. There will certainly be other sources of uncertainty depending on the environmental influences. These can be realistically modeled by lumping them in with Dispersion uncertainty or by adding a greater base threshold.

Distance Uncertainties

- **Error (job distance units).** The fundamental accuracy of the instrument in distance along the shot line.
- **Parts per Million.** The error propagation value which is used to define the rate of error expansion with distance.

Default Instruments Uncertainty Settings

The first question that every instrument reseller asks when they start to use SA is “why are the default instrument uncertainty values so high”. It’s important to remember that the default Instrument Uncertainty Variables are *not* intended to match the manufacturer instrument specifications. Rather, they are intended to encompass all the uncertainty in the measurements. These values have been determined to be a conservative first guess as to the overall error resulting from the instrument accuracy and real-world variability resulting from work in the field.

There will certainly be other sources of uncertainty due to refraction through air at varying temperatures, thermal plumes, vibration, inconsistent target placement, target mount stability, etc. But in many cases these values have turned out to be accurate depictions of the true measurement uncertainty.

It’s helpful to keep in mind that the default values are intended to be an initial estimate. Through the use of a USMN network alignment it is possible to experimentally determine the true instrument uncertainty parameters for your instrument in your environment. These values can then be entered into the Instrument Properties and to provide a better estimation of further point uncertainties within those specific condition. For more information on the process refer to the Chapter on USMN (see “Using USMN” on page 790).

Basic Uncertainty Computation Process for a Tracker/Total Station:

A basic point uncertainty computation for a tracker or total station or other spherical measurement device is performed as follows:

1. The linear distance from instrument to the coordinate is de-

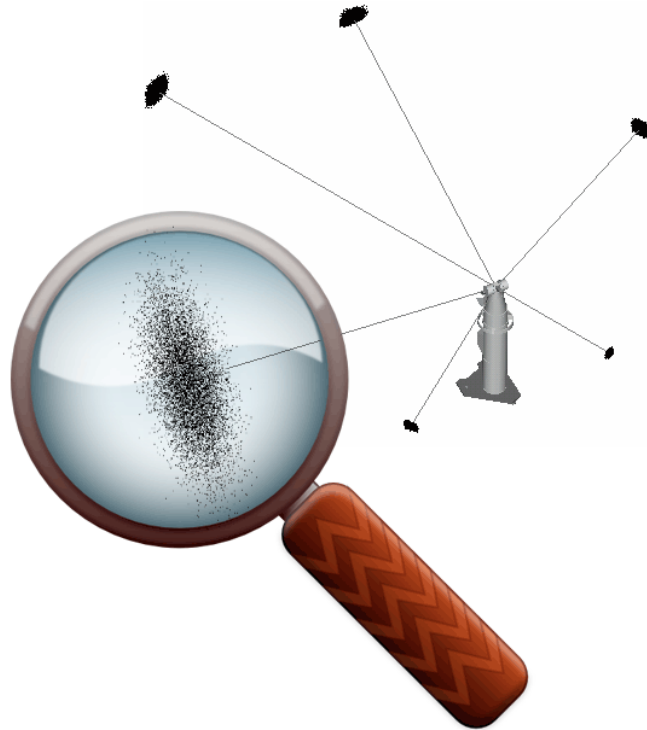
terminated.

2. The Confidence Interval set in the Users Options is used in conjunction with the Uncertainty Variables set in the instrument properties to determine the range of potential uncertainty in each parameter (Horizontal Angle, Vertical Angle, and Distance).
3. The point coordinates values are perturbed using a Gaussian random number generator (utilizing a Box-Muller algorithm) to build a new point coordinate, injecting random error into each of these instrument variables.
4. The process is repeated as many times as is necessary to build an uncertainty cloud as large as is requested.

The distribution of the resulting uncertainty points can be viewed in 3 dimensions and will have different characteristics depending on the instrument uncertainty variables and the distance. A direct comparison between a laser tracker and a total station provides a good example of the differences in cloud shape with different uncertainty parameters.

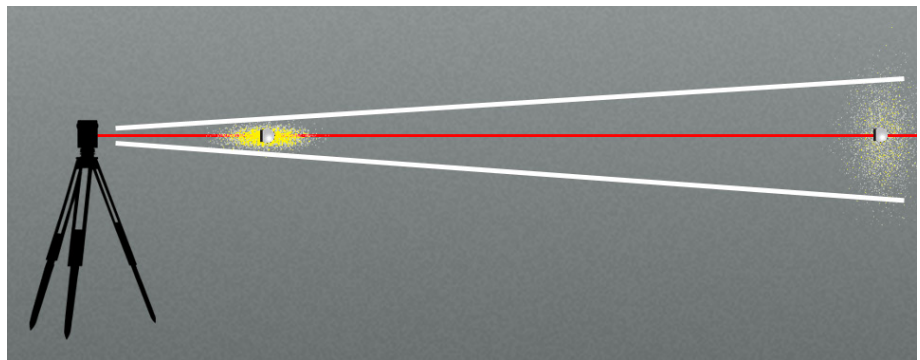
Figure 29-11 depicts a magnified uncertainty cloud for a laser tracker. Laser trackers have accurate distance measurements, but often the angular measurements are not nearly as accurate. As a result the uncertainty clouds inherits a flat “pancake” shape--there is less uncertainty in distance than angle, so the uncertainty cloud tends to be thin (along the line of sight) and wide (perpendicular to the line of sight).

Figure 29-11. A tracker measurement's uncertainty cloud.



Notice also that as measurements get further away from the tracker, the uncertainty becomes larger, manifested as larger uncertainty clouds (Figure 29-12).

Figure 29-12. Changes in Cloud shape with Distance



On the other hand, a total station's azimuth/elevation angle measurements can be far more accurate compared to its distance measurements. Notice how the uncertainty clouds are elongated and cigar-shaped: there are high uncertainties along the line-of-sight, but low uncertainties perpendicular to the line-of-sight (Figure 29-13).

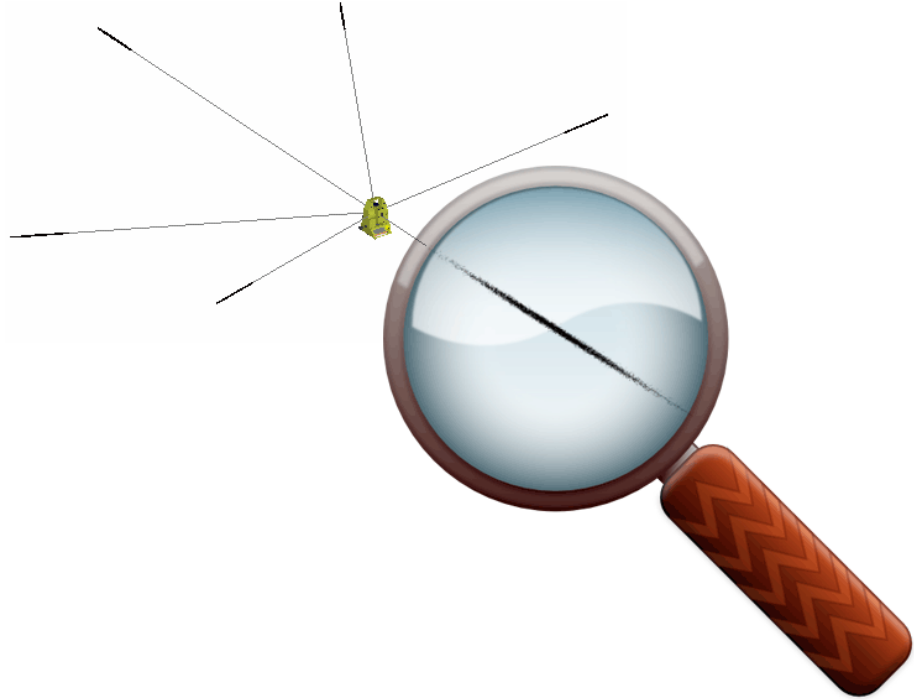


Figure 29-13. The uncertainty cloud for measurements from a total station.

Viewed side-by-side (Figure 29-14), the differing characteristics of each instrument result in drastically different measurement uncertainties (in size, shape, and density):

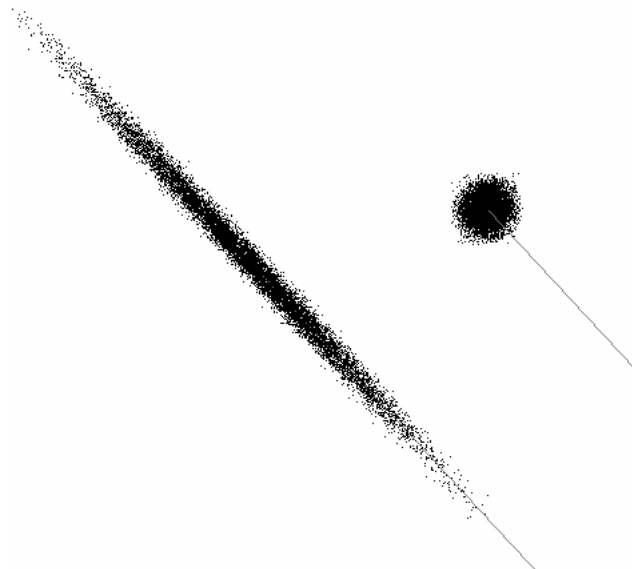


Figure 29-14. Comparing the shapes of the two uncertainty clouds.

Basic Uncertainty Computation Process for a PCMM Arm

The uncertainty characteristics of a portable CMM arm are even more complex, depending on the behavior of the encoder in each joint and the configuration of the arm when the measurement is taken. However, the computation process follows the same sequence:

1. The individual joint angles and lengths for each segment are recorded with each measurement along with the final point

coordinate relative to the instrument base.

2. The Confidence Interval set in the Users Options is used in conjunction with the Uncertainty Variables set in the instrument properties to determine the range of potential uncertainty in each parameter (Each individual joint's uncertainty)
3. The point coordinates values are perturbed using a Gaussian random number generator (utilizing a Box-Muller algorithm) to build a new point coordinate, injecting random error into each of these instrument variables.
4. The process is repeated as many times as is necessary to built an uncertainty cloud as large as is requested.

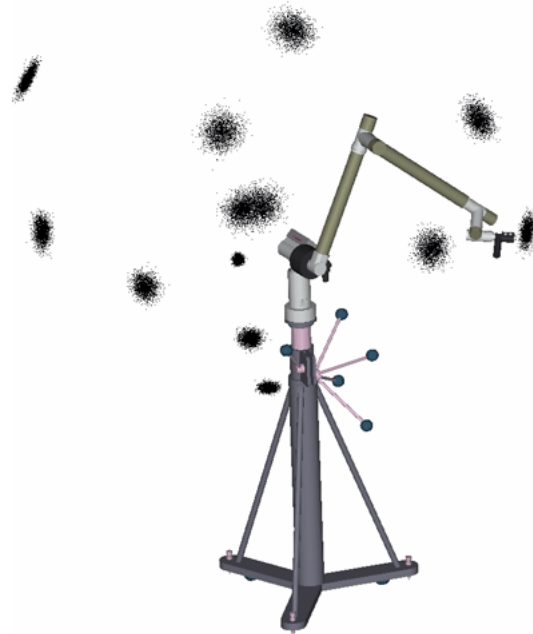


Figure 29-15. The uncertainty cloud shapes for a portable arm.

Uncertainty for Points with Multiple Observations

For instruments that do not provide a distance component, points must be built with more than one observation. It is also possible to combine additional shots into any point computation. The uncertainty of that point therefore is based upon 2 factors:

1. The network or instrument placement uncertainty for each instrument measuring the point. This uncertainty needs to be determined as part of the network alignment operation.
2. The composite point uncertainty. Which is calculated based up on the size of the overlapping point uncertainty fields.

Because the instrument placement uncertainty plays such an impor-

tant role in point uncertainty calculations for combined point uncertainties this subject is covered as part of the weighted bundle section of the USMN chapter (see “[Weighted Bundle](#)” on page 786).

Measurement Analysis with Uncertainty Fields

This section addresses applications of coordinate uncertainty fields beyond the uncertainty in a measured coordinate value. Often, the measurement task is not only to determine XYZ coordinate values of points as measured by the instrument, but also to evaluate the true, as-built, shape of an object. In these cases, the coordinate values measured by the instrument are only an intermediate result in the metrology job.

This means that in order to determine the uncertainty of the actual geometrical analysis operation, other methods must be used. The discrete point cloud uncertainty fields are well-suited for such an analysis.

Point to Point Distance Uncertainty

The uncertainty in the distance between two points can be computed with a function within SpatialAnalyzer. Use the menu selection **Query >> Sensitivity Cloud to >> Sensitivity Cloud**. Select any two points that have uncertainty fields associated with them to complete the process. SA uses the uncertainty cloud from each point to compute the geometric uncertainty between the points ([Figure 29-16](#)).

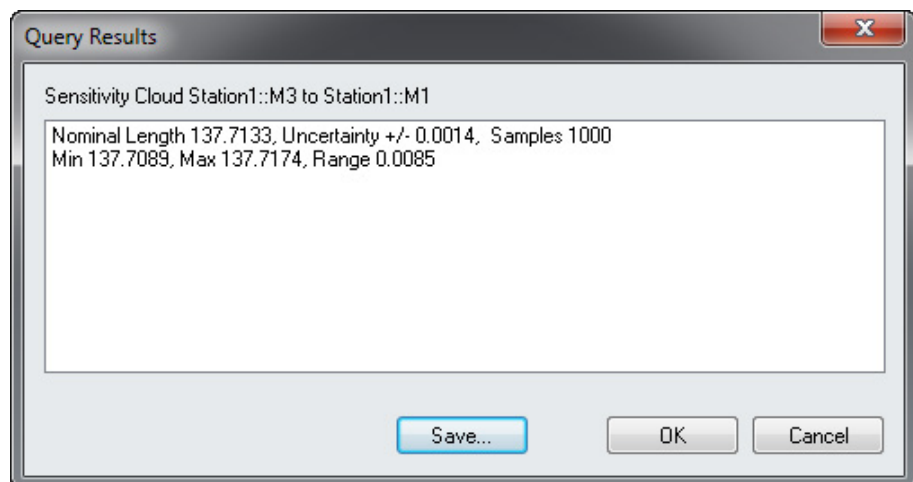


Figure 29-16. Query Sensitivity Cloud to Sensitivity Cloud

The analysis includes:

- **Nominal Length.** The length between the points
- **Uncertainty.** The uncertainty in that nominal length reported based on the standard deviations of all the distances pairs between the two uncertainty fields.

- **Min & Max.** The minimum and maximum distances between the points.
- **Range.** Distance between the min and max distance.

Geometry Fit Uncertainty

The fitting of measured data to geometrical shapes is quite common in measurement analysis. One of the most common is the sphere fit. Tooling ball measurement is very common since it is often used to corresponding common measurements or transform into nominal coordinate systems. The measurement device may measure several hundred points on the tooling ball, perform a best-fit sphere analysis, and return the center of the sphere as the XYZ value. In this case, a statement of the uncertainty in the sphere fit should accompany the XYZ center point of the sphere.

Using the discrete point cloud representation of the uncertainty field, it is a straightforward task to determine the effect of the measurement uncertainty on the resulting geometrical fit. A description of this process follows:

First, a geometrical fitting operation is performed on the actual measurement values, or the optimal common points as determined in the simultaneous system combination portion of the USMN solution. This result is stored as the baseline for the geometrical properties. In the case of a sphere-fit, the results are a center point (X, Y, and Z), and a radius.

Next, the geometric fit operation is repeated for all the points in the uncertainty fields. The result is a list of geometrical parameters representing the “uncertainty field” of the geometry. These sets of parameter values may be evaluated using methods similar to the uncertainty field numerical evaluations. By computing the standard deviation of each of the variable lists, and applying the desired confidence interval, a fit uncertainty statement may be given.

If the uncertainty fields for the points in the fit contain a different number of uncertainty field points, the largest quantity is used in the analysis. When the uncertainty field index exceeds the total number of points in the smaller clouds, they just loop back to the first point and reuse the cloud.

Sphere Fit Uncertainty Example

The measurement of a tooling ball with a laser tracker is accomplished by sweeping the laser tracker target across the sphere to obtain sufficient coverage for a sphere fit. Typically, laser tracker targets are 1.5 inch spheres with a center-mounted retro-reflector. [Figure 29-17](#) shows two sets of measurement data on a ½ inch tooling ball. One set shows typical coverage of the sphere while the other shows inad-

equate coverage.

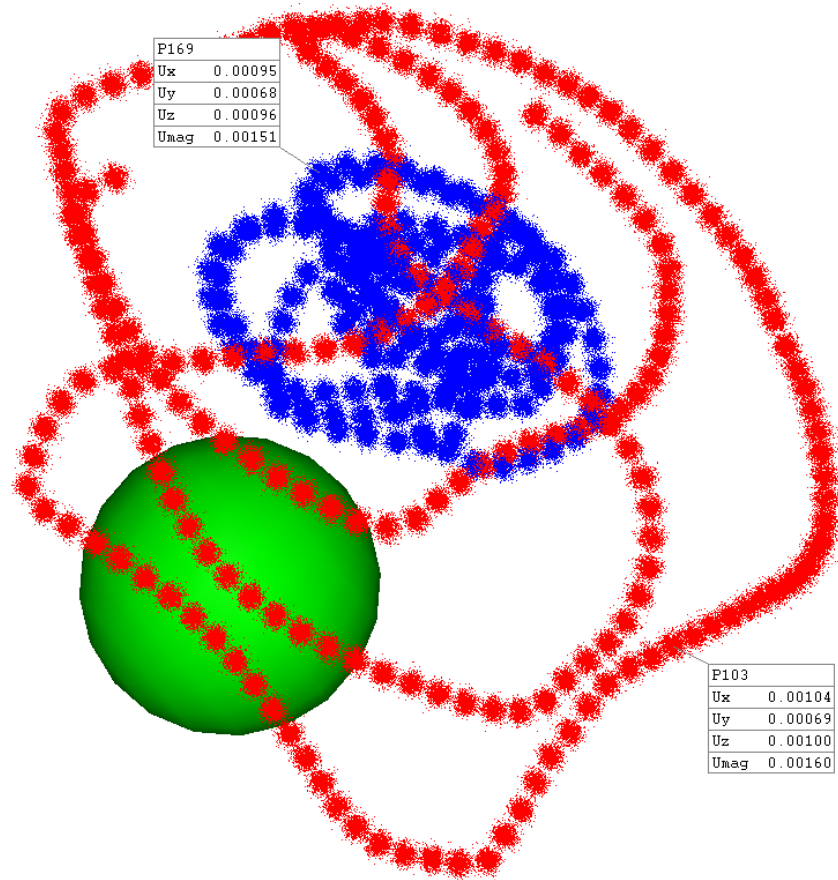


Figure 29-17. Tooling Ball measurements using an SMR

Notice that the data is offset from the actual sphere. This target offset is one of the factors that must be properly handled in all measurement analysis. In the case of a sphere fit, this is accomplished by adjusting the objective function so that the optimal solution is one where the measured points are $\frac{1}{2}$ of the target diameter away from the best-fit sphere.

As discussed previously, the analysis begins with a computation of the best-fit sphere for all the actual measured points. This is stored as the nominal value to which all subsequent fits are compared. Then, the first point from each cloud is used and another fit is performed. The results are tabulated. This process is repeated for the second point, third, and so on until all the points in the cloud are exhausted. The result is a tabulation of sphere fit results. By applying basic statistics to these results, a statement of their uncertainty is made. In this particular case, the entire process was repeated twice, once for the small data set, and then again for the larger data set. The results of this analysis are shown in Table 11.6.

Figure 29-18. Sphere Fit Analysis

	Small Coverage	Large Coverage
Sphere Center:		
Ux	0.0018	0.0003
Uy	0.0027	0.0004
Uz	0.0012	0.0002
Umag	0.0035	0.0005
Udia	0.0067	0.0008
Analysis time (P-4 1.3 GHz)	3.3 sec.	3.7 sec.
Points in Fit	186	222
Field Points	1000	1000

Sphere Measurement Coverage Comparison

This example not only underscores the importance of adequate measurement coverage when fitting a sphere, but it also demonstrates the need for uncertainty analysis. Depending on the measurement instrument and the shape of the uncertainty fields, the uncertainties in geometrical fits can be significant.

Cylinder Fit Uncertainty

Applying these methods to other geometrical shapes is a straightforward process. In this example, two sets of cylinder measurement data are investigated. The first maintains a thin area of coverage on the surface of the cylinder, while the second covers a larger portion of the surface. The analysis methods are identical to those in the previous example with the exception of the parameters that are tabulated. [Figure 29-19](#) shows the actual measurement data sets used in this example and the resulting best-fit cylinder. [Figure 29-20](#) shows the numerical results of the analysis and provides a comparison of the two measurement geometries. Notice the reduction in the uncertainty of both the axis of the cylinder and the diameter in the case of the

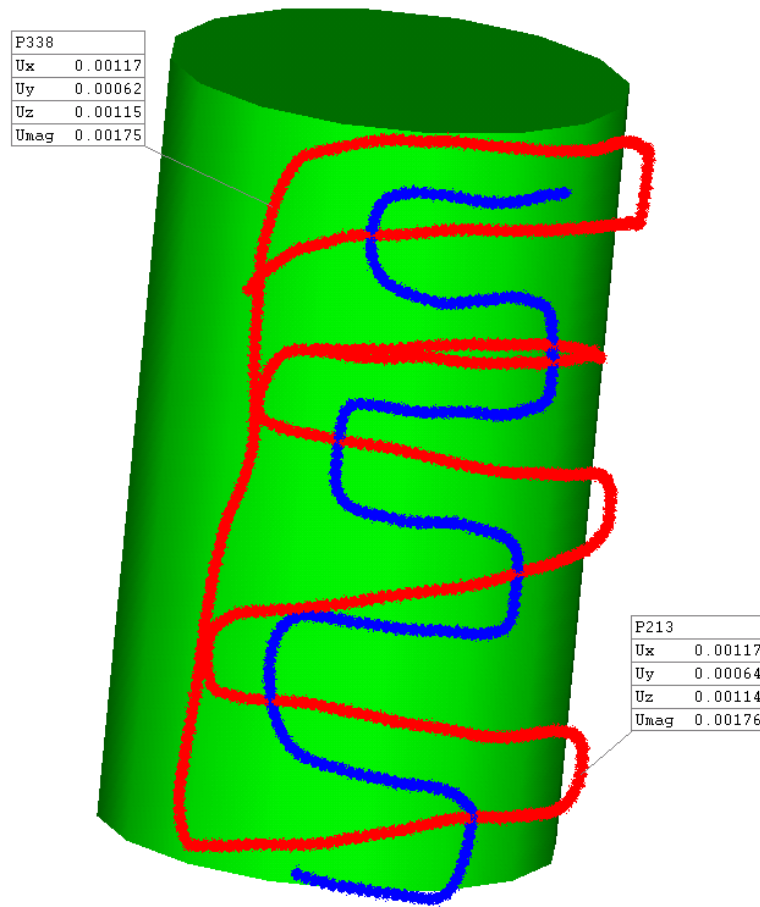


Figure 29-19. Cylinder Measurements

	Small Coverage	Large Coverage
Cylinder:		
U Axis (deg)	0.0021	0.0009
U Diameter (inches)	0.0027	0.0005
Analysis time (P-4 1.3 GHz)	14.7 sec.	32.3 sec.
Points in Fit	217	495
Field Points	1000	1000

Figure 29-20. Cylinder Fit Uncertainty Analysis

How to Perform a Geometry fit with Uncertainty:

In order to calculate the uncertainty in the Geometry fit

1. Use **Analysis>Coordinate Uncertainty>Create Point Uncertainty Fields** to build the uncertainty clouds for your measurements.
2. Use **Analysis>Coordinate Uncertainty>Fit Geometries using Uncertainty Fields** to begin the analysis.
3. Select the geometry type to use for the Fit.

The analysis will be performed and the results reported for review. To save the results externally select the **Save to File** option or simply choose **Accept**. This will build an event in the tree with the recorded statistics that can be added to a report (Figure 29-21).

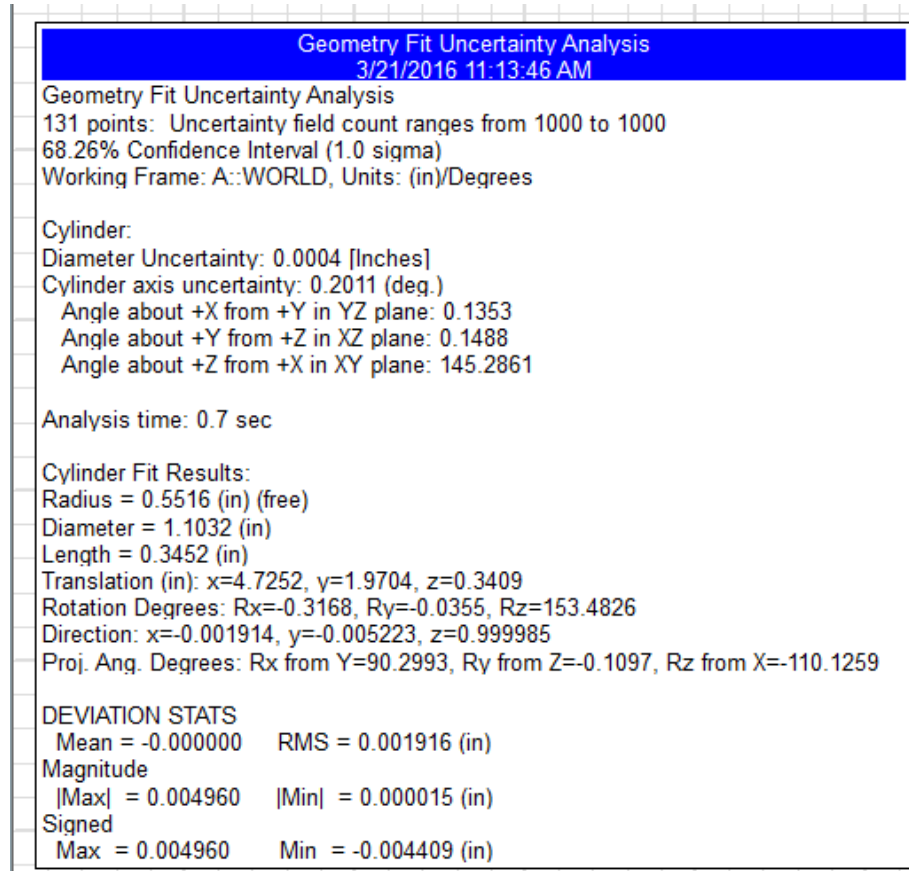


Figure 29-21. Cylinder Fit Event

Interpreting the Results

The following information is reported when performing a Geometry fit with Uncertainty analysis:

Fit summary:

- **Number of points.** (131 points) The number of point selected for the analysis.
- **Uncertainty field count ranges.** (1000 to 1000) Each point has it's own uncertainty cloud build prior to the analysis. This field states the min and max number of uncertainty points build for list of point using in the computation.
- **Confidence Interval.** (1.0 sigma or 68.26%) As uncertainty is statistic process, we give the result at 1 sigma in this case. You could change this value inside **User Option>Analysis>Uncertainty**
- **Working Frame.** (A::WORLD) Result are always given in specific

axis, this is the frame used for the result (the active one when the analysis was done).

- **Units.** ((mm)/(deg)) Values without units mean nothing, so the results are given using.

Feature Uncertainty Results :

The computed uncertainty values for the feature:

- **Feature Type.** (Sphere, Plane, Circle, Cylinder, Line, Paraboloid)
- **Center Uncertainty.** (X=4.7252, Y=1.9704, Z=0.3409) The component uncertainty of the center point along each axis of the working frame (Circles & Sphere's only).
- **Diameter Uncertainty.** (0.0004in) The uncertainty in the diameter of the feature (Circle, Sphere, and Cylinder only)
- **Normal Vector Uncertainty.** (0.2011(deg.)) 3D angle uncertainty in the normal vector of the feature described.
- **Projected Angles.** The component representation of the normal vector of the feature axis:
 - **Angle about +X from +Y in YZ plane.** (0.1353) Angle uncertainty about the X axis, trigonometric orientation
 - **Angle about +Y from +Z in XZ plane.** (0.1488) Angle uncertainty around Y axe, trigonometric orientation
 - **Angle about +Z from +X in XY plane.** (145.2861) Angle uncertainty around Z axe, trigonometric orientation. *Note that it's normal to have a big uncertainty in rotation about a cylinder's axis because this position is not constrained.*
- **D Uncertainty.** (0.0124 [Millimeters]) The 3D equation of a plane can be described by $Ax+By+Cz+D=0$, where D is the uncertainty in the placement/distance from the reporting frame origin (Planes only).
 - **Position.** (x=-0.258814, y=-0.000017, z=-0.965927) reported origin or centroid location of the plane also resolved by the ABCD notation.
 - **Rotation.** (Rx=-169.2707, Ry=-10.5442, Rz=134.0061) representation of a planes normal vector relative to the working frame.
- **Percent Coverage.** The extent of surface coverage provided by the measurements (Sphere's only).
- **Focal Length Coverage.** (0.001) The uncertainty in the focal

length (Paraboloid only).

Computation Statistics:

- **Analysis time.** (0.1 sec) Statistics calculation take time, especially if you have a lot of points and larger numbers of uncertainty samples (minimum 300, generally 1000).

Feature Fit Results:

The computed statistics of the best fit feature. This section describes the position, orientation, and size of the best possible feature fit determined by the uncertainty analysis.

Deviation Statistics :

The deviation statistics for the best fit feature.

- **Mean.** The mean deviation of the points from the feature.
- **RMS.** RMS = Root Mean Square or the quadratics mean if you prefer.
- **Magnitude:**
 $|Max| = 0.049358$ $|Min| = 0.049352$ (mm) min and max error value in absolute (without + or -)
- **Signed**
 $Max = 0.049358$ $Min = -0.049356$ (mm) min and max error value

This Page Intentionally Left Blank.

Unified Spatial Metrology Network

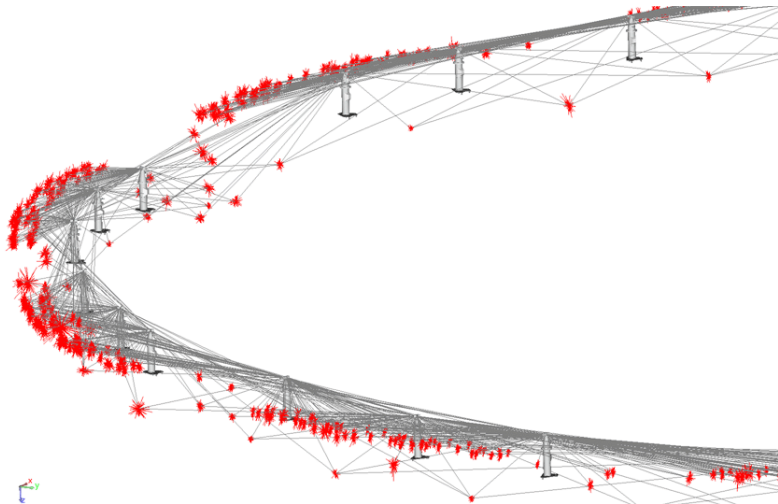
CHAPTER

30

Spatial Analyzer provides one of the most advanced network alignment tools on the market (USMN). Through the use of advanced weighting algorithms, customized instrument uncertainty and by calculating the intersections between 3D uncertainty clouds it provides the most robust network alignment possible.

What is USMN

Figure 30-1. Part of a large USMN network.



What is USMN?

USMN (Unified Spatial Metrology Network) is an advanced tool for combining common measurements for two or more instruments into a single instrument network. It has several major benefits over other analysis tools, such as a best-fit:

- It provides a means for calculating a “weighted bundle”: the most likely position and orientation for all instruments and targets in a measurement network.
- It provides the capability of calculating realistic uncertainties for measurements in a measurement network.

- It can be used to evaluate the real-world uncertainties of instruments on an individual basis.

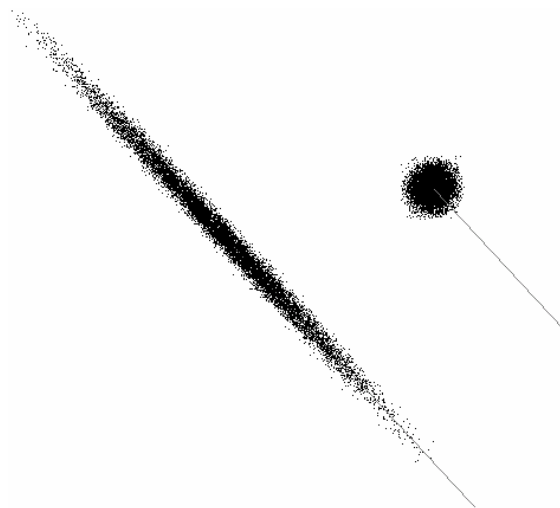
Let's consider each of these benefits in order to get a true grasp on the benefits of USMN.

Spatial Analyzer Tutorial: USMN Overview - <https://www.youtube.com/watch?v=kgviSEkMu70>

Weighted Bundle

As discussed in the previous chapter on Measurement Uncertainty (see “[Measurement Analysis with Uncertainty Fields](#)” on page 776), each instrument type has its own uncertainty characteristics. By simulating measurements of a point while inducing normalized random error using those uncertainty variables it's possible to build uncertainty clouds for points measured by that instrument. This provides an accurate estimate of the point's uncertainty both visually in 3D as well as numerically. Viewed side-by-side ([Figure 30-2](#)), the differing characteristics of each instrument result in drastically different measurement uncertainties (in size, shape, and density):

Figure 30-2. Comparing the Shapes of Two different Uncertainty Clouds

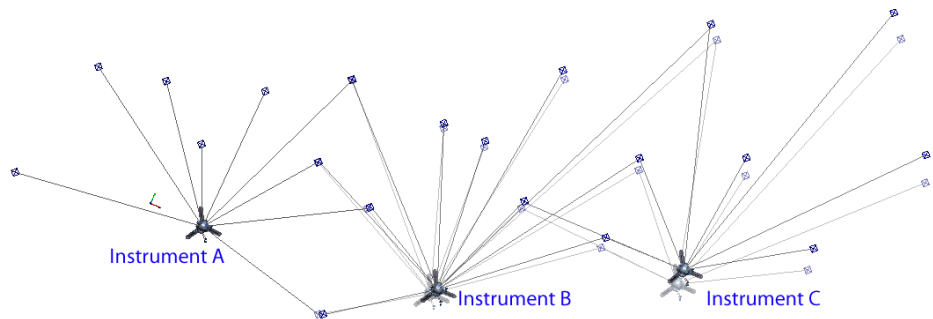


Now, consider the situation in which a long part is to be measured. Due to its shape, this part might require multiple instrument stations along its length. In order to tie all of the measurements into a single coordinate system, a number of common points are measured between adjacent sets of instruments, and then these points are best-fit together to determine the overall instrument network. For some cases, this method is a perfectly valid approach, and in fact, this is the approach taken by other metrology software packages for tying a network of instruments together. In particular, fairly small instrument networks with low uncertainty requirements can use this approach

without problems.

However, using best-fit introduces error stack-up, very similar to the idea of tolerance stack-up in engineering drawings. This error stack-up can be significant. **Figure 30-3** depicts an instrument network using a best-fit compared to the true instrument positions (ghosted). A slight error for the first set of common points (between Instruments A & B) will move the position and orientation of Instrument B out of its true position. By fitting Instrument C to Instrument B, not only does Instrument C inherit the error from the first set of common points (which has now been “leveraged” by its distance from those common points), but now errors in the common points between Instruments B & C cause even more error to stack up. The end result is a network that could potentially be quite different from reality.

Figure 30-3. Positional errors resulting from leap-frogging an instrument along a length.



When more accuracy is desired—or when the measurement network is elongated, large scale, fairly 2-dimensional, or C-shaped (non-closed, meaning the first instrument is not tied back to the last), then best-fitting is usually inadequate. Some examples of real-world measurements that get significant benefit from USMN include particle accelerator measurements (very large and mostly 2-dimensional), large linear measurements with high accuracy requirements (such as the catapult rail on an aircraft carrier), and large “open” networks (such as measurement of two sides of a building).

Instead of using a best-fit, USMN takes an intelligent weighted bundle approach. USMN examines each common measurement and considers the characteristics of the instruments that measured them and their positions in space. Components of instrument observations which are considered to have low uncertainty are assigned a higher weight than components that are considered to have higher uncertainty. If a total station and laser tracker measure a common point from different angles, then the angular measurement from the total station will be assigned more weight than that of the tracker, and the distance measurement from the laser tracker will be assigned more weight than that of the total station.

Once these weights are assigned, the instrument transforms are adjusted to calculate their most likely positions based on all available in-

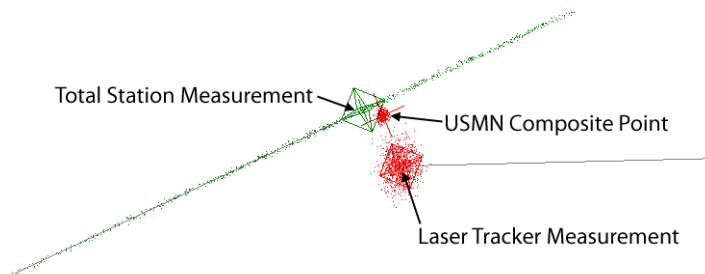
formation. The net result is an instrument network that is significantly closer to reality than the best-fit network described above.

In summary, USMN considers each instrument's uncertainty variables and perturbs instrument positions in order to get minimal measurement closures on the common points. This is similar to a traditional bundle process, except the optimization process uses the estimates of the instrument uncertainties and the range of each observation to weight the individual contributions for each measurement.

Network Measurement Uncertainties

Instrument measurements (observations) can have uncertainties calculated that indicate the uncertainty of a given measurement relative to the instrument. But when instruments are networked together, this simple uncertainty no longer applies, because the common point position in a networked measurement is really determined by a combination of instruments that measured the common point. In fact, the computed common point in the real world never lies on top of any of the component observations (because the world is not perfect--unfortunately!). This highlights the second major advantage to USMN: the ability to calculate "net uncertainties" for common points.

Figure 30-4. The USMN composite point. (Note: The error between the total station and laser tracker measurements are exaggerated for clarity).



Refer to [Figure 30-4](#). A total station and laser tracker have measured a common target in a nest. Due to noise, operator error, and a number of other factors reminding us that the world is not perfect, each instrument "sees" the target in a slightly different position in space. As a result, the measured points end up in slightly different positions.

Not only can USMN calculate the most likely position of the real target based on instrument characteristics (referred to as a USMN Composite Point), but it can also calculate the uncertainty of that composite point. In [Figure 30-4](#), the composite point is closer to the total station's line-of-sight position (due to lower angular uncertainties of the total station), but is also much closer to the laser tracker's distance measurement (due to less uncertainty in the laser tracker's distance measurement). The uncertainty of the composite point itself has less uncertainty than either of the instrument observations, because much of the total station's uncertainty along the line of sight is rendered extremely unlikely based on the laser tracker's measurement.

“Real-World” Instrument Uncertainties

As mentioned earlier, instrument uncertainty values as published by the manufacturer are a good way to compare different measurement systems, but they are a poor way to predict real-world measurement uncertainty. A number of factors, such as the calibration/performance of an individual instrument, the technique of the operator, and the measurement environment have significant effects on uncertainty.

Each instrument in SA has associated with it Instrument Uncertainty Variables (Figure 30-5). These can be found in the Instrument Properties window. Uncertainty variables are used to model an individual instrument’s uncertainty characteristics.

Figure 30-5. Uncertainty variables for a tracker.

If you look at the uncertainty variables for a laser tracker or total station, four values are provided (see image). These values together dictate the uncertainties for the horizontal angle, vertical angle, and distance components for an instrument. They are used in concert with a measurement’s relative position from the instrument to determine the uncertainty of the individual measurement.

By default, the values assigned to uncertainty variables are experimentally determined, conservative values for a given class of instrument. In other words, they generally give a result that is “in the ballpark”, but are probably a bit worse (higher uncertainty) than can truly be achieved with the instrument. This is done to a) err on the safe side, b) avoid conflict with instrument manufacturers, and c) avoid the need to continually update data as instruments evolve, which could be a time-consuming task.

In order to get the most realistic uncertainties for measurements and USMN, it is necessary to use uncertainty variable values that closely match the individual instrument, operator, and measurement environment characteristics. But how is this done? Manufacturers don’t (and can’t possibly) publish these values, and they differ from one instrument to the next and (to a lesser extent) from one operator/measurement situation to the next.

USMN itself can be used to determine the most accurate values for

these variables. This is done by setting up a set of monuments, and then moving an instrument around through the workspace, measuring each of those monuments in turn.

Consider a situation in which an instrument, its operator, and the environment is perfect (and therefore would have zero uncertainty). A perfect instrument, after having measured all monuments from different positions, would see a given monument in the exact same location in space. When fit together (using a traditional best-fit), there would be zero error. The observations would lie on top of each other perfectly. USMN would see this “perfection” and would report uncertainty variables of zero (no error).

Of course, in reality this never happens. In the real world, an instrument can measure a monument two different times and “see” it in a slightly different place. By analyzing the error between measurements of a common point from the same instrument, in different positions, USMN can perform a complex calculation and determine the real-world uncertainty variables for a given measurement situation.

Using USMN

A USMN analysis can be performed by choosing **Analysis>Coordinate Uncertainty>Unified Spatial Metrology Network**. In this same menu is a second setting: **Unified Spatial Metrology Network with Point Groups** which is specifically for aligning your network solution to a nominal point group.

Observations and Point Group Requirements for USMN

When setting up a network for analysis with USMN, there are several organizational requirements for instruments, measurements, point groups, and point names. The requirements for USMN analysis are as follows:

Point Naming and Observation Requirements:

As with any measurement network, common targets must be measured by more than one instrument to form a network, therefore:

1. Individual point names must be common between groups as USMN will associate measurements from different instruments based on point names.
2. Points in different groups with the same name must represent the same target. For example, MeasStation1::P1, MeasStation2::P1, ... MeasStationN::P1 will be combined to compute the composite point P1.

Points without measurements are ignored, unless the point group is

included as a 'Nominal Group.' Therefore:

3. A specific point in a given point group should have observations from only one instrument.

An instrument can make more than one measurement to a specific point. As an example, an instrument can have a front and back sight to a point or an instrument might have N front sight measurements to a point. Observations that are marked as 'ignored' are not included in the USMN network. The measurement details dialog for a point provides an interface to determine whether or not an observation is ignored.

Consider an example tree for a job ready for a USMN network as seen in the following figure (Figure 30-6).

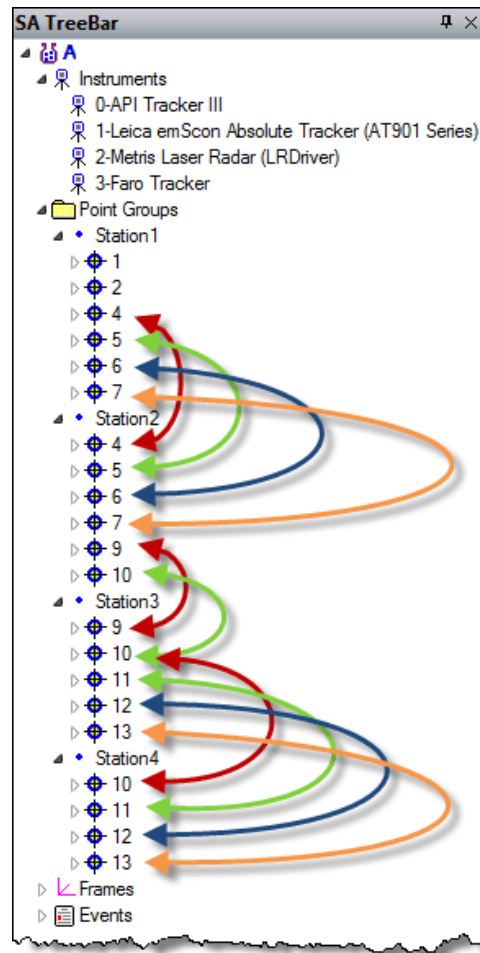


Figure 30-6. Organizational setup for a USMN.

- Points 1 and 2 (in the Station1 group) have no common measurements with other instruments.
- Points 4, 5, 6, and 7 are shared with Station2.
- Point 9 is shared by Station2 and Station3.
- Point 10 is shared by Station2, Station3, and Station4.

- Points 11-13 are shared by Station3 and Station4.

To help identify potential problems with point names, a warning dialog will display showing the Network Build point and instrument listing. This dialog helps to identify any potential problems when you first start the USMN process (Figure 30-7). You may see something like the following:

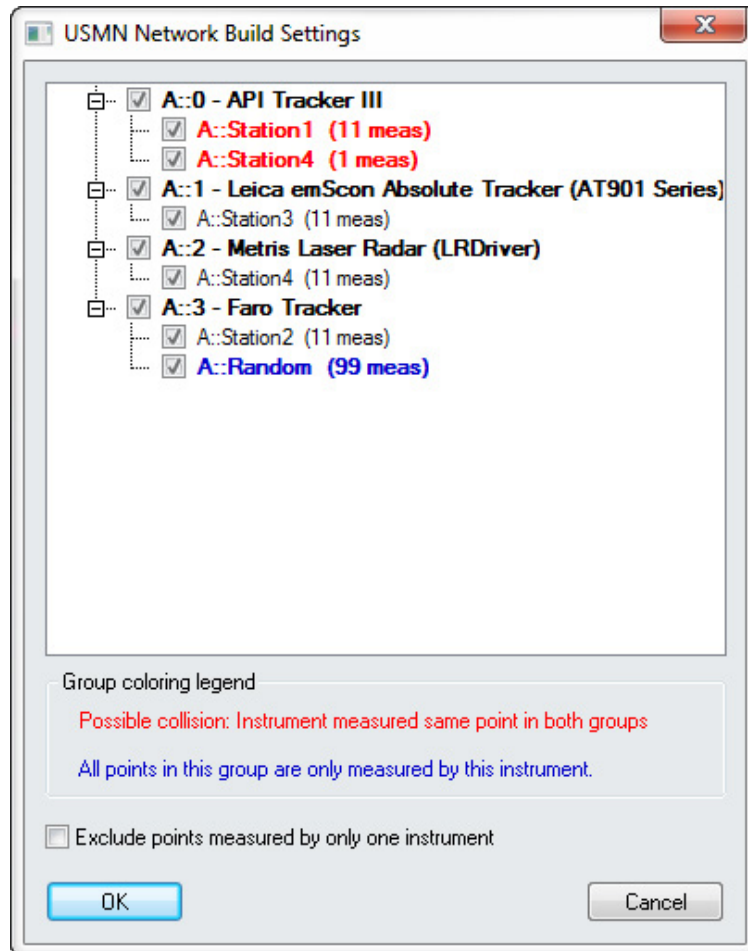


Figure 30-7. USMN Network Build Settings

In the above dialog point groups are marked in red or blue to provide a warning that a user may want to exclude them.

- Red.** Point groups marked in red indicate that there could be a potential collision. The instrument measured a point of the same name multiple groups. These points must represent the same target or be removed from consideration.
- Blue.** Points only measured by a single instrument are marked in Blue. These points do not contribute to the network solution and excluding them will increase the solution speed. Use the Check box to easily exclude all points measured by a single instrument.

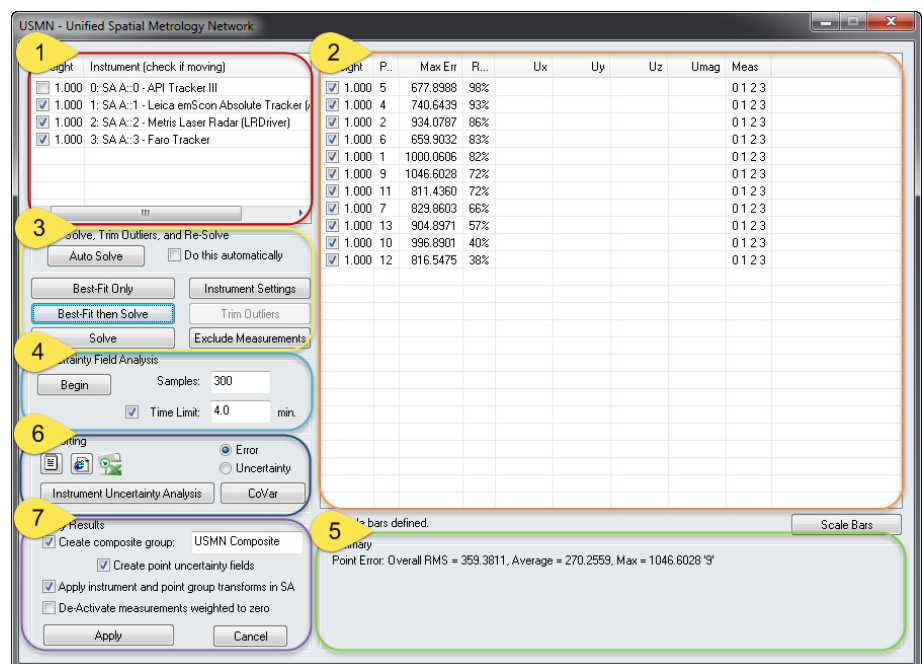
The primary output from USMN is a Composite Point Group. This group contains the optimal point coordinates (the most likely posi-

tion of the points) after solving the network. Inputs to USMN include both the instruments and their measurements. For a measurement to affect the calculation, it must have been measured from multiple instrument stations. The observations from different stations are made into different point groups, but should have the same target name.

USMN Dialog

The primary purpose of the USMN dialog is to help you optimize the network confidently and get the best Composite Point Group possible from the instrument stations and measurements. To do this, it provides many controls all accessed directly through the main dialog. The window is composed of 7 main parts (Figure 30-8).

Figure 30-8. The USMN dialog.



1. Instrument Section

This section lists the instruments whose measurements are included in the USMN network solution. Instruments can have different weights applied. So, for example, if one instrument station was set up in an unstable location, i.e. near a HVAC blower, up high on a large tripod, on a separate concrete pad, etc. The influence of that particular station on the entire network could be reduced by reducing the instrument's weight relative to the other stations.

Each instrument has a checkbox indicating whether that instrument is allowed to move during the calculation. A USMN bundle (or any bundle, or that matter) is a relative solution. That is, the solution can be reached relative to any instrument. As a result, all instruments may be checked (meaning all instruments will be perturbed to find the solution) or one may be unchecked (so that the solution is reached

by perturbing all instruments EXCEPT the unchecked instrument). In both cases, the same end result is reached for the network itself, although the networks will end up in different places in space (and the uncertainties will be relative to different baselines). Regardless, allowing all instruments to move will typically take longer to solve, but the resulting point uncertainties will be more uniform and representative of a network in which no one instrument is more important than another. Although the network solution is the same, the calculated uncertainties for the USMN Composite points will be different, because they are relative to different baselines. However, the relative uncertainties (differences between uncertainties) between each pair of points will remain the same.

More than one instrument may be fixed (unchecked) during a solution, although this is typically only done in rare cases and is generally not recommended, as the end result is a solution that is not an ideal spatial representation of an instrument network.

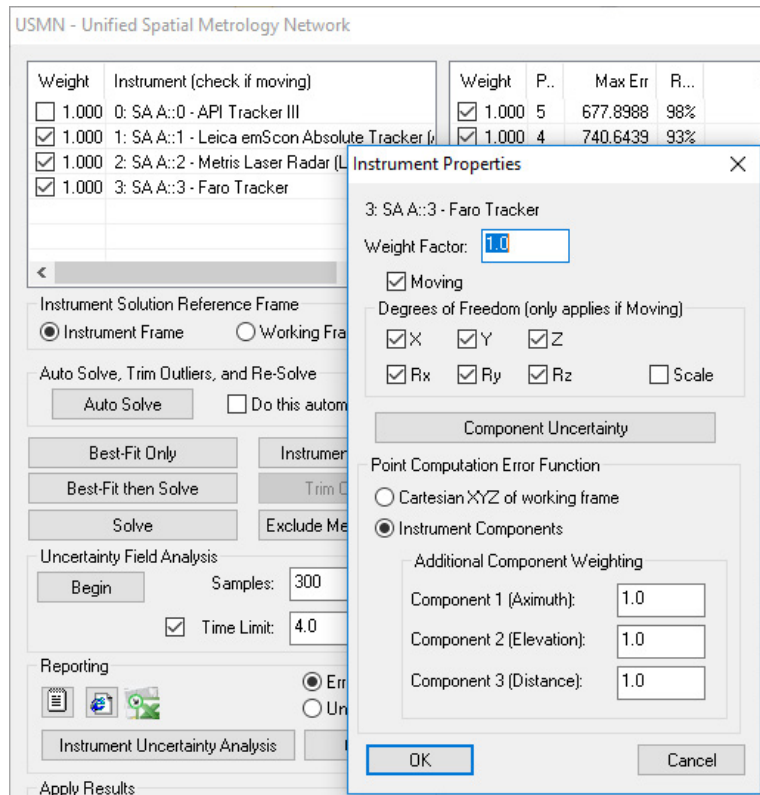


Figure 30-9. Instrument Section

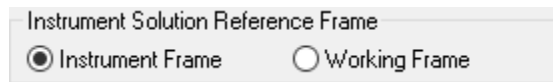
An individual instrument in this list may be double-clicked to access specific instrument properties (Figure 30-9). This instrument Properties dialog provides access to the following settings:

- **Weight Factor.** The overall weight applied to the instrument. (weights can also be edited directly in the USMN dialog).
- **Moving.** This sets the check box determining if a particular instrument is considered to be moved as part of the network.

- **Degree of Freedom.** This set of 7 controls allows restriction of motion of a particular moving instrument as part of the network. Only checked fields are allowed to move. (note: XYZ component restriction are relative to the working frame, for instrument frame reference align the instrument model with the working frame prior to running the USMN).
- **Component Uncertainty.** Provides direct access to the instruments Uncertainty Variables.
- **Point Computation Error Function.** Error can be computed using the instruments uncertainty variables. If you choose to do so weights an be applied independently to each variable. Alternatively error can be computed with respect to the XYZ of the working frame.

In addition to individual instrument configurations there is also an option to choose between two **Instrument Solution Reference Frame** settings.

Figure 30-10. Instrument Solution Reference Frame Setting



- **Instrument Frame.** In most cases, this is the best choice. This allows each instrument to be rotated independently about their own standing axis to find the ideal network solution and is the setting SA has always offered.
- **Working Frame.** This option provides a means to align instruments relative to each other in any orientation and hold that orientation independent of their own standing axis.

2. Measurement Section

This section shows the current statistics with respect to each point in the network. These statistics will recompute as the network is adjusted and recomputed. This measurement-specific information is provided in number of columns which are described below (Figure 30-11).

Figure 30-11. Example Measurements

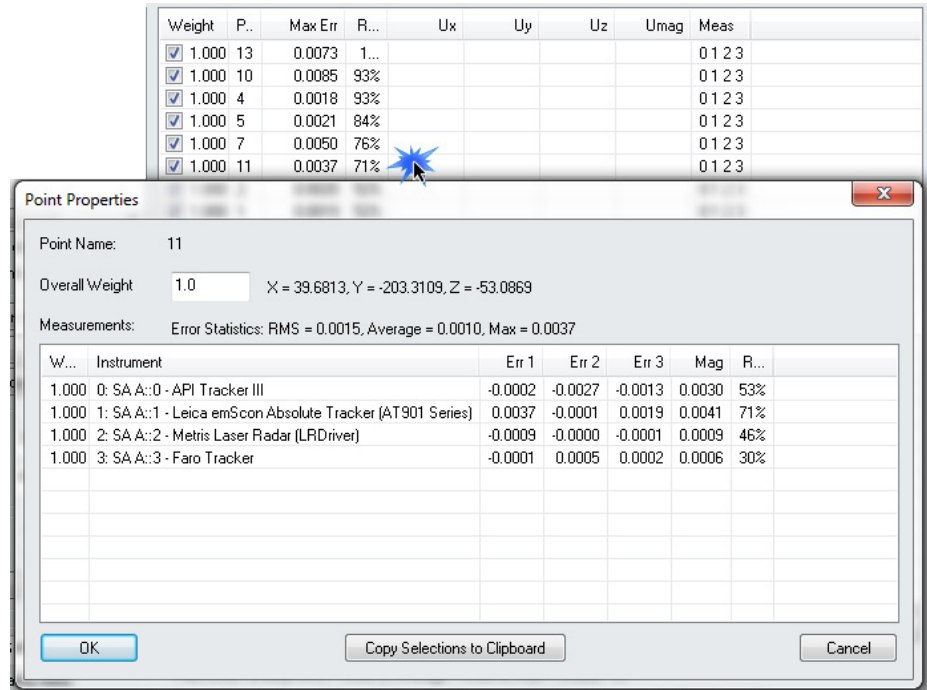
Weight	Point	Max Err	Ranking	Ux	Uy	Uz	Umag	Meas
<input checked="" type="checkbox"/> 1.000	5	677.8988	98%	0.0003	0.0003	0.0003	0.0006	0 1 2 3
<input checked="" type="checkbox"/> 1.000	4	740.6439	93%	0.0002	0.0003	0.0003	0.0005	0 1 2 3
<input checked="" type="checkbox"/> 1.000	2	934.0787	86%	0.0003	0.0004	0.0004	0.0007	0 1 2 3
<input checked="" type="checkbox"/> 1.000	6	659.9032	83%	0.0004	0.0004	0.0004	0.0007	0 1 2 3
<input checked="" type="checkbox"/> 1.000	1	1000.0606	82%	0.0004	0.0004	0.0005	0.0008	0 1 2 3
<input checked="" type="checkbox"/> 1.000	9	1046.6028	72%	0.0004	0.0005	0.0006	0.0009	0 1 2 3
<input checked="" type="checkbox"/> 1.000	11	811.4360	72%	0.0004	0.0003	0.0005	0.0007	0 1 2 3
<input checked="" type="checkbox"/> 1.000	7	829.8603	66%	0.0006	0.0006	0.0008	0.0012	0 1 2 3
<input checked="" type="checkbox"/> 1.000	13	904.8971	57%	0.0006	0.0005	0.0007	0.0010	0 1 2 3
<input checked="" type="checkbox"/> 1.000	10	996.8901	40%	0.0007	0.0007	0.0011	0.0015	0 1 2 3
<input checked="" type="checkbox"/> 1.000	12	816.5475	38%	0.0007	0.0005	0.0008	0.0011	0 1 2 3

The point list is automatically organized in the interface. Points with the highest ranking are listed first. A high ranking indicates a point with a higher chance of problems. Points with lower ranking suggest the target solution is better. However, each column can be sorted from either high to low or low to high by clicking on the column title.

- **Weight.** Each measurement can have a weight applied. If a point is considered less important or perhaps more erroneous, its weight can be reduced.
- **Point.** Each point measured by an instrument in the USMN network is listed by name.
- **Max Err.** The maximum error is listed for each point that has been measured by more than one instrument. This depicts the maximum error between any individual measurement of a point and its optimal value.
- **Ranking.** Displayed as a percentage, this indicates the percentage of the expected 3-sigma envelope consumed by the maximum measurement residual. This is an efficient method of determining which points may have measurement outliers. A point with a ranking of 100% has observations which are consuming the entire 3-sigma envelope. Points with smaller rankings are more tightly clustered around the optimal value. Lower rankings indicate better measurements.
- **Ux/Uy/Uz/Umag.** The component and magnitude uncertainties (in the active coordinate frame) for the optimal point. (These values only appear after an uncertainty field analysis has been performed).
- **Meas.** A list of the instruments that have observations for a given point. For instance, if a point shows 0 __ 3, this indicates that instrument index 0 and 3 each have a measurement for that point, while instruments 1 and 2 do not.

Further detail for a single point can be obtained by double-clicking the point in the list ([Figure 30-12](#)).

Figure 30-12. Measurement Section- Expanded



The Point Properties dialog displays the contributing shots used to compute the network point. It provides control over:

- **Overall Weight.** The weight used in the USMN for the point.

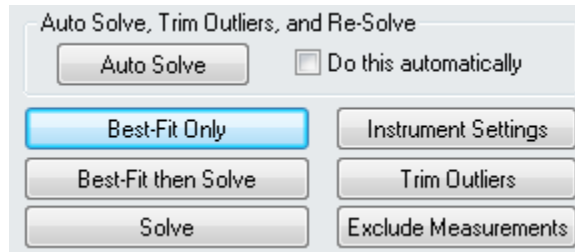
Specific details on each point are displayed in the following columns:

- **Weight.** The weighting factor applied to the particular point, this weight is used in the composite point computation.
- **Instrument.** Instruments that measured the point.
- **Error.** Component error (Err 1, Err 2, Err 3) shows the error contribution per component.
- **Mag.** The magnitude error for each point used in the composite point computation.
- **Ranking.** The percentage of the expected 3-sigma envelope consumed by the maximum measurement residual.

3. Solve Section

This section contains the parameters and controls for actually calculating the USMN solution (Figure 30-13). They are summarized below:

Figure 30-13. Solve Section Controls

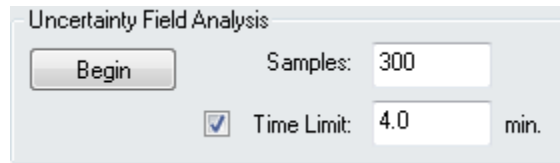


- **Auto Solve.** This button performs a “Best-Fit then Solve” (see below), trims outliers at 100% (see below), then performs a “Solve” (see below). This automated process is recommended for well-understood networks and in applications where the process has to be automated. It is not recommended when first optimizing an unfamiliar network of measurements. Anytime outliers are eliminated from the network, the causes and details should be characterized and documented.
- **Do This Automatically.** When the USMN dialog is displayed, an Auto Solve is automatically performed once.
- **Best-Fit Only.** Performs a best-fit of the networked points. This is equivalent to performing a traditional best fit. Performing a best-fit prior to initiating the USMN solve drastically speeds up the calculation, because it provides for a much better initial starting point for the calculation.
- **Best-Fit then Solve.** Performs a Best-Fit (see above) and Solve (see below) in one step.
- **Solve.** Initiates a USMN calculation from the instrument’s current positions. R-Click to Enable/Disable CUDA devices processing (see “[NVIDIA CUDA Devices](#)” on page 802).
- **Instrument Settings.** Allows you to modify properties for all instruments at one time.
- **Trim Outliers.** Removes points or observations exceeding a specified Ranking threshold from the USMN solution. Recall that ranking is an indicator of the quality of a point. Points above the ranking threshold can either be weighted to zero or removed from USMN entirely. Therefore, this is an effective means for preventing poor measurements from tainting the USMN solution.
- **Exclude Measurements.** Any measurements that should not be included in the USMN calculation can be excluded with this button.

4. Uncertainty Field Analysis Section

USMN does not by default calculate the uncertainties of the resulting optimal (USMN Composite) points, because this calculation can take a fair amount of time. As a result, you must initiate this calculation yourself (Figure 30-14).

Figure 30-14. Uncertainty field analysis controls.



Uncertainty field analysis works by using the instrument uncertainty characteristics to slightly perturb measurements following a realistic Gaussian distribution. Each iteration of perturbed measurements is then treated as a separate USMN network. Each iteration is solved in turn, which allows for statistical generation of optimal point uncertainties.

- **Samples.** Specify the number of perturbations, or iterations, to perform. Higher numbers take proportionately longer to solve, but have more accurate uncertainty values. It has been statistically determined that 300 or more samples will provide results within 5% of the “statistically true” value, so 300 is considered the default. About 10,000-15,000 samples are required to get a result within 1% of the statistically true value.
- **Time Limit.** When checked, the uncertainty field analysis will time out at this limit, regardless of the number of samples requested. In this case, the number of completed iterations will be used.
- **Begin.** Initiates the Uncertainty Field Analysis.

5. Summary

The summary section lists a general summary of the USMN calculation with the current settings. Each time the USMN network point list is updated or edited these value should be used as a review of the overall solution. The summary provides the following results (Figure 30-15):

Figure 30-15. Example Summary

```
Summary
Point Error: Overall RMS = 0.0015, Average = 0.0010, Max = 0.0085 '10'
System Solution Time: 0.0 sec, Robustness Factor = 0.130979, Unknowns 18, Equations 132
Uncertainty Magnitude: Average = 0.0009, Max = 0.0014 '10'
68.26% Confidence Interval (1.0 sigma), Samples: 300, WCF: A:\WORLD
Uncertainty Analysis Time: 4.7 sec
```

Pointing Error:

- **Overall RMS.** The RMS deviation of each measured point to the solved network points.
- **Average.** The average of each measured point to the solved network points.
- **Max.** Largest Point Error for the network. The point name with this showing this error is listed in parenthesis.
- **System Solution Time.** Time to arrive at the network solution.
- **Robustness Factor.** Is an indicator of the numeric instability of the solution and is the inverse of the Conditioning Number.

To understand the Condition Number, it becomes necessary to know a bit about singular value decomposition, singular values, eigen vectors, and eigen values – the sort of stuff that causes eyes to glaze over. The short description is that the process of numerical error minimization produces a system of matrices that are subjected to Jacobian operators and SVD to produce a set of orthogonal gradient vectors defined in solution parameter space that provide a direction towards a minimum (hopefully the global minimum) in solution space. Once the solution has moved to the solution point, these gradient vectors should be such that any movement away from the solution point will make the error worse. Ideally there will be as many orthogonal unique gradient vectors as there are solution parameters – the direction of steepest gradient will be a combination of these vectors.

The influence of each of these vectors is scaled by singular values, a large value means that small changes in the associated vector direction will have large impact on the solution result.

A small value means that the associated vector direction will have less impact on the solution result – in other words, the solution could wander further along this direction without substantially changing the solution result. Consider a point on a plane, (T_x , T_y , and R_z) are still unconstrained and could take on any value and still retain its status as a planar point – in solution space, a similar condition specific to a particular solution parameter means that that particular solution parameter (or some combination of solution parameters) is redundant.

The ratio between the largest and the smallest singular value is the Condition Number – condition numbers nearer to unity are more robust and those that are very large are not as robust. A condition number of 10000 (robustness = 0.0001) means that the largest contributing vector has 10000 times more effect than the smallest. This means that if the smallest of the solution gradient vectors was only dependent upon one of the solution parameters which was not otherwise used by any of other solution gradient vectors, then that parameter could vary significantly without significantly changing the

solution results – this would be a poorly resolved parameter. Even worse would be that one or more of the solution gradient vectors might not affect the solution result at all – each of these vectors will result in a negative integral robustness contribution.

A small Robustness (large Condition Number) value greater than zero is considered satisfactory for USMN.

A negative Robustness Factor indicates a reduced number of degrees of freedom in solution parameter space, along the solution gradient, at the solution point. Generally this means that redundant solution parameters exist relative to measurements. Alternately, an insufficient number of measurements exist to resolve solution parameters.

For example, a Robustness Factor equal to -3 would mean that three of these vectors had no effect on the solution, at the solution point specifically. This could indicate that the measurements taken are insufficient to resolve all of the solution parameters or that the solution is at a singular location. In either case a closer inspection of the results is recommended.

- **Unknowns.** The number of instruments in the network that are free to move times the 6 degrees of freedom considered in the solution (the first instrument in the list is not counted).
- **Equations.** The sum of the number of points from each of the instrument used in the network, times 3 to provide a 3 dimensional solution.

Uncertainty Magnitude:

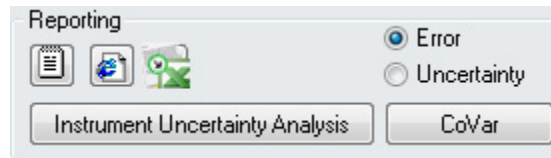
- **Average.** Average of the uncertainty magnitudes for each of the included network points
- **Max.** Largest point uncertainty for the network. The point name with this showing this error is listed in parenthesis.
- **Confidence Interval.** The Sigma uncertainty interval used for the calculation.
- **Samples.** Number of samples used for each point in the computation.
- **WCF.** Working Coordinate Frame used to determine XYZ uncertainty.
- **Uncertainty Analysis Time.** Time to compute the uncertainties for the network.

It also provides access to information about scale bars included in the USMN calculation.

6. Reporting Section

This section provides a set of simple export options and some extremely sophisticated analysis tools for reporting results (Figure 30-16).

Figure 30-16. Reporting Section



- **Reporting.** Three buttons provide a means to export the USMN results as a text file, and HTML file or an Excel file. These results are the same as those saved in the Event.
- **Instrument Uncertainty Analysis.** This button computes the individual instruments uncertainties based upon the network solution (It requires Uncertainty Field Analysis to be performed first).
- **CoVar.** Analytical uncertainty covariance matrices for both points and instrument base locations can now be generated as *.csv file output.

The covariance can be reported as either the Error or Uncertainty. For large networks that take a prohibitively long time to solve with the Monte Carlo approach (**Uncertainty Field Analysis > Begin**), the point uncertainties can be approximated with the Uncertainty Covariance matrices.

NVIDIA CUDA Devices

Using NVIDIA CUDA (Compute Unified Device Architecture) cores for USMN calculations provides an enhanced parallel processing option. It allows the use of the hundreds to thousands of cores contained within the GPU. On a decent NVidia graphics card (or alternately a Tesla module) this can significantly reduce the solve time for larger USMN networks. This requires a quality card and 64-bit SA.

There is an on/off option for this setting as a right-click function on the **Solve** button. This will activate CUDA devices if any are present and save the configuration to the registry so that it will persist between jobs. When CUDA devices are enabled/disabled, this transaction is logged so that the user can inspect the log for status.

Small USMN systems will not benefit from using CUDA devices. Each USMN solution sequence will take approximately 4 iterations to complete and the setup time to initialize and transfer data to the NVIDIA CUDA device simply adds more overhead time than is gained. So even if CUDA processing is enabled it will be ignored if an iteration takes less than 2 seconds.

Large USMN systems (30 instruments or more with 30 points or more per instrument) will benefit greatly from using NVIDIA CUDA devices.

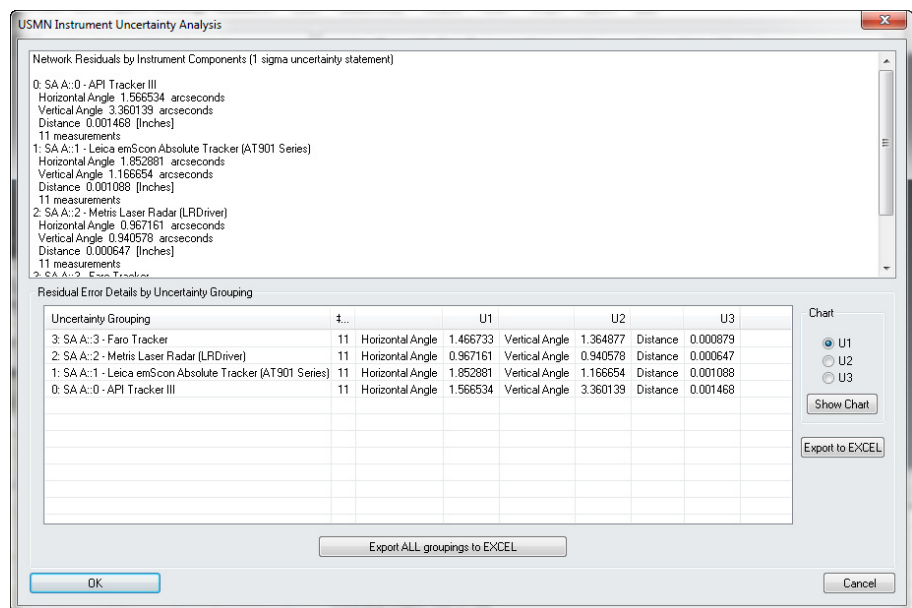
A test showed the following results:

A network featuring 551 instruments with an average of 30 points per instrument and solving this system on a dual XEON (8 CPU cores) running at 2.39 GHz required 40 hours to solve (5 iterations at ~8 hours each). Solving this same system using NVIDIA CUDA devices (QUADRO P6000) reduced the solution time from 40 hours to 23 minutes.

Instrument Uncertainty Analysis

Once an Uncertainty Field Analysis is performed for a network this network solution can then in turn be used to determine the resulting uncertainty parameters in the Instruments that measured those points based upon the solved network points. Press the **Instrument Uncertainty Analysis** button to access the results (Figure 30-17).

Figure 30-17. Instrument Uncertainty Analysis



This uncertainty is then reported per instrument for each of 3 Uncertainty Variable in the instruments properties (or XYZ for instruments with many variables).

The default instrument uncertainty values are generic values which have proven over time to provide an accurate estimate of the uncertainty of an instrument in an unknown environment under unknown conditions. However, if you wish to determine the best possible accuracy values for a particular instrument under your conditions you can use USMN to do so, using the following procedure:

1. Establish network of 8 or more fixed monuments to serve as control points.

2. Measure all the control points from at least 4 different physical locations using the same instrument, adding new stations to the job file for each instrument plant.
3. Perform a USMN on the network, compute the Uncertainty Field Analysis and check the Instrument Uncertainty Analysis results.
4. Copy the resulting values into your instruments properties, updating the instruments Uncertainty Variables.
5. If desired, rerun the USMN Analysis with the new Instrument Uncertainty Variables and compare the resulting Instrument Uncertainties.

Here is an example graph of the change in accuracy (Vertical & Horizontal Angle and Distance) as a result of 10 iterations of this process with an inhouse instrument. As you can see the values converge to a stable condition with repeated iterations.

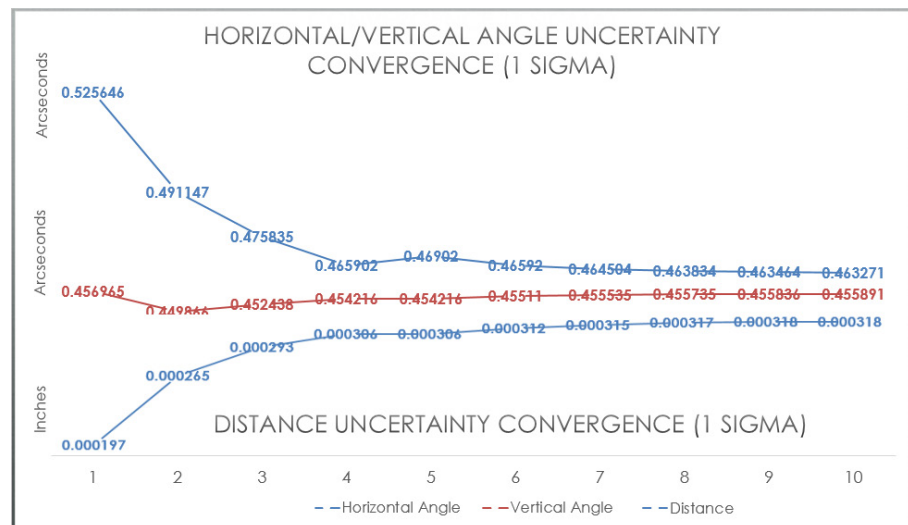


Figure 30-18. Angular & Distance Uncertainty Convergence

If you look at the change over the ten iterations, it quite small—less than the resolution of the encoders (Horizontal Angle Change ~0.06, Vertical Angle Change ~0.001 arcseconds and Distance Change ~0.00012”). The initial “guesses” for this solution (which are the default values in SA) turn out to be good enough to get close on the very first iteration.

Tip: Uncertainty Settings can easily be copied between instruments using the following command: **Analysis>Coordinate Uncertainty>Instrument Uncertainty Settings>Copy Settings Between Instruments**

This provides a means of evaluating the performance of your particular instrument under particular real-world conditions. Those values can then be used for the instrument uncertainties when you perform your true measurements and provide an even greater confidence that the uncertainty in your points reflect the results you will see with your instrument.

The Instrument Transform Uncertainty Fields are also computed and reported in the event as part of this process. A graphical representa-

tion of the position and orientation of this transform uncertainty can be displayed using **Analysis > Coordinate Uncertainty > Show Instrument Transform Uncertainty Fields (USMN)**. This will hide the instruments and display only the uncertainty graphics while enabled.

Covariance Analysis

Analytical uncertainty covariance matrices for both points and instrument base locations can now be generated as *.csv file output (as of SA version 2015.11.06).

Methods employed are those as described in the paper “On the Representation and Estimation of Spatial Uncertainty” by Randall C. Smith and Peter Cheeseman published in “The International Journal of Robotics Research”, Vol. 5, No. 4, Winter 1986.

χ^2 is the Chi-Squared error

M = number of observations (number of rows in A matrix)

N = degrees of freedom (number of variables to solve – number of columns in A matrix)

Nominally, $(M - N) = \chi^2$ when expected error is consistent with observed error.

Uncertainty covariance report includes:

- Composite point uncertainty covariance matrix data relative to working frame
- Instrument base uncertainty covariance matrix data relative to world frame
- Chi-square “Goodness of Fit” assessment (values $\ll 1$ indicate data too good to be true, values $\gg 1$ indicate measurement errors much greater than expected)
- Robustness: Solution solver SVD ((min(singular values) / max(singular values)) -- negative values reflect the input data does not provide enough information to resolve all of the degrees of freedom available to the model. Note that model degrees of freedom does not mean a one-to-one mapping to solution parameters. Eigen vectors are a composition of solution parameters similar to the I, J, K vectors of a normal vector. This only defines a single degree of freedom, but it requires input for the displacement along each of the coordinate system XYZ axes.
- Singular value tolerance threshold (minimum value an singular value is permitted to assume to be considered non-zero)
- Number of input equations
- Number of ignored equations

- Number of active equations
- Number of variables
- Number of independent variables
- Working frame
- Composite instrument base parameter covariance matrix
- Composite instrument base parameter correlation matrix

The input points used to develop a composite point are used to generate an error covariance matrix. This data may be helpful in terms of evaluating the data spread of the input points about the composite point. This data will not be generated for any composite point with less than three input points.

R-Click on “CoVar” button will allow the user to experiment with different values of singular value tolerances (default is 0.01).

7. Apply Results Section

The Apply Results section controls how the calculated results are applied to the job file (Figure 30-19).

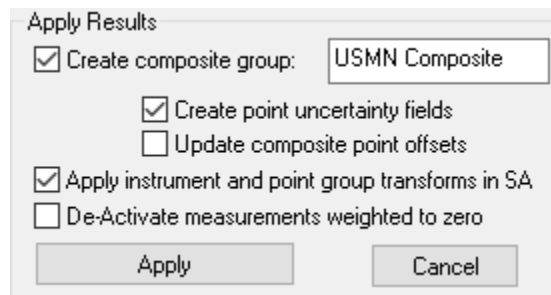


Figure 30-19. Apply Results Section

- **Create Composite Group.** When checked, creates the optimal (USMN Composite) point group, using the name specified in the user entry field.
- **Create Point Uncertainty Fields.** When checked, applies uncertainty fields (calculated via the Uncertainty Field Analysis) to the USMN composite points. These fields represent the resulting uncertainty of the network points.
- **Update composite point offsets.** This option provides the ability to pass point offsets on to the composite point group provided that all the input shots have the same offsets.
- **Apply instrument and point group transforms in SA.** If unchecked, no instruments or points in SA will be physically moved in the file.
- **De-Activate measurements weighted to zero.** If checked, any measurements that have been weighted to zero in the

Measurement Section will be deactivated in SA.

Basic USMN Solution Process

The following describes the steps for a basic USMN solution process, without calculating uncertainty fields for the USMN composite group.

1. Ensure that the job file matches the requirements in Observations and Point Group Requirements for USMN.
2. Select **Analysis>Coordinate Uncertainty>Unified Spatial Metrology Network**.
3. When prompted, select the instruments to include in the USMN network, then press **Enter**.
4. The *USMN* dialog will be presented (see USMN Dialog).
5. Ensure that the instrument settings are correct (moving/non-moving, weights, DOF constraints, etc).
6. Click **Best-Fit then Solve**.
7. Examine the max error and ranking values.
8. If desired, click the **Trim Outliers** button and trim any measurements beyond your desired ranking.
9. Click the **Solve** button and observe the results.
10. Verify that **Create composite group** is checked.
11. Click the **Apply** button. The instruments/point groups will be located, and a USMN Composite point group will be created representing the ideal position of each measurement.

For a more in-depth example on basic USMN, see the tutorial Basic USMN.

Uncertainty Context Manager

USMN has been the industry standard for network alignment for many years. However, it has had its critiques, the largest of which is the solution speed. The Monte Carlo simulation process used for uncertainty modeling by USMN can be slow for large networks and the uncertainty clouds generated by the process require additional storage. It is also difficult to use USMN for further uncertainty propagation or modeling purposes beyond the network solution.

The intent of the Uncertainty Context Manager is to provide an alternative means to use point uncertainties in an optimal manner, using covariance matrix analysis. This mathematical approach increases computation speed and reduces data storage. It has also been developed through a context dependency structure, much like the SA tree bar, that facilitates greater control over the progression of uncertainties. It allows a user to build a series of dependent solutions, providing more comprehensive modeling capabilities.

Building a Network Solution

To use the Context Manager to perform a network alignment, select **Alignment > USMN > Uncertainty Context Manager** (or in the classic menus: **Analysis > Coordinate Uncertainty > Uncertainty Context Manager**) (Figure 30-20).

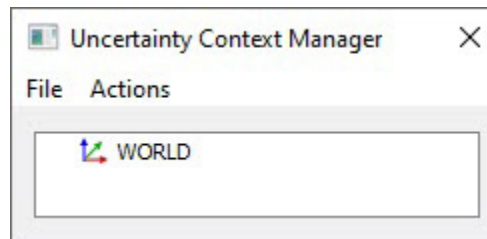


Figure 30-20. Context Manager

This unimposing dialog provides the foundation within which the uncertainty context for your job file will be built. It offers a tree structure for visualizing the dependencies in that process.

Use the **Actions** option to add a new network...(Figure 30-28).

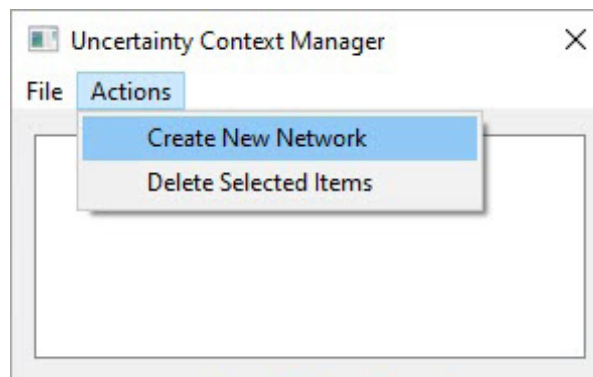


Figure 30-21. Actions tab offers the ability to Create a New Network.

Doing so will trigger the following prompts:

1. *Select Network Working Frame.* This provides an option to choose the base frame upon which the uncertainty computation is built.
2. *Select Instruments in the Network.* Select all the instruments that you wish to include in the solution.

You will then get an opportunity to exclude points measured by the selected instruments within the Network Build Settings dialog (Figure 30-22). Only include points you wish to use with the network solution (the same option is also available through USMN).

The expectation is that points from different groups with the same names included in this network represent separate observations on the same target and will be aligned as part of the computation.

Be sure to exclude points with the same name as the alignment points that do not represent the network points. If SA detects this condition it will mark the “Possible Collision” in **red**.

Points measured by only a single instrument station are marked in **blue**. These points are identified to help you rename them if they are part of the network solution. Addition points measured by only a single instrument station can also be included for uncertainty analysis if desired but will not impact the network solution.

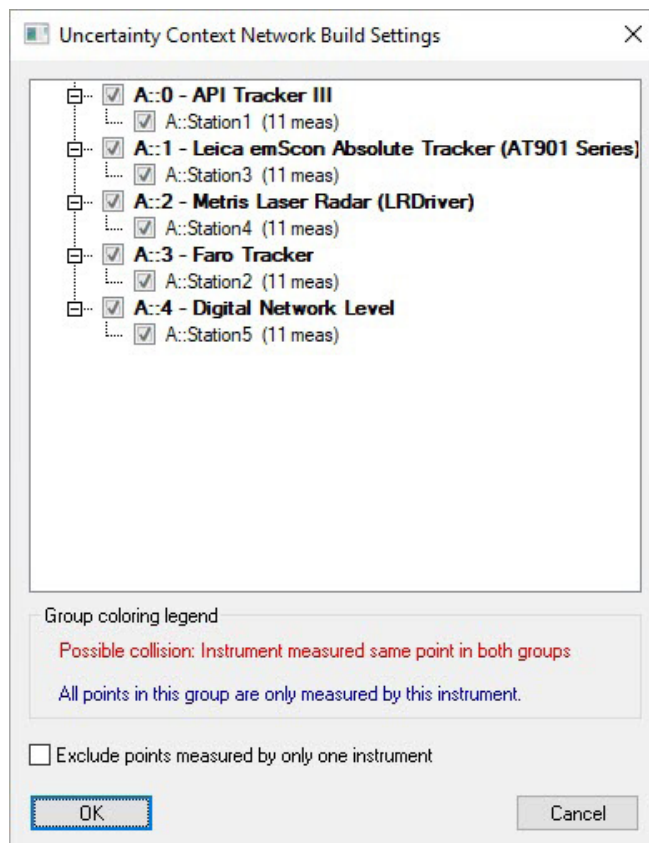


Figure 30-22. Uncertainty Context Network Build Settings offers a means to select which points are included for analysis within the network.

3. *Select Nominal Point Groups in the Network.* Much like USMN with Point Groups this option allows you to select a reference point group.
4. *Select Scale Bars.* Scale bars can also be included in the network.
5. The resulting network name that is added within the Uncertainty Context Manager. The default name is: "Uncertainty Context Network" which can be renamed as desired.

Accessing the Solution Interface

The **Solution Interface** can be directly accessed by right-clicking on the network in the Uncertainty Context Manager (Figure 30-23).

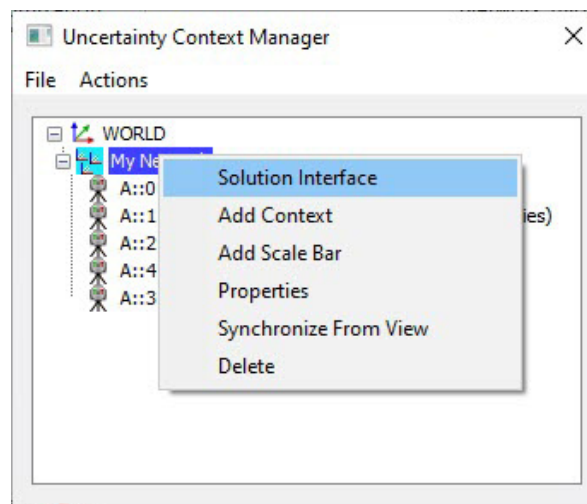


Figure 30-23. Accessing the Solution Interface for a single network.

Additional controls also offer the ability to update the existing network definition and include:

- **Add Context.** Provides a means to add an additional instrument or point group with uncertainties to the network solution.
- **Add Scale Bar.** Add an additional scale bar to the network solution.
- **Properties.** Inspect the properties of the network.
- **Synchronize From View.** Points included in the network solution are not automatically removed if a point is deleted from within SA A primary view. This offers an option to synchronize the results.
- **Delete.** Provides a means to remove items from the solution.

When the Solution Interface opens, it should look quite familiar in that it is based on the USMN network interface control and offers many of the same options. One advantage you will notice right away is that the point uncertainties are immediately displayed (Figure 30-

24) based upon potentially unsolved current context status.

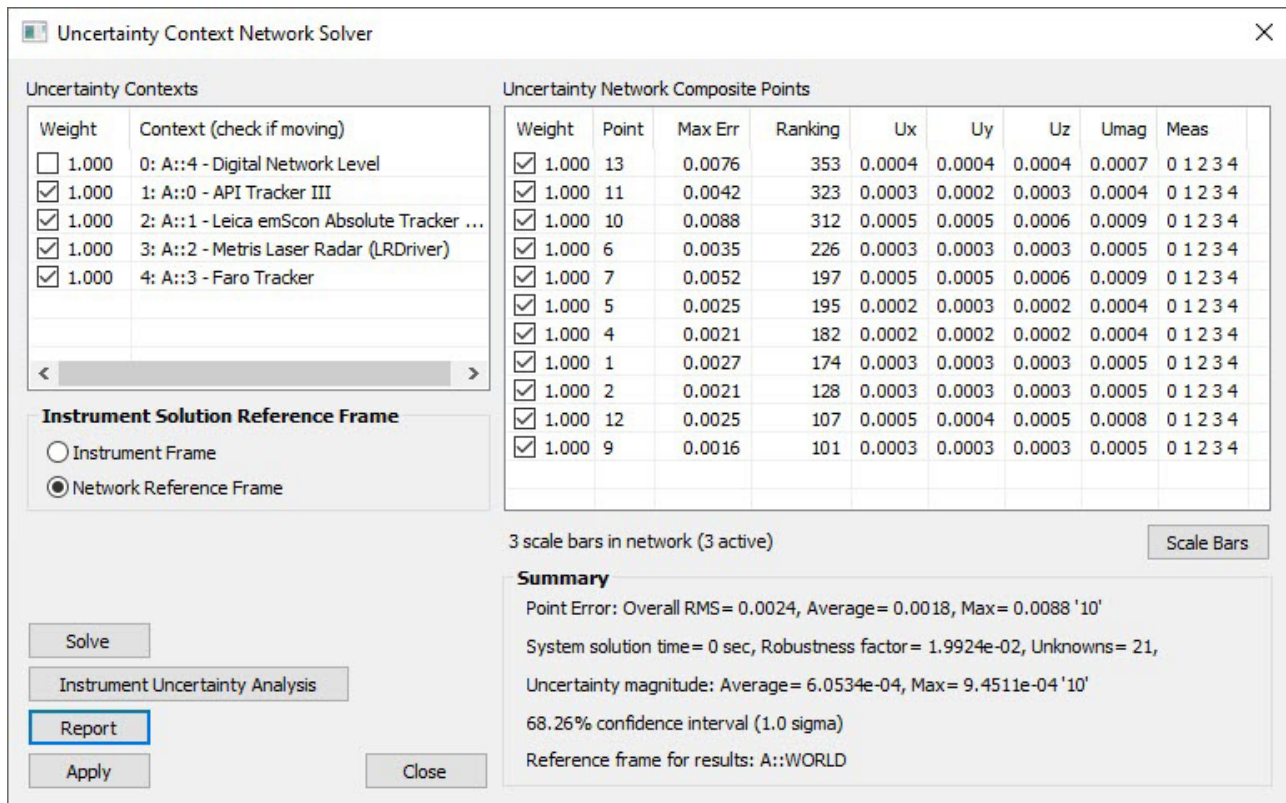


Figure 30-24. Solution Explorer

Just like USMN, the contributing instruments or context are shown in the top left panel while the composite points are listed on the right.

- Max Error is an excellent measure of the quality of a composite point as it reports the greatest contributing deviation.
- The Ranking represents the worst case error for the input points with respect to the composite point divided by the 3-sigma uncertainty magnitude for the composite point. Large values are not necessarily bad points if one of several input points happens to have been far more distant from its parent context base.

The solution summary information and statistics are on the bottom right once the network has been processed by pressing the **Solve** button. Important notes include:

- The RMS reported here is in regards to a 3D Magnitude deviation as opposed to only the component deviation. This means that the values will be larger than those reported from USMN by a factor of $\sqrt{3}$.
- A Goodness of Fit value is reported that reflects how well the actual error agrees with expected measurement uncertainty. Ideally this number should be close to 1 and either very large

or very small values warrant closer scrutiny.

The **Report** button will generate a full report in Excel format.

The **Apply** button can be used to apply the results to the SA job file and build a composite point group using the current solution. This will also update the instrument base covariances with respect to the world frame for all participating instruments.

Like USMN the decision to fix one or more instrument in the list, by unchecking it and preventing them from moving in the solution, will result in the uncertainties propagating outward from those fixed points (Figure 30-25).

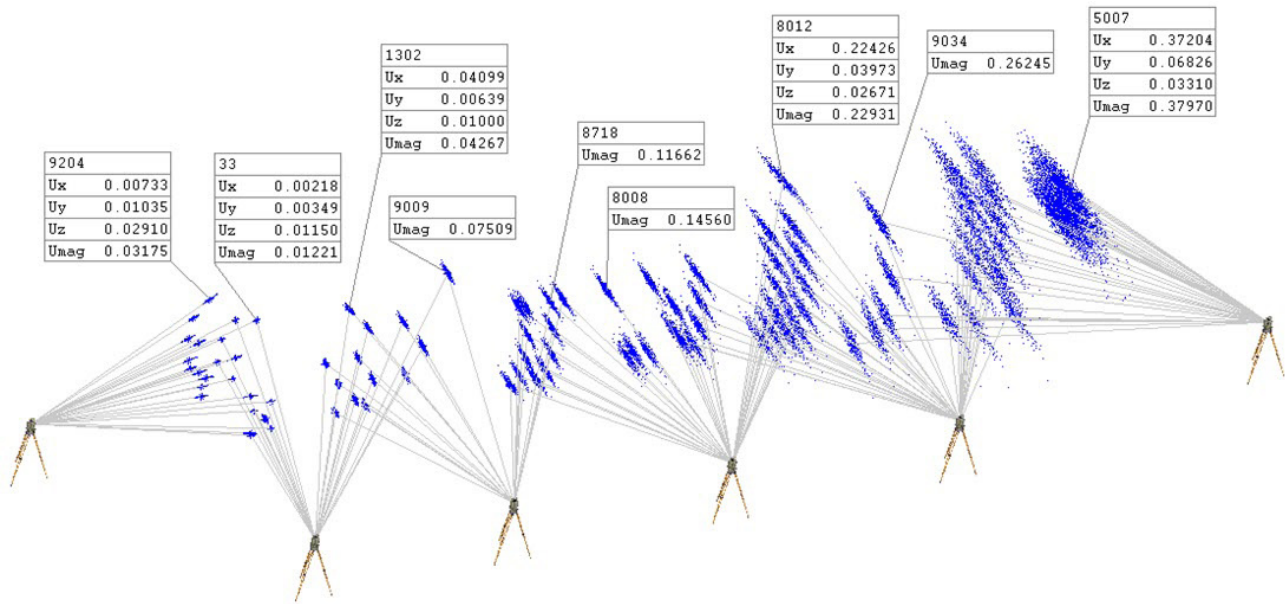


Figure 30-25. Example of error expansion from a single fixed instrument

Accessing More Details

Point Uncertainty Information

Right-click on a point to access its properties (Figure 30-26).

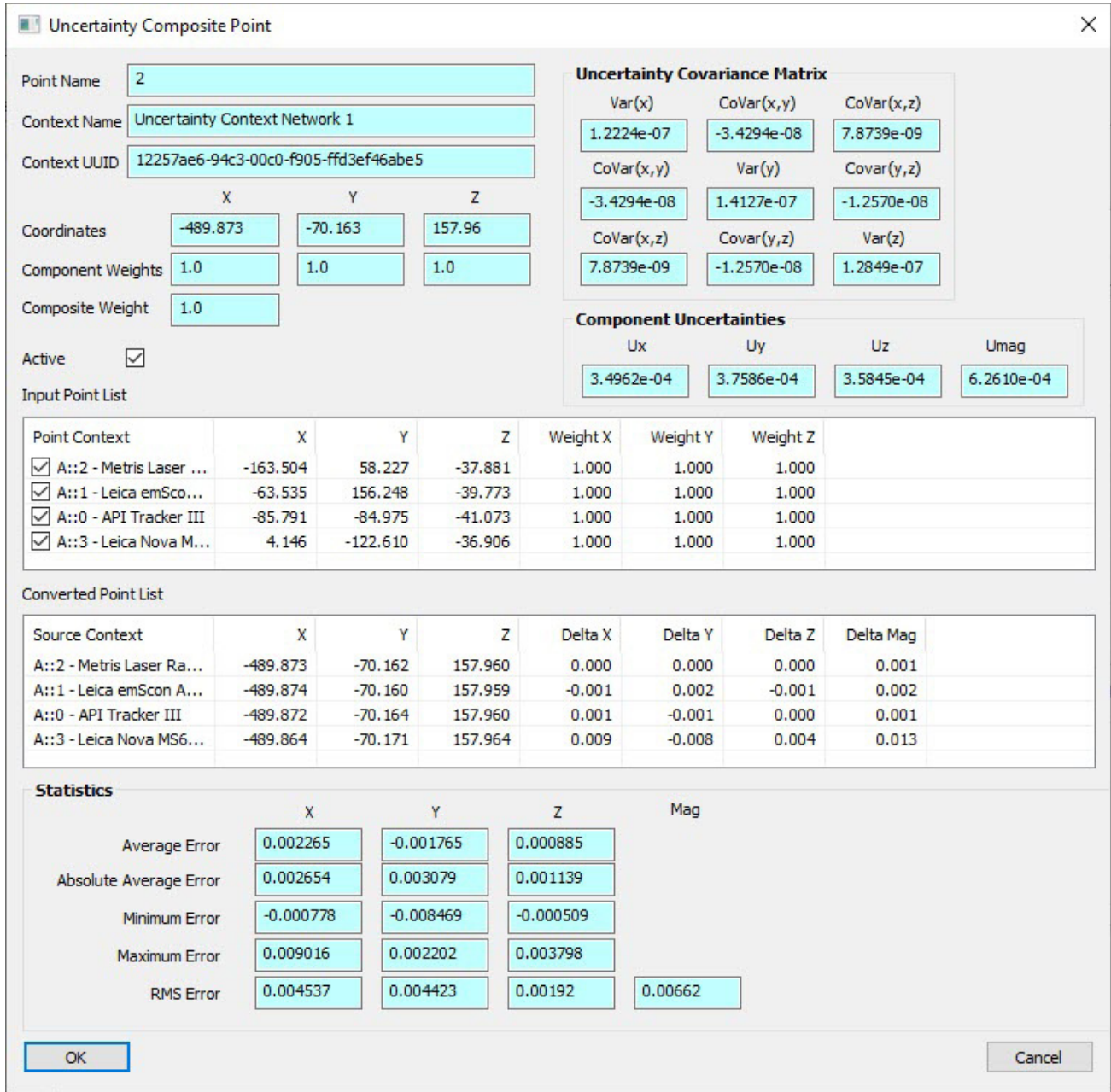


Figure 30-26. Composite Point details accessed by right-clicking on a point in the network list.

The Uncertainty Composite Point dialog offers a comprehensive list of the contributing information for the composite solution.

At the top of the dialog is the computed uncertainty information.

The **Input Point List** provides control over what points are included in the solution and their original coordinates.

The **Converted Points List** Defines the resulting point locations following the solution or where they will end up.

Right-click again on any of the input points or converted points to ac-

cess their individual covariances (Figure 30-27).

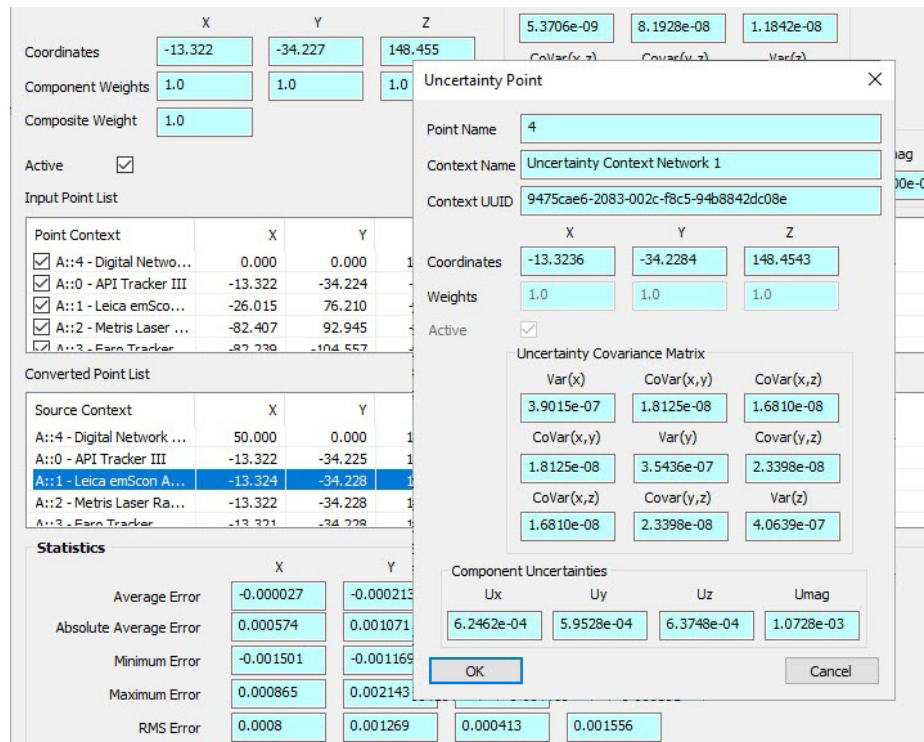


Figure 30-27. Point covariance

The resulting composite point statistics are on the bottom of the page including max, min, and average statistics.

Instrument Uncertainty Information

From the Solution Interface you can right-click on any individual instrument or "context" to access its properties and covariance information (Figure 30-28).

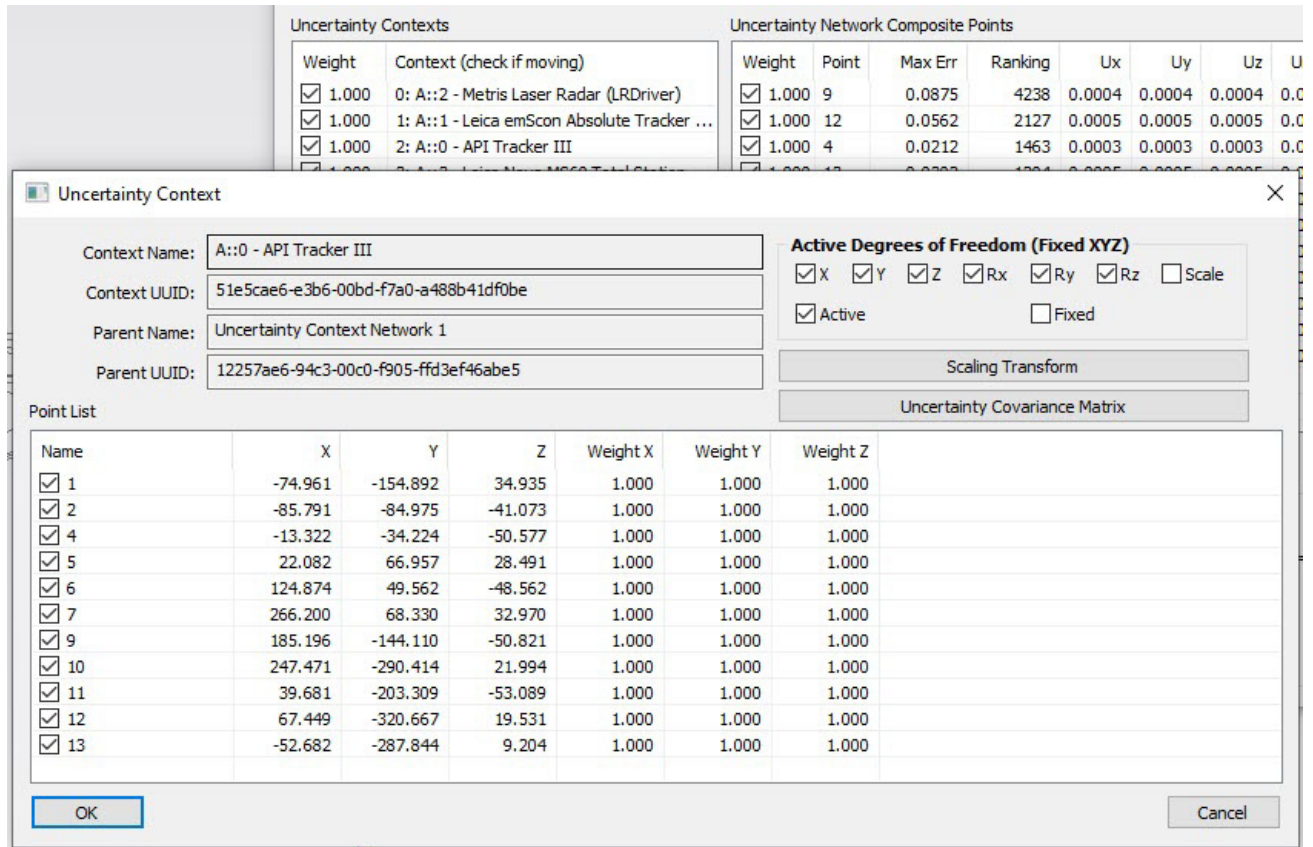


Figure 30-28. Instrument properties

This dialog provides access to the *Active Degrees of Freedom* for the instrument. These are the degrees of freedom controlled by the network solution.

The **Scaling Transform** and **Uncertainty Covariance Matrix** reflect the position of the instrument and its base uncertainty.

This dialog also offers a means to directly access the points measured by the instrument and included in the solution. Right-click on a point to access its covariance matrix. This will match the covariance of the input point also available through the composite point properties.

Instrument Solution Reference Frame.

Like USMN there are two options to consider when restricting degrees of freedom:

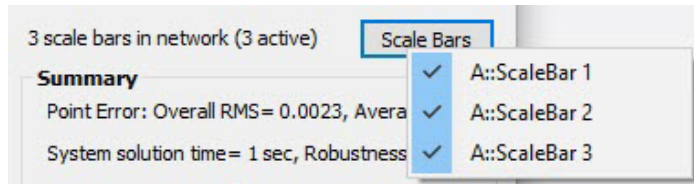
- **Instrument frame.** This option allows each instrument to rotate about their own standing axis, and should be considered if instruments are constrained to level and have level compensators.
- **Network Reference Frame.** This option provides a means to

align instruments relative to each other in any orientation and hold that orientation independent of their own standing axis.

Working with Scale Bars

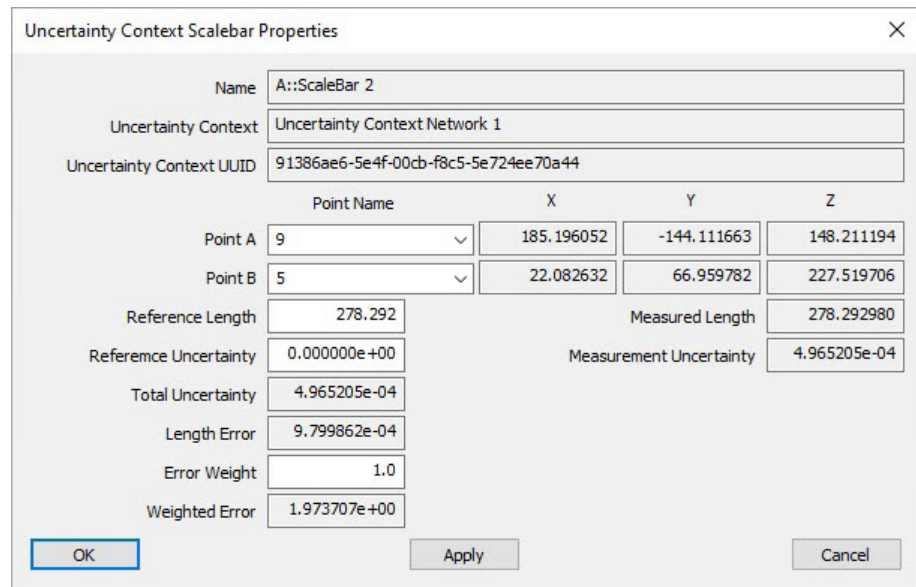
Scales bars are independently selected when building a new solution, can be incorporated within a network, and are included by default in the solution (which is a difference from USMN). The scale bar status and control button are available under the Composite Points List (Figure 30-29).

Figure 30-29. Left Click on the Scale Bars button to access the active scalebar list



Right-click on any of the individual scale bars to access the Scalebar Properties dialog(Figure 30-30).

Figure 30-30. Scalebar Properties



Incorporating Digital Levels

Digital levels are unique instruments in that they are intended to constrain the network in Rx, Ry, and Z. When you add a digital level it is fixed by default in the instrument list, for this reason. However, it does not fully constrain the solution, meaning that the solution is free to float and will not solve without additional constraints.

This can be solved in either of two ways. You can manually fix an additional instrument's location. This is done by opening an instruments properties from the network list and modifying its degrees of freedom. Fix R, Y and Rz as shown (Figure 30-31).

Figure 30-31. Fix or uncheck the degrees of freedom not controlled by the digital level as a reference for the network.

Active Degrees of Freedom (Fixed XYZ)

X Y Z Rx Ry Rz Scale

Active Fixed

Alternatively, the solution will do this for you automatically using the next instrument in the list if the network remains under constrained.

Expanding a Network Solution

The Uncertainty Context Manager offers the ability to add additional instruments to an existing network. This is accomplished by right-clicking on the network and selecting **Add Context**. The additional instrument will then be added to the network and the solution Interface will reflect this.

Alternatively a dependant network can be added. For example if you have an established network using multiple instruments you can add an instrument that is effectively best fit into the composite group. To do so,

1. Select *Actions* and add an additional network.
2. Select the instrument you want to add
3. Select the composite group from the first network as the reference group.

This will build a second network solution in the Uncertainty Context Manager (Figure 30-32).

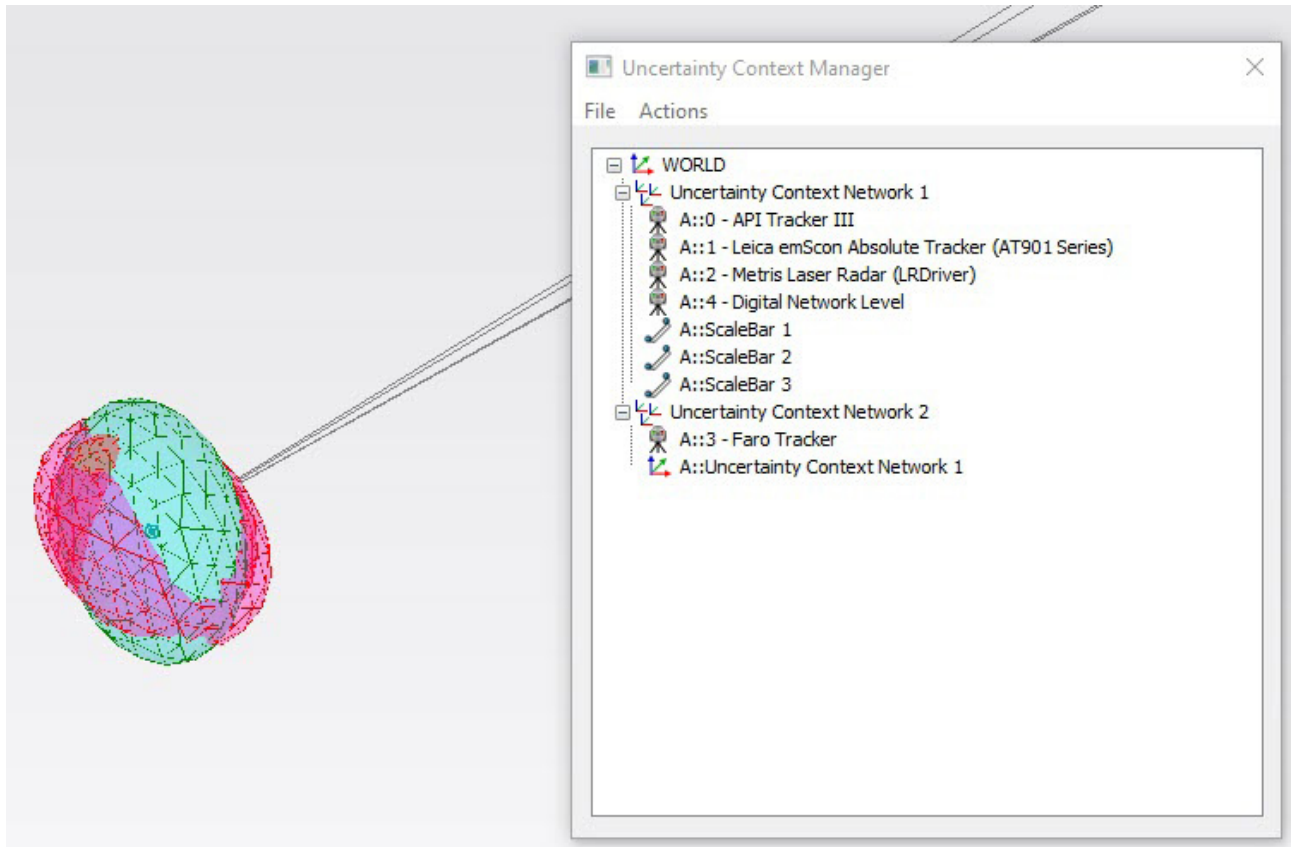


Figure 30-32. Context manager with two network definitions, the second based on the first.

CoVariance with SA

Observation Properties

The instrument properties dialog now includes a **CoVariance** button which offers direct access to the covariance matrix for the instrument base. This is updated based upon the last covariance analysis performed, either through USMN or the Uncertainty Context Manager.

The covariance for a single point observation can also be directly accessed from within the point's properties ([Figure 30-33](#)).

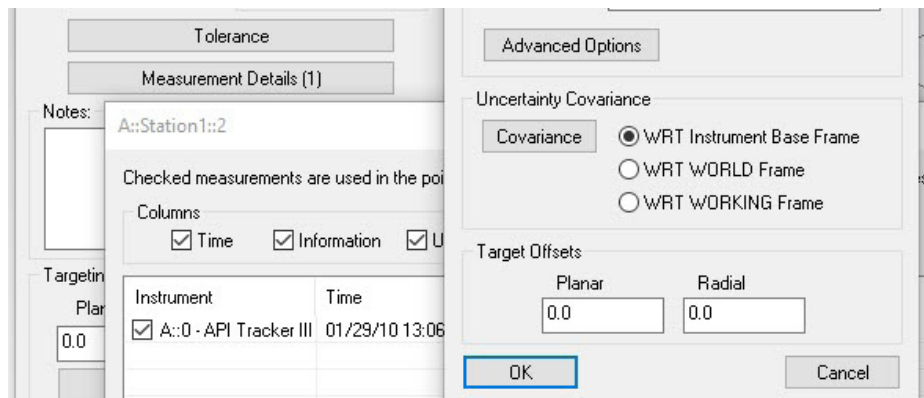


Figure 30-33. Point Covariance

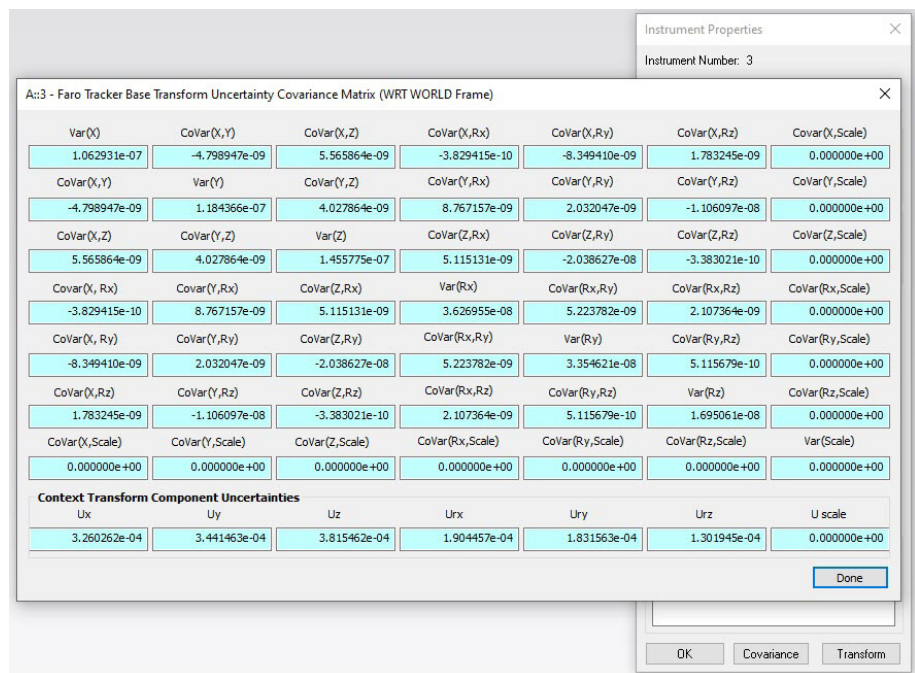
The point covariance can be displayed in either of 3 forms:

- **WRT Instrument Base Frame.** With Respect To (WRT) the instrument's base. This option represents the points uncertainty only.
- **WRT World Frame.** With Respect To (WRT) the World Frame. This option includes both the point uncertainty and the instrument's placement uncertainty with respect to the World frame.
- **WRT Working Frame.** With Respect To (WRT) the Working Frame. This option includes both the point uncertainty and the instrument's placement uncertainty with respect to the Working frame.

Instrument Properties

At the bottom of the instrument properties dialog there is now a **Covariance** button. This option offers direct access to the saved covariance or uncertainty in both position and orientation for the instrument. These values are saved following the last covariance evaluation from either USMN or the Uncertainty Context Manager (Figure 30-34).

Figure 30-34. Example Instrument Covariance



This Page Intentionally Left Blank.

Working With Surfaces

CHAPTER

31

Working with CAD surfaces is a discipline in itself. As more and more industries move toward computer model-based representations and away from paper drawings, the ability to work with surfaces grows every year.

Surface Grouping

Surfaces in SA are entities that exist in the SA tree bar and can be selected and colored individually. Each surface may be composed of many *surface faces*. These individual faces can only be selected or handled independently from the entire surface in certain modes or with particular commands. In general when querying points to a surface for example, the points are compared to the closest face included in the selected surface.

Therefore surface grouping is a process of tree management and ease of access.

Any set of surface faces or set of surfaces may be combined into a single surface or split into separate surfaces. The process of combining surface faces is referred to as *merging surfaces*, and the process of separating surfaces is referred to as *dissecting surfaces*.

The advantage of have a merged surface is that it simplifies the tree. When surfaces are merged, they appear as a single surface in the tree, therefore there are less surface entities to wade through when moving around in the tree. This also can make selection easier, because selecting a single surface selects all of the faces that are part of that merged surface.

However, merged surfaces have some limitations. A merged surface cannot have its normals corrected—so, if you need to reverse some surface faces, the surface must be dissected first.

Constructing a Merged Surface

The process of merging a surface involves selecting individual surfaces and combining them into a single surface in the tree that is treated as a single surface. When merging surfaces, you have the option of keeping the original surfaces in addition to the new merged surface, or deleting the original surfaces. In addition, you also have the option of hiding the original source surfaces as well.

To Merge Surfaces:

1. From the menu, select **Construct>Surfaces>From a Collection of Surfaces**.
2. Select all of the surfaces that you'd like to merge.
3. The **Group Surfaces** dialog appears (see Figure 31-1).
4. Select the desired options, then click **OK**. A new surface named **Merged Surface xxx** will be created.

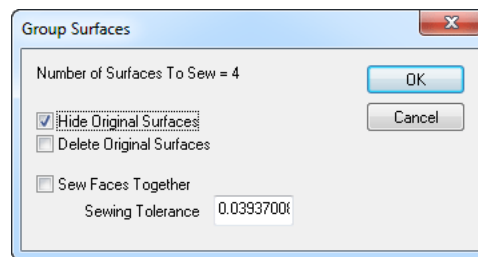


Figure 31-1. Merging a set of surfaces.

- **Hide Original Surfaces.** When selected, the original source surface faces will be hidden, but not deleted.
- **Delete Original Surfaces.** When selected, the original source surface faces will be deleted.
- **Sew Faces Together.** When selected, adjacent surface faces which lack tangency (they do not smoothly flow from one surface face to another) will be stitched together in accordance with the *sewing tolerance*.
- **Sewing Tolerance.** The maximum normal deviation from one surface face to the next in which those adjacent surface faces will be stitched together if *Sew Faces Together* is enabled.



Warning: Since sewing faces together can change the topology of a CAD model (usually undesirable in metrology), it is strongly recommended that you avoid this option unless you have a very valid reason for using it.

Dissecting Surfaces

Dissecting surfaces is the inverse operation of merging. Given a merged surface containing two or more surface faces, dissecting that surface will extract the selected faces and separate them as unique surfaces in the tree. Dissecting a surface leaves the original surface intact—it will be hidden for you in the tree.

To Dissect an Entire Surface:

1. From the menu, select **Construct>Surfaces>By Dissecting Surfaces>Entire Surface**.
2. Select the surface to dissect.

To Dissect Specific Surface Faces from a Surface:

1. From the menu, select **Construct>Surfaces>By Dissecting Surfaces>Select Faces**.
2. Select the individual faces to extract. The original surface will remain untouched, and the extracted faces will be created as separate surfaces.

Show Selected Faces Command

In the right-click menu of the Surfaces category in each collection is an option to **Show Selected Faces**. This option provides a shortcut to accessing only the surface faces of interest from a larger CAD model. When selected it will:

1. Prompt the user to select CAD Faces graphically from any of the visible surfaces, and press enter when done.
2. When selection is complete a new surface will be created using a copy of the selected CAD faces in the working collection and hide all the other surfaces.

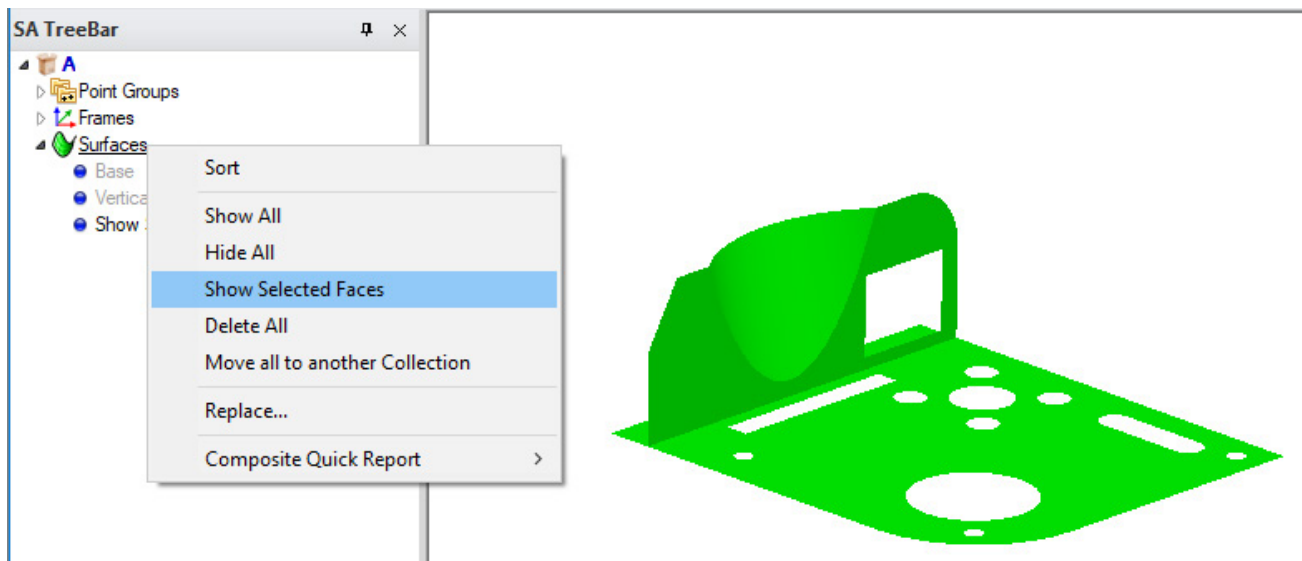


Figure 31-2. Show Selected Faces

Duplicate Surface Faces

Occasionally, you may encounter one or more surfaces that have overlapping surface faces. This may be a problem in the source mod-

el, a result of how the surface was imported, or simply a result of accidentally duplicating surface faces.

Duplicate surface faces can cause confusion in analysis, so it's best to clean these surfaces up before using them. Fortunately, SA has a command to detect and delete duplicate surface faces.

To Delete Duplicate Surface Faces:

1. From the menu, select **Edit>Remove Duplicate Surface Faces**.
2. Select the surfaces to examine.
3. Provide a coincidence tolerance. This indicates how close two overlapping surfaces must be to be considered duplicates.
4. If duplicate faces are found, confirm the deletion.

Surface Normal Conditioning

As explained in "SA Fundamentals", the normal direction of surfaces is critical to how offset compensation is applied to your measurements. An incorrectly-oriented surface normal can lead to dramatically incorrect analysis results. CAD surfaces should always have their normals facing away from the material of the part. That is, the side on which you approach the surface with a probe must be defined as the positive side of the surface.

The use of surface backside coloring (see "The Graphical View") enables you to determine if any surface normals are reversed. Whenever you import a CAD model, you should therefore check to ensure that surface normals are correct, and any incorrect normals should be fixed.

There are three methods for flipping surface normals: indicating point on positive side, reversing normals, and auto-reversing from viewpoint. All of these methods require a surface to be dissected into its individual surface faces before they are reversed.

Indicating Point on Positive Side

One method for correcting surface normals is to use a reference point as an indicator for whether that point should end up on the positive or negative side of a surface. This is similar to some metrology software that expects an additional measured point to indicate the direction of offsets for measured geometric surfaces.

To Condition Surface Normals by Indicating Point on Positive:



Note: This method currently compares the reference point to the middle of each surface face. Therefore, it is not 100% reliable.

1. Dissect the surface as described in "[Dissecting Surfaces](#)" on page 822.

2. From the menu, select **Edit>Surface Normal Conditioning>Indicate Point on Positive (or negative) side**.
3. Select the point to serve as the reference point.
4. Select the surfaces to consider.
5. At the *Decision* dialog, indicate whether the reference point should end up on the positive or negative side of the surface.


Reversing Surface Normals

The most straightforward method for reversing surface normals is to explicitly select the surfaces you'd like to reverse. The normal direction of each selected surface will be flipped.

To Reverse Surface Normals:

1. Dissect the surface as described in “[Dissecting Surfaces](#)” on page 822.
2. From the menu, select **Edit>Surface Normal Conditioning>Reverse Surface Normals**.
3. Explicitly select the surface faces you'd like to reverse.


Auto-Reversing Surface Normals


 **Note:** You may need to perform this operation from several different viewpoints.

SA can use the current view orientation to automatically reverse a set of surface faces. This has the advantage of reversing a large number of surfaces at once, without requiring you to select the specific surfaces you'd like to fix.

With this command, you orient the view so that you can see the surfaces you'd like to correct. You then select these surfaces, and the key is that you can also select surfaces which are already facing the proper direction. SA will go through each selected surface, determine if it's visible from the current viewpoint, and then reverse it if its normal does not generally point toward the viewpoint.

To Auto-Reverse Surface Normals from Viewpoint:

 **Note:** Occasionally, you may need to manually reverse a stray surface face.

 **Tip:** If there is not a large number of surfaces, it's generally faster to just select all surfaces using rectangular selection or the F2 window versus trying to select only those surfaces that need to be fixed.

1. Dissect the surface as described in “[Dissecting Surfaces](#)” on page 822.
2. From the menu, select **Edit>Surface Normal Conditioning>Auto Reverse Surface Normals from Viewpoint**.
3. Orient the view so that you can see the surface faces you'd like to correct.
4. Select the surfaces you want the command to consider. (It's OK if these surfaces are already facing the correct direction).

Surface Construction Options

From a Collection of Surfaces

Merges selected surface into one surface entity.

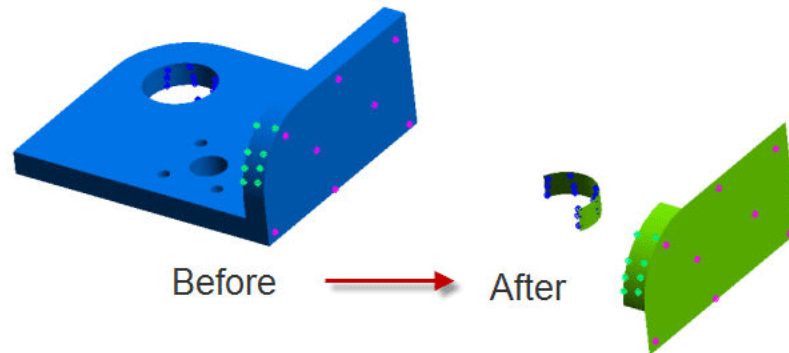
By Dissecting Surfaces

Dissects the surface into individual surfaces. The user has the option to dissect any surface or pick surface faces to dissect.

Select Faces by Projecting Points

Creates separate surfaces by checking point projections. This can be useful for speeding analysis when trying to reduce surface faces to analyze.

Figure 31-3. Creating surfaces by projecting points.



From Objects

Creates surfaces from supported SA Geometry. Examples are plane, cylinders, etc...

From Plane, Cylinder, Sphere, and Cone

Performs same functionality as above, these are just geometry specific. The above command is more universal.

Tubular Surface from a B-Spline

Creates a tubular surface that uses the B-Spline path. Users has the ability to enter Tubular Diameter.

Advanced Surface Fitting

There are several more advanced surface construction tools that are available and discussed in more detail in the Analysis Chapter. Please refer to [“Advanced Surface Fitting”](#) on page 434.

- **From Point Groups.** This option builds a surface using sequential point groups.
- **From B-Splines.** This option builds a surface from sequential B-splines.
- **From Plane and Closed B-Spline.** This function builds a bounded surface from a plane and a closed B-spline.
- **Fit from Bounding Curves and Internal Points.** This option builds a surface from a curve and measurements.
- **Fit from Nominal Face and Actual Data.** This function deforms an existing surface to fit measurements.

Meshing Options are also available. While not a true surface, a mesh provides a means to create a triangulated representation of a true surface in a light weight format “[Meshing Point Clouds](#)” on page 601.

B-Splines

Properties

A B-Splines, which can be referred to as a curve interchangeably, defines a polynomial trajectory through space. It visually expresses a polynomial curve of a given degree and is built from a series of “control points” and the minimal smooth bends between them. Curve properties provide basic spline definition parameters and allow directional arrows to be drawn.

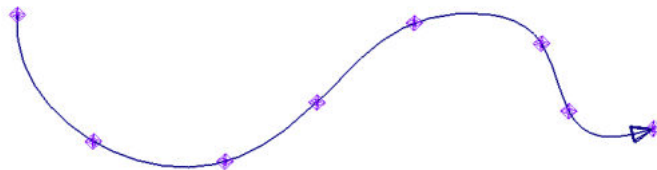


Figure 31-4. The B-Spline curve properties dialog.

B-Spline Construction

From Points

Figure 31-5. A B-Spline fit to a series of measured points.



Creates a B-Spline curve from selected points using either an interpolated fit or cubic fit (see [Figure 31-6](#) and [Figure 31-7](#)).

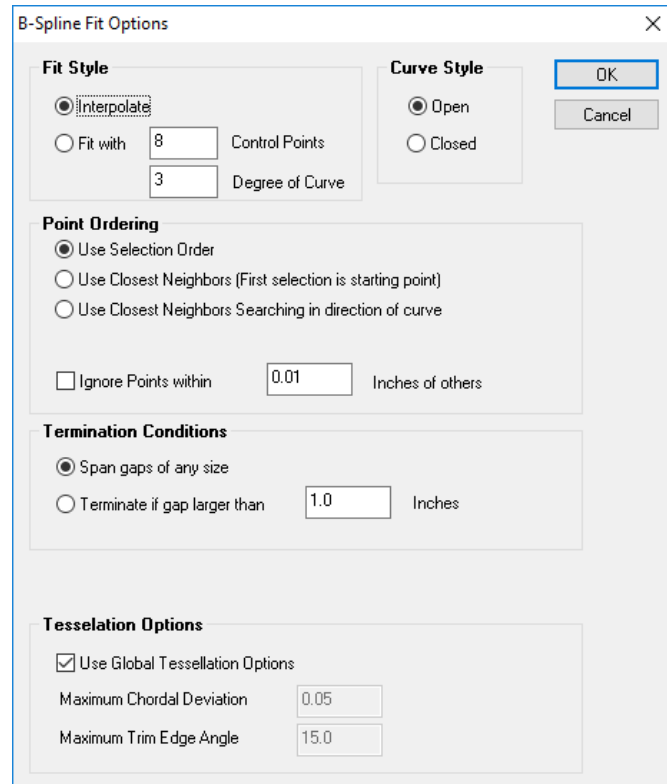


Figure 31-6. B-Spline Fit options.

Fit Style. A B-Spline can be fit to points exactly with a specified number of control points and degree of curvature.

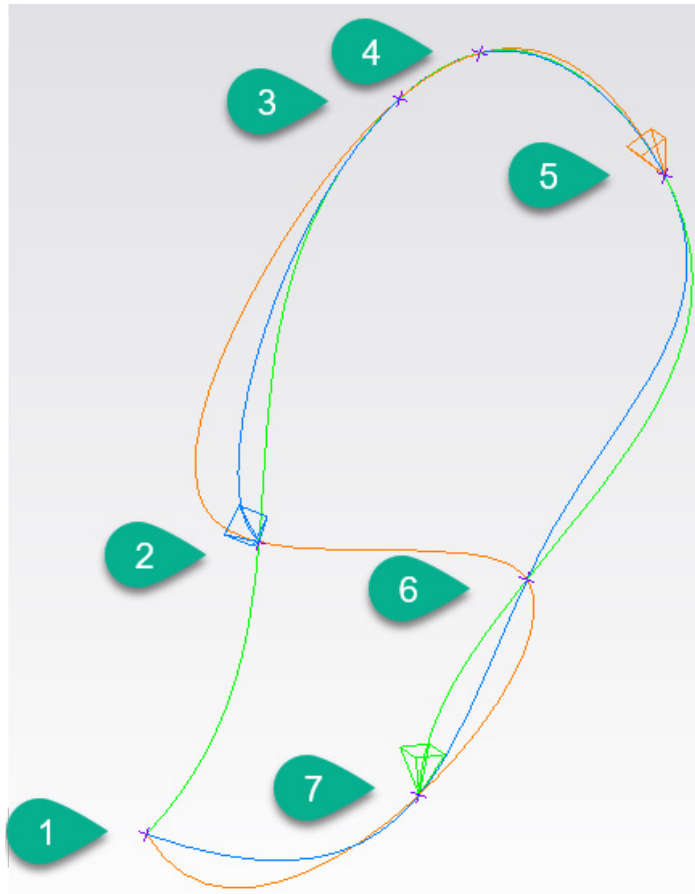
- **Interpolation.** An interpolated curve will fit each selected point exactly and find the best solution to get from one to the next in order to minimize the degree of curvature.
- **Fit with.** Fitting with a specified number of Control Points and Degree of Curve allows a user to define the complexity of the curve. The points will be fit as best they can within these constraints.
- **Curve Style.** An **Open** curve starts at the first point and ends at the last. **Closed** B-splines will end at the beginning point of the B-spline.

Point Ordering. The direction and path of the B-spline is determined by the ordering of the points. The user can select the point ordering as follow:

- **Use Selection Order.** The curve will fit the selected points in the order selected by the user. ****Note that when F2 selection is used the database (tree) order will be applied.**
- **Use Closest Neighbors (First selection is starting point).** This option starts with the selected point and fits the points based upon sequential proximity.
- **Use Closest Neighbors Searching in direction of curve.** This

option looks for the closest neighbor following the direction of the curve as it is built. This may ignore a closer point to prioritize a point closer to the direction of the curve.

Figure 31-7. Fit results when applying different Point Ordering options: 1. Use Selection Order (Green); 2. User Closest Neighbor (Red); 3. Searching in direction of curve (Blue).



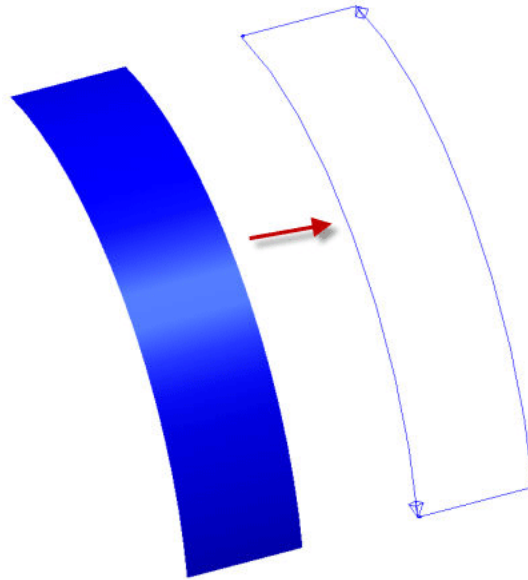
- **Ignore Points within.** This option allows points in very close proximity to be ignored, reducing the complexity of the resulting curve.
- **Termination Conditions.** When enabled, the B-Spline creation will terminate if a gap of a prescribed distance is exceeded.

Tessellation Options. B-Splines are always mathematically perfect but, due to rendering limitation, must be drawn graphically as interconnected line segments. This additional control allows particular splines to be drawn at higher or lower resolution.

From Surface

Creates B-Splines from surface boundaries (Figure 31-8).

Figure 31-8. Creating B-Splines from the boundaries of a surface.



From Circles

This command directly converts a circle or collection of circles into closed b-splines.

From a Collection of B-Splines

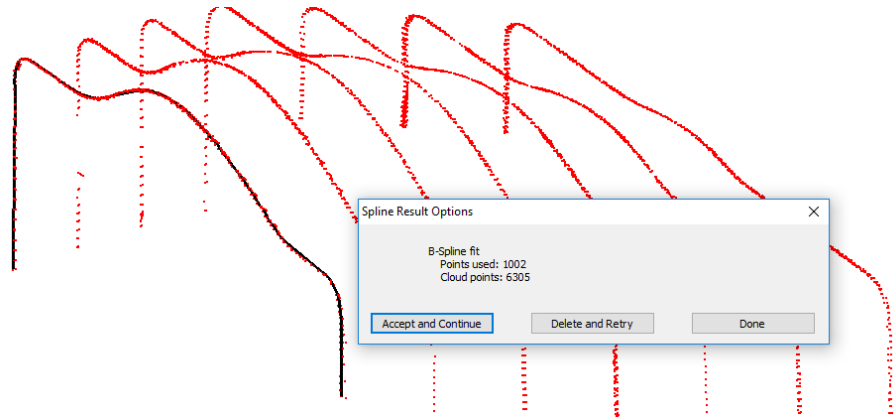
Creates one single B-Spline from the selected individual B-Splines.

From Cloud Cross Section

This command is designed to provide a means to build a fit b-spline to a cross section. To do so, it uses a seed (click-point) to define a starting search point. It expects this point to define the end of the cross section to build. It then asks for a proximity tolerance used to define how large a gap in the cloud data to span and then the number of control points to use in the fit. With a larger number of control points you will get a closer fit to each of the cloud points, and with a smaller number you will get a smoother curve.

The fit solution will then search through the cloud data in the selected cross section until a gap is reached that is bigger than the proximity tolerance. Cross Sections that are intended to bridge holes in the part for example will need to be built by running the fit command a second time and selecting a starting point at the far side of the hole to complete the curve (Figure 31-9).

Figure 31-9. Building additional b-splines from cross section clouds

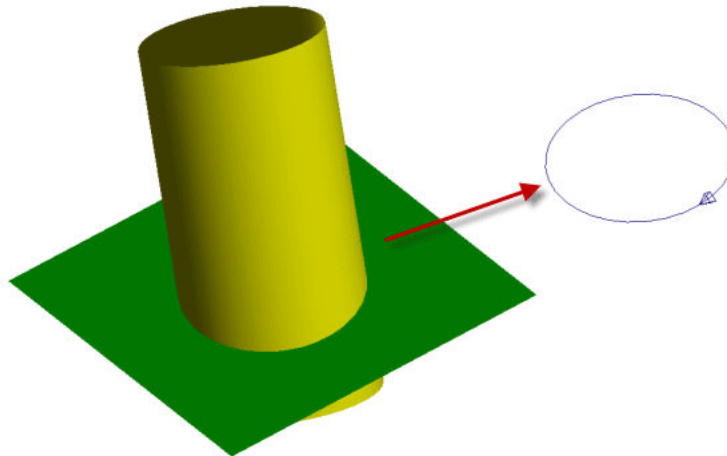


To facilitate building subsequent b-splines the newly constructed b-spline will be highlighted and an option provided to Accept or Retry provided.

Intersect Surface

Creates a B-Spline from the intersection of two surfaces (Figure 31-10). This function builds a series of splines defining the direct contact curves of each and every surface face within the selected surfaces.

Figure 31-10. This B-Spline has been created by intersecting cylindrical and planar surfaces.



Intersect Plane and Surface

Creates a B-Spline at the intersection of a plane and surface. This function creates the same spline that would be created from intersecting surfaces built from the geometries but allows for unbounded geometry to always intersect.

Intersect Cone and Cylinder

Creates a B-Spline at the intersection of the surfaces of a cone and cylinder. This function creates the same spline that would be created from intersecting surfaces built from the geometries but allows for unbounded geometry to always intersect.

Intersect 2 Cones

Creates a B-Spline at the intersection of 2 cones. This functions creates the same spline that would be created from intersecting surfaces built from the geometries but allows for unbounded geometry to always intersect.

Advanced Surface Fitting

From Point Groups (Interpolate)

Note: Points in the selected point groups must be in order or the b-spline will be created incorrectly.

Creates a interpolated surface by fitting b-splines to each selected group (Figure 31-11) and then lofting the b-spline to create a surface (Figure 31-12).

Figure 31-11. Fitting B-Splines to a series of points.

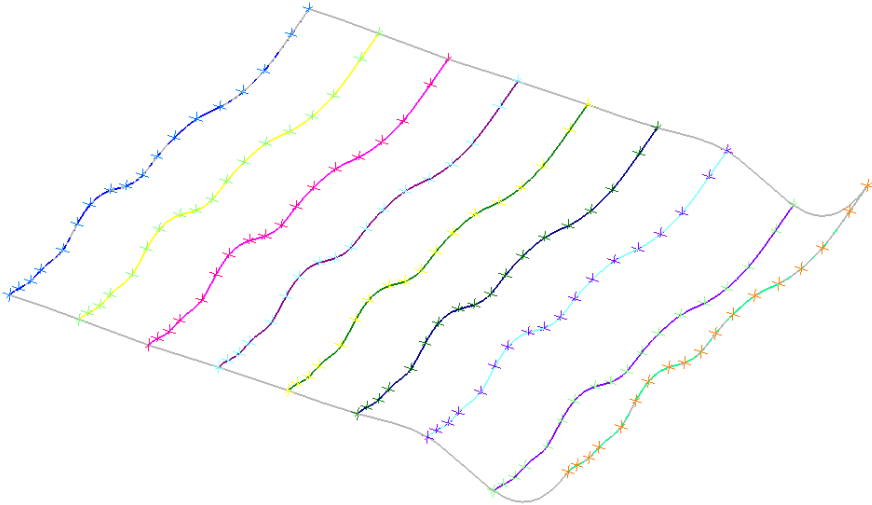
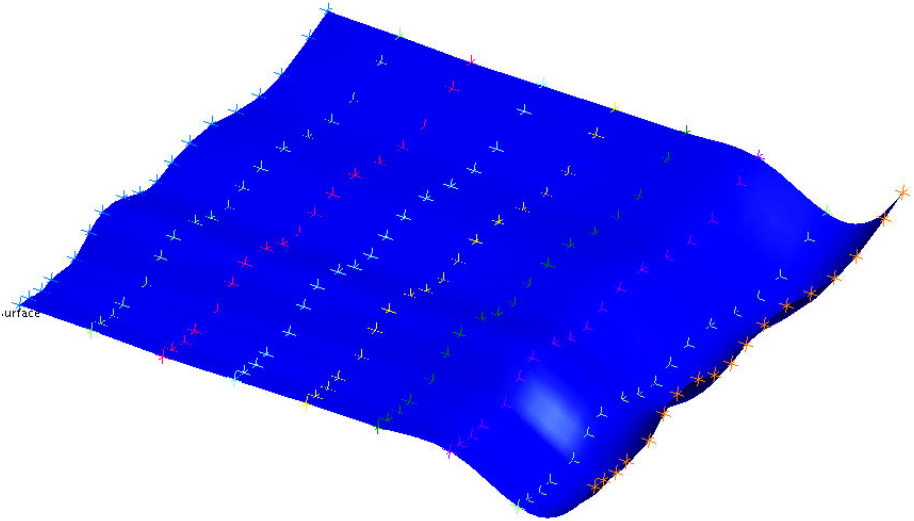


Figure 31-12. A surface has now been fit from the B-Splines.



From B-Splines (Approximate)

Similar to the above command, this time each b-spline is an approximate fit. The user has control of the control points, degree of curve

and the skinning tolerance.

Figure 31-13. Defining the number of intermediate control points.

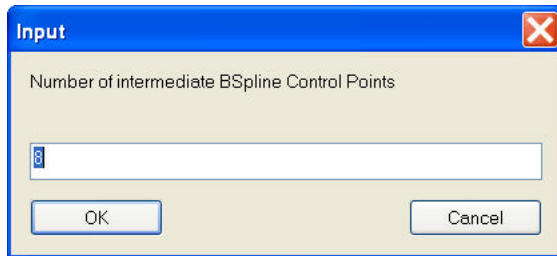


Figure 31-14. Specifying the degree of intermediate curves.

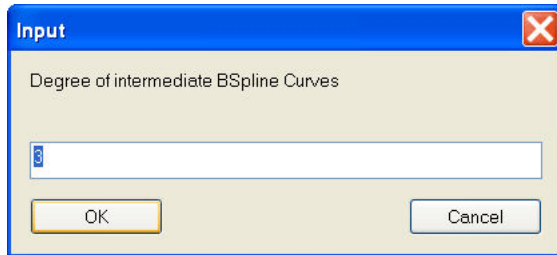
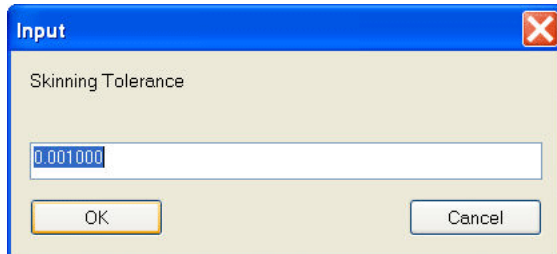


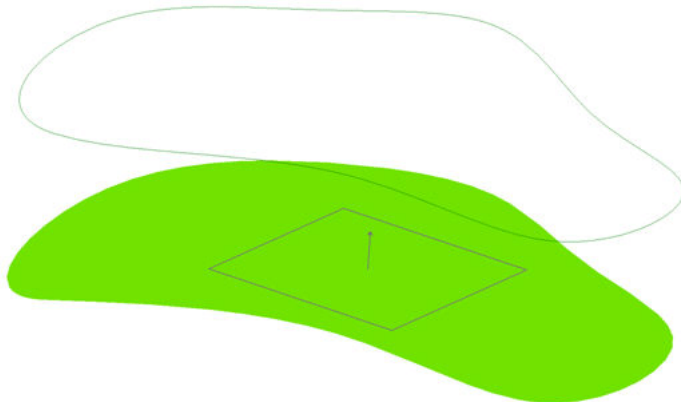
Figure 31-15. The skinning tolerance indicates how far the surface may deviate from the B-Splines.



From Plane and Closed B-Spline

Projects selected closed B-spline to selected plane and creates a surface from the projection (Figure 31-16).

Figure 31-16. A closed B-Spline has been projected to a plane.



Fit From Bounding Curves and internal points

Use for B-splines that bound a group of points (Figure 31-17) to create a surface (Figure 31-18).

Figure 31-17. Four bounding B-Splines around a surface.

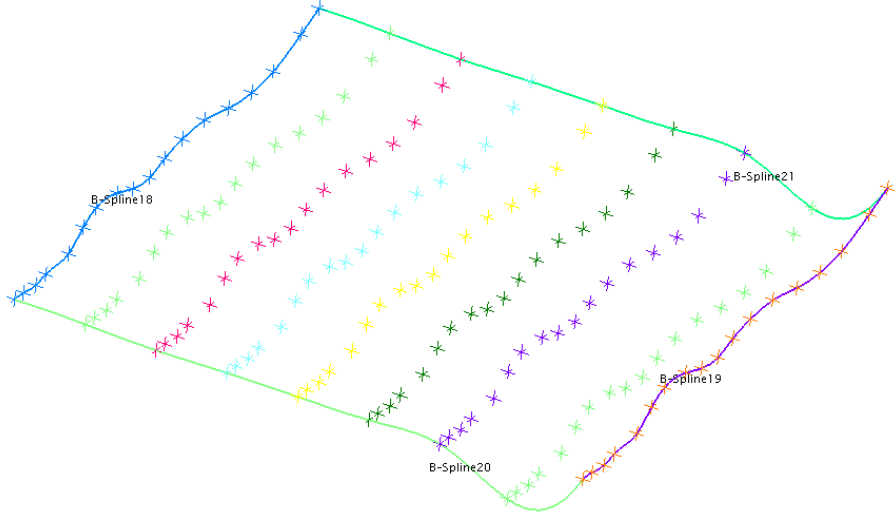
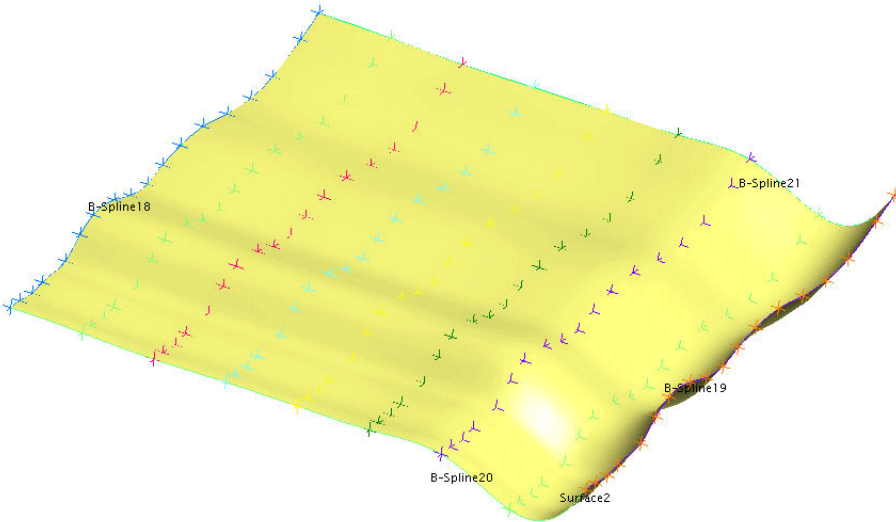


Figure 31-18. The resulting surface.



Fit From Nominal Face and Actual Data

A surface will be created by fitting the selected surface face to the selected data (Figure 31-19).

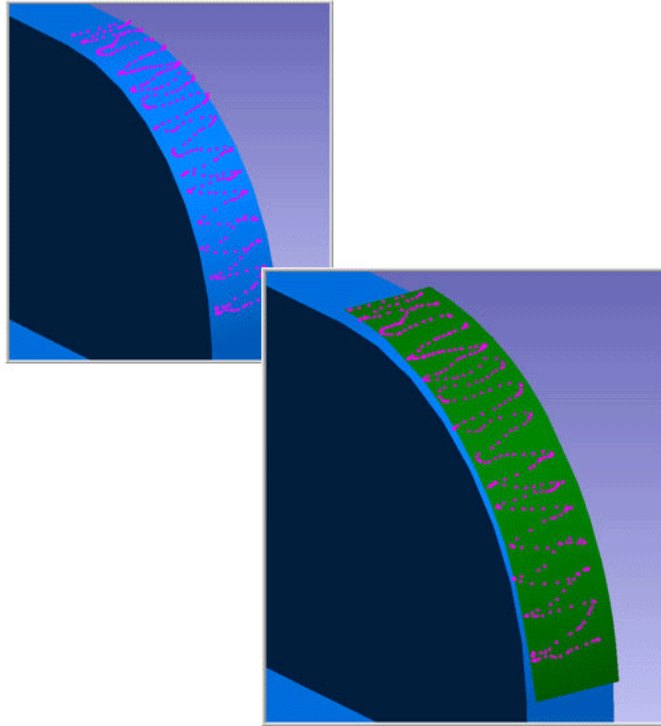


Figure 31-19. Fitting a surface face to selected data.

Polygonized Meshes

From Surfaces

Converts selected surface to a Polygonized Mesh.

From Cloud Points and from Selected Cloud Points

Generates a Polygonized Mesh from selected point clouds or points. By default the mesh orientation uses the current Point of View, however, the working frame can be used as well (Figure 31-20). Enter the grid resolution based on point cloud density. Rule of thumb, the data density should be four times greater than the grid resolution.

Figure 31-20. Polygonized mesh generation settings.

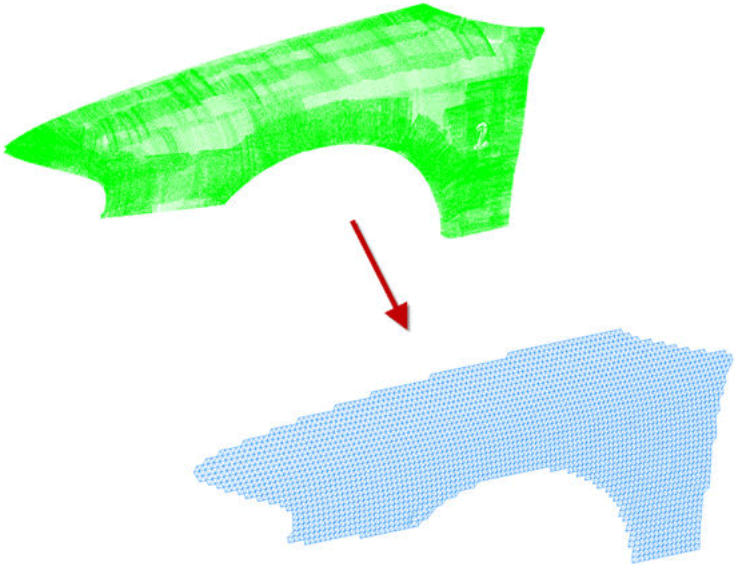
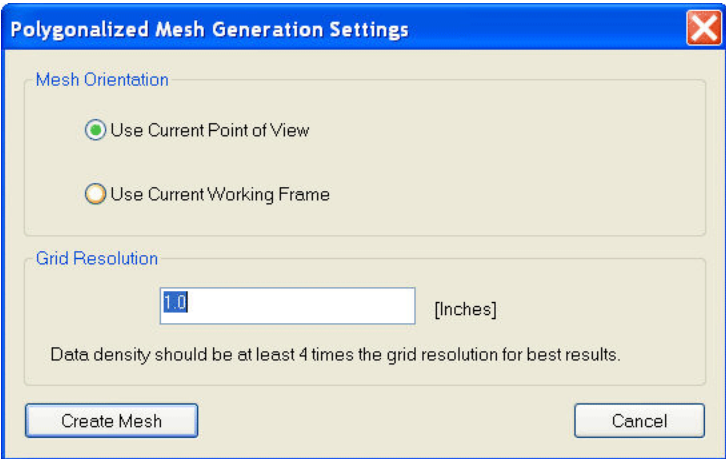


Figure 31-21. Generating a polygonized mesh from a point cloud.

Colorized Mesh from Point Proximity

Creates a Colorized Polygonized Mesh depicting the error between the selected points and the selected surface (Figure 31-22). The colorization options can be accessed through the Polygonized Surface properties.

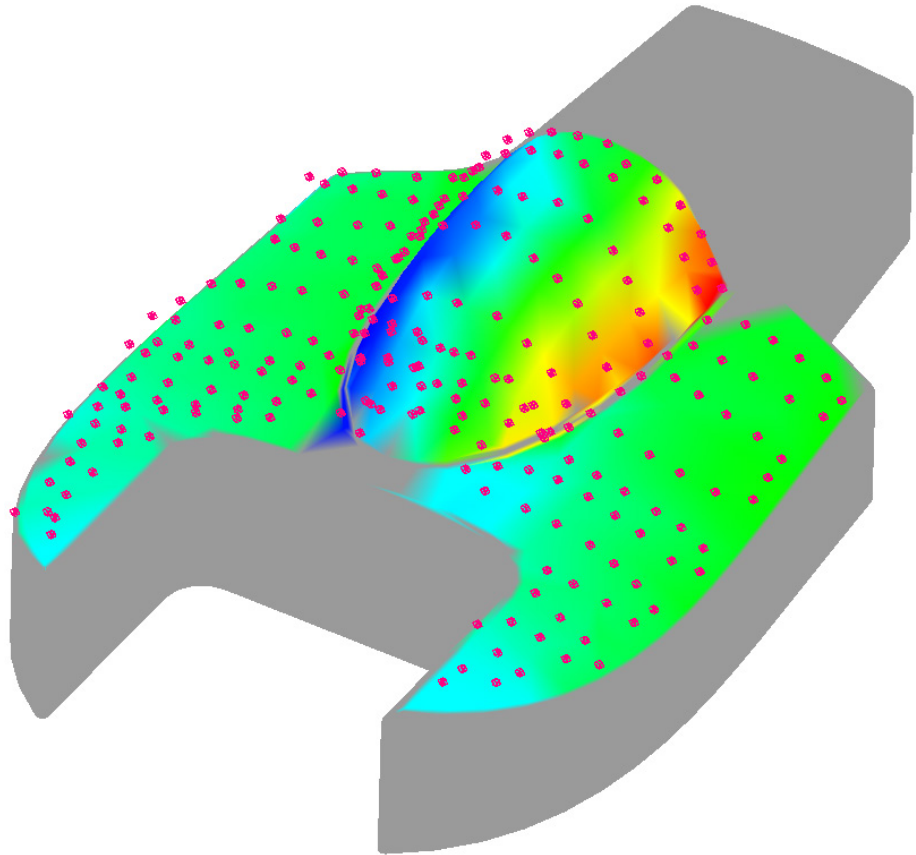


Figure 31-22. A colorized graphical mesh created using proximity of measured points to a nominal surface.

Measurement Plans

CHAPTER

32

SpatialAnalyzer contains a powerful scripting environment that gives you the ability to automate tasks ranging from measurement, to analysis, and reporting. Scripts can introduce efficiencies into your metrology processes that cannot be overstated.

About Measurement Plans

Measurement Plans (MPs) is the name given to the scripting capability built into SpatialAnalyzer. MPs are similar to macros in that they can repeat a series of steps for you—but the similarities end there. MPs actually have significantly more capability—scripts make calculations, branch to different behavior based on decisions, gather user input, read and write to files, and execute custom code. In that sense, scripts share more traits with a programming language than a macro.

However, true programming languages have a complex syntax, often take many years to master, and have a high learning curve. MPs are designed to share as much capability as possible with programming languages while still dramatically lowering the learning curve compared to learning a programming language.

Scripts can vary from simple utilities constructed in just a few minutes to comprehensive inspection plans that provide go/no-go automation and drive other machinery such as robots or CNC machines.

MPs are available for editing and executing on SA Ultimate or higher. They cannot be created, edited, nor executed on SA Professional.

Running a Measurement Plan

Scripts can be created in two formats:

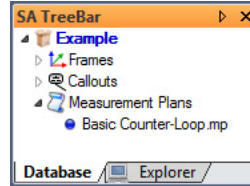
- **Embedded.** An embedded MP is stored in the SA job file and

listed in the tree (Figure 32-1).

- **External.** An external MP is stored as a separate .MP file on the file system. It is completely independent of any job file.

Both embedded and external scripts have essentially the same functionality and behavior (although there are some advanced reasons why you might choose one over the other).

Figure 32-1. An embedded MP.



If you would like some example scripts to view or run, they are included with the SA install in the Samples/MP Scripts folder.

To Run an External MP:


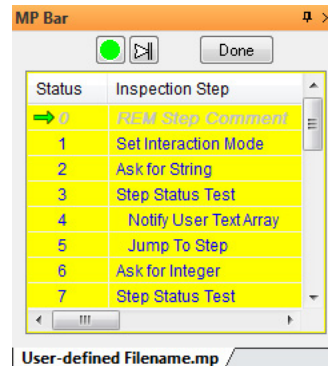

1. From the SA menu, select **Scripts > Run Measurement Plan.**
2. Navigate to the .mp file and select it.
3. The script will load into the *MP Bar* (Figure 32-2). If the *Auto-Run MPs* option is enabled (see “To Enable/Disable Auto-Running of Scripts:” on page 841), the script will begin executing immediately. If this option is disabled, you must click the run button  in the *MP Bar* to run the script.



Figure 32-2. A script ready to run in the *MP Bar*.




To Run an Embedded MP:

1. In the tree, locate the script in the **Measurement Plans** category (see Figure 32-1) and double-click it, or right-click it and select **Run** from the context menu.
2. The script will load into the *MP Bar* (Figure 32-2). If the *Auto-Run MPs* option is enabled (see “To Enable/Disable Auto-Running of Scripts:” on page 841), the script will begin executing immediately. If this option is disabled, you must click the run button  in the *MP Bar* to run the script.

To Run a Script from the MP Editor:

1. Load the script into the editor.
2. Do one of the following:
 - Click the run button  at the top of the editor,
 - From the editor's menu, select **File > Run**, or
 - Press **Ctrl + R**.
3. The script will load into the *MP Bar* (Figure 32-2). If the *Auto-Run MPs* option is enabled (see “To Enable/Disable Auto-Running of Scripts:” below), the script will begin executing immediately. If this option is disabled, you must click the run button  in the *MP Bar* to run the script.

To Enable/Disable Auto-Running of Scripts:

1. Click the User Options icon  or press **Ctrl+U** (default keyboard shortcut).
2. Navigate to the *Automation* tab. Toggle the *Auto-run MPs (applies for double-click in tree and Scripts>>Run Measurement Plan)* option.

Quick MPs

Warning: Making changes to Quick MP slots modifies your User Interface Profile. Changes must be saved to your UI profile in order to retain the settings the next time SA is run.

You can assign Measurement Plan scripts to the menus (Figure 32-3), so that any script can be executed by selecting it specifically from the SA menu (or from a toolbar, if you've customized the SA user interface). You can assign up to 30 scripts to Quick MP slots for easy execution.

This immediate execution of Quick MPs is excellent for listing a library of commonly-used scripts or for testing a script during development.

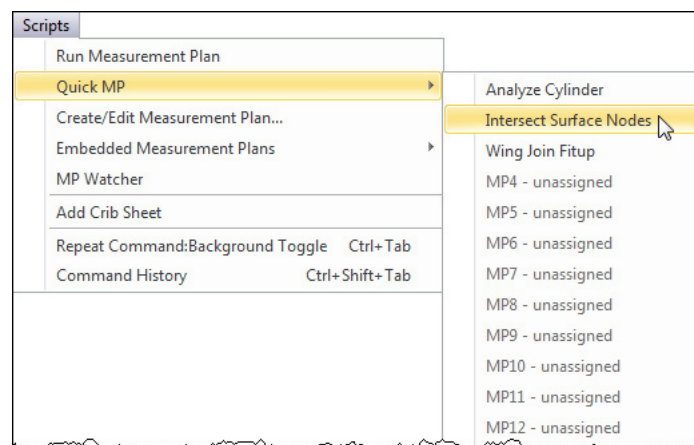


Figure 32-3. Scripts have been assigned to Quick MP slots 1, 2, and 3.

To Assign a Script to a Quick MP Slot:

1. In the menus, select **Scripts > Quick MP > Configure Quick MPs**.

● **Tip:** In most cases, you should use external scripts with Quick MPs.

2. In the *QuickMP Assignments* dialog, click the **Add** button.
3. In the *Quick MP Assignment* window, specify the parameters for the script:
 - **ID.** Which of the 30 slots (from one through 30) should be used for the script.
 - **Display Name.** The name for the script that you'd like to use in menus or toolbars.
 - **Location.** Select *Embedded MP* if the script is embedded, or *External MP File* if the script is an external file.
 - **Collection/Name/File.** If the script is embedded, specify the collection containing the script and the name of the script itself as it appears in the tree. If the script is external, the path to the script file to run.
4. Click **OK** to accept the settings, then **OK** again to close the *QuickMP Assignments* window.
5. Save the changes to the UI profile as specified in "Custom UI Profiles" on page 63.

If you now select **Scripts > Quick MP**, you should see your script listed.

To Edit a Script's Quick MP Slot:

1. In the menus, select **Scripts > Quick MP > Configure Quick MPs**.
2. In the *QuickMP Assignments* dialog, select the Quick MP to modify and click the **Edit** button.
3. Specify the desired settings as indicated in "To Assign a Script to a Quick MP Slot:" on page 841.
4. Click **OK** to accept the settings, then **OK** again to close the *QuickMP Assignments* window.
5. Save the changes to the UI profile as specified in "Custom UI Profiles" on page 63.

To Remove a Script from a Quick MP Slot:

1. In the menus, select **Scripts > Quick MP > Configure Quick MPs**.
2. Select the Quick MP script to delete, then click the **Delete** button.
3. Save the changes to the UI profile as specified "Custom UI Profiles" on page 63.

To Run a Script from a Quick MP Slot:

1. In the menus, select **Scripts > Quick MP**.
2. Choose the script you'd like to execute.


3. The script will execute immediately, even if the *Auto-Run MPs* option is disabled (see “[To Run an External MP:](#)” on page 840)

To Assign a Script to Any Menu or Toolbar:

1. Assign the script as a Quick MP as described in “[To Assign a Script to a Quick MP Slot:](#)” on page 841.
2. Assign the Quick MP slot’s menu item to a toolbar or menu as described in “[Customizing Toolbars & Menus](#)” on page 65.

Working With Measurement Plans


Measurement Plan Scripts are not text-based. Instead, SA uses a graphical editor for writing scripts, which is not as basic as a macro recorder but also not as complicated as a text-based scripting environment.

 **Tip:** In general, we recommend using external script files unless your script is fairly short and simple.

As described in “[Running a Measurement Plan](#)” on page 839, you can create your script as an embedded script or as an external script. There are advantages and disadvantages to each.

Embedded Measurement Plans:

- Are stored as part of the SA file, therefore are easier to find and bundle with a script.
- Are easier to execute and edit, since you can access them directly from the tree.
- May not work if you need to create a new SA file or load an SA file as part of the script, since such actions will clear the embedded scripts from the tree.
- Make version maintenance, reuse, and distribution of scripts more difficult, since you need to export the script to a separate file in order to use it with other SA files.

 **Warning:** You cannot use embedded scripts that have more than one script (such as a script with a subroutine) with an “Open SA File” or “New SA File” command. An external script is necessary in such a scenario.

External Measurement Plans:

- Are stored as a separate .mp file, therefore you must navigate to the file to execute it.
- Are able to work with any scripting commands—creating a new SA file or loading an existing file does not create problems.
- Are much better suited to script maintenance, reuse, versioning, and distribution.

You can start writing an embedded script or start writing a script in an external file. Likewise, you can export an embedded script to an external file or import an external script file as an embedded script.

To Start a New Embedded Measurement Plan Script:

1. In the menus, select **Scripts > Embedded Measurement Plans > Create**

New.

2. Specify a name for the new script. This is how the embedded script will be named in the tree.
3. The MP Editor will appear, ready for input.

To Start a New Measurement Plan Script in an External File:

1. In the menus, select **Scripts > Create/Edit Measurement Plan**.
2. The MP Editor will appear, ready for input.

To Import an External Script File as an Embedded Script:

1. Select **Scripts > Embedded Measurement Plans > Embed existing .MP file**.
2. Navigate to the script file to embed and select it.

To Export an Embedded Script to an External File:

1. Right-click the script in the tree and select **Export** from the context menu.
2. Specify a file name for the new external file, then click **Save**.

To Edit an Existing External Script File:

1. From the menu, select **Scripts > Create/Edit Measurement Plan...**
2. The *MP Editor* will appear.
3. Do one of the following:
 - In the editor's menu, select **File > Load**, or
 - Press **Ctrl + L**.
4. Select the desired file.

To Edit an Existing Embedded Script:

1. Right-click the script in the SA tree and select **Edit** from the context menu.

To Start a New Script in an Existing Editor Window:

- From the editor's menu, select **File > New**, or
- Press **Ctrl + N**.

To Save Your Script:

- From the editor's menu, select **File > Save** or **File > Save As...**, as applicable, or
- Press **Ctrl + S** (Save) or **Ctrl + Shift + S** (Save As...).

Protecting Scripts

Once you have finalized a script, you will often want to protect it from accidental (or intentional) modification. You can password-protect a

script so that it cannot be modified. Once protected, the script will no longer load into the editor. However, protecting a script does not prevent it from being executed, or from viewing the commands used to create the script.

To Password-Protect a Script:

1. Load the script into the editor.
2. From the editor's menu, select **Protect > Set Password**.
3. In the **Set Password** dialog, enter the desired password twice, then press **OK**.
4. Save the script.

To Open A Protected Script:

1. Load the script into the editor as usual.
2. When prompted, enter the password for the script.

To Remove Protection from a Script:

1. Load the script into the editor. You will need the password for this operation.
2. From the menu, select **Protect > Remove Password**.
3. Confirm the operation.

Generating SDK Code

The MP Editor has the capability of automatically generating the SDK code needed to repeat a script's functionality in both the C++ and Visual Basic languages. This code can then be copied, pasted into an SDK project, and (with limited changes) be executed directly in compiled code. For more on the SDK, refer to the next chapter (SA SDK).

To Generate SDK Code for a Script:

1. Load the desired script into the editor.
2. From the editor's menu, select **SDK > View SDK Code**.
3. In the **SDK Code Viewer** dialog, select the desired language, *Visual C++* or *Visual Basic*.
4. The code will automatically be copied to the Windows clipboard and can be pasted directly into any text editor or Integrated Development Environment.

The MP Editor

If you will be writing Measurement Plans, you'll be doing it in the MP Editor (Figure 32-4). Some companies even have employees whose primary (or only) job responsibility is to write Measurement Plans—these people may spend most of their working hours in this editor!

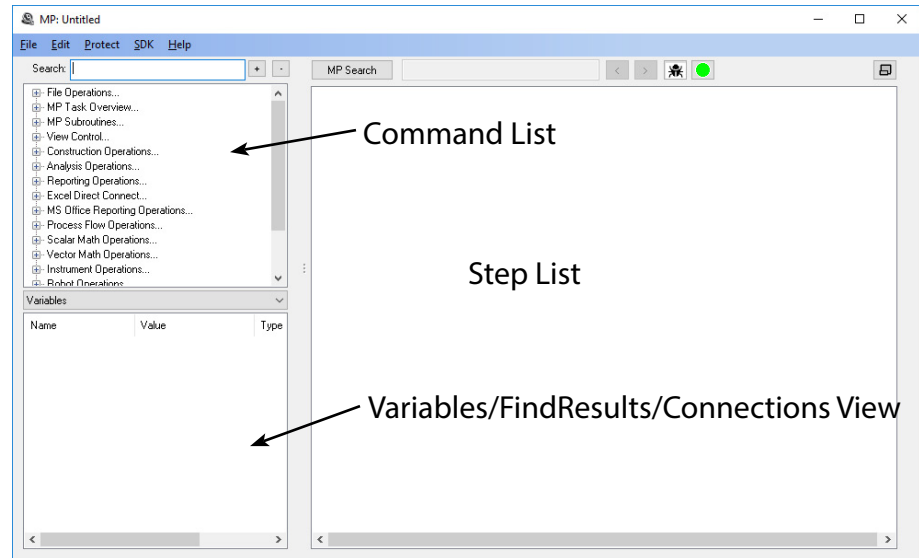


Figure 32-4. The MP Editor.

For those used to the legacy MP Editor in SA 2014.04.15 and earlier, this new editor has an enormous number of improvements. The most notable are:

- Command list and script are now integrated into one window. More screen real estate is devoted to your script.
- Multiple commands can be displayed at once for easy reference.
- Every command is a collapsible tree item, allowing easy expand/collapse behavior. This is particularly useful for “keeping track” of commands that you haven’t finished editing.
- Commands can be rearranged through easy drag and drop.
- Commands can be locked to prevent changes once they’re finalized.
- Changing entry methods (such as switching from “Enter Method” to “Reference” is dramatically faster than before with quick-select buttons.
- Most references can be selected directly in the step editor (instead of the large table that was displayed in older versions), causing less disorientation.
- Long comments can be seen completely, whereas they would be obscured in the old editor.
- A variable list shows any defined variables and their types for

- reference.
- A debug mode allows you to get “MP Watcher” functionality directly in the editor. You can see argument values directly in the editor as the script executes.
 - Current variable values are displayed while in debug mode.
 - A Connections viewer allows you to easily view commands that reference or are referenced by the current command, and jump to those steps quickly.
 - Lots of keyboard shortcuts for loading, saving, running, expanding and collapsing, selecting, etc. are available—making for a fast workflow.
 - Automatic expansion and collapse of applicable commands (an option) makes for quick selection of references.
 - An “expert mode” suppresses warning messages about deletion of commands.
 - Configurable alternating row colors for less eyestrain.
 - Automatic refresh of subroutine arguments.
 - Faster setting of breakpoints/comments.
 - More obvious indentation.
 - For new commands, if a step reference becomes invalid, it will switch to -1, forcing a command to fail instead of jumping to the start of the script. This makes it easier to find incomplete arguments in a command.

Overview

The MP Editor provides a graphical interface for writing scripts. It carries a few traits from a traditional IDE (Integrated Development Environment) that experienced programmers might recognize. But most importantly, it is not a text editor. Scripts are written in a point-and-click, drag-and-drop fashion instead of writing text-based commands. This makes the learning curve more acceptable for most users.

There are three main sections in the editor ([Figure 32-4](#)):

- **Command List.** This is a hierarchical tree showing every command that could possibly be added to a script, organized by type.
- **Connections/Find Results/Variables View.** This view provides an optional display of 3 different screens:
 1. **Connections.** This screen shows a two way list of interconnectivity with the currently selected step. This could also be named references because it displays a list of steps that the current step references and a second list of steps that refer-

ence this step.

2. Find Results. Working in conjunction with the MP Search button, this screen displays the resulting steps from the current search.
3. Variables. This screen displays either a list of the variables, variable types, and (if debugging) variable values defined in the current script.
 - **Step List.** The majority of the window is devoted to the *Step List*, which displays the individual steps of the script itself and allows for editing of the script's arguments.

A Measurement Plan is a procedural script that executes one line at a time. Normally, the script executes in a linear fashion, proceeding from one command to the next. However, there are commands that cause execution to jump to another location in the script. There are also commands that branch to different parts of the script (or to different scripts altogether) based on some value.

Everything in Measurement Plan scripting is zero-based. That is, all counting begins with zero, whereas in the regular world we typically start counting from one. The first step in a script is step zero, the first argument in a command is argument zero, the first item in a list (array) is item zero, and so on. If you have any programming experience, you will be used to this convention.

About Commands

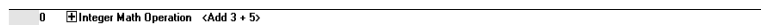
When a command is first added to a script, it will be in expanded form so that it can easily be modified (Figure 32-5).

Figure 32-5. A newly-added command, in expanded form.



Commands can also be collapsed to one line at any time to save space or simplify your view (Figure 32-6).

Figure 32-6. A collapsed command.



To understand the command's parameters, let's look at this command in more detail. Each item in the command's detail view is numbered in Figure 32-7 and discussed below.

Figure 32-7. A command's parameters.



1. Step Number

Every command in a script has a specific step number, used to order and identify steps in the script. Step numbers start from zero, so the first step in a script is step zero. You can rearrange single steps or entire groups of steps.

2. Command Name

The name of the command is listed at the top of the entry. This is a read-only field that allows you to view the name of the command being examined. The name of the command in [Figure 32-7](#) is **Integer Math Operation**.

3. Comment

Each step can have a user-defined comment (**Add 3+5** in this case). This comment is simply a single line into which you can type anything you'd like. Comments are typically used to provide "human-readable" text to accompany the commands, so that someone reading a script (often times, that's you!) can understand the intent of the command.

Comments can only be one line in length. When you collapse a command, the comment appears to the right of the command name. When expanded, the comment appears below the command name in a field that can be edited.

A few tips regarding comments:

- **Keep them brief.** Long comments clutter up the screen, take time to enter, and take longer to read.
- **Comment Only When Necessary.** New script writers have a tendency to add a comment to every line. Don't. Only add comments if you feel like they're needed.
- **Don't be redundant.** If the command itself is clear, there's no need to add a comment. For instance, in [Figure 32-7](#), it's clear that we're adding 3+5, so the comment is redundant.
- **The above three rules are not set in stone.** If you feel like a comment is needed, add one. For example, it may be clear what a command is doing when you're looking at its details, but when it's collapsed, you can't see what it's doing. So it may be acceptable to add a comment in that case. Use your own discretion.

4. Arguments

Every command added to a script has zero or more *arguments*. Arguments are the vehicle by which you can communicate with a command, and by which a command can communicate with you as the script's author. There are two types of arguments:

- **Input Arguments.** These are values that can be supplied to a

command as input. They help the command determine what it is supposed to do. Most input arguments are required, but some are optional.

- **Return Arguments.** Return arguments are used to return information or results back from a command after it executes.

In [Figure 32-7](#), the command has three input arguments: zero through two (A0 through A2). It has one return argument: A3 (notice the grayed-out “Result only” fields).

In this example, we want to perform a math operation on two integers (whole numbers). We obviously need to provide both of those numbers, in addition to the math operation that we want to perform. Those are the input arguments.


Once the command completes, we need a place to store the result of the operation so that we can refer to it later. The result of the addition in this case is stored in A3.

5. Data Type

Every argument has a data type (this will not be new for those with programming experience). A data type indicates the type of information that an argument is expecting. You must feed an argument data of a compatible data type—otherwise, SA will prevent the operation or the script will fail at runtime. For instance, you cannot provide a letter to an argument expecting a number. After all, what is the numeric value of that letter?

There are just four primary data types:

- **Integer.** Whole numbers, such as 10, 10000, 0, and -502. These values are represented exactly, but there is a limit to the range of this number. In general, you need not worry about this limit for scripting applications.
- **Double.** Decimal numbers, such as 3.1415926535, 2.781, 0.0, or -1.125. These numbers are in general not represented exactly, so expect numerical errors to creep into your calculations. These errors are typically well beyond the precision of portable metrology work, so they’re not usually a worry, but you should take care when trying to compare one double value to another. You may expect them to be the same but they may actually be very slightly different. Double values also have limits on the range of values they can represent, but in general you need not worry about this limit for scripting applications.
- **Boolean.** A value that represents one of two possible values, such as true/false, on/off, left/right, or yes/no.
- **String.** A sequence of one or more alphanumeric characters, such as a sentence or prompt.

 **Note:** Some data types are simply more descriptive names for other data types. These are called *Proxy data types*. **Collection Name** is an example of a proxy data type.

There are many more data types in scripts in addition to these four, but virtually all other data types in scripts are derivations of these four main types. For example, there is a **Collection Name** data type, but it is really a string. This means that any argument that expects a **Collection Name** can also be fed a string, and vice versa.

6. Description

Every argument has a description field. This is more of a human-readable description of what the argument is for.

7. Entry Method

A dropdown which appears when clicking in this column allows you to specify *how* you want to supply the information for the argument. For example, do you want to type it in, or feed it from some other command?

There are many entry methods. The most common are covered here:

- **Enter Value.** The argument's value is typed directly into the command—known as *hard coding* the value.
- **Reference.** The value is fed into the argument from some other command. For example, the result of a math operation might be fed into another command for further processing.
- **Variable.** The value for the argument is retrieved from a variable by specifying the variable's name. The variable must be of a compatible data type for the given argument.
- **Pick.** The value is selected from a limited dropdown list of values.
- **Dialog.** The value is specified graphically in a dialog.


8. Argument Value

The argument's value is specified in this field.

- Hard coded values appear as typed.
- References appear as a combination of step number and argument number. For instance, **Ref {S3 A4}** indicates that you are feeding the value stored in step 3, argument 4 into the current argument.
- Variables are specified by entering the variable name.

9. Entry Method Quick-Select

A series of buttons for the most common entry methods are listed to the right of the argument value. By clicking these buttons, you can immediately change the entry method for the argument and directly enter the value. Users familiar with SA's legacy script editor will find this new method of argument entry dramatically faster than the old method, while also requiring less clicking.

 **Note:** If you cannot find a button for the appropriate entry method, look for it in the entry method dropdown (7).

Only buttons applicable to a given argument are displayed, although often times there are more entry methods than buttons. (In other words, only the most common entry methods have buttons).

Entering Commands

Before you add a command to a script, you must find it in the *Command List* first. When you are just starting out, you will probably find commands by browsing the list. However, once you have more experience under your belt, you will begin memorizing key phrases of common commands and find them by simply typing a phrase to search for in the *Search* field above the *Command List*.

To Add a Command to a Script:

1. Select the step after which you'd like the new command to be added. (If no step is selected, the new command will be added to the end of the script).
2. Find the desired command in the command list, or search for it by typing into the *Search* field. Use the + and - buttons to expand or collapse categories in the command list.
3. Double-click the desired command, or select it and press **Enter**.

To Delete a Command from a Script:

1. Select one or more commands by clicking on their command name. (Standard Windows contiguous and non-contiguous selection is supported).
2. Press the **Delete** key.

Editing Commands

Once you have commands in your script, various editing operations can be performed.

To Select a Single Command:

- Click in the row containing the command's name, or
- Use the up/down arrow keys on the keyboard while the step list has focus.

To Select Several Contiguous Commands:

1. Click the row containing the first command's name.
2. Hold down **Shift** and click the row containing the last command's name.

To Select Several Non-Contiguous Commands:

1. Click the row containing the first command's name.

2. Hold down **Ctrl** and click additional rows containing subsequent command's names.

All clicked commands will now be selected.

To Select All Commands:

- Use contiguous selection above, or
- Choose **Edit > Select All** from the editor's menu, or
- Press **Ctrl + A**.



To Clear Your Selection:

- Click away from any commands in the step list, or
- Choose **Edit > Deselect All** from the editor's menu, or
- Press **Ctrl + Shift + A**.

To Rearrange Commands:

- To move a single command, drag it by clicking on its command name and drop it anywhere on the *command you'd like the command to follow*.
- To drop a command BEFORE another command, hold down the **Shift** key while dropping.
- To rearrange multiple steps, select them (standard Windows contiguous and non-contiguous selection is supported), then drag and drop them onto the step after which you'd like the group of steps to appear. Hold down **Shift** when dropping to place the group of steps before the command.

To Expand or Collapse Commands:

1. Select one or more commands (standard Windows contiguous and non-contiguous selection is supported).
2. Do one of the following:
 - Click the  or  button,
 - Double-click the row containing the command name, or
 - Press the right arrow (expand) or left arrow (collapse) keys on the keyboard.

To Copy Commands:

1. Select one or more commands.
2. Do one of the following:
 - From the editor's menu, select **Edit > Copy**, or
 - Press **Ctrl + C**.

To Paste Commands:

1. First copy one or more commands as described above.
2. Select the command after which you'd like to paste.

3. Do one of the following:
 - From the editor's menu, select **Edit > Paste**, or
 - Press **Ctrl + V**.

Resizing Columns

Although it may not be obvious, the MP Editor has a number of resizable columns. You can adjust these to better fit the column's contents, or if you have made the editor window large.


To Resize Columns:

1. Expand any command so that its arguments are visible.
2. Place the cursor just to the left of the Description, Entry Method, Value, or Entry Method Quick-Select Button columns. The cursor will change to indicate that you can drag left and right.
3. Drag the column to the desired width.

Hard Coding Argument Values

Hard coding involves manually entering values into an argument. This is done when the value for an argument is known at development time and is not expected to change.

To Hard Code an Argument Value:

1. Ensure the entry method dropdown for the desired argument is set to *Enter Value*, or click the  button.
2. Enter the value into the argument's value field, as specified below (the most common cases are described):
 - **Simple Data types.** Doubles, strings, integers, and other simple data types can be entered by typing the values directly into the field.
 - **Boolean types.** For arguments of type **Boolean**, you can either type **TRUE** or **FALSE** (case insensitive) or, as a shortcut, type a nonzero value to represent **TRUE** and **0** to represent false. (It will automatically convert this value to **TRUE** or **FALSE** when you press **Enter**.)
 - **Collection Object Names.** If you're entering the name of an object, type the collection name, followed by two colons, followed by the object name. For example, **Measured::AxisLine**. If you leave off the collection name and simply type **AxisLine**, the command will try to find the object named **AxisLine** in whichever collection is active at the time the command is executed.
 - **Point Names.** To specify a point name, type the collection,

group name, and point name separated by a pair of colons. For instance, **Measured::Tie-Ins::1**. You can leave off the collection name to indicate that you want to find the group and target in the active collection at the time the command is executed. For example, **Tie-Ins::1** will look for point **1** in a **Tie-Ins** group for the active collection.

- **Reference Lists.** Some commands, such as **Concatenate Strings**, expect reference lists, which are simply lists of items. To enter these, click the entry field, which will open the **List Editor** (Figure 32-8). Use this to add, rearrange, or remove items from the list.

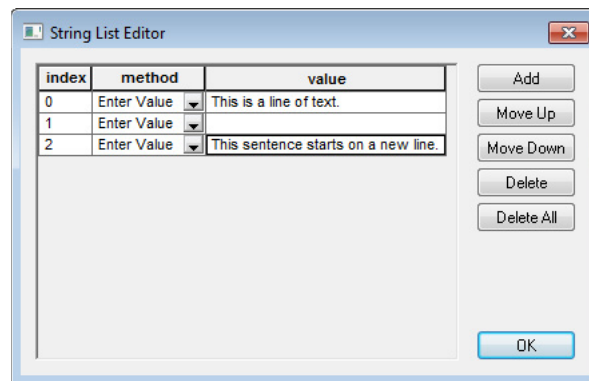


Figure 32-8. The *String List Editor*.

Referencing Arguments

The true power of scripting comes not from a script that repeats a sequence of steps exactly as before, but in making decisions and behaving differently based on parameters that may be different from the last time the script was executed.

Tip: If multiple arguments from different commands are going to refer to the same value, it is generally best for all of them to reference the same argument. Doing so ensures that future changes only require changing one argument.


The most common way of supporting this “non-hard coded” behavior in scripts is through the use of references. A reference is essentially a link or a connection to another value in your script. It is a way to feed a value from some other command in your script into the current argument.

For example, one step may multiply two values together. You can then feed the result of that calculation into an input argument in another command to further operate on it.

To Reference an Argument:

Note: An argument cannot reference itself. Such circular

Warning: Do not attempt to reference an argument that will not contain a value at the time the command is executed. This results in undefined (and generally incorrect) behavior.

1. Ensure the entry method dropdown for the desired argument is set to *Reference*, or click the  button.

The editor will highlight yellow all arguments that are of a compatible type for the given argument. If an argument is already being referenced, it will be highlighted blue. If the *Auto Expand/Collapse Steps* option is enabled in the MP Editor options, any commands containing compatible arguments will be automatically expanded to show

them. These commands will collapse again when the reference is selected.

2. Click on the highlighted argument to reference, or **Esc** to cancel.

Using Variables

Most if not all programming languages support the concept of variables. Variables allow you to define a name that is associated with a value. This allows you to refer to that value by name. The advantage of this is that while a script is running, you may change or overwrite the value of that variable, but anywhere you refer to it by name, you'll always retrieve its current value. In many scripting situations, this can lead to dramatically simpler scripts.

Variables must be of the appropriate data type to store a given value. For instance, an integer variable can store a whole number, but it cannot store a double. Likewise, a point name variable can store a point name, but it cannot store a vector.

You can define a variable by using the **Set [____] Variable** commands. (There is a different command for each variable type). These commands create a variable with the specified name that stores the specified value. You can then refer to that value by its variable name.

To overwrite or change the value stored under a variable, simply use another **Set [____] Variable** command using the same variable name.

To Enter a Variable Argument:

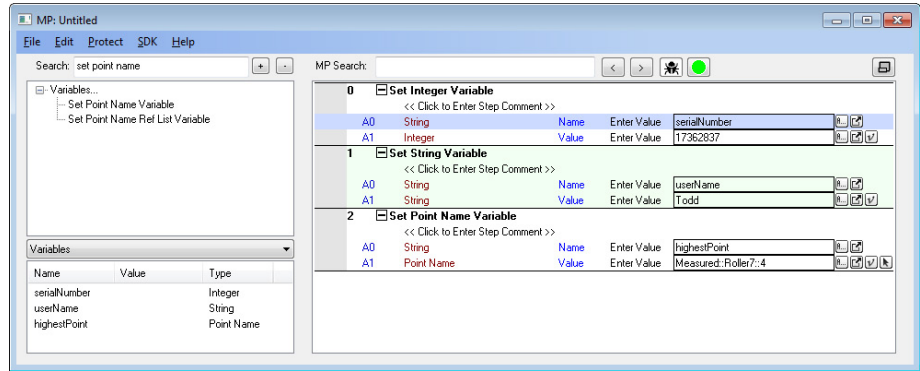
1. Ensure that the entry method dropdown for the desired argument is set to *Variable*, or click the button.
2. Type the name of the variable into the argument's value field.

The value of the variable will be retrieved at the time the command is executed and used as the argument's value.

Variables View

In the bottom-left-hand side of the *MP Editor* is a dropdown that allows you to select the *Variables* view (Figure 32-9).

Figure 32-9. A list of the defined variables.



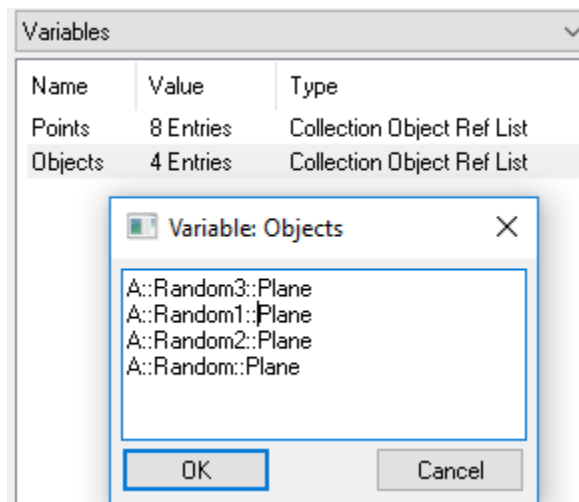
Note: For simplicity, variables defined in other scripts are not displayed. Variables defined in subroutines are not visible to a calling script, so listing a subroutine's variables would be pointless. While editing a subroutine, the editor doesn't know which script is considered its parent, so a parent script's variables are not displayed either.

The Variables view allows you to see all of the defined variables in the current script, as well as their data type. Additionally, while debugging, the value of the variable at the current point of execution is displayed in the *Value* column.

In addition to debugging uses, having a list of defined variables available allows you to easily reference the name when typing a variable in by hand.

Some lists, such as Point Name Reference Lists and Collection Object Name Lists also support a special Ctr+Shift double click option to display a new pannel with the names of the items defined in the variable while in debug mode (Figure 32-10).

Figure 32-10. Accessing Lists in the Variables View



Connections View

Occasionally, you may wish to see or jump to commands which are somehow related to the currently selected step. For example, if you delete a command, it would be nice to see which commands reference that step. After all, by deleting that step, any commands that reference it will need to be corrected.

As another example, you may have a command that jumps to a given step. If you want to insert a new command before that step and have

the command jump to it instead, you will need to adjust the step reference. These types of situations are where the Connections View comes in handy.

The Connection View provides two lists of steps:

- A list of steps that have references to the selected step
- A list of steps that are referenced by the selected step

Simply click on any of the step references in the Connection View to navigate to that step in the Step List editor.

MP Find and Replace

It is often important to be able to find specific entries within a long script quickly and efficiently. The **MP Find and Replace** functions were added to fill that need. The MP Search button (also accessible through Ctrl+F for find or Ctrl+H for Replace) will bring up a dialog for this application (Figure 32-11).

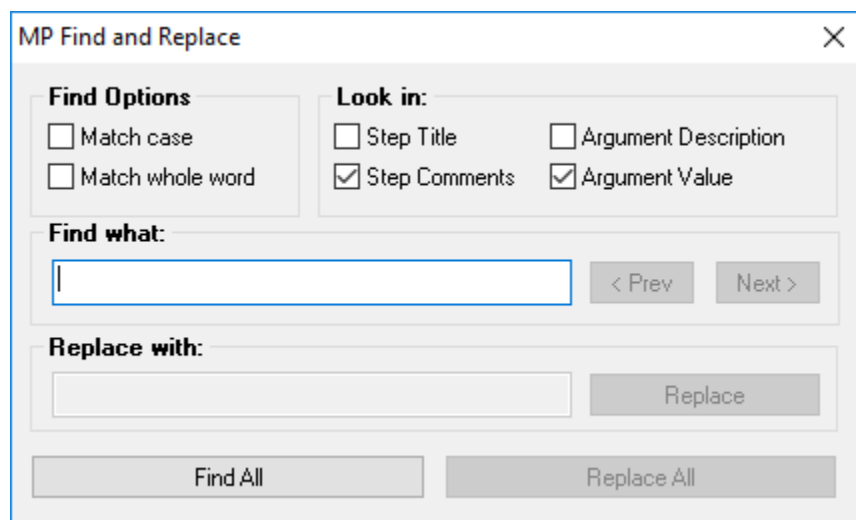


Figure 32-11. MP Find and Replace Dialog

This dialog is designed to work in conjunction with the Find Results View panel (Variables, Find Results and Connections View). To find the entry you are looking for consider the following:

1. Open the Find and Replace dialog (Ctrl+F).
2. Select from the check boxes to identify what you are looking for. You can search all fields by checking each box or isolate your search to just one field such as the "Argument Value" which is the entry field for most arguments.
3. Press the Find All button once to build a list.

At this point you can either use the **Prev** and **Next** to navigate through the results or close the Find and Replace dialog entirely and use the forward and backward buttons at the top of the editor.

These buttons will move the focus in the script from one entry to the next in sequence. However, the fastest way to navigate through your search results is to look in the Find Results View panel. This panel will show you a list of steps and individual arguments within those steps that include the search results. Simply click on an entry to navigate to that specific argument in the script (Figure 32-12).

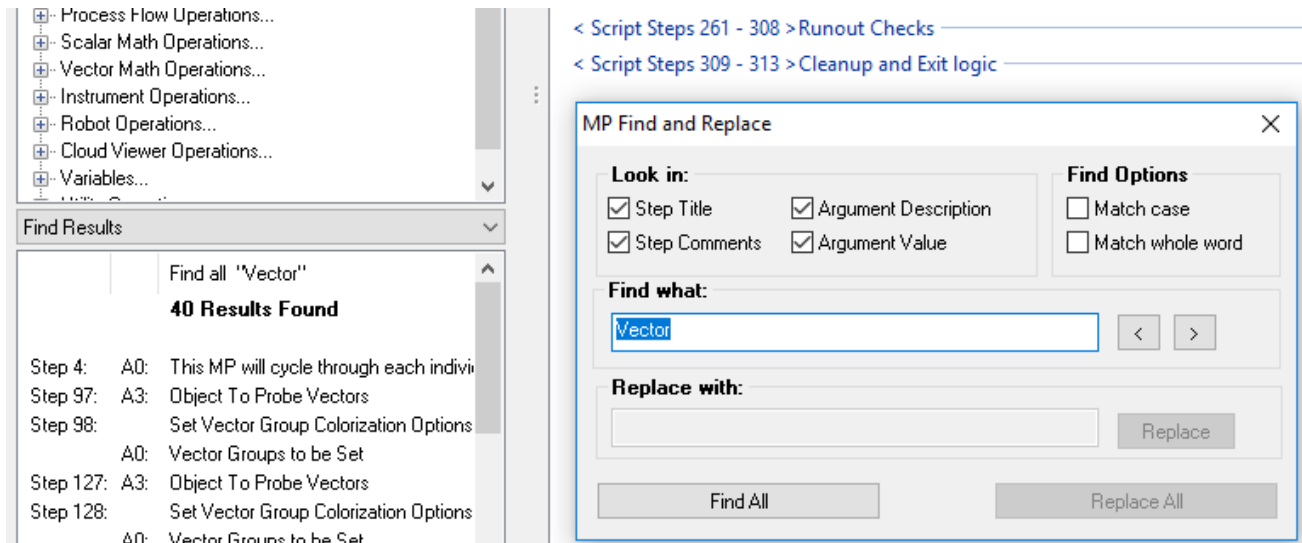
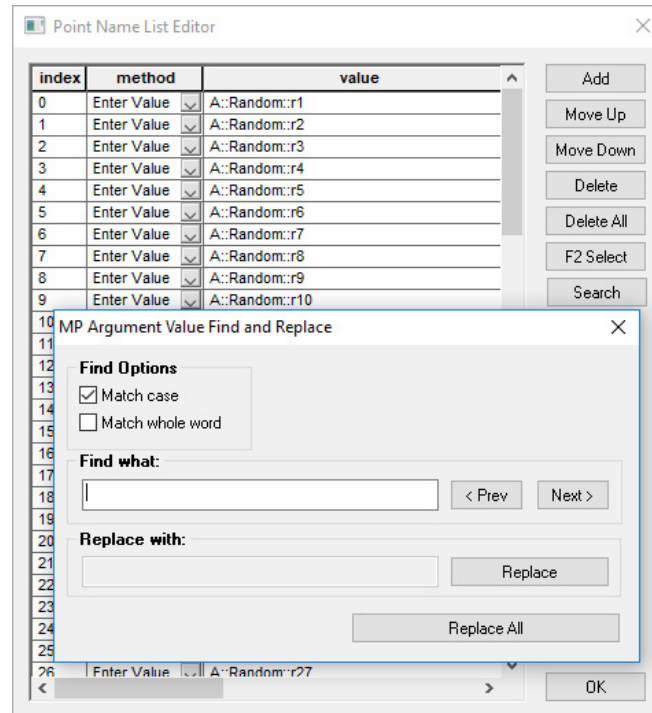


Figure 32-12. Find and Replace Search Results

The MP Find and Replace dialog will identify items that are both in the primary script and within a list. However, it will not allow you to navigate through a list to find a single entry. In order to find a single point name in a long list of names for example a second search button has been added to the List dialog (Figure 32-13).

Figure 32-13. Additional List Search Function



Commenting Out Commands

Sometimes, you may wish to “comment out” commands. Commenting out commands will temporarily disable them—they remain in your script, but the script will skip them during execution. As a result, you can easily remove a portion of your script and then restore it into the script with a simple click. This is extremely useful while testing your script—you may wish to try something without a section of commands, or you may wish to remove a block of your script simply to shorten the length of time it takes for a script to execute. Deleting these commands instead would not be a great approach, since adding them back in would require a lot of work.

Although commented commands are not executed, they are still a part of your script—references to those commands remain. If other commands reference a commented command, they will most likely fail at runtime, because the commented command is not executed, so its values may not be calculated.


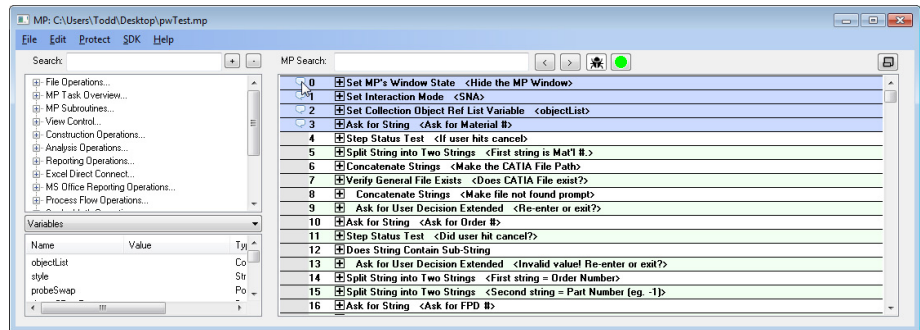

Steps which have been commented out will have a speech bubble icon  next to them (Figure 32-14).

Figure 32-14. Steps 0 through 3 are commented out.



To Comment Out Commands:

1. Select the commands you'd like to comment out.
2. Do one of the following:
 - Click in the right half of the gray column to the left of the selected step numbers until a small comment bubble icon appears , (Figure 32-14), or
 - Right-click the selection and select **Comment Line (REM)** from the context menu.

To Uncomment Commands:

1. Select the commands to uncomment.
2. Do one of the following:
 - Click in the right half of the gray column to the left of the selected step numbers until all icons are cleared, or
 - Right-click the selection and deselect **Comment Line (REM)** in the context menu.

To Clear All Comments:

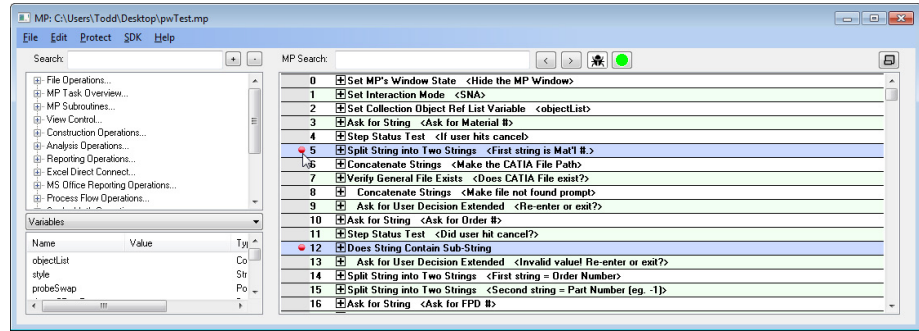
- Right-click any step and select **Clear All Comments** from the context menu.

Setting Breakpoints


When testing, you often want to run a script at “full speed”, then have the script pause at a certain point so that you can slowly step through a certain area of your script. For example, if there is a tight loop at the start of a script that executes thousands of times, it would be exhausting to step through the loop manually one line at a time. Instead, what is needed is a way to run the loop at full speed, then pause afterward.

Breakpoints enable this capability. Any line can be set as a breakpoint, and you can set as many breakpoints in your script as you'd like (Figure 32-15). When set as a breakpoint, an executing script will pause prior to executing the line on which the breakpoint is set. You can then choose to step forward one line at a time, or continue running “at full speed”.

Figure 32-15. Steps 5 and 12 have been set as breakpoints.



To Set Breakpoints:

1. Select the commands you'd like to set as breakpoints.
2. Do one of the following:
 - Click in the right half of the gray column to the left of the selected steps until a breakpoint icon  appears (Figure 32-15), or
 - Right-click one of the selected steps and choose **Breakpoint (BRK)** from the context menu.

To Clear Breakpoints:

1. Select the steps whose breakpoints you'd like to clear.
2. Do one of the following:
 - Right-click one of the selected steps and uncheck **Breakpoint (BRK)** in the context menu, or
 - Click in the right half of the gray column to the left of one of the selected steps until no icon is displayed.

To Clear All Breakpoints:

- Right-click any command and select **Clear All Breakpoints** from the context menu.


Locking Commands

When you are satisfied one or more command's parameters, you can lock those commands to prevent accidental modification. Locking a command provides a number of advantages:

- They cannot be reordered in the script, so you cannot accidentally move them around.
- They cannot be deleted.
- Changes cannot be made to their arguments.

To Lock or Unlock Commands:

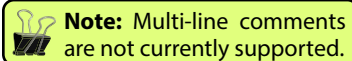
1. Select the commands you'd like to lock/unlock.
2. Do one of the following:

- Right-click one of the selected commands and toggle the **Lock** entry in the context menu, or
- Click in the left half of the gray column to the left of one of the selected steps until the lock icon appears  (to lock) or disappears (to unlock).

To Unlock All Commands:

- Right-click any command and select **Clear All Locks** from the context menu.

Entering Comments



Note: Multi-line comments are not currently supported.

When a command is expanded, a line directly beneath the command's name says << **Click to Enter Step Comment** >>. In this field, you can type any comment to help document your script. Simply click in the field and start typing.

When you collapse a command, its comment will be placed next to the command name.

Indenting Commands

You can indent commands to help set them off as being part of a loop or branch of code, similar to how programmers indent their code to improve readability of the source code.

To Indent Commands:

1. Select the commands you'd like to indent.
2. Press **Tab**. The command will shift over by one tab space.

To Remove Indentation:

3. Select the commands you'd like to unindent.
4. Press **Shift + Tab**.

Setting Command Colors

You can change the background color of one or more commands in the editor to help differentiate a section of your script.

To Change Command Colors:

1. Select the commands you'd like to change.
2. Right-click one of the selected commands and choose **Set Background Color** from the context menu.
3. Choose a new color, then click **OK**. To reset the color back to its default, choose white.

Grouping Commands

When working with longer scripts it can often helpful to group steps. This can make it much easier to navigate a large script and provides a convenient way to isolate and identify key sections of code.

To Group Steps Together

Select the steps you wish to group together by first selecting the first step in the group and then Shift+Left Click on the last step in the group. These steps should become highlighted. Then right-click and select Group Steps.

A script either has grouping or does not. Therefore when you build a group from selected steps it will create 3 groups. The steps prior to your selection will be in an initial group, the steps you selected will be in a group, and the remainder of the script will fall into a final group. Additional groups can then be build by subdividing these initial groups by selecting an additional set and again grouping them.

Each group can be labeled with a group title to clearly identify what is contained within that group and the step ranges are automatically displayed (Figure 32-16).

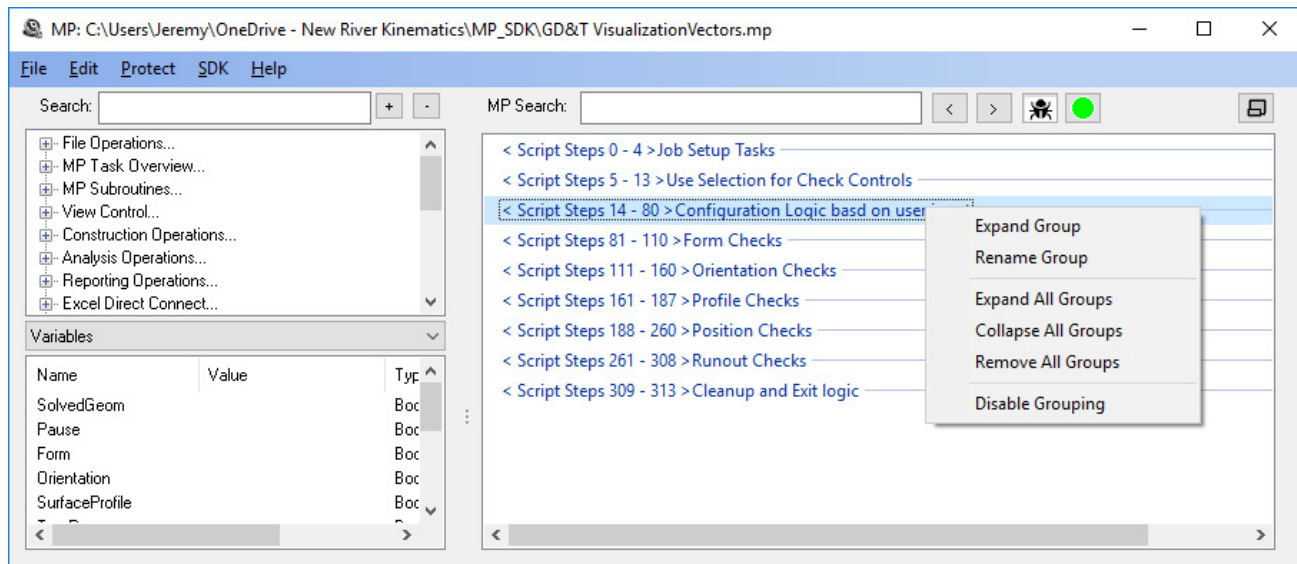


Figure 32-16. Grouping Steps

Enabling/Disabling Grouping

Grouping can be turned on and off as desired without losing the grouping definition and groups can be removed or added as needed. The grouping definition is for display only and will not impact the performance of an MP when it is running.

Working with Subroutines

Subroutines are a powerful means of expanding a script. It allows a sub-section of code to be saved separately and called any time it's needed. This can greatly reduce the size of single MP and allow you to use a common subroutine in several applications.

Subroutines are structured as follows:

- Within your master script add a step called **Run Subroutine**.
- Build a second script that has an initial step **Define Subroutine Input Values** and a final step **Define Subroutine Return Values**. The addition of these two steps makes a script a subroutine that can be called from the master script.

Sharing Information with Subroutines

Adding Input and Return Arguments

The **Run Subroutine** step is unique in that the number of arguments in the step will change based upon the number of input and return arguments defined in the subroutine.

Lets say for example that you want to pass a user's name to a subroutine. To do this, select the **Define Subroutine Input Values** step in your subroutine and press the **Add** button (Figure 32-17) which will appear at the top of the editor when the set is selected.

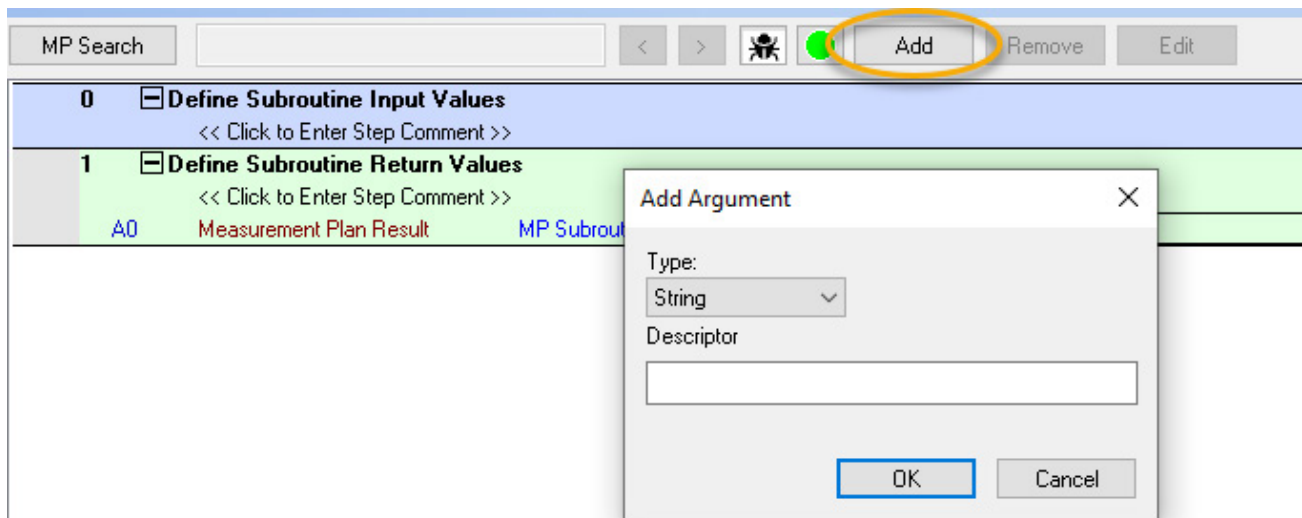


Figure 32-17. Adding an Input Argument in a Subroutine

The same is true for the *Return Values*. You will want to add any inputs to the subroutine and any returns you want to use and then when you connect it to the main script those arguments will appear in the Run Subroutine step in the main script. Continuing this example an input string "User Name" and a return argument "Report Name" were added to the subroutine and you can see they both appear as part of

the Run Subroutine command within the main script:

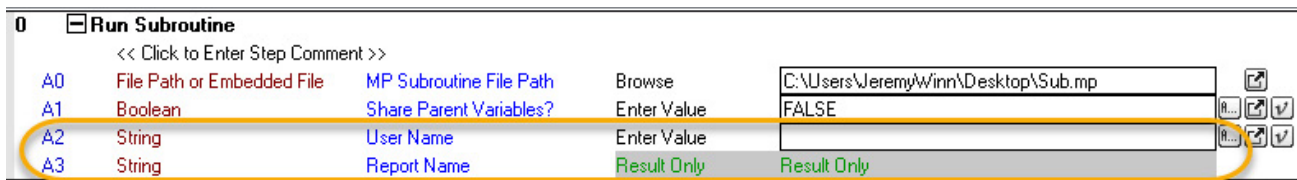


Figure 32-18. Run Subroutine
Input and Return Arguments appear
when defined within the Subroutine

This means that your workflow will be to:

1. Build your subroutine and add all the input and return arguments you want first.
2. Within your main script add the **Run Subroutine** command and point it at the subroutine.
3. Wire up all the inputs and returns to that subroutine.

Sharing Variables with a Subroutine

Within it the **Run Subroutine** step, Argument A1 is “*Share Parent Variables*”. This option is off by default so that you can use the same variable names in the main script and the subroutine without fear of confusion, but if you want you can enable it. When you do a variable in set in the Main script should then be available in the subroutine as well.

Writing Your First Script

Let’s go through the process of writing a short simple script just to see how it’s done. This script will ask for a number between 1 and 10, and it will count up to that number starting from 1. For instance, if the user enters 2, our script will display 1, 2, then exit.


1. Start a new script in the editor.


We first need to ask the user for the number to count to. We’ll do this using the **Ask for Integer** command, which allows the user to enter a whole number and returns the value entered.

2. In the search field above the command list, type **Ask For I**.


The search field will filter the available commands down—the **Ask For Integer** command should be the only one remaining.

3. Double-click the command in the command list, or select it and press **Enter**. It will get added as the first line in our script.

 **Tip:** For any boolean argument, you can type **0** to represent **FALSE** or any nonzero number to represent **TRUE**. The value will be converted automatically when you hit **Enter**.

4. In the *Question to Ask* argument, click in the box and type **Please Enter a number from 1 to 10**, then press **Enter**.
5. In the *Enforce Min/Max Values?* argument, click in the box and type **1**, then press **Enter**. The value becomes **TRUE**. This allows us to ensure that the user can only supply a value within a given range.
6. Set the *Min Value* to **1** and the *Max Value* to **10**.
7. In the search field/command list, find the command **Exit Measurement Plan** (again, starting to type this name in the search field is the fastest way to find the command). Add it to the script (it will appear at the end).
8. Back in the **Ask for Integer** command, set the *Step to jump to if Canceled* argument to **1**. If the user clicks the cancel button, the script will jump to the **Exit Measurement Plan** command in step 1 and exit.
9. Make sure a line in step 0 is selected, then add the command **Create Counter**. In the *Counter Value* argument, set the value to **1**. This will be the initial value of our counter.
10. Next, add an **Integer Comparison** command. This will allow us to check the current counter value against the value the user entered so we can determine when to stop counting.
11. Click below the command name where it says << **Click to Enter Step Comment** >>. Type **Keep Counting?** or a similar comment.
12. In the *Integer A* argument, click the reference quick-select button , then click the yellow box in Step 1, argument 0. The argument should now say **Ref {S1 A0}**.

We have now set this argument to reference the current counter's value in step 1, argument 0. This will feed the current counter value into this argument at runtime.

13. In the *Comparison Type* argument, click the box and select **<=** from the dropdown list.
14. For *Integer B*, again click the reference button  and select step 0, argument 7.

Essentially, this command is now comparing the “current” counter value against the number that the user entered. If the current counter is less than or equal to the supplied number, the command will jump to the *Step if TRUE* step. Otherwise, the command will jump to the *Step if FALSE* step. However, we don't yet have these steps written, so we can't yet fill them in.

Let's consider what happens if the current counter is less than or

equal to the supplied value. In this case, we want to display the current counter value.

15. After the **Integer Comparison** command, add a **Notify User Integer** command. Click the command name to select it and press **Tab** to indent the line over once (we're now inside the body of a loop).

Your script should now appear as depicted in [Figure 32-19](#).

Figure 32-19. Our first script, so far.

0 Ask for Integer						
<< Click to Enter Step Comment >>						
A0	String	Question to ask	Enter Value	Please Enter a number from 1 to 10	[Icons]	
A1	Integer	Initial Value	Enter Value	0	[Icons]	
A2	Boolean	Enforce Min/Max Values?	Enter Value	TRUE	[Icons]	
A3	Integer	Min Value	Enter Value	1	[Icons]	
A4	Integer	Max Value	Enter Value	10	[Icons]	
A5	Font Type	Font	Enter Value	MS Shell Dlg:8	[Icons]	
A6	Step ID	Step to jump to if Canceled (-1 will fail step on Cancel)	Enter Value	4	[Icons]	
A7	Integer	Answer	Result Only	Result Only	[Icons]	
1 Create Counter						
<< Click to Enter Step Comment >>						
A0	Counter	Counter Value	Enter Value	1	[Icons]	
2 Integer Comparison						
Keep counting?						
A0	Integer	Integer A	Reference	Ref {S1 A0}	[Icons]	
A1	Comparison Type	Comparison Type	Pick	<=	[Icons]	
A2	Integer	Integer B	Reference	Ref {S0 A7}	[Icons]	
A3	Step ID	Step if TRUE	Enter Value	-1	[Icons]	
A4	Step ID	Step if FALSE	Enter Value	-1	[Icons]	
3 Notify User Integer						
<< Click to Enter Step Comment >>						
A0	String	Leading Text	Enter Value		[Icons]	
A1	Font Type	Font	Enter Value	MS Shell Dlg:8	[Icons]	
A2	Integer	Reported Value	Enter Value	0	[Icons]	
4 Exit Measurement Plan						
<< Click to Enter Step Comment >>						

16. In the *Reported Value* argument, click the reference button [Icon] and select step 1, argument 0 (the counter value). This will display the current counter value to the user at runtime.

We need to increment the counter, or else we'll be stuck in an infinite loop.

17. Add an **Increment Counter** command after the **Notify User Integer** command. Click the reference button [Icon] and select the counter in step 1, argument 0. Press **Tab** to indent the command name.

When this command is reached, the current counter value will automatically be incremented by one.

Almost done. Now that we've incremented the counter, we need to jump back to our test—the comparison—to see if we should continue the loop.

18. Add a **Jump to Step** command. In the *Step ID* argument, enter 2 to jump back to the **Integer Comparison** command.

The last thing we need to do is fix the step references in the **Integer Comparison** command.

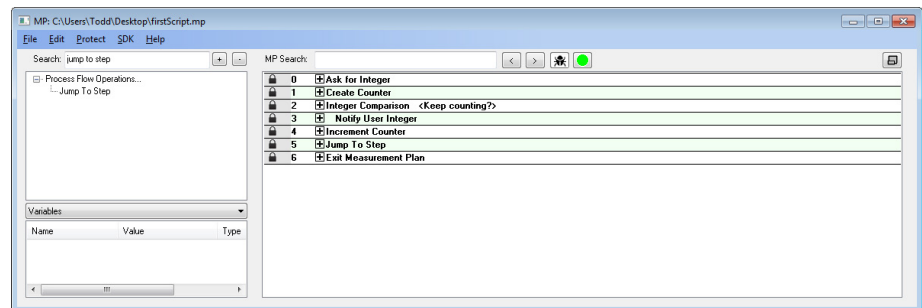
19. In the **Integer Comparison** command, set the *Step if TRUE* value to 3. This will enter the body of the loop if we're still counting.

20. Set the *Step if FALSE* value to **6**. When we're done counting and the counter becomes greater than the entered value, we'll exit the script.
21. Select all commands by pressing **Ctrl + A**. Then press the left arrow key to collapse all commands at once.


Let's lock the commands so we don't accidentally change them.

22. Ensure all commands are still selected, then click the left side of the column to the left of the step numbers. A padlock should appear on every line, indicating the lines are now "read-only" (Figure 32-20).

Figure 32-20. All steps are now locked.



That's it! Our script is done. Let's test it.

23. Click the run button  in the editor. If auto-run is disabled (see "To Enable/Disable Auto-Running of Scripts:" on page 841), click the same button in the *MP Bar*.

Did your script run as you expected? If not, try to find any errors and correct them.

Debugging



A large portion of the time spent writing a script is often spent on debugging—that is, finding any bugs and errors in the script and correcting them so that the script runs flawlessly in an execution environment.

There are several ways to test and evaluate your script's behavior.

Stepping Through a Script

One of the most obvious but still useful techniques for testing a script is to simply step through its evaluation one line at a time. This gives you the opportunity to view the effect of each command as it's executed—typically, scripts execute too quickly to observe what's happening.

When a script is loaded into the MP Bar, you can step forward one line at a time by clicking the Next Step  button.

In the *MP Editor*, you can enable debugging mode by clicking the Debug button . When you do this, the Next Step button  will appear in the editor window, allowing you to step through the script.

Debug Mode

The *MP Editor* has a debug mode that you can enable. Debug mode (Figure 32-21) provides several benefits for testing:

- You can step through a script one line at a time.
- Each command's arguments will display their runtime values, so that as you step through the script you can see the actual runtime values stored with each argument.
- The Variables view shows the current value of each variable as you step through the script.

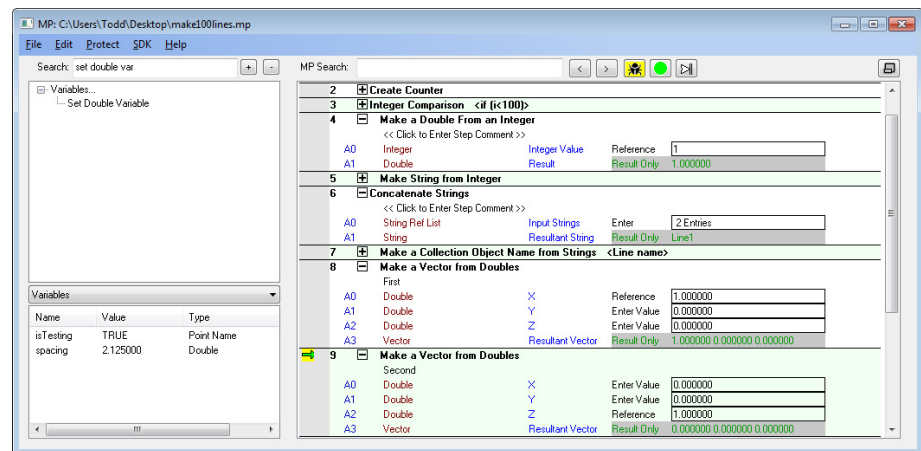




Figure 32-21. Debug mode shows variable values (left column) and current argument values.

These three items provide most of the tools you need to discover errors in your script.

 **Note:** Scripts will run noticeably slower in debug mode. This is typically fine, since you're testing—but it is something to be aware of.

Debug mode only works with a single script. This means that if a subroutine is called, you will not see any changes in the *MP Editor* window until the subroutine returns and execution continues in the current script. To view values in parent scripts *and* subroutines as a script executes, use the *MP Watcher*.

To Enable Debug Mode:

1. In the editor, click the Debug button . In debug mode, all arguments will become read-only and they will display their current values.
2. Run the script or step through it using the controls at the top of the window.

MP Watcher

The *MP Watcher* is very similar to debug mode in the editor. It is a separate window that allows you to view the current values of all com-

mand arguments as a script is executing. However, the MP Watcher adds each line to a list as it is executed. This means that unlike debug mode in the editor, if a script calls a subroutine, the subroutine's values can be seen while that subroutine is executing. Moreover, in a loop each iteration through the loop will add additional lines to the list, whereas in debug mode the previous iteration is "overwritten" with the current iteration's values.

The MP Watcher is shown in [Figure 32-22](#). It will only show commands that are executed starting from the time that the window was opened. In other words, it is possible to run partway through a script, then open the MP Watcher. Only commands from that point on will be displayed.

Scripts will run noticeably slower with the MP Watcher open, so make sure that this window is not open when execution speed is important. (Typically, this is fine, since the window is only used during the testing phase of script development).

To Print the Contents of the MP Watcher Window:

- Click the **Print** button.

To Export the Contents of the MP Watcher to an Excel Spreadsheet:

- Click the **Excel Export** button.

To Export the Contents of the MP Watcher to a PDF File:

- Click the **PDF Export** button.

To Clear the Contents of the MP Watcher Window:

- Click the **Clear** button.

To Search for Content in the MP Watcher Window:

- Click the **Search** button. You can search for text in the title, the comment field, for a specific value, or search among a specific command.

To Limit the Number of Steps Displayed in the MP Watcher:

- Click the **Options** button. Set the *Max Displayed Steps* option, then click **OK**.

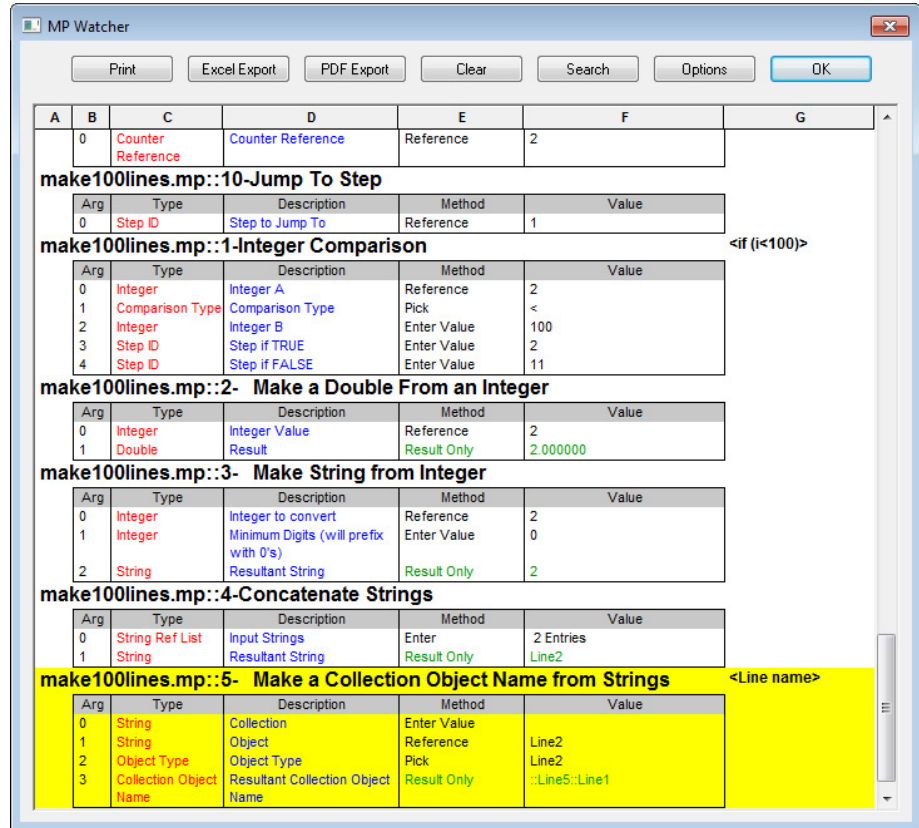


Figure 32-22. The MP Watcher in mid-execution.

The ability to automate measurement, analysis, and reporting tasks in SA can prove to save an enormous amount of money, time, and effort. Automation through Measurement Plans or the SA SDK can introduce efficiencies in your metrology processes unmatched by anything else.

SA SDK

The SpatialAnalyzer SDK provides a means by which to write custom applications that utilize Measurement Plan functionality within Visual C++, VB.NET, or other development environments that support ActiveX controls. Once the SDK engine has been added to the development environment, Measurement Plan steps can be executed from within the programming language.

The advantages of the SDK over Measurement Plans include the following:

- Since the code is written in a full-featured programming language (such as Visual C++ or VB.NET), it offers the full flexibility and power of those programming languages.
- Complex math operations and data manipulation is often easier than when written in MPs.
- There is no need to write routines to transfer data back and forth between MPs and custom external applications--all data is handled within the SDK application itself.
- SDK code is compiled and can be distributed as a true application.
- SDK code is more difficult to reverse engineer than MPs, so proprietary processes can be kept more secure.

The disadvantages of the SDK when compared to MPs include the following:

- It is more difficult to understand for a user not familiar with

programming. Creating automated routines is less intuitive.

- It requires at least basic knowledge of a real programming language and development environment.

SDK Engine

Whenever a client application that you have developed is executing, the SpatialAnalyzer SDK engine will automatically start and appear minimized on the Taskbar. [Figure 33-1](#) shows the engine in its maximized state:



Figure 33-1. The window displayed when the SDK Engine is running.

Automatic Code Generation

SpatialAnalyzer supports the automatic generation of SA SDK code for Visual Basic and Visual C++ languages through the MP Editor. This is an excellent way to prototype SDK applications, or convert MPs into full SA SDK applications. To access this sample code, set up an MP with your desired arguments. Then, press the SDK Code Generation button in the MP Editor ([Figure 33-2](#)).

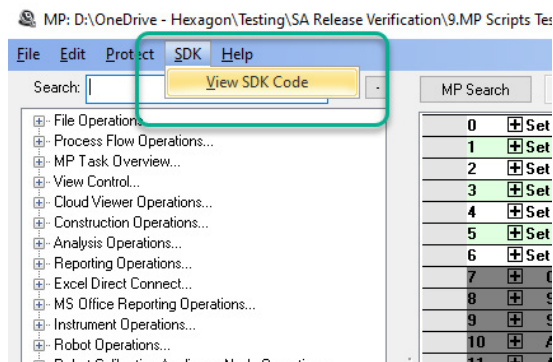


Figure 33-2. The SDK Code Generation button.

The necessary code will be displayed in a window and automatically

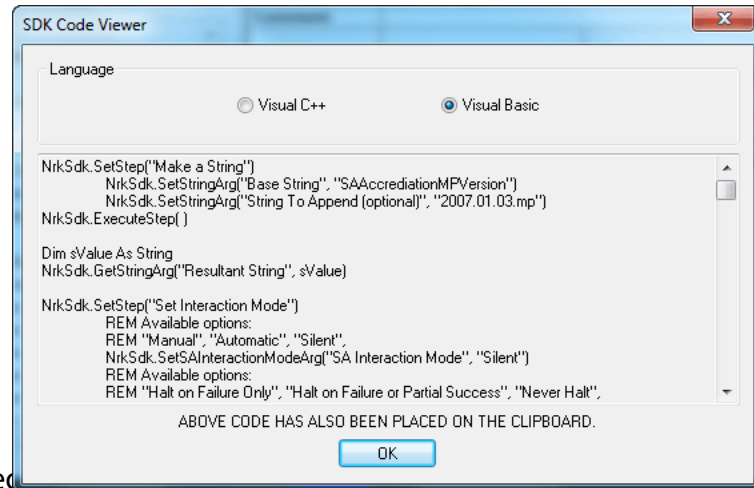


Figure 33-3. The SDK Code Viewer.

copied

You can then paste this code into your SDK application.

Setting Up an SDK Project

A selection of examples are available for different programming languages and can be downloaded from our website here:

<https://www.kinematics.com/ftp/SA/Install/Driver%20Downloads/SA%20SDK/Examples/>

The listing currently includes:

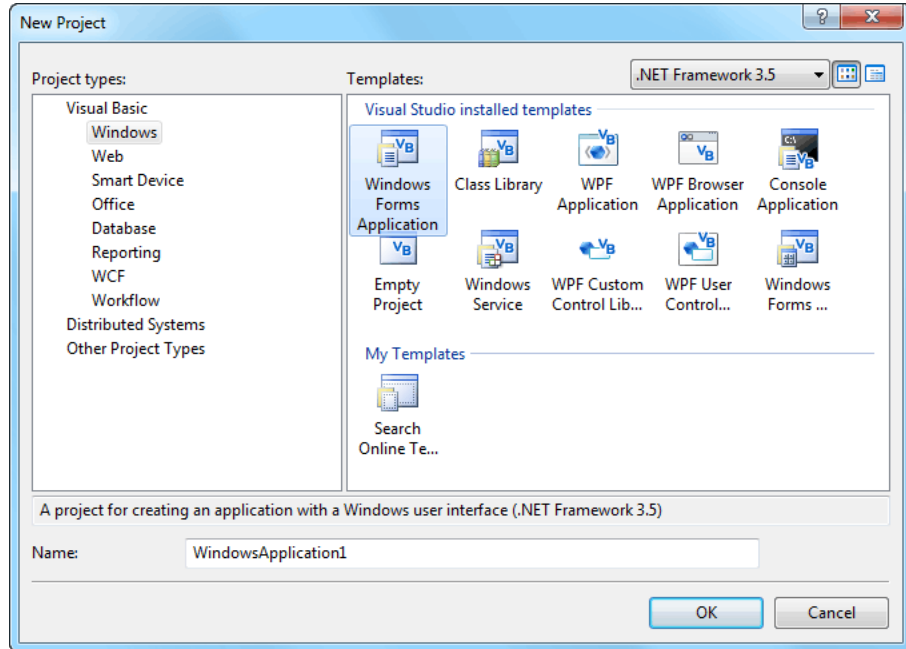
- SDKTesterVB
- SDKTester C++ vs10
- SDKTester C++ vs19
- SDKTesterCSharp

SA also offers a couple of alternative SDK's that users may want to be aware of for instrument integration. If you are interested in writing your own instrument interface for your own hardware we can provide an alternative, more comprehensive option for that. For more information reach out to support@kinematics.com for additional details.

VB.NET

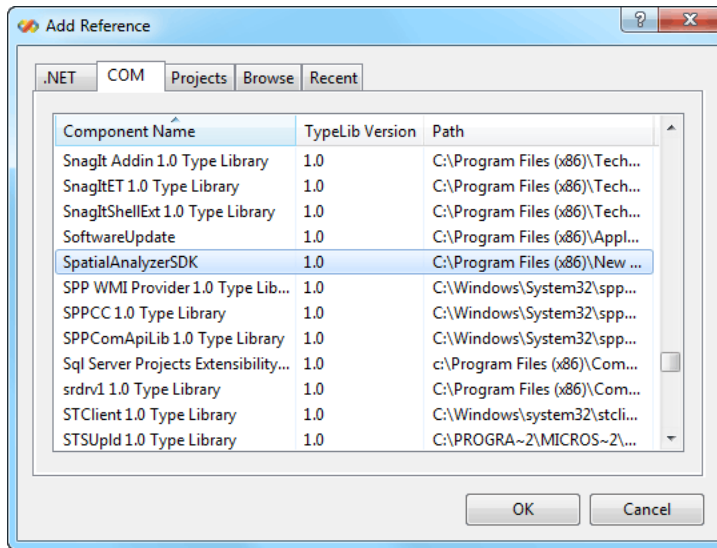
1. Create a new Visual Basic Windows Forms Application project

Figure 33-4. Creating a new VB.NET project.



2. Type a name for your application, then press **OK**.
3. Select **Project>Add Reference...**

Figure 33-5. Adding the SpatialAnalyzerSDK COM object.



4. In the **Add Reference** dialog, select the **COM** tab, then scroll down in the list and choose **SpatialAnalyzerSDK**. Click **OK** (Figure 33-5).
5. Double-click the form in your project. You should be presented with the code for the `Form1_Load` method of your form (Figure 33-6).

Figure 33-6. The default form.

```
Public Class Form1
    Private Sub Form1_Load(ByVal sender As Object, ByVal e As EventArgs) Handles MyBase.Load
    End Sub
End Class
```

6. Replace this class code with the following:

```
Public Class Form1
'Declare an object for the SA SDK Interface
Private NRKSdk As SpatialAnalyzerSDK.SpatialAnalyzerSDK

'Declare an enumerated type that defines the various return codes from the SA
'SDK Interface.

Enum MPStatus
    SDKERROR = -1
    UNDONE = 0
    INPROGRESS = 1
    DONESUCCESS = 2
    DONEFATALERROR = 3
    DONEMINORERROR = 4
    CURRENTTASK = 5
End Enum

'Allocate the SDK Interface object
Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.
EventArgs)
Handles MyBase.Load
    NRKSdk = New SpatialAnalyzerSDK.SpatialAnalyzerSDK
End Sub
End Class
```

7. Ensure SA is running, then compile and run the application.

Visual C++

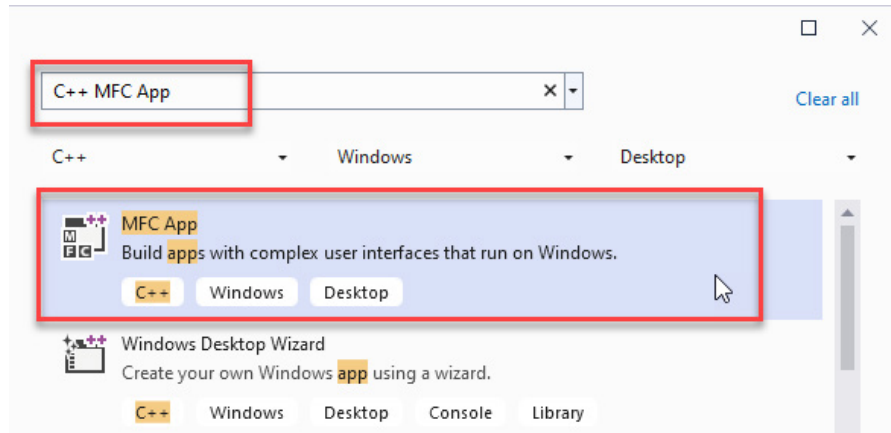
Prior Visual Studio 2022, C++ programmers could use “MFC Class from TypeLib” Visual Studio C++ wizard to add the SDK engine to their C++ projects. The “MFC Class from Type Lib” wizard is not available in Visual Studio 2019 anymore.

Steps for creating a new SA SDK C++ MFC App project

First, setup a new C++ Project (the following steps build a Dialog App basedd project)

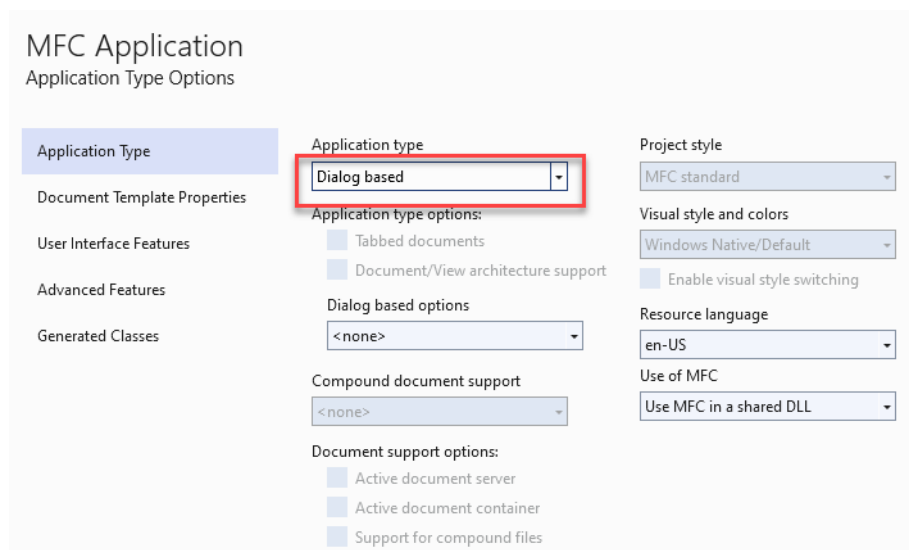
1. Create a new Visual C++ MFC project within Visual Studio ([Figure 33-7](#)). Select **File>New Project...**

Figure 33-7. Creating a new MFC C++ project.



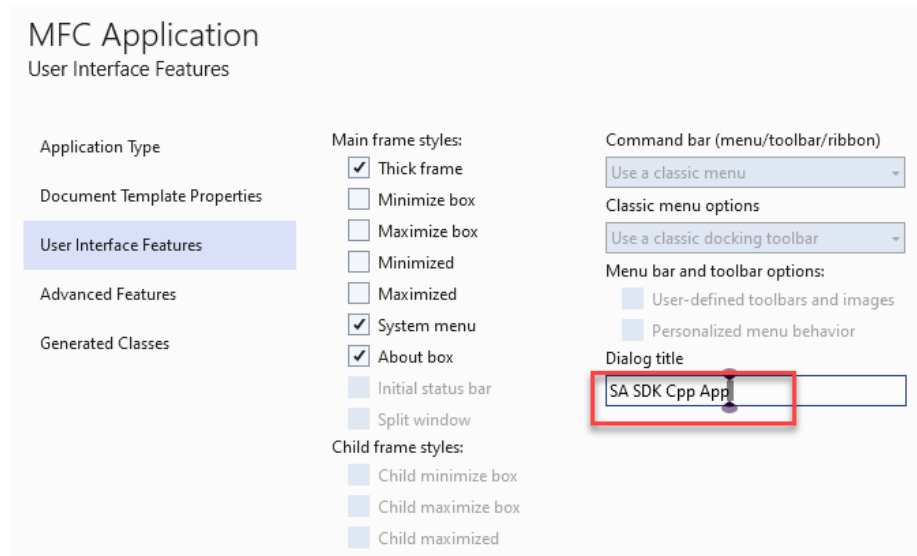
2. Type a name and location for your project, then click **OK**.
3. For purposes of illustration, let's build a dialog application (Figure 33-8). To do so, in the Application Type Options dialog select *Dialog based* Application Type (Figure 33-8).

Figure 33-8. The MFC Application Wizard.



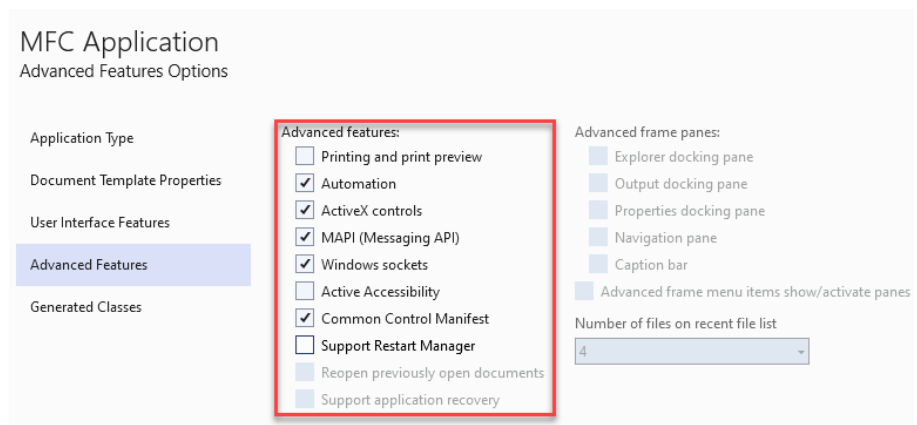
5. No options are needed on the Document Template Properties dialog. Select *Next*.
6. Enter the *Dialog Title* into the User Interface Features dialog and select any Main frame styles you would like (optional) (Figure 33-9).

Figure 33-9. Naming the new Dialog.



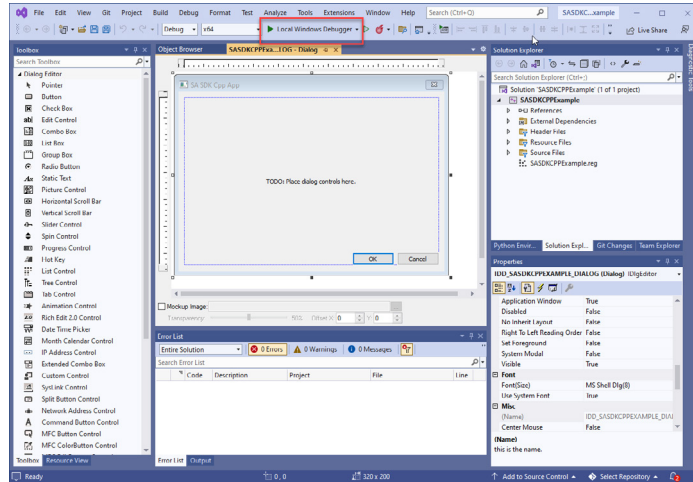
7. In the Advanced Features Options dialog, ensure that *Automation* is selected. An example of the Advanced Feature Options is shown. Select *Next* after configuring these options (Figure 33-10).

Figure 33-10. Setting Advanced Feature options.



A new Dialog Project should now be setup and ready to build (Figure 33-11).

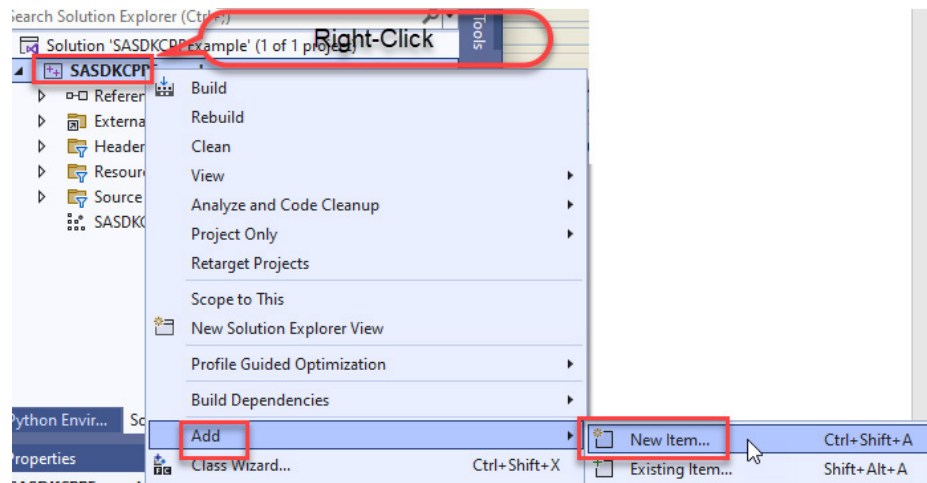
Figure 33-11. Completed MFC Dialog App, ready to begin development.



Once the C++ project is setup add the SA SDK Class

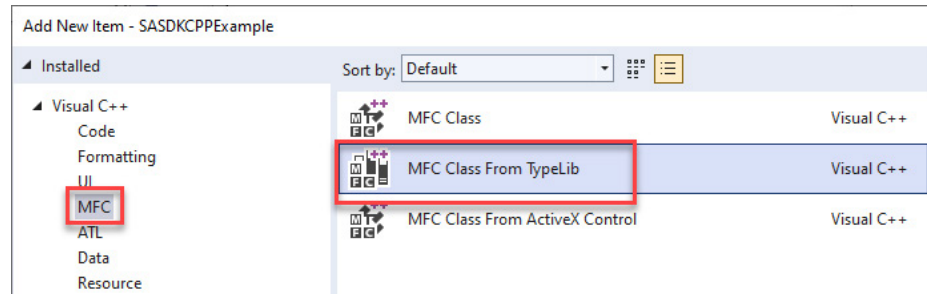
1. Right-click on the Project Name in the Solution Explorer. Select Add >> New Item... (Figure 33-12)

Figure 33-12. Add MFC from TypeLib



2. Select the MFC object and MFC Class from TypeLib, and then the Add button at the bottom (Figure 33-13).

Figure 33-13. Adding a new class.



3. In the Add Class From Typelib dialog, select the down arrow on the *Available Type Libraries*. Scroll down and select using the check box, the **SpatialAnalyzerSDK <1.0>** option (Figure 33-14).

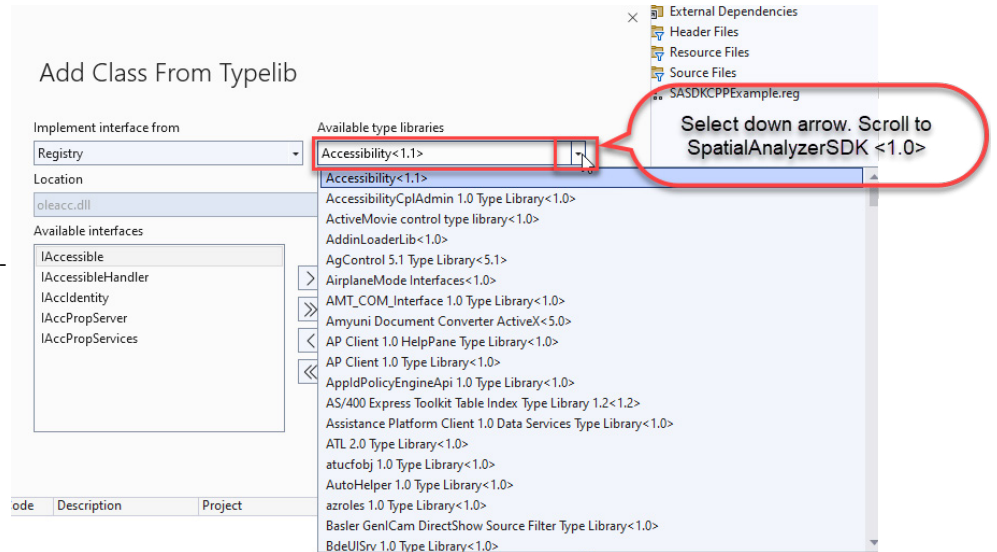


Figure 33-14. Selecting SpatialAnalyzerSDK <1.0>

4. Select the ISpatialAnalyzerSDK interface and add it to the Implement Interfaces list box. Then select Ok (Figure 33-15).

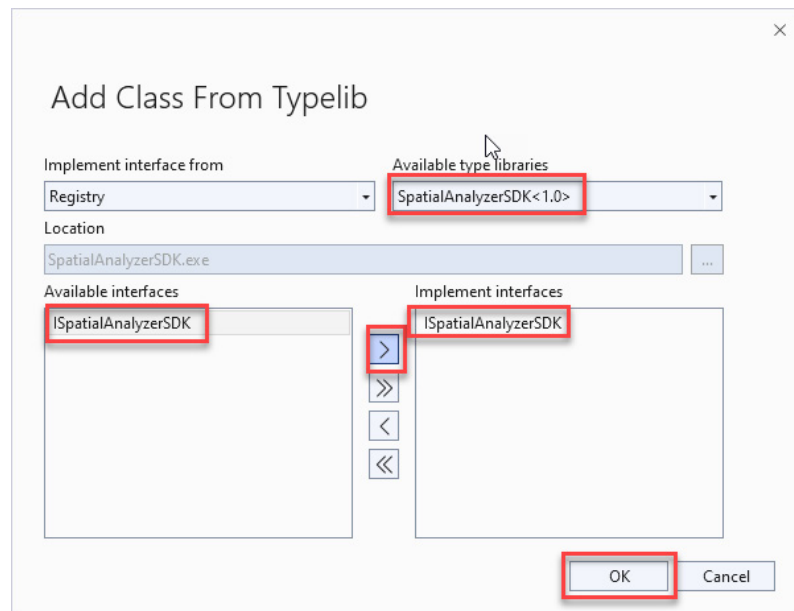
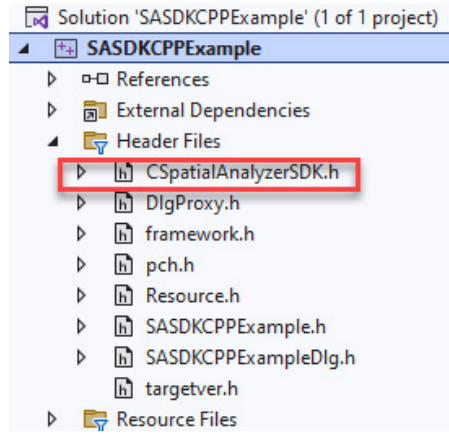


Figure 33-15. Adding the ISpatialAnalyzerSDK Interface.

CSpatialAnalyzerSDK.h will automatically be added to your project and contains all of the available SDK declarations (Figure 33-16).

Figure 33-16. CSpatialAnalyzer.h should now be available to you.



Visual Studio 2019 C++ Notes

The notes below present an alternative way how to generate C++ wrappers for the objects/interfaces declared in a typelib in Visual Studio C++ 2019 by using #import directive and smart pointers for COM object.

1. Create dialog-based project in Visual Studio 2019 (Figure 33-17) and (Figure 33-18):

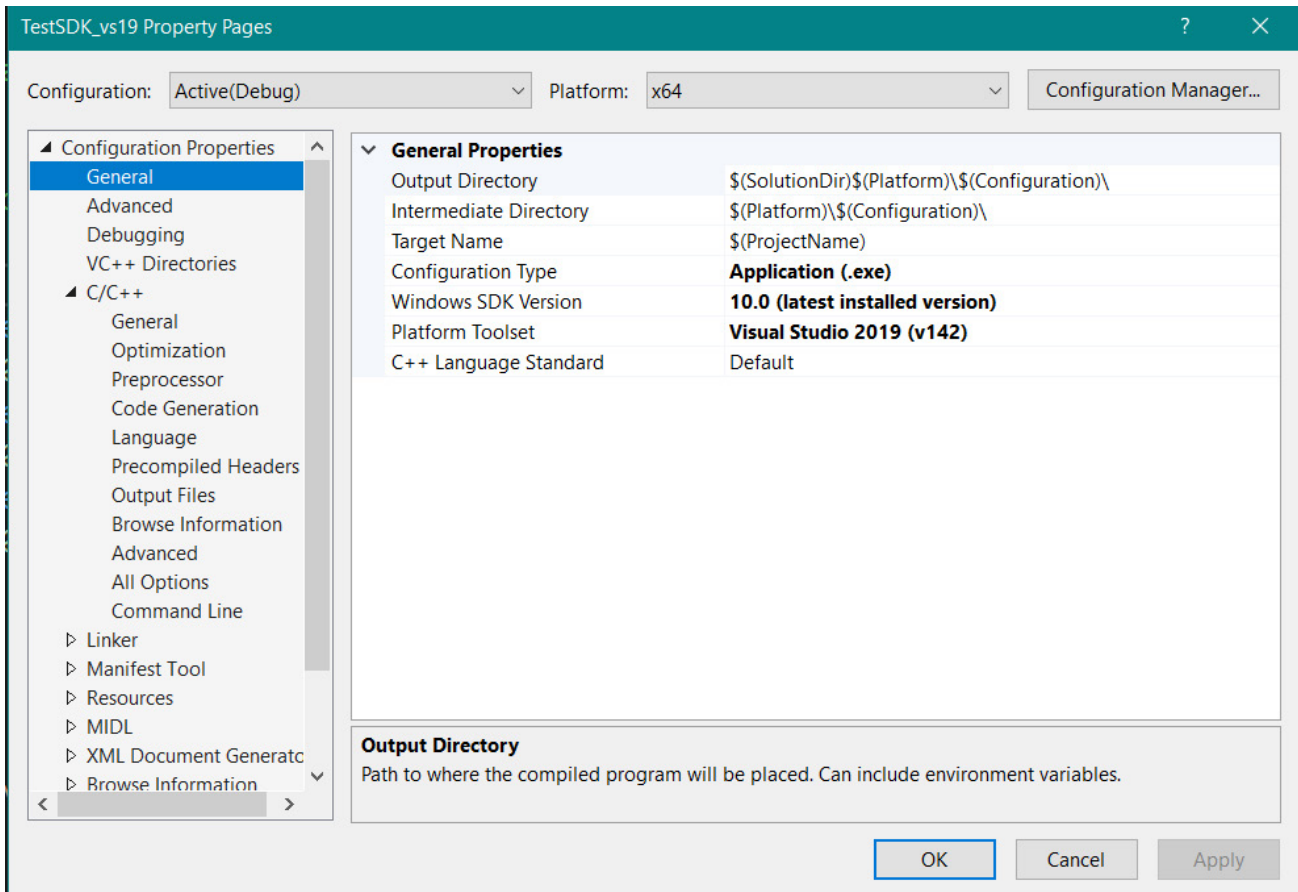


Figure 33-17. Dialog based project in Visual Studio 2019 - General

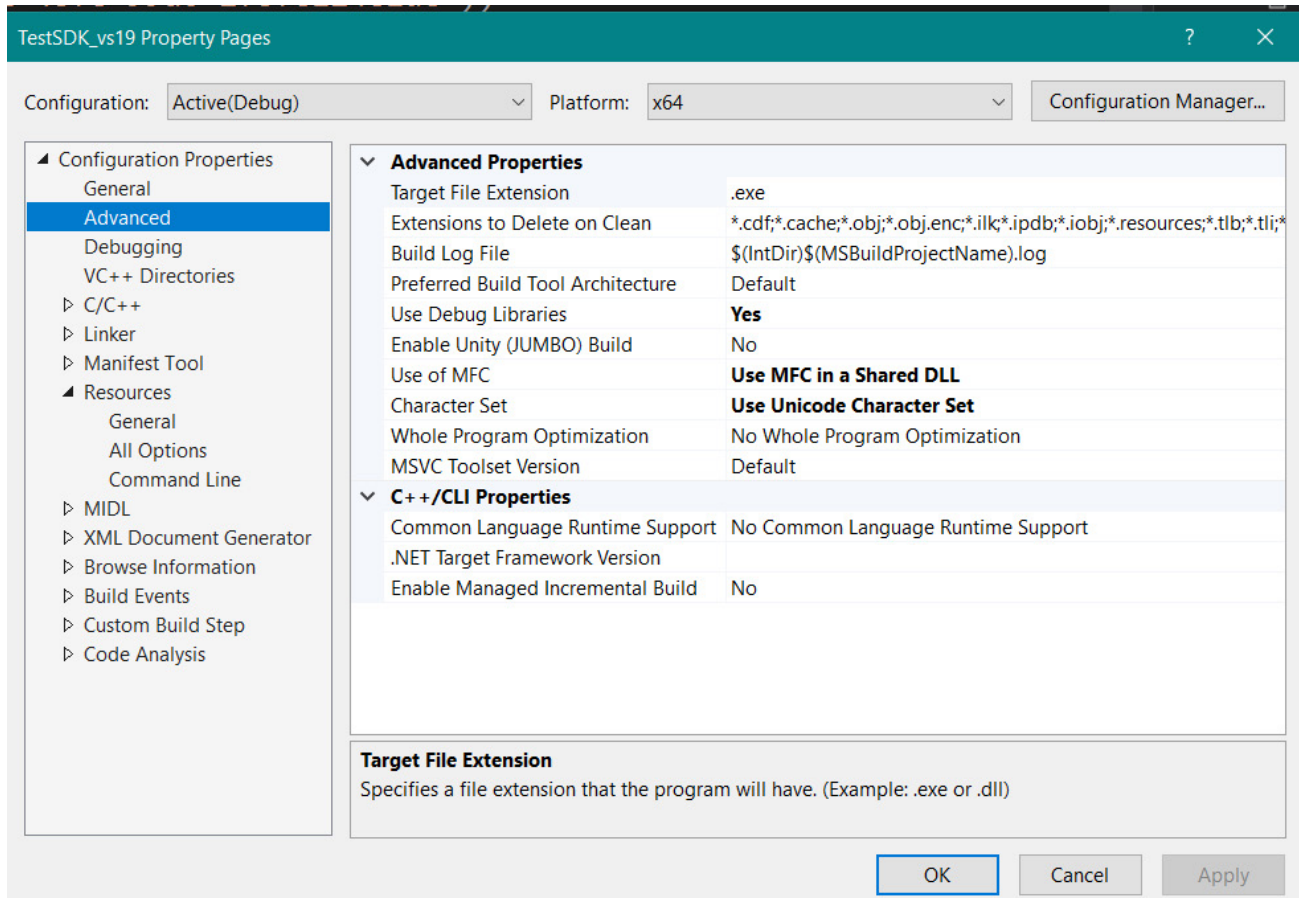


Figure 33-18. Dialog based project in Visual Studio 2019 - Advanced

2. Use the smart pointer technology of Visual Studio and import the server's type information using `#import` directive in `framework.h` file as the following:

```
#import "C:\\Program Files (x86)\\New River  
Kinematics\\<SpatialAnalyzer XXXX.XX.XX_XXXX>\\SpatialAnalyzerS-  
DK.exe" no_namespace named_guids
```

where two attributes are:

- `no_namespace` - specifies that the compiler doesn't generate a namespace name
- `named_guids` - tells the compiler to define and initialize GUID variables in `.tlh` file:

```
LIBID_SpatialAnalyzerSDK
```

```
IID_ISpatialAnalyzerSDK
```

```
CLSID_SpatialAnalyzerSDK
```

Placing `#import` for typelibs in the precompiled header, ensures we don't unnecessarily repeat generating the `.tlh` and `.tli` files each build.

3. Build your project. When the compiler processes the #import directive, it will generate two new C++ files *.tlh (type library header file) and *.tli (type library implementation file) and place them by default into your Intermediate Directory.
4. Open *.tlh file and take a look at smart pointer and named GUID constants sections(Figure 33-19):

```
//
// Smart pointer typedef declarations
//
_COM_SMARTPTR_TYPEDEF(ISpatialAnalyzerSDK, __uuidof(ISpatialAnalyzerSDK));
```

```
//
// Named GUID constants initializations
//
extern "C" const GUID __declspec(selectany) LIBID_SpatialAnalyzerSDK =
    {0x860f748c, 0xc80b, 0x4216, {0x93, 0x86, 0x18, 0x73, 0x57, 0xf5, 0x20, 0xf7}};
extern "C" const GUID __declspec(selectany) DIID_ISpatialAnalyzerSDK =
    {0xb2976e35, 0xe89b, 0x4862, {0xa0, 0x17, 0xdb, 0xf2, 0x9e, 0x19, 0x98, 0xcd}};
extern "C" const GUID __declspec(selectany) CLSID_SpatialAnalyzerSDK =
    {0x7c73dbf2, 0x1921, 0x48fe, {0x99, 0xa6, 0x2f, 0x8f, 0xe1, 0x24, 0xe1, 0xde}};
```

Figure 33-19. GUID constants

5. In order to create COM object using smart pointer, define a smart pointer variable *dlg.h file:

```
ISpatialAnalyzerSDKPtr m_pNRKSdk;
```

and call the intrinsic *CreateInstance()* method of *_com_ptr_t<>*, passing CLSID of the object (see *CTestSDKvs19Dlg::OnBnClickedConnect()*):

```
m_pNRKSdk.CreateInstance(CLSID_SpatialAnalyzerSDK);
```

6. Once a smart pointer created, you can access SA SDK interface methods. For example,

```
m_pNRKSdk->Connect(_T("localhost"));
```

or

```
m_pNRKSdk->SetStep(_T("Set Active Units"));
```

7. When finish, make sure to set the smart pointer to NULL. The *Release()* will be called automatically when it drops out of scope.

Sending Commands to SA

Establishing a Connection

The SDK consists of a set of function calls that allow the specification and execution of MP Steps. Before executing any commands in SA, your SDK code must first connect to SpatialAnalyzer. You can connect

to SA using one of two methods: `Connect` (deprecated) and `ConnectEx`. The deprecated `Connect()` method returns a simple boolean `YES` or `NO` answer, indicating whether the connection was successful:

```
'Connect to SA. If unable to connect, display a failure message.
If Not (NRKSdk.Connect("localhost")) Then
    MsgBox ("Failed to connect with automation server!")
End If
```

The `ConnectEx()` method allows you to pass a long integer variable (by reference) and obtain a status code from the command which indicates whether a connection was successful or not, and why:

```
'Connect to SA. If unable to connect, display a failure message.
Dim statusCode As Long
NRKSdk.ConnectEx("localhost", statusCode)
If (statusCode)
    MsgBox ("Connection error, code: " & statusCode)
End If

'Possible values for statusCode include:
'CONNECTED = 0,
'CONNECTION IN USE = -1    An SDK Client is already connected to SA
'INVALID_LICENSE = -2    Invalid SA SDK license
'COULD_NOT_CONNECT = -3    Failed to connect to specified host
'SDK_SOCKET_ERROR = -4    Other Socket error
```

Executing an MP Step

Once you've successfully connected, you can now execute any MP command by adhering to the following process:

Declaring the Command

The first step in executing a command is to tell the SDK which MP command you want to execute by using the `setStep()` method:

```
'Tell SA which MP step we're defining.
NRKSdk.setStep("Construct a Point in Working Coordinates")
```

This example would tell SA that you are in the process of defining the `Construct a Point in Working Coordinates` command. Note that the string provided as an argument to the `setStep()` method must match the name of the MP command exactly.

Setting the Arguments

Once you've declared the command, you need to provide input arguments using the argument setting methods such as `setVectorArg()` and `setPointNameArg()`:

```
'Set the "Working Coordinates" argument of the command.
NRKSdk.SetVectorArg("Working Coordinates", 10.1234, 20.2345, 30.5678)

'Set the Point Name argument of the command.
'We want to put this point in the active collection, in the TestGrp point
'group, and name it "TestPt".
NRKSdk.SetPointNameArg("Point Name", "", "TestGrp", "TestPt")
```

When doing this, the first string provided to the methods is from the *Description* field of the MP command. This string must match what is seen in the MP, in English. The remaining arguments to the method depend on the type of argument.

Executing the Command

Now that the MP command is fully defined, you need to tell SA to execute the step. This is performed using the `executeStep()` method:

```
'Create a boolean variable to hold the result of the execution.  
Dim bSendStatus As Boolean  
  
'Execute the step, and assign the result to our boolean variable.  
bSendStatus = NRKSdk.ExecuteStep()
```

Checking the Result of a Command

If the SDK fails to execute the command (for example, because you have not properly defined the arguments), then `bSendStatus` (above) will be `False`. If the command succeeds, the value will be `True` and you can further obtain the MP step result using the `getMPStepResult()` method. That is, you can determine whether the MP step passed, partially failed, or completely failed:

```

'Only do this if the MP command executed properly.
If (bSendStatus) Then
    'Declare a long integer to hold the result code.
    Dim result As Long
    'Get the MP step result, and assign it to result.
    NRKSdk.GetMPStepResult(result)

    'Check the result of the MP command.
    Select Case result
        Case MPStatus.DONESUCCESS
            MsgBox("Success!")
        Case MPStatus.DONEMINORError
            MsgBox("Partial Success!")
        Case MPStatus.DONEFATALERROR
            MsgBox("Failure!")
        Case MPStatus.SDKERROR
            MsgBox("There was an SDK Error!")

            'Create an object to hold the error message
            Dim msgArray As Object

            'Display the text returned from execution attempt to
            'user...
            If (NRKSdk.GetMPStepMessages(msgArray)) Then
                'Declare a variable to hold the array index.
                Dim i

                'Declare a string to hold the message
                Dim msg As String = ""

                'Loop through the array to assemble the
                'message.
                For idx = LBound(msgArray) To
                    UBound(msgArray)
                    msg += msgArray(idx) + Chr(10) +
                        Chr(13)
                Next

                'Now display the message.
                MsgBox(msg)
            End If
        Case MPStatus.UNDONE
            MsgBox("Command hasn't been executed!")
        Case MPStatus.INPROGRESS
            MsgBox("The command is currently being executed...")
        Case MPStatus.CURRENTTASK
            MsgBox("The step is the current task.")
    End Select
End If

```

Getting Return Arguments

If an MP contains a "Result Only" argument, then the result argument can be retrieved by calling the appropriate argument get methods, such as `getVectorArg()` and `getPointNameArg()`. For example, if the MP step successfully executes and a vector result argument is returned, the following code could be used to retrieve the result:

```
If (bSendStatus) Then
    'Declare a long int to hold the return code.
    Dim result As Long
    NRKSdk.GetMPStepResult(result)
    If (result = MPStatus.DONESUCCESS) Then
        'Get the return values and store in variables.
        Dim xVal, yVal, zVal As Double
        NRKSdk.GetVectorArg("Vector Representation", xVal, yVal, zVal)
    End If
End If
```

The OPC Interface

CHAPTER

34

SA supports the OPC standard, enabling a wide-variety of possibilities for real-time communication between itself and other applications or process control hardware. This chapter discusses the capabilities and use of this interface.

What Is OPC?

OPC is an acronym that originally stood for “OLE for Process Control”, but has since been renamed to “Open Platform Communications”. It is a standard originally developed in 1996 to enable communication of real-time factory data between various hardware devices and software applications from different manufacturers.

OPC as a communication standard enables various devices in a manufacturing environment to communicate among themselves—for instance, allowing real-time measurement data to be monitored by other hardware, or to track part progress through an assembly line in a real-time database.

SA & OPC

The OPC DA (Data Access) server functionality is available in SA Ultimate (64-bit only). SA’s implementation is based on version 2.05 of the OPC DA specification. Any client that complies with this specification can communicate with SA’s server—that is, that client has the ability to read from and write to SA’s OPC server.

Prerequisites

In order to activate and use the OPC DA server in SA, the following prerequisites must be satisfied:

- The OPC Core Components Redistributable (x64) 3.0 package must be installed on the system,
- 64-bit SA Ultimate must be installed,
- SA's OPC server must be registered and enabled, and
- The OPC server DCOM and network settings must be configured for remote client access.

The OPC Foundation (<http://www.opcfoundation.org>) provides a compliance test client for the data access 2.05 specifications to all of its members. A sample client which can be used to try out new SA functionality can be downloaded for free from one of the following websites:

- www.softing.com/opc (select OPC Classic options)
- www.matrikonopc.com/products/opc-desktop-tools/opc-explorer.aspx

Configuring the OPC Server

● **Tip:** It is recommended that you create a system restore point prior to changing the system configuration.

Step 1: Creating a DCOM User

Windows 7 Home Premium

► **Note:** For Windows 7 Home Premium, this must be an administrator account.

1. Login to Windows 7 under an administrator account.
2. Under the Windows Control Panel > User Accounts, select **Manage another account**. Then click **Create a new account**.
3. Add a new administrative user **OPCUser**, password **nrkopc**.

Windows 7 Professional

1. Under the Windows Control Panel, select **Administrative Tools**, then select **Computer Management**.
2. Expand the **Local Users and Groups** item in the tree.
3. Right-click **Users** and select **New User...** from the context menu.
4. Enter **OPCUser** for the *User Name* and **nrkopc** for the password. Enter a description as well ([Figure 34-1](#)).
5. Click **Create**, then **Close**.

6. Add the new user to the **Distributed COM Users** group (Figure 34-2).

Step 2: Disable Windows Firewall

1. Under Windows Control Panel, select **Windows Firewall**.
2. From the sidebar, select **Turn Windows Firewall On or Off**.
3. Select the *Turn off Windows Firewall (not recommended)* option.

Figure 34-1. Setting up an OPC user account.

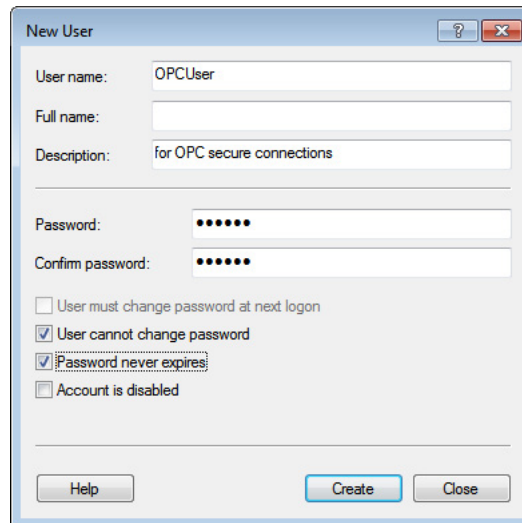
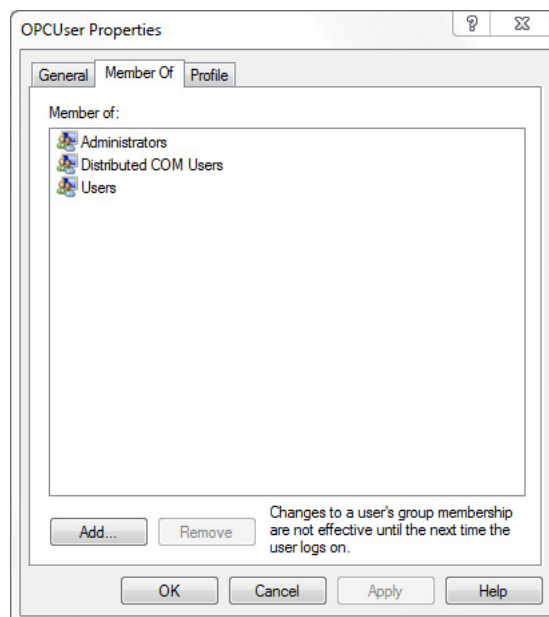


Figure 34-2. Making the new user a member of the **Distributed COM Users** group.



Step 3: Adjust User Account Control Settings

1. As administrator, type **uac** in the start menu.
2. Set the UAC to *Never Notify*, then click **OK**.

3. Restart the computer.

Step 4: Install SA

1. Log into the **OPCUser** account.
2. Install SA as usual.
3. If SA automatically starts at the end of the install, close it.


Step 5: Install OPC Core Components

1. Run the OPC Server Utility (C:\Program Files (x86)\New River Kinematics\SpatialAnalyzer <version>\x64\OpcServerUtility.exe).
2. If the core components are missing on your system, the utility will notify you—continue with the following. Otherwise, skip to step 6.
3. To install the missing components, you download the OPC Core Components Redistributable (x64) package from the OPC Foundation website (<https://opcfoundation.org/>).
4. After installing the core components, restart your computer.

Step 6: Register the NRK OPC DA Server

1. Login to the **OPCUser** account.
2. Ensure SA is not running, then run the OPC Server Utility (see Step 5, item 1).
3. Click the **Register SA OPC Server** button.

Step 7: Enable the OPC Server in the SA Configuration

1. Run SA Ultimate (x64) or greater.
2. In SA's **User Options**  dialog, select the *Machine Configuration* tab.
3. Turn on the *Enable OPC Server* option. The server will now run in the future whenever SA is running.

Step 8: Test the Server with a Local Client

1. Install a client onto the machine. For instance:
 - www.softing.com/opc (select OPC Classic options)
 - www.matrikonopc.com/products/opc-desktop-tools/opc-explorer.aspx
2. Start the client application.
3. Verify that the **NRK OPC DA Server** is presented in the list of

► **Note:** At least one tag (**NRK SA Version**) is always available in the OPC server's address space. If you add an instrument to your job file, SA automatically adds the instrument's coordinates to the address space.

local v2 data access servers.

4. Connect to the SA OPC server and browse its tags.
5. Add items to the client's group and verify that data is updating as appropriate.
6. Once proper local operation is verified, close the local OPC client and SA Ultimate before continuing on.

Step 9: Configuring System-wide DCOM Settings

1. Ensure you're logged in as **OPCUser**.
2. Run the OPC Server Utility (step 5, item 1).
3. Click the **View DCOM Configuration** button.
4. Right-click on **My Computer** and select **Properties** from the context menu.
5. In the *Default Properties* tab, ensure that the *Enable Distributed COM on this computer* option is enabled. The *Default Authentication Level* should be set to *Connect*, and the *Default Impersonation Level* should be set to *Identify* (Figure 34-3).

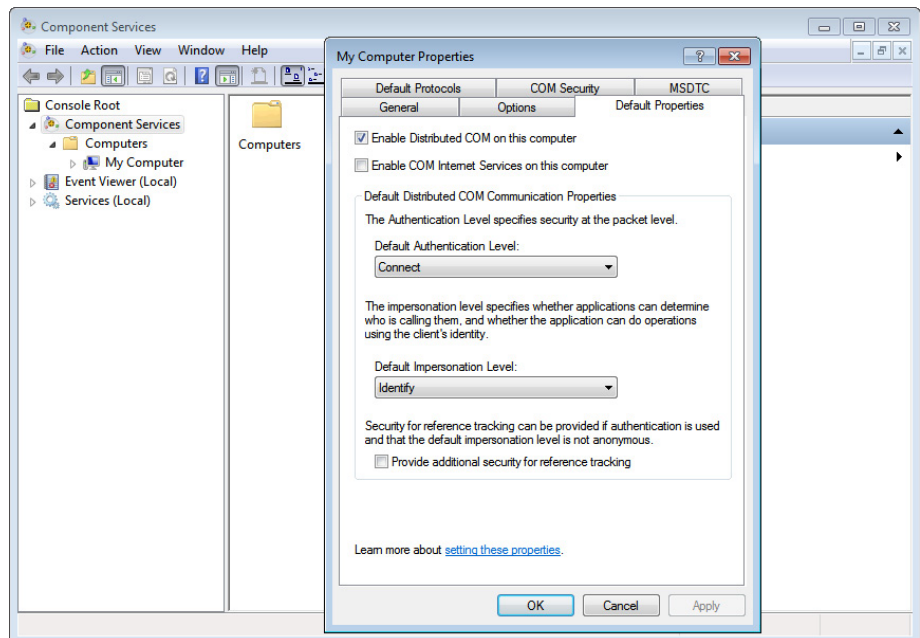


Figure 34-3. Setting the computer's default component services properties.

6. Select the *COM Security* tab. Click the **Edit Default...** button in the *Access Permissions* group.
7. In the *Access Permission* dialog, add permission for **Everyone** and the **ANONYMOUS LOGON** users, then click **OK**.
8. Click the **Edit Limits...** button in the *Access Permissions* group.
9. In the *Access Permission* dialog, add permission for **Every-**

► **Note:** Both of these are required for OPC functionality.

► **Note:** Do not adjust system-wide COM security Launch and Activation permissions limits (the *Edit Limits...* button).

one and the **ANONYMOUS LOGON** users, then click **OK**.

10. Click the **Edit Default...** button in the *Launch and Activation Permissions* group.
11. In the *Launch and Activation Permission* dialog, add permission for **Everyone** and the **ANONYMOUS LOGON** users, then click **OK**.
12. Click **OK** again to close the *My Computer Properties* dialog.

Step 10: Configure the OPC Server's Specific DCOM Settings

1. In the *Component Services* window, open the properties of the **NRK SA OPC DA Server** (Figure 34-4).

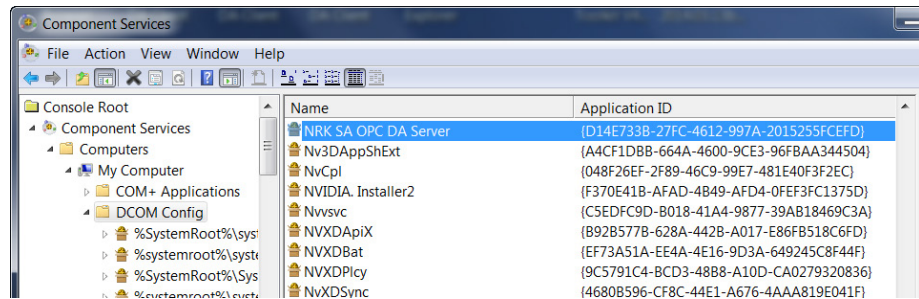


Figure 34-4. Opening the properties for the NRK SA OPC DA Server component.

► **Note:** The OPC clients are not allowed to launch SA. For this purpose the *Local Path* parameter is not configured and the *Location* tab's parameters are not set.

2. In the *General* tab, set the *Authentication Level* to *Default*.
3. Verify that the *Security* tab's *Launch and Activation Permissions* are set as indicated in Figure 34-5 (no launch is allowed).

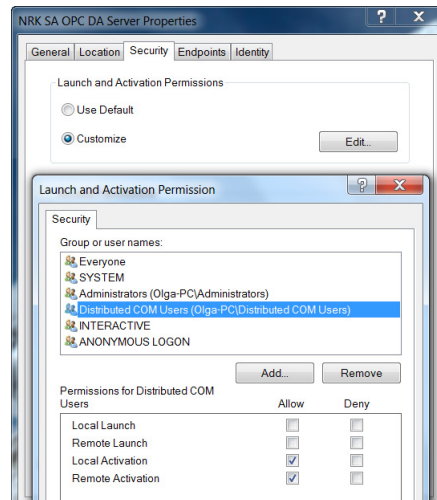
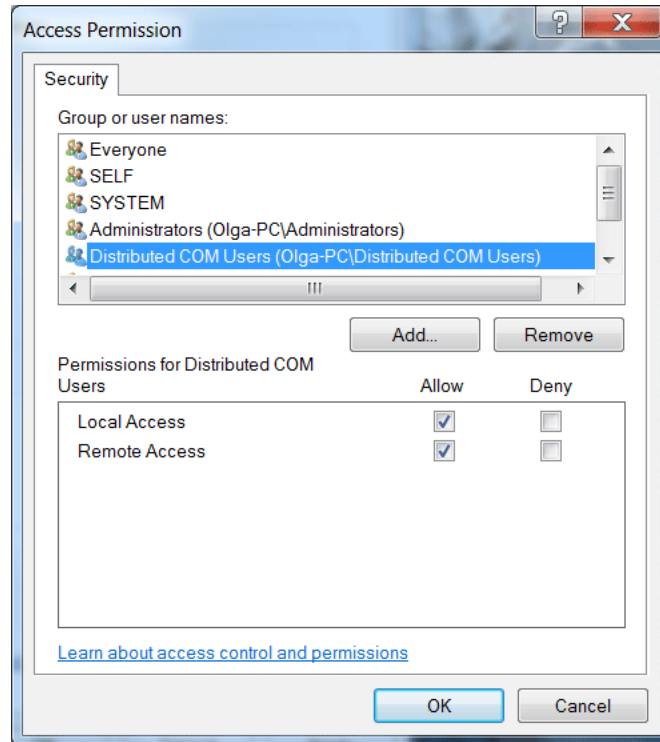


Figure 34-5. Settings for the *Security* tab.

4. Verify that the *Security* tab's *Access Permissions* are set as indicated in Figure 34-6.

Figure 34-6. The *Security* tab's *Access Permissions*.



5. In the *Endpoints* tab, verify that *DCOM Protocols* and *Endpoints* are set to defaults (the default is TCP/IP).
6. Set the *Identity* to *Interactive User* to allow displaying of the SA GUI elements on the server side.

Refer to the following section to configure the client side.

Configuring the OPC Client

Step 1: Create an OPC User Account & Install Components

1. Follow the instructions described in “[Step 1: Creating a DCOM User](#)” on page 890.
2. Follow the instructions described in “[Step 5: Install OPC Core Components](#)” on page 892.

Step 2: Configure System-Wide DCOM Settings

1. Ensure you’re logged in as **OPCUser**.
2. Run the OPC Server Utility (step 5, item 1).
3. Click the **View DCOM Configuration** button.
4. Right-click on **My Computer** and select **Properties** from the context menu.

5. In the *Default Properties* tab, ensure that the *Enable Distributed COM on this computer* option is enabled. The *Default Authentication Level* should be set to *Connect*, and the *Default Impersonation Level* should be set to *Identify* (Figure 34-3).
6. Select the *COM Security* tab. Click the *Edit Limits...* button in the *Access Permissions* group.
7. In the *Access Permission* dialog, add permission for **Everyone** and the **ANONYMOUS LOGON** users, then click **OK**.
8. Click the *Edit Limits...* button in the *Launch and Activation Permissions* group.
9. In the *Launch and Activation Permission* dialog, add permissions for **Everyone**, **Distributed COM Users**, and **ANONYMOUS LOGON**, then click **OK**.

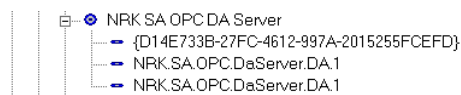
Step 3: Test the OPC Server Connection

1. Restart the computer.
2. Install a client onto the machine. For instance:
 - www.softing.com/opc (select OPC Classic options)
 - www.matrikonopc.com/products/opc-desktop-tools/opc-explorer.aspx
3. Start the client application.
4. Browse the remote OPC server to verify that data is being received.

Using SA's OPC Server

Once the server is up and running, the server's identifications (class ID, description, and program ID) are displayed in the client (Figure 34-7).

Figure 34-7. The SA OPC DA server identifications.



The following parameters are available to the OPC server:

- Active coordinate values for each live instrument as a 3-entry array of doubles (XYZ).
- Active coordinate values for each live instrument as separate values (X, Y, and Z).
- The current SA version.
- The full name of each instrument.

The tag name for each coordinate includes the collection and instrument ID for the associated instrument—for example, **A : 0 . x** (Figure 34-8).

Figure 34-8. Tags for single-coordinate items in the OPC server.

Item	Value	Quality	TimeStamp
A:0.x	-195.745613286698	GOOD	10:18:53.401
A:0.y	9.02854736148172E-14	GOOD	10:18:53.402
A:0.z	-172.072930905314	GOOD	10:18:53.403
NRK SA Version	SA 2014.04.07b (x64)	GOOD	10:04:37.883

Point coordinates are presented with A::0.xyz tag names in the array format (Figure 34-9):

Figure 34-9. A coordinate as an array.

Item	Value	Quality	TimeStamp
A:0.xyz	[0,2] (-195.745613286698, 9.02854736148172E-14, -172.072930905314)	GOOD	10:18:53.404
NRK SA Version	SA 2014.04.07b (x64)	GOOD	10:04:37.883

MP Support

Measurement Plan commands have been added to support interfacing with SA's OPC server—both setting and retrieving values from the server's address space. These commands include:

- Set OPC DA Tag Value Double
- Get OPC DA Tag Value Double
- Set OPC DA Tag Value Integer
- Get OPC DA Tag Value Integer

These commands are detailed in the MP Command Reference. The “set” commands will create a double or integer in the server's address space under the specified tag name. This value can be retrieved from the address space using the corresponding “get” command, or can be updated (overwritten) by specifying another “set” command.

This Page Intentionally Left Blank.

Machine Calibration

CHAPTER

35

SpatialAnalyzer has the unique ability to integrate portable metrology measurements for the calculation of true metrology-compensated robot and CNC machine calibrations.

SA Machine

SA Machine contains all of the features of SA Ultimate + Native CAD, but also allows interfacing with robots and CNC machines. This permits calibration and compensation of robots and large volume CNC machines to high degrees of accuracy. It also enables robotic scripting, teach pendant behavior, and linear or joint space robotic control.

Usually, robots are delivered by the manufacturer with default kinematic models that are accurate to on the order of about 1/2 - 1/4". This depends on the manufacturer, product, calibration level, and other factors. Many manufacturers offer calibration packages that might improve this accuracy to perhaps 0.1". CNC machines (especially large-volume machines) can also encounter the same sorts of issues (although they are usually more accurate). Regardless, this default accuracy is nowhere near what's necessary for precision applications. There are several reasons for this. One might be that the manufacturer's kinematic model is not quite right or doesn't incorporate things like the stiffness of the linkages or the weights of tools on the machine very well (if at all).

SA's Robot and Machine calibration functions have a few purposes. The first benefit is that they can be used to calculate a significantly more accurate kinematic model than is provided by the manufacturer by default. The following discussion will refer to robots, but it applies equally well to CNC machines.

To calculate a new kinematic model, a robot is commanded to go to a set of poses throughout the desired volume of use and whose posi-

tions in the robot controller's coordinates are known. For each pose, measurements are taken on the robot's end effector to determine the actual position of the tool compared to where it was "asked" to go. These measurements can be taken with a single measurement from a 6-DOF measurement device, or they can be taken by measuring 3 targets with a 3-DOF measurement device. (Any measurement method that results in 6-DOF information can be used, including photogrammetry).

Once a set of commanded positions and measured "actual" positions are determined, this data is run through a series of kinematic algorithms to determine an accurate kinematic model for the specific robot (machine) of interest. Using this kinematic model, any position that you'd like to send the robot to is first backed through this kinematic model to give the controller a new set of positions. In other words, based on the calculated kinematic model, SA Machine tells the controller where the robot ought to go (based on its kinematics) to end up where you want it to be. This results in a drastically improved result, (to accuracies of about 0.010", for example). Although this is a big improvement, it can be taken a step further.

Now that the robot has gone where we think it should go, the end-effector can be measured again, which tells us where the robot actually went compared to where we expected it to be. We can then iterate through the kinematic model to give a small correction to the robot, getting it even closer to the desired target. We have demonstrated accuracies to about 0.001"-0.002" using this method, which is quite accurate for a robot. Given the accuracy of the encoders on the robot's servos, this can't be beat without more accurate hardware encoders. This process is a great way to position tools and other items with very high accuracy/repeatability.

Robots in SA

Adding a Robot to SA

There are two ways to add a robot to SA.

- To load an existing SA Robot use **Analysis>SA Machine - Robots and Machines>Add Machine>Load .SAMachine File (kinematics and graphics)**.
- To create a new Robot use **Analysis>SA Machine - Robots and Machines>Add Machine> Add Serial Robot>Using Kinematic Definition File**.

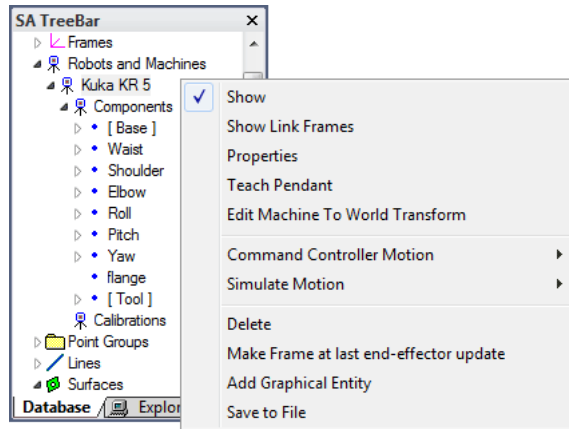
Once a robot is added to the tree it is displayed under the *Robots and Machines* category.

Robot Category

This category contains the specific Robot or Machine with the respec-

tive components and calibrations (Figure 35-1).

Figure 35-1. The properties of a Kuka KR 5 robot.

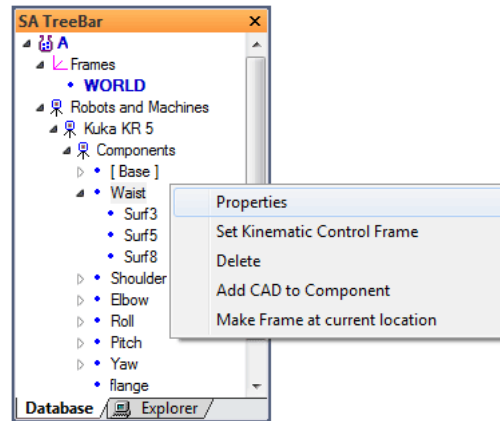


- **Show.** Show/Hide all components
- **Show Link Frames.** Show/Hide Link Frames
- **Properties.** Displays properties
- **Teach Pendant.** Displays teach pendant dialog in which user can control robot via several different methods.
- **Edit Machine to World Transform.** Displays machine base transform dialog allowing for the machine to be transformed with respect to the current frame.
- **Command Controller Motion.** Sends move commands to machine based on controller kinematics or SA kinematics.
- **Simulate Motion.** Simulates move commands to machine based on SA kinematics.
- **Delete.** Delete machine from SA TreeBar.
- **Make Frame at last end-effector update.** Creates a SA coordinate frame at the last end-effector position.
- **Add Graphical Entity.** Adds a new component.
- **Save to File.** Saves current machine to .SAMachine file.

Components

The Components sub-category contains each link of the robot with its respective graphical representations (Figure 35-2).

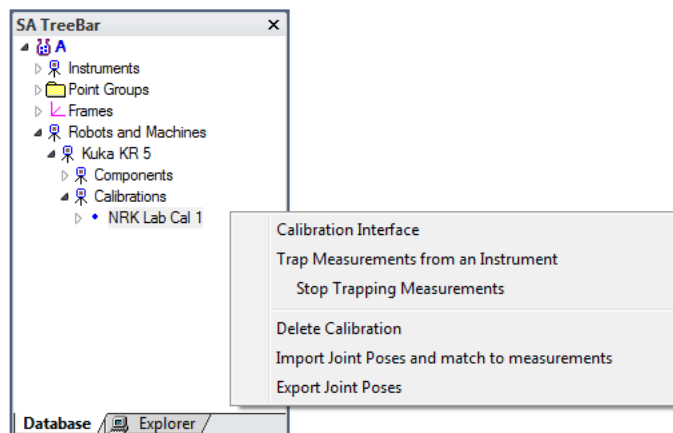
Figure 35-2. The properties of the robot's waist joint.



- **Properties.** Displays the Manipulator Segment Properties - Denavit-Hartenberg Dialog. Here the Segment Transform and Joint Variables can be modified.
- **Set Kinematic Control Frame.** Select the Link Frame for the component.
- **Delete.** Deletes the selected component.
- **Add CAD to Component.** Select surfaces to attach to component at current location.
- **Make Frame at current location.** Creates a SA coordinate frame at the current component location.

Calibrations

Figure 35-3. A calibration in the tree.



- **Calibration Interface.** Displays the Calibration Interface for the respective calibration set.
- **Trap Measurements from an Instrument.** Allows incoming measurements to be paired with current joint sets.
- **Stop Trapping Measurements.** Stops the trapping of incoming measurements.
- **Delete Calibration.** Deletes selected calibration.

- **Import Joint Poses and match to measurements.** Imports Joint Poses and pairs with selected measurements. Often used when a live interface to a Robot is unavailable.
- **Export Joint Poses.** Exports the Joint Poses from the selected calibration set.

Controlling a Robot from SA

There are several ways to control a Robot from SA. The following sections will cover the various methods.

SA Teach Pendant

The SA Teach Pendant (Figure 35-4) allows for Robot control from SA instead of the physical teach pendant. The SA Teach Pendant allows for individual actuator inputs, Cartesian moves and graphical input.

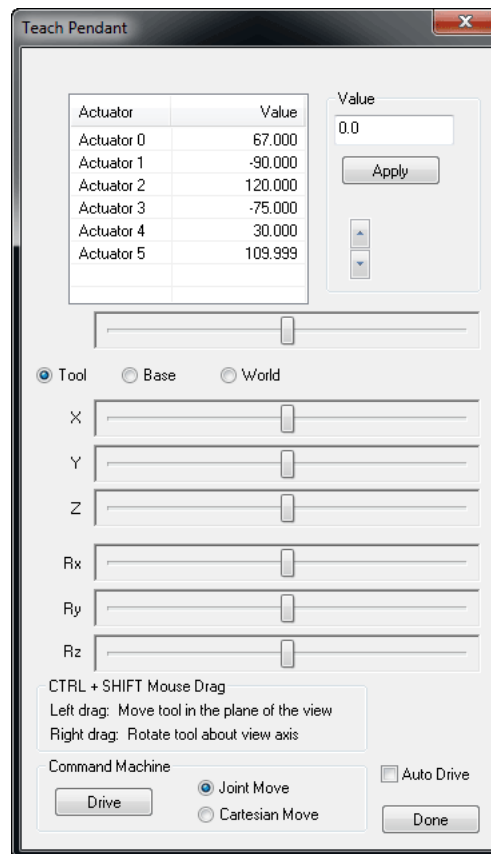


Figure 35-4. The Teach Pendant dialog.

- **Actuator Movement.** Select the actuator to move and either enter the value or use the slider to increase/decrease the value. Select the **Drive** button to apply the movement to the Robot Controller.
- **Cartesian Movement.** Here the TCP can be moved in Cartesian space about either the Tool, Base or World frame. Simply slide the respective sliders and then click the **Drive** button to apply the movement to the Robot Controller.

- **Graphical Movement.** Allows the user to graphically move the Robot and then apply the movement to Robot Controller by clicking the **Drive** button.
- **Command Machine.** The **Drive** button sends over the specified movement in either Joint space or Cartesian space.
- **Auto Drive.** Automatically sends movement to the Robot Controller when values change.

Move to Frame

The Move to Frame commands (Figure 35-5) allow robot movements to be calculated from the selection of Coordinate Frames in SA. When used, the user will be prompted to select a frame. From here, the robot will be moved to the selected position either in controller kinematics or SA Kinematics.

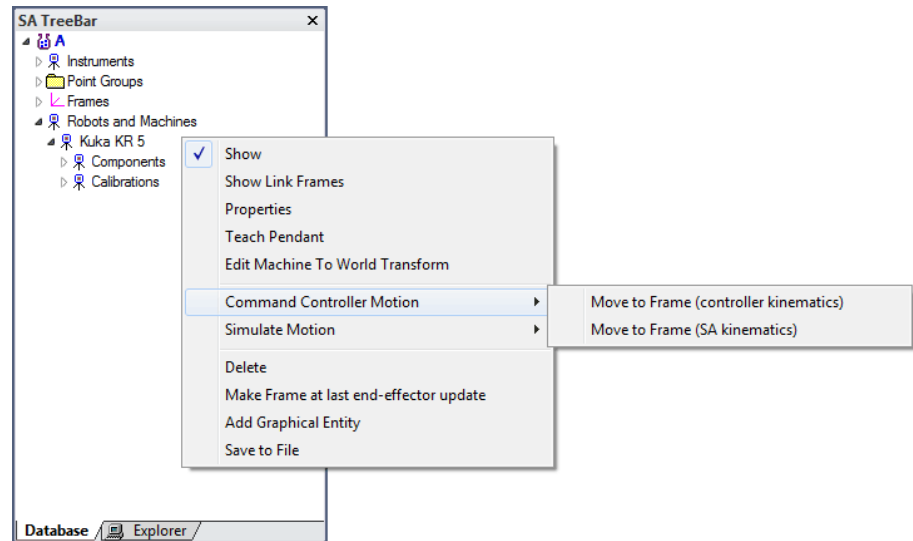


Figure 35-5. The Move to Frame commands.

Measurement Plan

The SA Measurement Plan scripting language also includes several commands for controlling Robots. As the above section describes Move to Frame, similar commands are available in MP. MP is a great tool for automating data acquisition and robot movement.

- Move Robot/Machine to Frame
- Move Robot/Machine to Named Destination
- Set Robot/Machine Parameter
- Compute Robot/Machine Adjusted Goal Frame

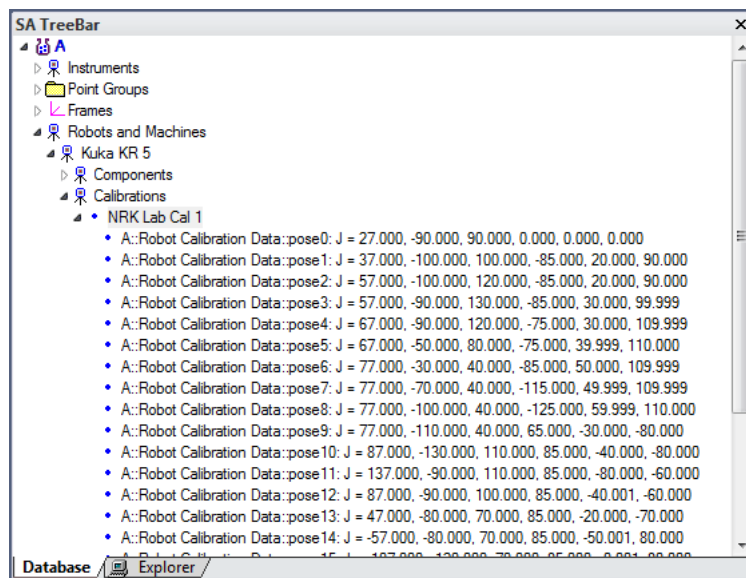
Collecting Calibration Measurements

Collecting data for a Robot calibration only takes a few steps. Two scenarios exist: Collecting joint sets from a live robot or importing joint sets.

To Collect Measurements from a Live Robot

1. Ensure that SA and the Robot are connected via the SA Robot Driver.
2. Add a new calibration by right-clicking Calibrations and select **Add Calibration**.
3. Right-click the newly added calibration and select **Trap Measurements from an Instrument**.
4. Now the calibration set is ready to receive measurements and joint sets. As a measurement is received a corresponding joint set is recorded. Simply move the Robot to various poses and record measurements (Figure 35-6).

Figure 35-6. Pose data for a calibration set.



To Import Measurements and Robot Poses

1. Ensure the proper Robot poses are formatted in a .txt or .csv file:

<name>, <j1>, <j2>, etc.
2. Collect measurements for each pose and ensure the point names match the respective pose names.
3. Add a new calibration by right-clicking Calibrations and selecting **Add Calibration**.
4. Right-click the newly-added calibration and select **Import Joint Poses** and match to measurements. You will be prompted to select the points and pose file.

Calibration Interface

The calibration interface is used to compute the new kinematic model for the robot based on the recorded poses and measurements (Fig-

ure 35-7).

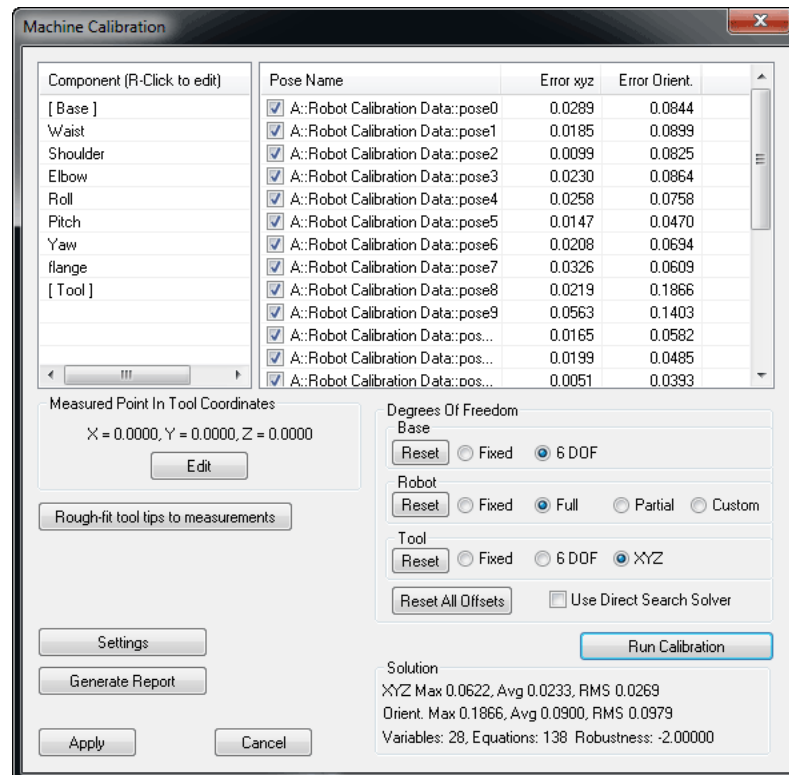


Figure 35-7. The Machine Calibration dialog.

- **Components.** Displays the current components for the current Machine. Right-click a component to access properties.
- **Pose Name and Errors.** Displays each pose and corresponding positional and rotational errors. Uncheck a pose to remove from solution.
- **Measured Point in Tool Coordinates.** Used to define the measured target in the TCP frame when using single point measurements. Not needed when using a 6 degree of freedom target.
- **Rough-fit tool tips to measurements.** Performs a best-fit of the tool tip locations to the measurements to ensure a good initial condition for the optimization.
- **Settings.** Displays the tool orientation weighting factor.
- **Generate Report.** Generates a final report to Excel with solution results and kinematic properties.
- **Degrees of Freedom.** Degrees of freedom for the base, linkages and tool of Robot.
- **Run Calibration.** Executes the optimization.
- **Solution.** Displays current position and orientation errors and RMS values.

Best Practice for Running a Calibration

1. With measurements and poses captured, open the calibration interface by right-clicking the calibration set of interest and choosing **Calibration Interface**.
2. Ensure a good initial starting condition by pressing **Rough-fit tool tips to measurements**. The overall errors will reduce greatly as the measurements and tool tip positions are best-fit to one another.
3. The initial Degrees of Freedom (DOF) are to allow the base and tool to move freely while maintaining the nominal kinematics (Figure 35-8). The tool DOF should only be set to 6 DOF when 6 DOF measurements are present. Press **Run Calibration** to start the optimization.

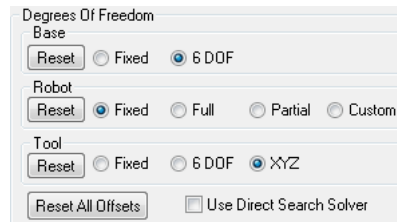


Figure 35-8. The Degrees of Freedom box.

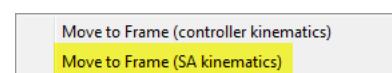
4. Now, if desired, allow the Robot kinematics to vary by choosing **Partial** or **Full** (Figure 35-9). Press **Run Calibration** to start the optimization.



Figure 35-9. Choosing a partial robot DOF solution.

5. At any time, make sure to validate any potential outliers and remove by un-checking the pose and re-running the optimization by pressing **Run Calibration**.
6. Once satisfied with optimization, generate a report for external sources and/or apply the new calibration to SA by pressing **Apply**.
7. This calibration will now be used when moving the robot with SA Kinematics.

Figure 35-10. The Move to Frame option.



Kuka Robot Deployment

Below are the necessary components to operate SA and SA Robot Driver with Kuka Robots.

XML Channel configuration and Data format files (C:\KRC\Roboter\INIT)

- RAC.xml
- RAC+.xml
- XmlApiConfig.xml (set tcp/ip address and port of remote access computer – secondary fixed ip on remote computer)

KRL Client Communication (C:\KRC\Roboter\KRC\R1\Program)

- SAKRL.dat
- SAKRL.src

KRL Client motion programs (called by SAKRL.SRC C:\KRC\Roboter\KRC\R1\Program)

- MoveToFrame.SRC and .DAT
- MoveToJointSet.SRC and .DAT
- HitEStop.SRC

VxWorks Communication Configuration (C:\Windows)

- vxWin.ini (robot controller VxWorks tcp/ip address modified under e={.....})

Server application and data format files on remote computer

- SARobotDriver.exe
- EKX Server.exe
- RACData.xml
- Global Data (c:\KRC\Roboter\KRC\R1\System)
- \$config.dat

SA Robot Driver – Kuka Controller Requirements

- KR C2 Controller
- Kuka System Software
- KSS v.5.5
- Kuka Ethernet KRL XML
- v. 1.2
- 3COM Ethernet Card Option

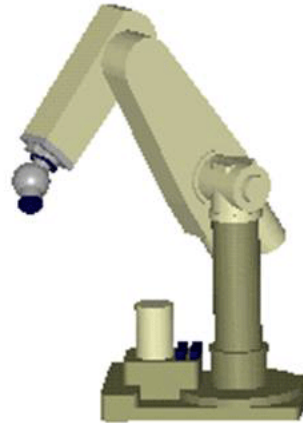
Denavit-Hartenberg Notation¹

The first step in calibrating a manipulator is to understand its nominal kinematic parameters, and the controller's ability to accept new, calibrated values for these parameters. This section will describe how to

¹ For a more detailed description of Denavit-Hartenberg notation see Craig, J. J., 1989, Introduction to Robotics: Mechanisms and Control, Addison Wesley, New York.

represent a robot's kinematics. Most manipulators are serial devices, meaning they are composed of a series of links and joints connected in an open chain. Denavit-Hartenberg notation is a standard means for specifying the kinematics of a manipulator. See the example table below, then a brief description of the meaning of the parameters.

Figure 35-11. A robot and its Denavit-Hartenberg (D&H) table.



Link	α	a	d	θ	Type
1	0.0	0.0	0.0	0	Revolute
2	-90.0	0.0	0.0	0	Revolute
3	0	8.0	-4.8	0	Revolute
4	-90.0	0.0	8.0	0	Revolute
5	-90.0	0.0	0.0	0	Revolute
6	-90.0	0.0	0.0	0	Revolute

Links are rigid members; joints allow relative motion between the links. The chain starts at the base, or ground link, and ends at the working tool. Except for the tool link and the base, each individual link in the chain begins and ends with a joint. The tool may be a gripper, paint sprayer, welding tip, electro-magnet or any of a wide variety of devices. Common manipulators use only revolute (turning) or a prismatic (sliding) joints to connect links. More complicated forms of joints can always be modeled as a combination of these simple elements. The function of a link is merely to fix the relationship between the joints at either end of the link. As long as the two joints are held at the same relative position and orientation with respect to one another, the link serves only as a rigid connector. The actual physical shape of the link can be changed without altering the kinematic relationship between the two interconnected joints.

A kinematic description of the manipulator is generated by SpatialAnalyzer using the standard Denavit-Hartenberg notation. This remarkably simple representation uses only two values to describe each link, and two additional values to describe the joint connection between adjacent links. Links are described in terms of their length (a) and the twist they maintain between adjacent joints (α), as explained below.

- **Twist Angle (α).** The angle between the first and second joint axes on a link. This angle is measured in a right-hand sense when viewed directly along the common normal.
- **Link Length (a).** The distance between the two joint axes of a link. This distance must be measured along a line that is normal (i.e. perpendicular) to both joint axes. This line is referred to as the common normal between the joint axes. If the joint axes intersect (a common arrangement in industrial manipulators), the link will have zero length. The joint connection between adjacent links is described in terms of the offset (d) and the joint angle (θ), as explained below.
- **Offset (d).** The distance between the common normal of one link and the common normal of the next, measured along the axis of the joint connecting the two links.
- **Joint Angle (θ).** The angle between the common normal of one link and the common normal of the next, measured about the axis of the joint connecting the two links.

The four parameters α , a , d , and θ , when taken for each link and joint in the chain, provide a complete kinematic description of the manipulator. When taken together with a description of the physical shape of each link and limits on joint motion, an accurate simulation of robot motion and operating workspace can be developed.

In addition, there is usually a *Joint Offset* value computed during the calibration. This value represents the difference between the “kinematic zero” of a joint, and the mechanical zero based on how the encoder was set in the joint mechanism.

For a more detailed description of Denavit-Hartenberg notation see:
Craig, J. J., 1989, Introduction to Robotics: Mechanisms and Control, Addison Wesley, New York

Keyboard Shortcuts

APPENDIX

A

The following page is a list of default keyboard shortcuts that ship with SpatialAnalyzer. To become a more efficient user of SA, it is recommended that you try to become familiar with as many of these shortcuts as possible.



KEYBOARD SHORTCUTS

General

Action	Shortcut
Repeat Command	Ctrl + Tab
Command History	Ctrl + Shift + Tab
Graphical Selection Mode	F3 (During Selection)
List Selection	F2
Undo to Restore Point	Ctrl + Z
Create Restore Point	Alt + Space
Pick Working Frame	Alt + W
User Options	Ctrl + U

View

Action	Shortcut
Show Point Labels	Alt + L
Hover	Ctrl + H
Cloud Display Control	Ctrl + T
Refresh	F5

File

Action	Shortcut
Open File	Ctrl + O
Save File	Ctrl + S

Edit

Action	Shortcut
Object Properties	Alt + Q
Copy then Transform	Ctrl + C
Delete Objects	Delete
Delete Points / Cloud Points	Alt + D / Ctrl + Alt + D
Add User Note	Ctrl + N

Geometry Fit

Action	Shortcut
Fit Geometry	Ctrl + G
Fit Line	Ctrl + Alt + L
Fit Circle	Ctrl + Alt + C
Fit Plane	Ctrl + Alt + P
Fit Cylinder	Ctrl + Alt + Y
Fit Sphere	Ctrl + Alt + S

Graphical View

Action	Shortcut
Rotate (Orbit)	(Drag)
Zoom In/Out	Pg Up/Pg Dn
	Scroll Up/Down
	Ctrl + + Drag Up/Down
Pan	↑ ↓ ← →
	Shift + (Drag)
	(Drag)
Recenter View	(Click to Center)
Zoom to Selection	+ Drag Rectangle
Autoscale	Alt+A
Clipping Planes	Alt+X
Previous View	Ctrl + Alt + ←
Next View	Ctrl + Alt + →

Queries

Action	Shortcut
Point to Point	Ctrl + D
Group to Group	Ctrl + W

Instruments

Action	Shortcut
Add Instrument	Alt + I
Point at a Point	(Double-click on point)

Geometry Construction

Action	Shortcut
Construct Points	Ctrl + P
Layout Random Points	Ctrl + Alt + Z
Patch Normal Shift	Ctrl + M
Construct Line 2 Points	Ctrl + L

Analysis

Action	Shortcut
Best Fit Points to Points	Ctrl + B
USMN	Alt + U

Inspection

Action	Shortcut
Inspection Bar Previous	Ctrl + F7
Insp. Bar Stop	Ctrl + F8
Insp. Bar Next	F8
Instrument Toolbar Measure	F6
Inst. Toolbar Iterate Profile	F9
Inst. Toolbar Iterate Tooling	F10
Inst. Toolbar Lock Beam	F12

Although not at all essential to get benefit out of the software, this section provides background knowledge on some of the math behind SpatialAnalyzer.

Linear Least Squares Adjustment

Least squares is a procedure for adjusting observations containing random errors. It was developed in the latter part of the eighteenth century by Karl Gauss. It was not used much until computers became commonplace because of the lengthy calculations required. SpatialAnalyzer uses it in many routines (transformations, best-fit lines-planes-circles etc), to provide the best and most consistent answer for fitting problems.

For a group of equally weighted observations (or measurements in our case), the basic condition that is enforced in least squares adjustments is that the sum of the squares of the *residuals* is minimized. Residual is another term for the difference between a measured value and its most probable (or “correct”) value. For 3D coordinate metrology, one example of a residual would be the difference between the true X-coordinate for a point and the actual X-coordinate measurement.

The method of least squares is a criterion for fitting a specified model to observed data. In this case, the specified model is the reference set of points (or expected values), and the observed data are typically your measurements of the reference set of points. Let’s use a common use of the least squares adjustment as an example. The least squares line routine is a commonly used method for computing the best-fit line from a series of measurements of a linear function. The distance your car travels when you’re driving at a particular speed is an ex-

ample of a linear function. In this example, you are trying to drive your car at a constant speed and measure how far you have gone and the time it took you to get there. While driving, it is likely that a little error is going to creep into the process—you're not going to be able to . That error is generally referred to as random error. Least squares adjustment techniques are really good at dealing with this type of error and are able to produce a reasonable and consistent answer in spite of it. An example is shown in the figures. The time is plotted on the X-axis and the distance on the Y-axis.

The points are the paired measurements made of time versus the distance traveled, which is linear in this case because you were trying to maintain a constant speed. To find what is considered the "best line" through the data the least squares line routine is used. In this case, the residual that gets minimized is the difference (squared) from a measured point to the line that best describes the distance you traveled versus the time it took. The least squares line routine will find a line, such that the sum of the distances (squared) from the data points to the line is the minimized. The figure shows a line that best fits those points; it was solved for with the best-fit line routine in SpatialAnalyzer.

The slope of the best-fit line is actually the average speed that you traveled on your trip. This line is the most consistent way of determining where you probably were at any point in time during your trip.

The least square line adjustment was able to use this set of measurements even through each point had some error, and it produced a high-quality and reliable answer. This ability of dealing with measurements that have at least some component of random error and still being able to produce consistent and representative answers is why the technique is used in SpatialAnalyzer.

Basics of Spatial Transformations

Homogeneous transformations—transforms for short—can be used to represent the position and orientation of one coordinate frame (system) with respect to another coordinate frame. A general transformation matrix can be conveniently represented as a four by four matrix; however, only twelve of the sixteen numbers are typically used in this application.

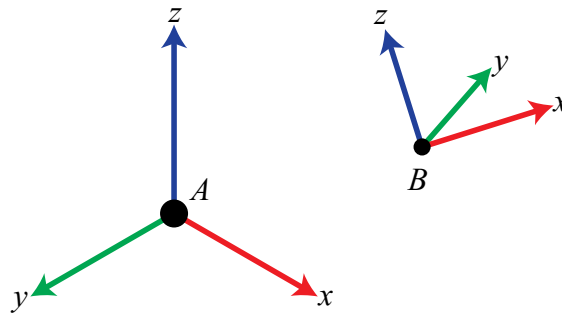
The four-by-four matrix is typically used because it is symmetric. Numerical routines can generally be made more efficient when the numerical pieces are symmetric.

This representation is unique—i.e., there is only one transformation matrix that unambiguously describes a particular relative position

and orientation. The problem with representing all transforms as a matrix is that this representation is over-constrained. We know, for example, that a rigid object in space has only six degrees-of-freedom—three positional and three rotational. You simply don't need all twelve numbers in the 4x4 matrix to represent the relative position and orientation. Let's look at other more compact representations.

First of all, we need to define which coordinate frame is described relative to the other. Let's assume we have frames **A** and **B**, and that we wish to describe frame **B** relative to **A** (Figure B-1).

Figure B-1. Two coordinate frames in space.



Position is easy—we simply describe the x , y , and z components of the origin of **B**. Note these terms are the distances along the x , y , and z axes of frame **A**. Also note the position of **A** relative to **B** would not, in general, be the negative of these values. (This would only be the case if the corresponding axes for the two frames were all parallel). Rotation is not as easy as position. To describe rotations, certain standard conventions have been adopted. These are by no means the only possible methods. The following sections will discuss different viewpoints that are commonly used to discuss transforms, then we will discuss simple unit rotations, and go on to develop compound rotation methods.

The old viewpoint debate: Are we describing or transforming?

This section only becomes necessary because of the language that individuals use to discuss transforms. This is one of the most common sources of errors in dealing with spatial transformations. Transforms can be used in two different ways:

- **As a descriptor.** This is the way we will always talk about transforms in SpatialAnalyzer. This is the description of one frame relative to another. For example, frame **B** relative to frame **A**.
- **As a transform operator.** In this case we ask the question: what operations we do to frame **A** to rotate and translate it to the position and orientation of **B**? In some cases, it's natural to think about transforms in this way. For example, if you want to describe the position of a box as it moves, it's natural to think

about the rotations and translations that it is subjected to over time.

The final result is a little counter-intuitive—but essentially, it doesn't matter which view point you use—they both produce the exact same results. You just need to be aware of the fact that despite the language used, the math is the same. Many people incorrectly interpret one as the inverse of the other.

Unit Rotation Matrices

Before we can discuss compound rotations, we need to develop the building blocks of unit rotations. These are simply rotations that occur about any given principal coordinate axis (X, Y, or Z). They are as follows:

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix} \quad R_y(\theta) = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix} \quad R_z(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Or, for the particular example of an X rotation of 30°, a Y rotation of 60°, and a Z rotation of 90°:

$$R_x(30^\circ) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.866 & -0.500 \\ 0 & 0.500 & 0.866 \end{bmatrix} \quad R_y(60^\circ) = \begin{bmatrix} 0.500 & 0 & 0.866 \\ 0 & 1 & 0 \\ -0.866 & 0 & 0.500 \end{bmatrix} \quad R_z(90^\circ) = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Compound Rotations

Using the building blocks of the above unit rotations, we can now proceed to develop more complex compound rotations. As we stated earlier, a rotation matrix is unique. However, the descriptive methods we often use to describe a rotation are not. The same physical orientation can be described in an infinite number of ways. We'll explore the most common below.

Fixed Angle XYZ

In this case, we will start with frame **A** and **B** coincident. We will then rotate frame **B** about the X-axis of **A** (X_A), then about the Y-axis of **A** (Y_A), then about the Z-axis of **A** (Z_A). We can express this as:

$$R_{xyz}(30^\circ, 60^\circ, 90^\circ) = R_z(90^\circ)R_y(60^\circ)R_x(30^\circ) = \begin{bmatrix} 0.000 & -0.866 & 0.500 \\ 0.500 & 0.433 & 0.750 \\ -0.866 & 0.250 & 0.433 \end{bmatrix}$$

Fixed Angle ZYX

In this case, we again start with frame **A** and **B** coincident. We then rotate frame **B** about the Z-axis of **A** (Z_A), then about the Y-axis of **A** (Y_A), then about the X-axis of **A** (X_A). We can express this as:

$$R_{zyx}(90^\circ, 60^\circ, 30^\circ) = R_x(30^\circ)R_y(60^\circ)R_z(90^\circ) = \begin{bmatrix} 0.000 & -0.500 & 0.866 \\ 0.866 & -0.433 & -0.250 \\ 0.500 & 0.750 & 0.433 \end{bmatrix}$$

Euler Angle XYZ

In this case, we will start with frame **A** and **B** coincident. We will then rotate frame **B** about the X-axis of **B** (X_B), then about the Y-axis of **B** (Y_B), then about the Z-axis of **B** (Z_B). Note that the axes of frame **B** rotate during the operations. We can express this as:

$$R_{x'y'z'}(30^\circ, 60^\circ, 90^\circ) = R_x(30^\circ)R_y(60^\circ)R_z(90^\circ) = \begin{bmatrix} 0.000 & -0.500 & 0.866 \\ 0.866 & -0.433 & -0.250 \\ 0.500 & 0.750 & 0.433 \end{bmatrix}$$

Euler Angle ZYX

In this case, we will start with frame **A** and **B** coincident. We will then rotate frame **B** about the Z-axis of **B** (Z_B), then about the Y-axis of **B** (Y_B), then about the X-axis of **B** (X_B). Note that the axes of frame **B** rotate during the operations. We can express this as:

$$R_{z'y'x'}(90^\circ, 60^\circ, 30^\circ) = R_z(90^\circ)R_y(60^\circ)R_x(30^\circ) = \begin{bmatrix} 0.000 & -0.866 & 0.500 \\ 0.500 & 0.433 & 0.750 \\ -0.866 & 0.250 & 0.433 \end{bmatrix}$$

Euler Angle ZYZ

In this case, we will start with frame **A** and **B** coincident. We will then rotate frame **B** about the Z-axis of **B** (Z_B), then about the Y-axis of **B** (Y_B), then again about the Z-axis of **B** (Z_B). Note that the axes of frame **B** rotate during the operations. We can express this as:

$$R_{z'y'z'}(90^\circ, 60^\circ, 90^\circ) = R_z(90^\circ)R_y(60^\circ)R_z(90^\circ) = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -0.500 & 0.866 \\ 0 & 0.866 & 0.500 \end{bmatrix}$$

Equivalent Angle Axis

This representation is a little different from the others given in that it does not result from the combination of unit rotations. This convention stems from the fact that for any given rotation there exists a unique axis of rotation and magnitude of rotation. If we rotate about a unit vector K and amount θ we find that the following matrix results:

$$R(K, \theta) = \begin{bmatrix} K_x K_x v_\theta + \cos \theta & K_x K_y v_\theta - K_z \sin \theta & K_x K_z v_\theta + K_y \sin \theta \\ K_x K_y v_\theta + K_z \sin \theta & K_y K_y v_\theta + \cos \theta & K_y K_z v_\theta - K_x \sin \theta \\ K_x K_z v_\theta - K_y \sin \theta & K_z K_y v_\theta + K_x \sin \theta & K_z K_z v_\theta + \cos \theta \end{bmatrix}$$

where $v_\theta = (1 - \cos \theta)$ and K_x, K_y, K_z are the components of unit vector K . Through an inverse analysis we can extract the (K, θ) for any rotation matrix.

Quaternions (also known as Euler Parameters)

A Quaternion is simply another way of expressing the information that is inherent in the equivalent angle-axis representation. Namely,

$$\varepsilon_1 = K_x \sin\left(\frac{\theta}{2}\right) \quad \varepsilon_2 = K_y \sin\left(\frac{\theta}{2}\right) \quad \varepsilon_3 = K_z \sin\left(\frac{\theta}{2}\right) \quad \varepsilon_4 = \cos\left(\frac{\theta}{2}\right)$$

Notes on Decimal Precision and Rounding

Decimal Precision

The precision with which a 64bit double value can be stored is roughly the machine epsilon * value of double, approximately 2.220446e-16.

Epsilon can be approximated by finding the smallest value for which $((1.0 + \text{epsilon}) - 1.0) <> 0$.

This means that a double can never be better than 15 decimal places (scientific notation).

As a practical matter, subsequent to a reasonably limited number of mathematic operations, a double is generally no better than 12 or 13 decimal places (scientific notation).

If a set of points are imported into SA into the world frame, then a point recorded as (-247.5, 0, -247.5) should resolve X and Z to the same displayed value when displayed with zero significant digits – in this case, the values for X and Z should be stored as identical double representations.

If, however, this point is imported into a working frame different from world, it will be subjected to a coordinate transform so that it will be stored in world coordinates – points are ALWAYS stored relative to the *WORLD* frame.

$$Pt_stored = T_world_wrt_working * Pt_imported$$

Now when the point is displayed in the working frame, X and Z values may no longer be precisely the same – they will have been twice transformed.

$$Pt_wrt_working = T_working_wrt_world * Pt_stored$$

$$Pt_wrt_working = T_working_wrt_world * T_world_wrt_working * P_imported$$

Every time a point goes through a transform, it will be subjected to multiple multiplications and additions for each coordinate component.

So, if you start out with a point recorded as (-247.5, 0, -247.5) such that X and Z are precisely equal and end up with a case where X and Z are not precisely equal, then this means this point has been subjected to coordinate transformations.

Rounding

Double values within SA are recorded and stored using full available decimal precision. The units control within the **Users Options>Display** tab (**Decimal Digits for Display**) offers an option to control the number of displayed digits, but this controls only the displayed representation of the values for reporting purposes.

Rounding for display is controlled by the standard C++ unit conversions. This means that a value rounded to 3, 1, 0 decimal places would appear as:

```
[-247.4999999999000124]
```

```
-247.500
```

```
-247.5
```

```
-247
```

Alternatively:

```
[-247.500000000099876]
```

```
-247.500
```

```
-247.5
```

```
-248
```

These results may appear inconsistent when comparing rounded values to each other but should be more understandable with respect to the full precision value.

This Page Intentionally Left Blank.