



PI System Management Tools

2023

© 2015-2025 AVEVA Group Limited and its subsidiaries. All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, photocopying, recording, or otherwise, without the prior written permission of AVEVA Group Limited. No liability is assumed with respect to the use of the information contained herein.

Although precaution has been taken in the preparation of this documentation, AVEVA assumes no responsibility for errors or omissions. The information in this documentation is subject to change without notice and does not represent a commitment on the part of AVEVA. The software described in this documentation is furnished under a license agreement. This software may be used or copied only in accordance with the terms of such license agreement. AVEVA, the AVEVA logo and logotype, OSIsoft, the OSIsoft logo and logotype, ArchestrA, Avantis, Citect, DYNSIM, eDNA, EYESIM, InBatch, InduSoft, InStep, IntelTrac, InTouch, Managed PI, OASyS, OSIsoft Advanced Services, OSIsoft Cloud Services, OSIsoft Connected Services, OSIsoft EDS, PIPEPHASE, PI ACE, PI Advanced Computing Engine, PI AF SDK, PI API, PI Asset Framework, PI Audit Viewer, PI Builder, PI Cloud Connect, PI Connectors, PI Data Archive, PI DataLink, PI DataLink Server, PI Developers Club, PI Integrator for Business Analytics, PI Interfaces, PI JDBC Driver, PI Manual Logger, PI Notifications, PI ODBC Driver, PI OLEDB Enterprise, PI OLEDB Provider, PI OPC DA Server, PI OPC HDA Server, PI ProcessBook, PI SDK, PI Server, PI Square, PI System, PI System Access, PI Vision, PI Visualization Suite, PI Web API, PI WebParts, PI Web Services, PRISM, PRO/II, PROVISION, ROMeo, RLINK, RtReports, SIM4ME, SimCentral, SimSci, Skelta, SmartGlance, Spiral Software, WindowMaker, WindowViewer, and Wonderware are trademarks of AVEVA and/or its subsidiaries. All other brands may be trademarks of their respective owners.

U.S. GOVERNMENT RIGHTS

Use, duplication or disclosure by the U.S. Government is subject to restrictions set forth in the license agreement with AVEVA Group Limited or its subsidiaries and as provided in DFARS 227.7202, DFARS 252.227-7013, FAR 12-212, FAR 52.227-19, or their successors, as applicable.

AVEVA Legal Resources: <https://www.aveva.com/en/legal/>

AVEVA Third Party Software Notices and Licenses: <https://www.aveva.com/en/legal/third-party-software-license/>

Contents

Welcome to PI System Management Tools	24
Start PI SMT	25
PI SMT behaves differently on different versions of Data Archive	26
Select a Data Archive server	27
Add a Data Archive server.....	28
View or save the session record	29
View Data Archive connection credentials	30
Set Windows or PI Time format	31
AF Link for migration to AF	32
PI Batch to event frame migration.....	32
Prepare batch data sources and consumers for migration	32
Batch data source	34
PI Event File (EVT) interface	34
PI BaGen interface	35
Batch interfaces 1.x 2.x 3.x	36
Your proprietary interfaces	36
PI Batch Subsystem interfaces	36
Batch data consumers	36
PI BatchView PI ProcessBook add-in	37
PI BatchView Microsoft Excel add-in (used with PI DataLink)	37
RtReports	37
PI WebParts Batchview Control	37
PI OLEDB	37
In-house PI SDK Batch applications	37
Your code in PI ProcessBook displays	37
Get ready to run the migration process	37
Analyze batch data before migration.....	39
Analysis error report	40

Test migration and cut over from current sources and consumers	46
Migrate batch data to event frames	46
Migration error report	48
Verify your event frame data	50
Event frame templates for batch data	50
How a batch object converts to an event frame	52
MDB to AF Synchronization	56
About MDB to AF Migration and Synchronization	56
About the Data Archive element	56
How PI MDB objects are represented in PI AF	58
What content is not synchronized	59
PI MDB edits that are not allowed	59
Tools for editing PI AF and PI MDB objects	60
Make PI AF content accessible in PI MDB	60
Enable PI MDB	61
Finding and fixing synchronization problems	61
Health status	61
Checking security synchronization	65
View the security synchronization report	65
Guidelines for good security synchronization	66
Possible security conflicts	66
Identity does not have a mapping	66
PI AF uses a Windows account that has no mapping	67
Element uses deny access	67
Element security does not match module security	67
Missing mapped principal	67
Troubleshooting synchronization	68
Cannot edit PI MDB	68
Element does not appear in PI MDB	68
Cannot edit access permissions on a module	69
Disabled a mapping identity but PI AF still allows access	69
Module and element hierarchies look different	69
..... 0	0
Force a remigration	70
Reset PI MDB	70
Identify the Data Archive element	71
Change the PI AF server settings	71
Change replacement characters or suffix	71
Replacement characters	71
Default replacement characters	72
Replacement suffix	72
Enable tracing	72
About tracing	73
PI MDB and PI AF object conversion details	73
Module to element conversion	73
How batch objects are migrated	74
Alias to attribute conversion	74
Alias to attribute type conversion	75
Property to attribute conversion	75

Property to attribute value type conversion	76
Element to module conversion	77
Attribute to property and alias conversion	78
Data type conversion for attributes to properties	78
Access permissions required by AF Link	80
PI AF Link access permissions	80
About the Windows account for PI AF Link	80
Configure PI AF Link access for cross-domain deployments	81
Changing the AF Link account	81
The AF Link to PI Windows group	81
Access permissions required to run the preparation wizard	82
Access permissions required by AF Link	82
Error messages	82
 Alarm Groups in PI SMT	93
View alarm groups and points	93
Change alarm groups and points	93
Alarm groups and points status	94
Specify point sources	94
Add alarm groups to a Data Archive server	94
Organize alarm groups and points	94
Alarm groups reference	95
 Archive Editor	96
Search for archived events	96
Use Tag Search	96
Define a time, event count or event range	97
Define a time range	98
Define an event range	98
Protect existing archive values	98
Define a boundary type	99
Filter search results	99
View and edit archived event values	100
Edit an archived event value	101
Refresh the list of archived events	101
Add an event to the archive	102
Delete an archive event	102
Annotations in PI SMT	102
Add or edit annotations	103
Import a file to an annotation	104
Export a file from an annotation	104
 Archives tool	105
Archives tool	105
Archives toolbar	106
Archive properties	106

Show or hide archive gaps	107
Archive management tasks in SMT	108
Archive creation with PI SMT	108
Archive names	108
Create an archive	108
Create multiple archives for backfilling	109
Register archives	109
Unregister archives	109
Display the header of an unregistered archive	110
Set an archive to writable or read only	110
Set availability of archives for shifts	110
Force an archive shift	110
Schedule archive shifts	111
Configure scheduled shifts	111
Force shifts	112
PowerShell support for scheduled shift	112
Manage archive gaps	112
Reprocess an archive	113
Enable auto-dynamic archive conversion	113
Export a list of archives to a file	114
Create a .bat registration file	114
 AutoPointSync List	 116
AutoPointSync List window	116
Synchronization status	117
Export the list of interfaces that use PI Auto Point Sync	117
 Backups	 118
View backup history of a Data Archive server	118
Data Archive backup types	119
Change the number of backups shown in the Backup History table	119
View backup information summary	119
View backup details	120
Backup details summary	120
Backed up file list	121
Troubleshooting Data Archive backups	122
View Data Archive backup logs and messages	122
Common issues with backups	122
Data Archive backup failure due to offline subsystem	123
Perform an on-demand Data Archive backup	124
Export Data Archive backup history	124
 Batch Custom Names	 125
Custom name sets	125
View custom name sets	125
Custom name set details	125

Properties	126
Batch terms of a custom name set	126
Description	127
Custom name examples	128
Create a new custom name set	130
Edit batch terms of a custom name set	130
Rename an existing custom name set	130
Delete an existing custom name set	130
Export existing custom name sets	130
Import existing custom name sets	131
Revert a custom name set	131
Show only customized batch terms	131
Sort the batch terms	131
Find information about a batch term or custom name	131
Custom names toolbar	132
Batch Database	134
Module Database tree	134
Creation of PI Batch Database items	134
Create a new PIBatch	135
Create a new PIUnitBatch	135
Searches for PI Batch Database items	135
Search for PIBatches	136
Search for PIUnitBatches	136
Results from PI Batch Subsystem batches	136
Copy batch items	137
Paste PIBatches	137
Paste PIUnitBatches	137
Paste PISubBatches	138
Search results list	138
View and edit batch items from search results	139
PIBatch details	140
PIBatch properties	140
PIBatch edits	142
PIProperty updates	142
PIUnitBatch details	143
PIUnitBatch properties	144
PIUnitBatch edits	145
Link or unlink a PIUnitBatch to a PIBatch	145
PISubBatch details	146
PISubBatch properties	147
PISubBatch edits	148
Create a new PISubBatch	148
Delete batch items	149
Remove search results	149
Batch Generator	150

Migration Note	150
View PIUnits and PI Module Databases	150
Configure PIUnits	150
Configure the PIBaGen interface	151
Configuration Module Name	151
Interface Debug Messages	151
Event Analysis Delay Time	151
PIBaGen Status Tag Name	151
Retry Timeout	152
Maximum Events in Event Pipe	152
Maximum Stop Time	152
Configure PIUnitBatches	152
Active Point (Required)	153
ActivePoint Behavior	153
Unit Batch ID Point (Optional)	153
Product Name Point (Optional)	153
Procedure Name Point (Optional)	154
Evaluation Delay	154
Recovery Options	154
Recovery Time	155
PIUnit Debug Messages	155
PI SubBatch Configuration	155
Add New SubBatch	156
Delete SubBatch	156
Refresh SubBatch Hierarchy	156
SubBatch Active Point (Required)	156
ActivePoint Behavior	157
SubBatch Name	157
PIHeading Set	158
PIBatch Configuration	158
PIBatch Index Point (Optional)	158
PIBatch Search Time	159
PIBatch Product Point (Optional)	159
PIBatch Recipe Point (Optional)	159
Save Configuration	160
Register or Unregister PIUnits	160
Migration	160
Migration errors in PI SMT	161
Add PIModules and PIUnits	162
Refresh MDB and Registered Units Only view	162
Right-click menu options	162
Current Values	164
Display current values	164
Update in real-time or freeze values	164
Customize the display	164
Export current values data	165
Refresh the current values	165

Current Values quick reference	165
Database Security	166
Edit database security settings	166
Database security for PI Server 2010 and later versions	166
Updated access permission model	167
Set access permissions with the Database Security tool	167
Set access permissions for versions earlier than PI Server 2010	167
Set default access for new PI points	168
Export database security settings	168
PIUserIncompatible user and PIGroupIncompatible group	169
Digital States	170
View digital state set properties	170
System digital state set	170
Built-in digital state sets	171
Custom digital state sets	171
Digital state set status icons	171
Search for digital states and digital sets	172
Verify that System digital state set is up to date	172
Create custom digital states	172
Add a digital state set using PI SMT	172
Add digital states to a set	173
Create digital points	173
Digital state set edits	174
Update only new digital state sets	174
Edit digital states in a set or add new states to a set	174
Delete a digital state	174
Delete a digital state set	174
Reconfigure points that use a deleted digital state set	175
Reconfigure events that use a deleted digital state set	175
Copy a digital state set	175
Rename a digital state set	176
Export or import digital state sets	176
Export digital state sets	176
Import digital state sets	176
Digital States quick reference	177
Firewall	179
PI Firewall database	179
PI Firewall connection protocol	179
Create a PI Firewall	180
Use an IP address mask	180
View PI Firewalls	180
Edit a PI Firewall	181
Save a list of PI Firewalls in XML format	181

Delete a PI Firewall	181
PI Firewall quick reference	181
Identities, Users, and Groups	183
About PI identities, PI users, and PI groups	183
About access permissions	184
Built-in PI identities, users, and groups	184
PIUserIncompatible user and PIGroupIncompatible group	185
View in single list or three tabs	186
Configure Data Archive authentication	186
Data Archive authentication methods	186
Set up Windows authentication	187
Manage PI identities	187
PI identities	187
Create a PI identity	187
PI identity configuration options	188
Delete a PI identity	188
Disable a PI identity	189
Create a PI identity mapping	189
Define a PI trust against a PI identity	190
Manage PI users	191
Create a new PI user	191
Change a PI user password	191
Delete a PI user account	192
Disable a PI user account	192
Disable explicit logins for a user account	192
Import Windows users	193
Create a PI user mapping	193
Define a PI trust against a PI user	194
Export a PI user list to file	195
Add a PI user to a PI group	195
Remove a PI user from a PI group	195
Manage PI Groups	196
Create a new PI group	196
Manage group memberships	196
Delete a PI group	197
Disable a PI group	197
Create a PI group mapping	197
Define a PI trust against a PI group	198
Export a PI group list to file	199
Interface List	200
View interface information	200
.....	0
Change the status of an interface service	200
Rename an interface service	201
Assign a Windows startup type to an interface service	201

Export a list of interfaces	201
Interface tool quick reference	201
IT Organizer	203
Configuring the IT Organizer	203
Data Archive configuration node	203
Connect to a Data Archive configuration node	204
Set up your PI Interface information	204
..... 0	0
Assign a role to a device	205
Add device roles	206
Add images to the icon list	208
Managing devices and tags	209
Update points	209
Monitored device list	209
Modify a monitored device	210
View point and device role details	211
View point details	211
View device role details	212
Manage the navigation tree	212
Add a group to the navigation tree	212
Add a device to the navigation tree	213
Device identification	214
Interface definition file	215
Interface instances	215
Clear configuration node settings	216
Licensing	217
View point and module statistics	217
Monitor point and module count together	217
View licensing information	218
License icons	218
View connection limitations	218
View licenses for Data Archive collective nodes	219
Check licensing for PI BatchView	219
Check whether you are licensed to use PIBatch	219
Check licensing for SQC	220
Mappings and trusts	221
Manage mappings	221
PI mappings	221
Windows side of the mapping	221
Data Archive side of the mapping	222
Create a mapping in PI SMT	222
Map a role or client ID to a PI identity using OIDC	223
Manage trusts	226

PI trusts	226
PI trust authentication process	226
Default trusts	227
PI SDK trusts	227
Windows authentication versus SDK trusts	228
Configure SDK authentication protocols in SMT	228
Create a PI trust	228
Trust wizard and Advanced Trust dialog box	229
Connection types	229
Application name	229
IP information as a network path or IP address and netmask	230
Windows account information (SDK only)	230
Remove a PI trust	230
Edit a PI trust	231
Copy a PI trust	231
Export trusts to a file	231
Import trusts from another Data Archive	231
 Message Logs tool	 233
Search for messages	233
Message time range	233
Refresh message list automatically	233
Filter messages	233
Filter by source program	234
Filter by message details	234
Message severity levels	234
Display advanced filter options	235
View message source fields	235
View identifying information about a process	235
View identifying information about the origin of a message	235
Find messages displayed in Message Logs tool	236
Find messages by file properties	236
View message logs	236
View Data Archive & PI SDK Log details	236
View PIPC log details	237
View other log files	238
Set Message Logs options	239
Export messages	239
Message Logs quick reference	239
 Module Database Editor	 242
About the Module Database	242
Query Module Databases	242
View the Module Database tree	242
Navigate the Module Database tree	244
Edit module hierarchy	246
Set security attributes	247

Module Database security permissions	249
Add or modify Module Database attributes	249
Add or edit module attributes	250
Add or edit module values	251
Delete module values	251
Add or edit alias attributes	251
Add or edit PIUnit attributes	252
Add or edit property attributes	252
Add or edit heading set attributes	253
Delete a heading set	254
Add or edit heading attributes	254
 Network Manager Statistics	 256
View connection details	256
Statistics in PI SMT	256
Set display refresh rate	258
Set Network Manager Statistics list display	258
Export statistics to a file	259
 Performance Counters tool	 260
Select a Performance Monitor interface	260
Select counters to build points and view details	261
Build Performance Monitor points on the Data Archive server	261
Build Performance Monitor points with PI Builder	262
Edit point attributes	262
Long tag names	264
Rename Performance Monitor points	264
Create a Performance Monitor points template	264
Deviation	265
Compression deviation	266
Exception deviation	266
Step	267
Shutdown	268
Compression	268
Load a Performance Counters template	268
Replace updated tag names	269
 Performance Equations	 270
Viewing performance equations	270
Create a PE point	270
Enter general point attributes	271
Define a PE calculation	272
Equation fields	272
Schedule a PE calculation	273
Event-based scheduling	273
Clock scheduling	273

Scan class and interface configuration	274
Scan class offsets for performance equations	275
Set PE point archive options	277
Step	277
Shutdown	278
Compression	278
Deviation	279
Compression deviation	279
Exception deviation	280
Set PE point security	280
Define classic attributes for PE points	281
Review PE Point system data	282
Set PE point validation options	282
Rename PE points	282
Delete calculated points	282
Performance equations (PE) syntax and functions reference	282
Performance equation syntax	283
Performance equation arguments	283
Performance equation operands	284
Number operands	285
Tagname operands	285
Tagnames in expressions	285
Tagnames as function arguments	285
Tagnames that are valid time expressions	286
String operands	286
Time expression operands	287
Numbers and strings as digital states	287
Comparing the value of digital and numeric points to strings	287
Comparing a digital state to a string point	287
Setting the digital state for a numeric or digital point	287
Performance equation operators	288
Arithmetic operators	288
Arithmetic operations on time values	288
Relational operators	289
Comparing bad values	290
Comparing operands of different types	290
Time comparisons	290
Prefix operators	290
Conjunction, disjunction, and inclusion operators	291
Inclusion operator examples	291
Using the inclusion operator with digital state functions	291
Time comparisons using inclusion operators	291
If-then-else expressions	292
Operator priority	292
Data types	293
Type checking	293
Error values	294
Testing the performance equation syntax	294
Run the pipetest utility	294

Using pipetest in interactive mode	294
Using pipetest in file input mode	294
Built-in performance equation functions	295
PE functions by type	295
Alphabetical reference for PE functions	301
Abs (Tag-based PE function)	301
Acos (Tag-based PE function)	302
AlmAckStat	303
AlmCondition (Tag-based PE function)	303
AlmCondText (Tag-based PE function)	304
AlmPriority (Tag-based PE function)	305
Arma	305
Ascii (Tag-based PE function)	307
Asin (Tag-based PE function)	308
Atn (Tag-based PE function)	308
Atn2 (Tag-based PE function)	309
Avg (Tag-based PE function)	310
Badval (Tag-based PE function)	311
Bod (Tag-based PE function)	311
Bom (Tag-based PE function)	312
Bonm (Tag-based PE function)	313
Char	314
Compare (Tag-based PE function)	314
Concat (Tag-based PE function)	315
Cos	316
Cosh	316
Curve (Tag-based PE function)	317
Day	318
DaySec	318
Delay	319
DigState (Tag-based PE function)	320
DigText (Tag-based PE function)	320
EventCount (Tag-based PE function)	321
Exp	322
FindEq (Tag-based PE function)	323
FindGE (Tag-based PE function)	323
FindGT (Tag-based PE function)	324
FindLE (Tag-based PE function)	325
FindLT (Tag-based PE function)	326
FindNE (Tag-based PE function)	327
Float	328
Format (Tag-based PE function)	329
Frac	329
Hour	330
Impulse	331
InStr	332
Int (Tag-based PE function)	333
IsDST (Tag-based PE function)	333
IsSet (Tag-based PE function)	334

LCase	335
Left	335
Len (Tag-based PE function)	336
Log	336
Log10	337
LTrim	338
Max (Tag-based PE function)	338
Median (Tag-based PE function)	339
MedianFilt	340
Mid	341
Min (Tag-based PE function)	341
Minute	342
Month	343
NextEvent (Tag-based PE function)	343
NextVal	344
Noon	345
NoOutput	345
ParseTime (Tag-based PE function)	346
PctGood (Tag-based PE function)	347
Poly	348
PrevEvent (Tag-based PE function)	348
PrevVal (Tag-based PE function)	349
PStDev (Tag-based PE function)	350
Range (Tag-based PE function)	351
Right	352
Round (Tag-based PE function)	352
RTrim	353
Second	354
Sgn	355
Sin	355
Sinh	356
Sqr	357
SStDev (Tag-based PE function)	357
StateNo (Tag-based PE function)	358
StDev (Tag-based PE function)	359
String	360
TagAvg (Tag-based PE function)	361
TagBad (Tag-based PE function)	362
TagDesc (Tag-based PE function)	363
TagEU (Tag-based PE function)	363
TagExDesc (Tag-based PE function)	364
TagMax (Tag-based PE function)	364
TagMean (Tag-based PE function)	365
TagMin (Tag-based PE function)	366
TagName (Tag-based PE function)	367
TagNum (Tag-based PE function)	368
TagSource (Tag-based PE function)	368
TagSpan (Tag-based PE function)	369
TagTot (Tag-based PE function)	369

TagType (Tag-based PE function)	371
TagTypVal (Tag-based PE function)	371
TagVal (Tag-based PE function)	372
TagZero (Tag-based PE function)	373
Tan	373
Tanh	374
Text	375
TimeEQ (Tag-based PE function)	375
TimeGE (Tag-based PE function)	376
TimeGT (Tag-based PE function)	377
TimeLE (Tag-based PE function)	378
TimeLT (Tag-based PE function)	379
TimeNE (Tag-based PE function)	380
Total (Tag-based PE function)	381
Trim	382
Trunc	382
UCase	383
Weekday (Tag-based PE function)	384
Year (Tag-based PE function)	385
Yearday (Tag-based PE function)	385
 PI Services	387
View PI services	387
PI process list	387
Add processes	387
View thread details	388
Determine service startup type	389
Service startup types	389
Start PI services	390
Stop PI services	390
PI Service display options	390
Export a list of PI services	391
Remote login	391
PI services quick reference	391
 PI Version	394
 Ping	395
Managing PI Ping points	395
PI Ping interfaces	395
PI Ping point configuration	396
Best practices for PI Ping points	396
Configure default point properties	396
Point attribute values for PI Ping	398
Create PI Ping points	398
Copy or move PI Ping points	399

Rename a PI Ping point	399
Delete a PI Ping point	399
Search for PI Ping points	399
View and edit PI Ping properties	399
Import lists into the PI Ping tool	400
Create an interface list batch file	400
Import an interface list batch file for editing	401
Create a CSV file interface list	401
Export the network node list to a CSV file	401
Add points from the network node list	401
 Point Builder	403
View PI points	403
Use Tag Search	403
Configure point attributes	404
General attributes	405
Archive attributes	405
Step	406
Shutdown	407
Compression	407
Deviation	408
Compression deviation	408
Exception deviation	409
Classic point attributes	409
Other point class attributes	409
Configure point security	410
PI point access permissions	410
Default access for new points and modules	411
Set point and data security permissions for PI Server 2010 and later	412
Set access permissions for versions earlier than PI Server 2010	412
View PI point system data	413
Turn off character validation	413
Rename PE points	414
 Point Classes	415
View point classes	415
 Point Source Table	416
Point Sources	416
Add a point source	416
Edit a point source description	417
Export a point source list	417
Built-in point sources	417
 Reason Tree	419
View reason codes	419

Add reason codes	419
Reason code title restrictions	419
Edit reason codes	420
Move reason codes	420
Delete reason codes	420
About moving reason codes between Data Archive servers	421
 Security settings.....	422
Understanding the security levels	422
Authentication options that are most secure	423
Configure security settings	423
 Snapshot and Archive Statistics	424
Set up automatic refresh	424
Freeze snapshot and archive statistics	425
Change the refresh rate	425
View snapshot statistics only	425
View archive statistics only	425
View snapshot and archive statistics together	426
Export snapshot and archive statistics	426
View snapshot and archive statistics	426
 SNMP	427
Information provided by SNMP	427
SNMP information exchange	427
COUNTER values	428
Build SNMP points	428
Select a PI SNMP interface and configure tag settings	429
SNMP agent configuration	430
Create SNMP agent profile	430
Select a saved agent profile file	430
View network SNMP interfaces	430
Validate SNMP agent	430
Get interface information	431
Map OIDs to PI Points	431
Select OIDs	431
Create PI SNMP points	432
Modify point attribute properties	432
Manage OID templates	433
Load an OID template	433
Remove an OID template	433
Apply an OID template	433
Create OID templates	434
Preview SNMP points	434
SNMP point preview options	434
Rename PI SNMP points	435

Point name restrictions	435
Export PI SNMP Point Configurations	435
Default point attributes	435
ifInOctets point attributes	436
ifOutOctets point attributes	436
ifInErrors point attributes	437
ifOutErrors point attributes	437
ifAdminStatus point attributes	438
ifOperStatus point attributes	439
SysUptime point attributes	439
SQC Alarms	441
SQC Alarms tool	441
Create SQC alarms	442
Edit SQC alarms	442
SQC Alarms wizard	442
Basic SQC alarm attributes	442
Automatic point generation	443
Source chart and sampling	443
Associated points	444
Specify conditions for the SQC pattern tests	446
Change the execution state	448
Change control limits	448
Delete alarms	448
Stale and Bad Points	450
Find malfunctioning points	450
Decommission a point with Point Builder	450
Decommission multiple points with PI Builder	451
Search parameters for stale or bad points	451
View Stale and Bad Points	452
Export a list of stale and bad points to file	452
TCP Response	453
Create a TCPResponse point	453
Using the PI TCPResponse plug-in	453
Point Browser	453
Save button	455
Delete button in Point Browser	455
Create point button	455
Edit default point properties button	455
Search points	456
Refresh button	456
Network node list	456
New node button	457
Delete button	457

Create button	457
Import button	457
Export button	458
Stop button	458
Resize button	458
DNS check box	458
Test Measure check box	458
Point Property pane	458
Supported Data Archive versions	460
Totalizers	461
Search for a PI totalizer	461
Key parameters for PI Totalizer tags	461
Types of summary calculations	462
Edit an existing PI totalizer	463
Create a new PI totalizer	463
PI totalizer configuration	464
Specify the sampling	464
Specify how to write results	465
Set archive attributes	466
Step	466
Shutdown	467
Compression	467
Deviation	468
Exception deviation	468
Compression deviation	469
Define security settings	469
Configure optional functions	470
Turn off character validation in the Totalizer tool	470
Copy a PI totalizer to another Data Archive	471
View information about a PI totalizer	471
Tuning Parameters	472
Configurable tuning parameters	472
Add a tuning parameter to the list	473
Edit tuning parameters	473
Export a list of tuning parameters	473
Update Manager	475
View consumer details	475
Consumer columns	475
Consumer details	476
View producer details	477
Producer columns	477
Producer details	478
Producers and associated subsystems	478

View statistics summary	479
Statistics columns	479
Change the update refresh rate	479
Reduce displayed content for high-latency connections	480
SMT security requirements	481
Alarm groups security permissions	481
Archive editor security permissions	481
Archives security permissions	482
AutoPointSync security permissions	482
Backups security permissions	483
Batch custom names security permissions	483
Batch database security permissions	484
Batch generator security permissions	484
Current values security permissions	485
Database security permissions	485
Digital states security permissions	486
Firewall security permissions	486
Identities, users, & groups security permissions	486
Interface list security permissions	487
IT Organizer security permissions	487
Licensing security permissions	488
Message Logs security permissions	488
Mappings security permissions	489
MDB to AF synchronization security permissions	489
Access permissions required to run the preparation wizard	490
Module Database security permissions	490
Network Manager Statistics security permissions	491
Performance counters security permissions	491
Performance Equations security permissions	492
PI Point Source Table security permissions	492
PI Services security permissions	493
PI Version security permissions	493
Ping security permissions	494
Point Builder security permissions	494
Point Classes security permissions	495
Reason Tree security permissions	495
Security security permissions	495
Snapshot and Archive Statistics security permissions	496
SNMP Points security permissions	496
SQC Alarms security permissions	497
Stale and Bad Points security permissions	497
TCP Response security permissions	497
Totalizers security permissions	498
Trusts security permissions	498
Tuning Parameters security permissions	499
Update Manager security permissions	499

AVEVA™ PI SMT 2023 Release Notes	500
---	------------

Welcome to PI System Management Tools

PI System Management Tools (PI SMT) is an easy-to-use graphical user interface application that provides tools you can use to perform many Data Archive administration tasks, including configuring security settings, managing archives, building and configuring points, and viewing message logs.

PI SMT is included in the Data Archive installation, but you can get the latest version of PI SMT on the [Customer Portal](#).

Start PI SMT

From the Windows **Start** menu, choose **All Programs > PI System > PI System Management Tools**.

PI SMT behaves differently on different versions of Data Archive

Features that are not available on older versions of Data Archive do not appear in PI SMT if you are connected only to an older Data Archive. This means that PI SMT can look a little different, depending on the version of Data Archive to which it is connected. Note these major version differences:

- PI Server 2015 and later versions include archive management for future archives that are used for forecast data.
- PI Server 2010 and later includes PI AF. Objects in PI Module Database are automatically synchronized with PI Asset Framework. The MDB to AF Synchronization tool in PI SMT allows you to monitor this synchronization.
- PI Server version 3.4.380 represents a significant change in security configuration. You might need to manage PI Server instances that use the old security model along with servers that use the new model. Using PI SMT helps you do that seamlessly.

Note: If you are installing, or upgrading to PI Server version 3.4.380, refer to the *Data Archive Security Configuration* section in the *AVEVA PI Server Installation and Configuration guide* for the security model and implementation options.

Select a Data Archive server

The Servers pane, in the upper left area of PI SMT, displays a list of all known Data Archive servers, sorted by collectives and individual servers.

- To choose a Data Archive server, click the corresponding check box.

Note: Clicking the check box does not immediately create a connection to the corresponding server. You connect with that server only when you take an action with a PI SMT tool. PI SMT shows you whether you are connected to servers and the type of connection.

- To connect to a server as a specific user, right-click the server and click **Connect As**.
- To find a server or collective, in the Search field, enter all or part of a server or collective name.
- To hide or show the Servers pane, click **View > Servers**.

Add a Data Archive server

Use the Servers list in the upper left corner of PI SMT to connect to one or more Data Archive servers. If PI SMT does not list the Data Archive server you want to connect to, use Connection Manager to add the server to the list.

Note: To add a Data Archive server to PI SMT, you must have access permissions to that server or collective.

1. Select **File > Add Server**.
2. In the **Network Path/FQDN** field, enter the network path (host name or IP address) of the Data Archive server. If the target server is a Data Archive collective, enter the path to a server in the collective. You can also choose a default user for the connection.
3. Click **OK**.

The Data Archive server appears in the Servers list.

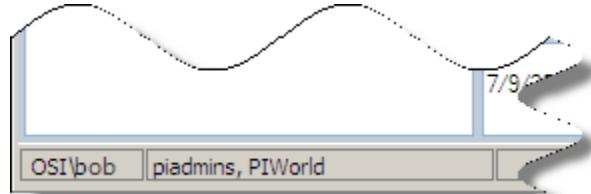
View or save the session record

The Session Record pane in the lower right area of the PI System Management Tools shows any errors and other messages for the currently selected tool. It provides a useful record of tasks during the current SMT session. The session record also shows all error messages, some of which are also logged in the local message log.

- To hide or show the Session Record pane, choose **View > Session Record**.
- To save the session record data to a text file, choose **File > Save Session Log**.

View Data Archive connection credentials

The lower left corner of the PI SMT window shows your Windows user name, followed by all the PI identities, PI users, and PI groups that you are identified with: either through a PI mapping, a PI trust, or a PI user account.



- If multiple Data Archive servers are selected, you will see *Multiple*. Your Windows account is always displayed, even if you are not connected to the Data Archive server through a mapping.
- When you double-click, you can see the full connection information for each connected Data Archive server. For example, if you are connected to MyDataServer1 through a mapping and to MyDataServer2 through a default user account, the information would look something like this:

```
MyDataServer1 via mapping: PIEngineers, PIWorld
MyDataServer2 via default user: pidemo
```

Set Windows or PI Time format

You can configure time stamps to display in PI Time or Windows format.

1. To select a format for displaying timestamps in the plug-ins, choose **View > Settings**.
2. In the Settings dialog box, click the **Time Format** tab.
3. Select a time format.

The options are:

- **Windows Format**

When selected, shows the default setting for the currently selected regional setting. A sample of the timestamp format appears in the **Windows Format** field. To specify another format for the date and time, select **Custom** and select from the **Date** and **Time** lists.

- **PI Time Format**

When selected, time stamps are formatted using PI Time format. A sample of the time stamp format appears in the **PI Time Format** box. You can also select the format for subseconds.

AF Link for migration to AF

The AF Link tool enables you to perform the following migrations:

- **PI Batch Database and PI Batch Subsystem data to PI Event Frames**

You can choose to migrate your batch data to event frames. Event frames help you capture, compare, and analyze important processes or business events over a repeatable time period. They can track many kinds of events, including batches. Event frames provide wider search options and greater flexibility in composition than batches.

After migration, there is no synchronization between the Batch Database and AF; the Batch Database becomes read only, and batch interfaces, version 4.0, automatically switch to writing to event frames.

- **Module Database (MDB) to AF**

After installation or upgrade to PI Server 2010 or later, migration occurs automatically unless you choose not to enable MDB during setup. To enable MDB after installation, you must manually initiate a migration.

Data Archive maintains a two-way synchronization between MDB and AF. Changes made in MDB are automatically reflected in AF, and changes made in AF are automatically reflected in MDB.

PI Batch to event frame migration

Starting with PI Server 2015, you can migrate your PI Batch Database and PI Batch Subsystem data to event frames. Event frames help you capture, compare, and analyze important processes or business events over a repeatable time period. They can track many kinds of events, batches being just one type. Event frames provide wider search options, and greater flexibility in composition than using batches. For more information about event frames, see the PI Server topic *Structure of event frames*.

Because you will need different client applications to manage batch data in event frames than you have been using with the Batch Database, do not migrate to event frames until the solutions you need for event frames are all in place. Also, once you start the migration, the Batch Database becomes read only, and you will not be able to revert to it.

We do not recommend that you run the PI Batch Database and use event frames in parallel for the same batch data. We envision that you will migrate all batch data at one time from the PI Batch Database. An exception is if you decide to create event frames to support non-batch data (such as for monitoring equipment downtime) at the same time as you create batches in the PI Batch Database.

Note: We recommend that, unless you have non-critical migration requirements or a very simple environment, you test the migration of your batch data to event frames and the cutover to event frame data sources and consumers.

Allow approximately 90 minutes per million event frames for the migration, bearing in mind that this time will vary greatly depending on network, hardware and other factors.

Prepare batch data sources and consumers for migration

To successfully transition your environment from using the PI Batch Database and PI Batch Subsystem, you need

to prepare all of your batch data sources and consumers, or devise alternate solutions for supporting your batch needs using PI Event Frames.

Note: If you plan to use RtReports with event frames, you need to use RtReports version 4.0.

First, inventory how your current batch system is constituted. You'll need to take actions for each batch data source and consumer to get it ready to switch to event frames.

1. For each Data Archive server that you plan to migrate, take an inventory of the current sources and consumers of PI batch data.

Sources of batch data might include:

- PI Event File (EVT) interfaces
- PI BaGen (PI Batch Generator interfaces)
- Batch interfaces 1.x, 2.x, 3.x, and 4.x
- Your proprietary interfaces
- PI Batch Subsystem

Consumers of batch data might include:

- PI BatchView PI ProcessBook add-in
- PI BatchView Microsoft Excel add-in (used with PI DataLink)
- RtReports
- PI WebParts Batchview Control
- PI OLEDB
- Your in-house PI-SDK batch applications
- Code you have implemented within PI ProcessBook displays
- Third-party applications

2. Prepare your batch data sources. Your site will be ready to migrate to event frames after you prepare each batch data source and consumer, as described here.

Caution: When you start the migration, the Batch Database becomes read-only, therefore do not proceed with migration until you have identified effective ways of adapting each data source and consumer for event frame data, or devise alternate solutions for supporting your batch needs using PI Event Frames.

After you initiate migration, most batch interfaces fail to create further PI batch database data. To continue generation of event frames after the migration, you must identify and update all data sources that support generating event frames. Newer versions of some interfaces, specifically batch interfaces version 4.0, automatically cut over from writing batch data to writing to the event frame database. For other interfaces, you must switch manually. Go to the [Customer Portal](#) for interfaces that are not listed here for which you need more information.

Data source	Actions
PI Event File (EVT) interface	See PI Event File (EVT) interface .
PI BaGen interface	See PI BaGen interface .
Batch interfaces 1.x, 2.x, 3.x	See Batch interfaces 1.x 2.x 3.x .

Data source	Actions
Your proprietary interfaces	See Your proprietary interfaces .
PI Batch Subsystem interfaces	See PI Batch Subsystem interfaces .

3. Prepare your batch data consumers.

Consumer	Actions
PI BatchView PI ProcessBook add-in	See PI BatchView PI ProcessBook add-in .
PI BatchView Microsoft Excel add-in (used with PI DataLink)	See PI BatchView Microsoft Excel add-in (used with PI DataLink)
RtReports	See RtReports .
PI WebParts Batchview Control	See PI WebParts Batchview Control .
PI OLEDB	See PI OLEDB .
In-house PI SDK Batch applications	See In-house PI SDK Batch applications .
Your code in PI ProcessBook displays	See Your code in PI ProcessBook displays .
Third-party applications	Explore solutions offered by third-party vendors.

Batch data source

Batch interfaces scan a data source for events of interest, such as the start or end of a level, and the acquisition and release of equipment.

PI Event File (EVT) interface

Upgrade EVT interfaces to PI batch interfaces, version 4.0, or later.

The PI interface for Batch Event File Monitor, V 3.8.7.0, or earlier, cannot write to event frames, so you must upgrade if you want to populate batch information into event frames.

If you require EVT files as a data source, you might be able to use a batch interface that handles EVT files and can write to event frames. All version 3.x interfaces can write to event frames. The following version 3.x Batch interfaces can read EVT files as a data source:

- Emerson DeltaV Batch (EMDVB) Interface
- Rockwell Factory Talk Batch (FTBInt) Interface
- GE iBatch Batch (GEIB) Interface
- Honeywell TotalPlant Batch (HWTPB) Interface

If you want to use a different interface, you must upgrade it to version 4.0, or later.

See the Knowledge Base article: [Migration from Batch Event File \(EVT\) Interface to Batch Interface Framework, DeltaV Batch Interface, or RtReports](#).

When a batch interface version 4.0 is running and you initiate migration, the following actions occur:

1. The migration process takes existing batches and writes them to event frames using templates in the OSIBatchMigration category.
2. The interface detects that the Batch Database is locked and switches to generating event frames. It writes any currently running batches to event frames using templates in the OSIBatch category.
3. The interface then loads any EVT files in the source folder that fall within the ABTO (abandoned batch time out) range into its cache. The interface reprocesses these batches to event frames using the OSIBatch templates.
4. From that point forward, the interface writes new batches to event frames using the OSIBatch category templates.

If you have batch applications you have customized in-house, be aware that with PI batch interfaces, version 4.0, PI Batch Properties might reside at different levels in event frames. Refer to your interface documentation for details about where these properties will be located. Consequently, you might need to upgrade any dependent client applications and reports to use the PI batch data created by these newer interfaces.

Note: The migration process will migrate existing EVT Interface data, but only PI batch interfaces, version 4.0, or later, support automatically switching from the PI Batch Database to event frames, after migration.

PI BaGen interface

Note: We recommend you install the latest version of PI EFG (4.0.11.104, or later) before migrating, because 4.0.11.104, or later, have significant migration-related improvements.

You cannot use PI BaGen after migrating. Therefore, before you migrate, make PI EFG (PI Event Frames Generator) ready to generate batch data in event frame format (but do not turn it on until after migration).

Note: PI EFG has additional functionality compared to PI BaGen, so you might want to explore those features before you migrate to see if you want to take advantage of them.

Test using PI EFG to produce data that is consistent with your existing PI BaGen interfaces. You can use the Converter tab in PI EFG to migrate existing PI BaGen configurations to PI EFG. Be certain to set PI BaGen compatibility mode. For more information, see PI Event Frames Generator.

Stop PI BaGen. Confirm that PIBaGen batches are completed and not open. If there are batches open, determine the best approach for closing these open batches.

After you have migrated your batch data, enable PI EFG to generate batch context data in event frame format.

If you cannot find a time when you can run migration without in-progress batches, you can use a feature of PI EFG to recover your in-progress batches. After migration, use PI EFG historical recovery to recover these batches:

1. Note the date and time that you stop the PI BaGen Interface, just prior to migration.
2. Migrate your data.
3. After migration completes, start PI EFG in Recovery mode so that PI EFG recovers any batches that were open during migration. Set PI EFG to recover from the earliest in-process batch.

Go to the [Customer Portal](#) if you have questions about how best to configure these options.

Note: Older versions of PI EFGen (3.0.20.567, and earlier) are not aware of migrated event frames, therefore, it is possible to have duplicate event frames in these cases. To avoid this problem, upgrade to PI EFGen 4.0.11.104, or later, before migrating.

Batch interfaces 1.x 2.x 3.x

Upgrade all interfaces that generate batch data to batch interfaces, version 4.0, or later, which automatically switch from writing to the PI Batch Database to writing PI Event Frames to the AF database, when the interface detects migration is complete.

Interfaces that cannot write event frame data, such as the PI Interface for Batch Event File Monitor, must be retired.

Your proprietary interfaces

Assess how you can upgrade any proprietary interfaces to use the AF SDK to write event frames. You need to make these interfaces ready to generate batch data in event frame format, but do not turn them on until after migration.

PI Batch Subsystem interfaces

For legacy PI Batch Subsystem data to be migrated, PI Batch Subsystem units must be mapped to the Module Database.

The pre-migration analysis that you run from the AF Link tool in PI-SMT checks for PI Batch Subsystem units (the PI Batch Subsystem must be running during the analysis). If the analysis shows:

- Units that are mapped to the Module Database, those batches will be migrated.
- Units that are not mapped to the Module Database, if you want this data migrated to event frames, you must map these units to the Module Database.

To map PI Batch Subsystem units to the Module Database, follow the instructions in the *Batch Database Support of the PI Batch Subsystem*, available on the [AVEVA Customer Portal Products page](#).

For each PI Batch Subsystem unit, only Batch Subsystem batches before the cutoff date will be migrated to event frames. Therefore, if there is no cutoff date, or if the cutoff date is set to *Current*, all Batch Subsystem batches for that unit will be migrated to event frames.

After migration, the PI Batch Subsystem no longer generates batch data. Therefore, you must install a new interface to generate event frame data.

Note: To migrate to event frames, you do not have to move your PI Batch Subsystem data to the Batch Database, but you must create mappings in the Module Database for any PI Batch Subsystem unit that you want to migrate.

Batch data consumers

PI Batch is used in conjunction with its companion Client application, PI BatchView, which allows you to search, select, trend, and compare events that have been collected by PI Batch and stored in the AVEVA™ PI System™. Earlier versions of PI BatchView were based on the PI API, and more recent versions are based on the PI SDK.

PI BatchView PI ProcessBook add-in

Use a third-party partner product, such as Mirabo System's Livepoint, for single-trend capability.

PI BatchView Microsoft Excel add-in (used with PI DataLink)

Upgrade to PI Datalink 2014, or later, if you are not already using it.

Existing Excel spreadsheets will continue to work for old data still in the Batch Database. However, you need to create new spreadsheets if you want to use PI BatchView Microsoft Excel add-in with PI DataLink 2014, or later, with event frames.

RtReports

Use RtReports, Version 4.x, which supports event frames.

PI WebParts Batchview Control

Explore implementing equivalent functionality using the PI Table web part and PI OLEDB Enterprise SQL queries to display event frame data.

PI OLEDB

Use PI OLEDB Enterprise with SQL queries to access event frame data, providing similar functionality to traditional batch data access. This approach aligns with the modern PI System architecture, enabling event frame analysis and reporting for batch data consumers.

In-house PI SDK Batch applications

Explore implementing equivalent functionality using the AF SDK, a software development kit that provides access to objects and features of PI Asset Framework.

Your code in PI ProcessBook displays

Overlay trends are supported in AVEVA PI Vision. You can use them to implement equivalent functionality. Or you can extend your current visualization solution to use the AF SDK to access PI Event Frames data.

Get ready to run the migration process

Caution: Before getting ready to run migration, you must assess all of your batch data sources and consumers and prepare them for the transition. See [Prepare batch data sources and consumers for migration](#).

To run the migration process, you need to have certain versions of software installed, and sufficient permissions set, as described here.

If you are running PI Server 2010 or later, with no data in the Module Database

If you are running PI Server 2010 or later and the following conditions apply, you must perform some tasks before migration:

- You have not added data to your Module Database.
- Use the MDB Editor in PI SMT to check your Module Database; if the only modules you see are in the %OSI branch, you have no user data in your Module Database.
- You have Batch Subsystem batches.
- The pre-migration analysis that you run from the AF Link tool in PI-SMT shows you if you have PI Batch Subsystem units (the PI Batch Subsystem must be running during the analysis).

If you have no Module Database and do have PI Batch Subsystem batches that you want to migrate to event frames, perform these steps before migration:

1. Migrate the Module Database to AF. Use the PI MDB to AF Preparation Wizard, which is included in PI AF Link in PI SMT.
2. Map PI Batch Subsystem units to the PI Module database. For information about how to do this, see [Prepare batch data sources and consumers for migration](#).

Ensure you have Data Archive 2015 or later

To migrate your batch data to event frames, you need to be running Data Archive 2015 or later, which includes the analysis and migration tools in PI-SMT. This release introduced support for migrating batch data to event frames.

Direct upgrade to PI Server 2015 or later is supported from all 64-bit versions of Data Archive (versions 3.4.370.99, and later).

If you have a version older than 3.4.370.99, first move your Data Archive server to a 64-bit computer and then upgrade to PI Server 2015 or later. For more information, see the Knowledge Base article: [Upgrading to 64-bit PI Data Archive while moving to 64-bit hardware](#).

Ensure you have PI AF Server 2015 or later

At migration time, you must have at least PI AF Server 2015 (version 2.7) or later installed on your system.

Configure batch interface security settings to access AF

Before starting migration, configure AF security settings to ensure that the account running the batch interface has permission to create event frames.

The account running the PI batch interface, version 4.0, must be able to connect to the target AF server and database, and have the following AF permissions:

- Database: Read and Write
- Categories: Read and Write
- Elements: Read and Write

- Element templates: Read and Write
- Event frames: Read and Write

Make offline archives available online

When you migrate to event frames, all historical archives that are registered with Data Archive will have their batch data migrated. If you register additional historical archives with batch data after you perform the migration, those archives' batch data will not be migrated. So be sure all batch data is registered with Data Archive before migration.

SQL Server Express insufficient for large data sets

If you are using the Express edition of SQL Server, we recommend you do not work with more than 3 million event frames. You can determine the number of event frames that will be created by running the Analyze function and reviewing the results of the analysis.

Analyze batch data before migration

Before you can migrate your batch data to AF event frames, you must analyze your batch data to find potential incompatibilities with AF. After fixing essential issues (such as creating any necessary PI identity mappings) and making any optional changes, you can migrate your data.

1. In PI AF Link in the Operations section of PI SMT, select the Data Archive server for which you want to migrate batch data.

Note: The field **Batch database status** indicates whether analysis has already been run, or if migration is complete.

2. Click **Analyze** to find potential incompatibilities with AF in your batch data.

The **Analyze** function checks that the user running PI AF Link has appropriate permissions (read-write permissions on the PI SMT AF Link Security database are required).

Analysis might take a few minutes, depending on the size of your online archives.

3. Click **View report** to obtain a list of issues.

The report summarizes the number of batch records analyzed and issues encountered, shows if there are any gaps in your archives, and provides details about issues such as missing links between batch objects, and any PI identity mappings that need to be created.

- Issues marked with a red X must be resolved before migration.
- Issues marked with a yellow bell warning icon are recommended to be resolved before migration.
- Issues marked with a blue i are optional.

4. If you have archive gaps that you want to fill, use the PI SMT Archives tool or command-line tools to register archives that cover the gaps. Click **View** in the Registered Archives box to find out more information about the gaps in your archives.

While avoiding archive gaps is recommended to ensure a complete Batch Database is online, it is not required to fill these gaps to complete the migration. Because batch data is migrated only once, filling the

gaps with batch data at a later time does not cause that batch data to be migrated to event frames. Batch data is only migrated once, so make sure all of your batch data is online before you perform the migration.

5. Security for migrated event frames is based on mappings of Windows Active Directory users and groups to PI identities. Therefore, you will be prompted to create a mapping for each PI identity that is part of the access control list (ACL) on any Batch Database record, if the mapping does not already exist. Click **Create mapping** and enter required information for each new mapping.

The mappings created for the Module database to AF synchronization are still valid for batch to event frame migration, but there might be some additional mappings needed.

Note: After migration, users who had read and write permissions to PI Batch data should have read and write access to event frames. However, in some situations, batch to event frame migration will not be able to set the security on event frames. If this happens, a message to that effect is recorded in the list of migration errors shown in AF Link. In such cases, the event frames will either inherit permissions from the parent or be assigned default permissions (defined by event frame templates).

6. There are some characters that are valid for names in the Batch Database but are not valid in AF. Before migration, you can modify these names in the Batch Database, or they will be replaced with a default character in AF (only) during migration.

For example, a semicolon (;) is replaced with a comma (,) and square brackets are replaced with parentheses. After migration, you can view the original character before it was replaced in the event frame's **Extended Properties**.

For further details about migration issues, see [Analysis error report](#).

7. The maximum length of an object name in AF is 259 characters. You can shorten long names, for example Batch IDs, before migration, or allow them to be truncated during migration.

After migration, the full original name is stored in the event frame **Extended Properties**.

8. If you made any changes, click **Rerun analysis** to re-analyze the Batch Database.
9. After you have fixed essential issues and have made all desired optional changes, follow the steps in [Migrate batch data to event frames](#).

Note: When you start the migration, the Batch Database becomes read-only, therefore you need to prepare thoroughly before migration. See [Prepare batch data sources and consumers for migration](#) if you have not already done so.

Analysis error report

Category	Severity	Description	Possible action	Sample message
PIArchss Subsystem Shutdown	Error	PI Archive Subsystem is shutting down, analysis is aborted.	Start the PI Archive Subsystem.	<i>Batch Database analysis is aborted due to PI Archive Subsystem shutdown</i>
Invalid HA Batch Database Access	Error	Unable to access batch records on a secondary Data Archive server. Analysis is not	Analysis is not allowed on a secondary Data Archive server.	<i>Invalid HA Batch Database access</i>

Category	Severity	Description	Possible action	Sample message
		initiated.		
Batch Database Not Supported	Error	The batch database is not supported in future archive sets, so no analysis can be done.	Analysis is only allowed for historical archive sets.	<i>Error detected while analyzing Batch Database records, Batch Database is only supported in Historical Archive Set</i>
Point Error	Error	Failed to get the point IDs for batch (PIBatch, PICampaign, PIUnitBatch), analysis stops.	If the analysis fails to get point IDs for batch points, there is a serious issue in Data Archive. Check if the Data Archive server is functioning properly.	<i>Failed to get points for PIUnitBatch and PI Batch Subsystem</i>
Invalid BDB Type	Warning	PI Batch Database (BDB). If a PI point obtained for analysis is not for a PIBatch, PICampaign, or PIUnitBatch, analysis skips it and continues with the next point.	None.	<i>Invalid Batch Database type</i>
Unable to Get Event	Warning	An error occurred while getting an event for a batch database point. If this event is the first event of this point, analysis skips all events for this point and continues with the next point. Otherwise, analysis skips this event and continues with the next event.	None.	<i>Failed to get first event</i>
Invalid UID	Warning	Failed to read UID for a batch database record.	None.	<i>Failed to get UID from the annotation</i>

Category	Severity	Description	Possible action	Sample message
Invalid BatchID	Warning	Failed to get Batch ID for a batch database record.	None.	<i>Failed to get PIBatch name</i>
Invalid Procedure	Warning	Failed to obtain a procedure name for a PIUnitBatch.	None.	<i>Failed to get Procedure name for PIUnitBatch</i>
Invalid PISubBatch Name	Warning	Failed to get a PISubBatch name.	None.	<i>Failed to get the name for PISubBatch</i>
Invalid PIProperties	Warning	Failed to get PIProperties for PIBatch or PICampaign.	None.	<i>Failed to get PIProperties collection for PIBatch or PICampaign name</i>
UID Mismatch	Warning	The UID stored in the event does not match the UID stored in the annotation.	None.	<i>The UID stored in the event does not match the one stored in annotation</i>
Invalid Child Reference	Warning	The child reference of a Batch Database record cannot be found or verified. The message logged in the analysis report shows details about the failure. If a child reference is not accessible due to an archive gap, it is logged under the "Archive Off-line" category. The child's stored parent UID might not match the actual parent UID, or the system may have failed to get the child's:	None.	<i>Failed to get child collection</i>

Category	Severity	Description	Possible action	Sample message
		<ul style="list-style-type: none"> • collection • point ID • UID • effective date • object • Batch ID • stored child collection • stored child UID 		
Invalid Parent Reference	Warning	<p>The parent reference of a batch database record cannot be found or verified. The analysis report message shows details of the failure. If a parent reference is not accessible due to an archive gap, it is logged under the "Archive Off-line" category.</p> <p>The parent might not contain the reference to the child, or the system may have failed to get the parent's:</p> <ul style="list-style-type: none"> • collection • point ID • UID • effective date • object • Batch ID • stored child collection • stored child UID 	None.	<i>Failed to get UID of parent PIBatch</i>

Category	Severity	Description	Possible action	Sample message
PIUnit Deleted	Warning	<p>Some PIUnits were deleted, but their related PIUnitBatches and child PISubBatches still exist and can be found through a child reference from PIBatch.</p> <p>These PIUnitBatches and PISubBatches are not migrated.</p>	None.	<i>PIUnit : deleted PIUnit UID has been deleted.</i>
IsPIUnit False	Warning	<p>In some PIUnits, the IsPIUnit flag is set to <i>false</i>, but their related PIUnitBatches and child PISubBatches still exist and can be found through a child reference from the PIBatch.</p> <p>These PIUnitBatches and PISubBatches are not migrated unless their IsPIUnit flag is set back to <i>true</i>.</p>	To migrate these PIUnitBatches and child PISubBatches, set the IsPIUnit flag to <i>true</i> for these PIUnits.	<i>PIUnit : TestPIUnit0001 PIUnit name and ID has IsPIUnit flag set to false.</i>
Archive Off-Line	Warning	Failed to get a child or parent object due to the target archive being offline.	Bring the archive back online.	<i>Failed to get object of child PIUnitBatch because the target archive is off-line.</i>
Null Name	Info	<p>For a PIBatch or a PICampaign, the BatchID is used as the element name in the event frame. If the BatchID is null:</p> <ul style="list-style-type: none"> For a PIBatch or a PICampaign, this message is logged in the 	Migration automatically names a null name with the string: <i><undefined></i> in the event frame. Or you can set a BatchID or procedure name in the batch database before migration.	<i>BatchID name for PIBatch is null</i>

Category	Severity	Description	Possible action	Sample message
		<p>analysis report.</p> <ul style="list-style-type: none"> For a PIUnitBatch, the procedure name is used as the element name in the event frame. <p>If the procedure name is null, the BatchID is used.</p> <p>If the Batch ID and the procedure name are both null, this message is logged in the analysis report.</p>		
Length Exceeds Limit	Info	The maximum length for a name in PI AF is 259 characters.	Migration automatically truncates the name. Or you can shorten the name in the batch database before migration.	<i>The maximum name length is 259 characters. PIBatch exceeds limit by n characters</i>
Disallowed Characters	Info	The following characters are allowed in MDB but not in AF: ; { } [] \	Disallowed characters are automatically replaced with the migration replacement characters that were set for the module database to AF migration. Alternatively, replace them in the batch database before migration.	<i>PIBatch contains disallowed characters</i>
Control Characters	Info	Control characters are not allowed in the name.	Control characters are automatically replaced with a !.	<i>PIUnitBatch has control characters</i>

Category	Severity	Description	Possible action	Sample message
			Alternatively, remove or replace them in the batch database before migration.	

Note: When the **Possible action** is *None*, you do not need to do anything before migration because the batch data is not migrated. This generally occurs when those unit batches or batches are either deleted or missing.

You can try to recover the batch data by reloading the data through the interface. Not all interfaces can recover partial batches or unit batches. Even after recovery, these same warnings are reported. Before proceeding, migrate into a test system and evaluate the specific migrated unit batch data.

We do not recommend manually altering PI Event Frames after migration, other than deleting verified duplicates (as can be encountered, for example, when moving from PIBaGen to EFGen).

Test migration and cut over from current sources and consumers

We recommend that you test both of the following procedures, unless you have non critical migration requirements or a very simple environment:

- The migration of your batch data to event frames
- The cut over to event frame data sources and consumers

We recommend you copy your production Data Archive server to a test environment. Go to [Customer Portal](#) for support on how to clone your production system. You clone your environment utilizing Data Archive documented procedures.

Ensure that the test includes:

- A thorough review of the pre-migration analysis report, see [Analyze batch data before migration](#).
- Verification that after migration, batch type event frame data is populated and properly presented from all data sources and by all consumers. See [Prepare batch data sources and consumers for migration](#) and [Verify your event frame data](#).
- Verification that after migration, all configured instances of PI batch interfaces, version 4.0, have switched over automatically, and manual interfaces, such as PI EFGen (PI Event Frames Generator), correctly populate data and properly present it for all consumers. See [Prepare batch data sources and consumers for migration](#) and [Verify your event frame data](#).

Migrate batch data to event frames

Before you migrate your batch data to AF event frames, first you must analyze your batch data to find potential incompatibilities with AF, as shown in [Analyze batch data before migration](#). After fixing essential issues and making any optional changes, you can migrate your data.

During migration, the Module Database is unavailable for updates or modifications. You can update elements in AF, but changes will not be synchronized with the Module Database until after migration completes.

Note: When you start the migration process, the PI Batch Database becomes read-only and the PI Batch Subsystem stops generating batches. Therefore you need to prepare thoroughly before migration. See [Prepare batch data sources and consumers for migration](#).

If you have upgraded to the latest PI batch interface, version 4.0, or later, and it is running when migration starts, it goes into a wait state and before resuming, adds the required AF settings to the initialization (.ini) file. To avoid inadvertently overwriting these settings, close the Event Frame Interface Configuration Manager before starting migration.

After migration, you can review the results. If you chose to utilize PI EFGen in Recovery mode (as explained in [Prepare batch data sources and consumers for migration](#), you might need to delete duplicate batches. You should delete in-progress batches that belong to Category *OSIBatchMigration*, as long as there are completed and correct batches belonging to the Category *OSIBatch*. For more information, see [Event frame templates for batch data](#).

Note: You do not have to migrate while the batch is running, but it is perfectly alright to do so. Your batch interface log will record when it detects the start of migration, while it is waiting for migration to complete, and when it starts generating event frame data.

1. Using the PI AF Link tool in PI SMT, after you have obtained an analysis report that contains no must-fix issues, and you have resolved all desired optional issues, click **Migrate** (available at the end of the Analysis report).

The **Migrate** function checks that the user running PI AF Link has appropriate permissions (read-write permissions on the PI SMT AF Link Security database are required).

Migration progress is shown, and the **Batch database status** field indicates when migration is complete. Migration can take a while, depending on the amount of PI batch data online. As a guideline, on average, migration can take approximately 1 hour for each year of PI Batch data. To obtain a better estimate, use the information provided when you clicked Analyze about the number of event frames you will be generating. Allow approximately 90 minutes per million event frames, bearing in mind that this time will vary greatly depending on network, hardware and other factors.

2. After migration completes, click **View migration report** to open the Migration report.
3. Click **View errors** to see information about any errors that occurred during migration.

Informational messages are also shown, for example about the following actions:

- Automatic name modification (truncation of names longer than 259 characters, and replacement of any characters that are invalid in AF).
- Event frames named with the string *<undefined>*. You can update this string to give the event frame a meaningful name. This might occur with batch data created by PIBaGen that is migrated to PI Event Frames.
 - The Procedure name of a PI Unit Batch is used as the name of the event frame. If the Procedure name is blank, the Batch ID is used as the event frame name. If both are blank, the event frame is given the name *<undefined>*.
 - The Batch ID of a PI Batch is used as the event frame name. If the PI Batch has no Batch ID, the event frame is given the name *<undefined>*.

Note: The Details tab shows the path to the file containing the migration error report.

For a list of migration messages, see [Migration error report](#).

4. You now have event frames in AF that represent your batch data. See [Verify your event frame data](#) for information about confirming the migration has gone smoothly and has resulted in event frame data in your desired format.

For example, when you view an event frame's properties, click **Extended Properties** to see information such as an original object name before truncation or character replacement took place.

Note: After you have migrated your batch data to event frames, you will no longer be able to re-migrate the Module Database to AF. Remigration used to be a workaround when the Module Database had lost synchronization with the AF database. The recommended workaround is to reset the Module Database from the AF database; in the AF Link tool in PI SMT, on the Details tab, click **Reset MDB**.

Migration error report

If migration fails, you need to contact Technical Support using the [Customer Portal](#) to determine the cause and have their assistance to reenable migration.

Category	Severity	Message	Action
Batch Database event generation: Migration fails	Error	Failed to get points for PIUnitBatch and PI Batch Subsystem Failed to add Unit Batch point ID to list Failed to add point for PIBatch Failed to add point for PICampaign Failed to get point from the array of BDB Points Failed to generate BDB events for EF Failed to get BDB points, BDB is only supported in Historical Archive Set Failed to add data Failed to post change notification for EF BDB event generation interrupted Failed to post change notification for EF Failed to get active BSS batch	Contact Technical Support using the Customer Portal .

Category	Severity	Message	Action
		Failed to get active BSS batches due to PIBatch Subsystem shutdown BSS active batch event generation interrupted BSS inactive batch event generation interrupted Failed to post change notification for EF Failed to get BSS batch BSS event generation interrupted	
Batch Database event generation: Migration continues	Warning	Cannot get first BDB event Cannot get first BSS event	None
Event frame generation	Error	InsertChildEventFrames: BDB event frame, EFID <i>event frame UID</i> , not found InsertChildEventFrames: BDB event frame, EFID <i>event frame UID</i> , not found: <i>exception description</i>	Migration continues
	Warning	Failed to delete processed change from the event queue due to shutdown or comm failure EventFrame for <i>object name</i> was not found. ID: <i>batch UID</i> UnitID: <i>unit UID</i> StartTime: <i>start time</i> Parent ID: <i>parent UID</i> CreateEventFrameForCampaign, Set event frame security failed: < <i>exception text</i> >	

Category	Severity	Message	Action
		CreateEventFrameForPIBatch, Set event frame security failed: <exception text>	

Verify your event frame data

To verify your resulting event frame data:

1. Confirm that PI batch interfaces, version 4.0, or later, have automatically switched to event frames.

The interfaces log file **PIPC.Log** will include messages about migration and cutover:

- When the interface fails to write to the PI Batch Database, because batch to event frame migration is occurring:

PI Batch Database to event frames migration is in progress.

- When migration is complete:

PI Batch Database is read-only because it has been migrated to event frames.
Interface will create event frames in the PI Asset database:
AFserver\\AFdatabase

Where *AFserver* is the name of the PI AF server, and *AFdatabase* is the name of the AF database, to which Data Archive is migrating batches.

2. Confirm that the PI Batch Database is read-only.

The AF Link tool shows you if the Batch Database is read-only. Also, if you attempt to edit batch data the edit will be rejected.

3. Enable or start up any PI EFGen (PI Event Frames Generator) or non-Batch Framework interfaces that create PI Event Frames data.
4. Confirm that event frames were properly migrated by looking at the **Batch database status** field in the AF Link tool and by checking the Migration errors report.

You might also want to perform a random check of a few event frames in PI System Explorer. To look at an individual event frame, right-click it and choose **Properties**, you can verify that the start time and end time make sense and use the Attributes tab to check values such as Batch ID, Procedure, and Product. You might also want to look at the overall event frame hierarchy in PI System Explorer to check it looks as expected.

Note: You can click **Extended Properties** to see information such as the original object name before any truncation or character replacement took place during migration.

5. In any reporting tools you use, confirm that batch data is being presented properly in event frame format to your end users.

Event frame templates for batch data

The migration process creates event frame templates in your AF database that define the minimum required attributes to represent a batch record, such as Batch ID, Product, Procedure Name, and Recipe. After the migration completes, the batch interfaces create event frame templates that match the interface requirements.

OSIBatchDB is a generic event frame category for batch data that contains parent templates for the other batch event frame template categories. The following categories of event frame templates derive from OSIBatchDB:

- OSIBatchMigration is used by AF Link for migrated batches.
- OSIBatch is used by event frame interfaces, such as EFGen and Batch 4.0 interfaces.

Templates for migrated batch data	Notes
PICampaignMigrated	This template is for migrated PI Campaigns
PIBatchMigrated	This template is for migrated PI Batches
PIUnitBatchMigrated	<p>For unit batches, the name of the procedure is used for the event frame name. If the procedure name is empty, the batch ID is used instead. If the batch ID is empty, the event frame name is the string <i><undefined></i>.</p> <p>PI Batch Subsystem batches are migrated to event frames derived from the PIUnitBatchMigrated template. The procedure name attribute of the event frame will be empty, so in this case, the batch ID is used to create the name of the event frame.</p> <p>A PI Heading associated with PI Batch Database batches is not migrated, because there is no equivalent object in the AF database.</p>
PISubBatchMigrated_L1	Level 1 PI Subbatches are migrated to event frames that derive from the template named PISubBatchMigrated_L1.
PISubBatchMigrated_L2	Level 2 PI Subbatches are migrated to event frames that derive from the template named PISubBatchMigrated_L2.
PISubBatchMigrated_L3	Level 3 PI Subbatches are migrated to event frames that derive from the template named PISubBatchMigrated_L3.
PISubBatchMigrated_L4	The template PISubBatchMigrated_L4 is used for levels 4, and deeper.

PI batch interfaces, version 4.0, or later, create event frames that derive from PICampaign, PIBatch, PIUnitBatch and PISubBatch. Due to how the templates derive, by searching against these templates attributes (Product, Recipe, BatchID, Procedure and Product), you can quickly find common event frames created by the migration process or PI batch interfaces, version 4.0.

Note: Transfer records are not migrated to AF. After the migration of batch data is initiated, PI transfer records become read-only and no new transfer records are generated.

How a batch object converts to an event frame

The PI Batch Database contains objects that are organized in a hierarchy: PI Campaign, PI Batch, PI UnitBatch, and PI SubBatch:

- A PI Campaign represents a logical group of PI Batch objects.
- A PI Batch represents a set of activities used in production of a batch and includes a collection of PIUnitBatches.
- A PI UnitBatch represents a period of processing or activity for a specific piece of equipment.
- PI Subbatches are subsets of a PI UnitBatch; they cannot exist as stand-alone objects.

In the PI Batch Subsystem there is just one type of object, called a Batch.

PI Campaigns are migrated to event frames as an instance of the PICampaign_Migrated template. Event frames that are derived from the PICampaign_Migrated template include a collection of event frames based on the PIBatches_Migrated template. For any given PI Campaign, the PI Batch collection should match the event frame collection in the corresponding event frame for that PICampaign, assuming all those PI Batches exist in Data Archive at the time of migration.

PI Campaign information	Equivalent information in the event frame
Start time	Start time
End time	End time
Campaign ID	Event frame name
PI Batch collection	Child event frame collection
PIProperties	AF attributes

PI Batches are migrated to event frames as an instance of the PIBatch_Migrated template. Event frames derived from the PIBatch_Migrated template include a collection of event frames based on the PIUnitBatch_Migrated template. For any given PI Batch, the PIUnitBatch collection should match the event frame collection in the corresponding event frame for the PIBatch, assuming that all those PIUnitBatches and the referenced PIUnit exist in Data Archive at the time of migration.

PI Batch information	Equivalent information in the event frame
Start time	Start time
End time	End time
Batch ID	Event frame name
Product	AF attribute named Product
Recipe	AF attribute named Recipe
PIUnitBatch collection	Child event frame collection

PI Batch information	Equivalent information in the event frame
PIProperties	AF attributes
Parent PI Campaign	Parent event frame

PIUnitBatches are tied to a PIUnit in the Module Database. PI UnitBatches are migrated to event frames as an instance of the PIUnitBatch_Migrated template. Event frames based on the PIUnitBatch_Migrated template also have a reference to the Element that represents the PIUnit in the Module Database. PI Unit Batches include a collection of PISubBatches. Event frames derived from the PIUnitBatch_Migrated template include a collection of event frames based on the PISubBatch_L1_Migrated template . For any given PIUnitBatch, the PISubBatch collection should match the child event frame collection in the corresponding even tframe for the PIUnitBatch, assuming that all those PISubBatches exist in Data Archive at the time of migration.

PI Unit Batch information	Equivalent information in the event frame
Start time	Start time
End time	End time
Procedure Name	Event frame name (if not blank)
Batch ID	AF attribute named BatchID. If Procedure Name is blank, the Batch ID is used as the event frame name.
Product	AF attribute named Product
Referenced PIUnit	Referenced Element
PISubBatch collection	Child event frame collection
Parent PI Batch	Parent event frame

PISubbatches are associated with a PIUnitBatch. PISubBatches can be nested any number of levels deep. PISubBatches at level 1 are migrated as event frames derived from the PISubBatch_L1_Migrated template. PISubBatches at level 2 are migrated as event frames derived from the PISubBatch_L2_Migrated template. PISubBatches at level 3 are migrated as event frames derived from the PISubBatch_L3_Migrated template. PISubBatches at level 4 and greater are migrated as event frames derived from the PISubBatch_L4_Migrated template.

PI Sub Batch information	Equivalent information in the event frame
Start time	Start time
End time	End time
Name	Event frame name
PIHeading	Not represented in the event frame
Child PISubbatch collection	Child event frame collection

PI Sub Batch information	Equivalent information in the event frame
Parent object (could be a PIUnitBatch or a PISubBatch)	Parent event frame

The PI Batch Subsystem is an older database than the PI Batch Database. Data in PI Batch Subsystem is different than PI Batch database data and the units referenced in the Batch Subsystem data are also different from the PIUnits in the Module Database. There is a way to "link" a Batch Subsystem unit to a PIUnit in the Module Database, but the two databases remain different (for information about how to set this up, see [Prepare batch data sources and consumers for migration](#)). Batch Subsystem data that is linked in this way is also migrated with PI Batch Database data to event frames. PI Batch Subsystem data has no hierarchy, however, it is functionally equivalent to PIUnitBatches. Therefore, Batch Subsystem batches are migrated as event frames derived from the PIUnitBatch_Migrated template.

PI Batch Subsystem information	Equivalent information in the event frame
Start time	Start time
End time	End time
BatchID	Event frame name
Product	Product
Referenced unit	Referenced Element

PI Properties are converted to AF attributes

In the Batch Database, PI Properties are stored at the root level of the batch hierarchy, only, in the batch properties collection. Each node in the collection is the name of the recipe level (procedure, unit procedure, operation, or phase). A PI Property has a name and a value, and the value has a data type. You can nest one batch property under another.

When you migrate your batch data to event frames, an AF attribute is created to represent the batch property; you can set AF attributes at any level of the batch hierarchy. For the migrated PI properties:

- The AF attribute name is the same as the name of the corresponding batch property.
- The batch property hierarchy, if any, is preserved in the hierarchy of the AF attributes.
- The type of the AF attribute, as shown in the **Value Type** field in PI System Explorer, is shown in the following table.

Batch property data type	AF attribute Value Type	For Value Type Anything, AF attribute Value points to object
Boolean	Boolean	
Byte	Byte	
Byte Array	Byte Array	
Date	DateTime	

Batch property data type	AF attribute Value Type	For Value Type Anything, AF attribute Value points to object
Double	Double	
Double Array	Double Array	
Float	Single	
Float Array	Single Array	
Long	Int32	
Long Array	Int32 Array	
Null	<Anything>	Null
Object	<ul style="list-style-type: none"> For PI Properties that store objects such as a PI Alias or a PI Module, Value Type for the AF attribute is set to <Anything> and the value of the attribute points to the same object to which the PI Property pointed. 	Not supported
PIAlias		PIAlias
PIHeading		PIHeading
PIModule		PIModule
PIPoint		PIPoint
PIProperty		PIProperty
PIUnitBatch		PIUnitBatch
PIValue		PIValue
Server		PIServer
Short	Int16	
Short []	Int16 Array	
String	String	

Setting the security descriptor on event frames

In the PI Batch Database, access control lists (ACLs) manage who can access and update a PI Batch object. These ACLs are based on records in the database security table (DBSECURITY). Each PI Batch object, PI Campaign, PI Batch, and so on, has a record that defines permissions. Note that permissions for PIUnitBatches and PI Subbatches are set by the record for PI Units.

Active Directory objects, such as event frames, have security descriptors that control access to the AF objects. Security descriptors are data structures that contain the ACL of the object, which includes all of the permissions that apply to that object. When someone tries to access the object, the operating system security-subsystem examines the object's security descriptor to see if access should be granted, and if the requested action is

permitted.

When you migrate to event frames, the ACLs on the batch objects are translated into a Windows security descriptor for the event frame. The migration process takes any PI Identities that are mapped to the users and groups in the ACL, and uses them to define the Windows security descriptor. This security descriptor is used within AF to control access to the event frame.

Migration process

When you initiate batch to event frame migration, the conversion follows this outline:

1. The archive subsystem walks through all the archives and generate events. PIUnits are explored first, and within each PIUnit the walk starts from the most recent PIUnitBatch and proceeds backwards in time.
2. The events for PI UnitBatches are posted to an event queue.
3. The archive subsystem generates events for all PI_batches, starting with the most recent and proceeding backwards in time. These events are posted to the queue.
4. The archive subsystem generates events for all PI Campaigns, starting with the most recent and proceeding backwards in time. These events are posted to the queue.
5. The AF Link tool starts reading these events and generating event frames.

MDB to AF Synchronization

The MDB to AF Synchronization tool shows you the synchronization status between PI MDB and PI AF.

About MDB to AF Migration and Synchronization

On PI Server 2010 and later, the Data Archive maintains a two-way *synchronization* between PI MDB and PI AF. Changes that you make in MDB are automatically reflected in AF. Similarly, changes that you make in AF are automatically reflected in MDB. This means that AF-based applications and MDB-based applications can access the same content.

MDB and AF synchronization begins with an initial *migration* and is thereafter automatic. Migration is the process during which Data Archive initially copies the PI Module Database to an AF database. Migration is typically a one-time operation, but it is possible to re-migrate if necessary.

Migration occurs automatically after installation of or upgrade to PI Server 2010 or later versions, unless you choose not to enable MDB during setup. If you do not choose to enable MDB during the installation of or upgrade to PI Server 2010 or later versions, then migration and synchronization do not occur and MDB is read-only. To enable MDB at any point after installation, you must manually initiate a migration.

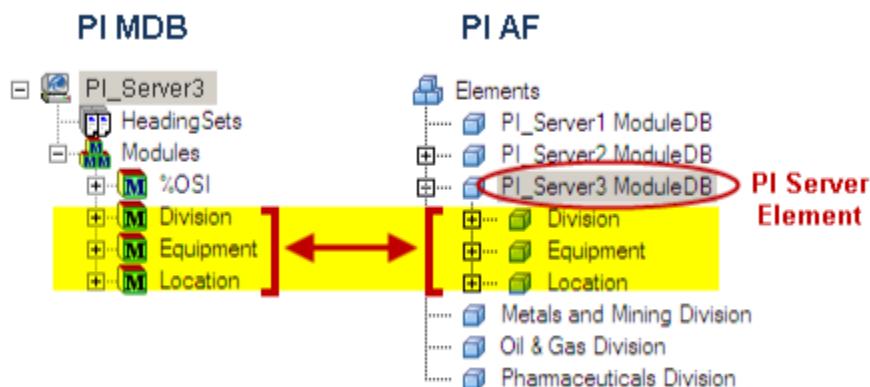
On PI Server 2010 and later, PI AF is the system of record. When PI AF Link cannot connect to the AF Server, MDB becomes read-only. Users can continue to make changes in AF, but cannot edit MDB until the connection is restored.

About the Data Archive element

AF Link synchronizes PI MDB with the content under the specified Data Archive element in PI AF. The Data Archive element designates the portion of a PI AF database that is synchronized with PI MDB. Any content in PI

AF that is outside a Data Archive element is not synchronized with PI MDB. (There are a few configuration modules that are not synchronized with PI AF. See [What content is not synchronized](#)).

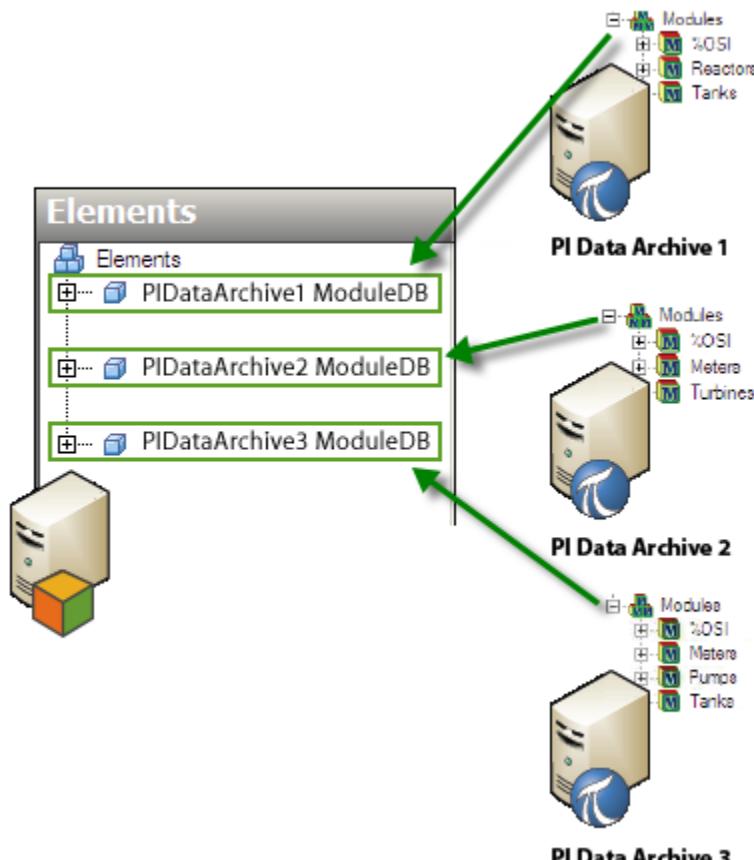
PI MDB contents synchronized under Data Archive element



This diagram shows the content of PI MDB on a Data Archive server called PI_Server3 with the corresponding Data Archive element in PI AF. Only the highlighted content is synchronized between the two.

A single PI AF server can be synchronized with multiple Data Archive servers. Each Data Archive server that is synchronized with PI AF has its own corresponding Data Archive element in PI AF. The PI AF content under each Data Archive element is synchronized only with PI MDB on the corresponding Data Archive server, as indicated in the following illustration.

One PI AF server synchronized with multiple Data Archive servers



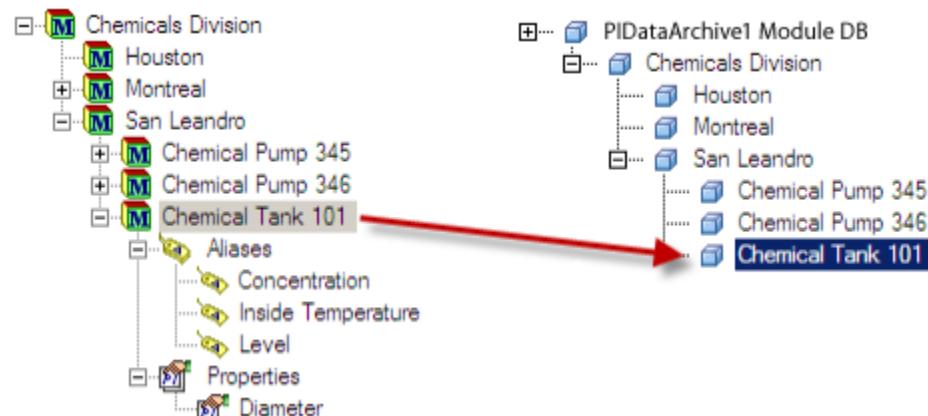
When multiple Data Archive servers are synchronized with a single PI AF server, it is important to choose descriptive names for each Data Archive element. For example, in the preceding illustration, the Data Archive elements are called PI_Server1 ModuleDB, PI_Server2 ModuleDB, and PI_Server3 ModuleDB.

How PI MDB objects are represented in PI AF

The top-level modules from PI MDB are located directly under the Data Archive element in PI AF. Each PI MDB module is represented by a PI AF *element*. A module's properties and aliases are represented in PI AF by *attributes* on the corresponding element. Objects have the same names in PI MDB and PI AF.

Under the Data Archive element, the PI AF elements reproduce the hierarchy of the PI MDB modules. However, the hierarchy of PI properties and aliases is represented differently in PI AF. In PI MDB, aliases and properties are represented as child objects in the tree structure, while PI AF attributes are not. The illustration below shows a migrated module with aliases and properties (shown in SMT's Module Database Editor), and the corresponding PI AF element (shown in PI System Explorer).

Migrated module shown as a PI AF element



To see the migrated properties and aliases, you need to look at the attributes for the element. In PI System Explorer, click to select the element and then click the **Attributes** tab from the set of tabs to the right. PI aliases are represented as attributes that have a data reference to a PI point; these attributes are marked with a tag icon.

Aliases shown as attributes in PI AF

General Child Elements Attributes Ports Version		
<input type="text" value="Search"/> 🔍		
	Name	Value
<input checked="" type="checkbox"/>	Concentration	1.23716259002686 kg/m ³
<input checked="" type="checkbox"/>	Diameter	5.2 m
<input checked="" type="checkbox"/>	Inside Temperature	0.409331083297729 °C
<input checked="" type="checkbox"/>	Level	0 m

What content is not synchronized

In PI MDB, the following objects are not migrated to or synchronized with PI AF:

- OSI proprietary Modules:

The modules %OSI, %OSI_MCN, and %OSI_ManagedPI store metadata for PI applications. These modules are not migrated to PI AF. This metadata remains available to PI MDB applications; PI AF applications do not need it and cannot access it.

- PI Heading Sets, PI Headings:

PI Heading Sets and PI Headings are not migrated to PI AF. PI AF elements are not assigned a PI Heading even if the corresponding PI module references a PI Heading. PI MDB clients still display the PI Heading associated with a PI module but PI AF clients cannot display PI Headings.

- Batch Data and Batch Subsystem (BSS) Units:

After the migration, you can continue to access batch data on the Data Archive server. See [How batch objects are migrated](#) for more information.

In PI AF, all objects under a Data Archive element are synchronized with PI MDB on the corresponding Data Archive server. Any content that is not under a Data Archive element is not synchronized with PI MDB.

PI MDB edits that are not allowed

On PI Server 2010 and later, you cannot make changes to PI MDB that are incompatible with PI AF. There are also some PI MDB edits that you should not make, even though you may be allowed to.

Changes you cannot make in PI MDB:

- You cannot create an PI MDB object with a name longer than 259 characters. PI AF limits the length of object names to 259 characters. If you try to create a module with over 259 characters, you get an error message.

Note: For PI property names, the maximum length is 259 characters minus the length of the replacement suffix (for more information, see [Replacement suffix](#)). For example, if the replacement suffix has 5 characters, then the maximum length of a property name is 254 characters.

- You cannot grant access permissions to an PI MDB object for a PI identity (or PI user or PI group) that does not have a mapping.
- You cannot edit properties or aliases on a PI Module if the corresponding PI AF element is based on a PI AF template. PI AF templates impose restrictions on the elements that use them. MDB allows no edits on properties or attributes of the corresponding modules.
- You cannot create links into or out of %OSI, %OSI_MCN or %OSI_ManagedPI. The same is true for %OSI_MCN and %OSI_ManagedPI hierarchies. You can create links only if both objects are under the %OSI, %OSI_MCN or %OSI_ManagedPI hierarchies or if neither object is under the %OSI, %OSI_MCN or %OSI_ManagedPI hierarchies.
- When multiple versions of a module exist, you cannot change the effective date of an existing version to be inconsistent with dates on other versions.
- Specifically, the effective date cannot be modified such that it is older than the previous version (if it exists) and newer than the next version (if it exists). The obsolete date for any of these versions cannot be older than the effective date for that version.

- There are no restrictions on adding new versions and what effective date is used for these new versions.

Changes you *can* make in MDB, but should avoid:

- Create a PI property or alias name that contains any of the following characters:
`; { } [] \`
- The above characters are not valid in PI AF. PI AF Link uses *replacement characters* (see [Replacement characters](#)) when it creates the objects in PI AF. In some rare cases, it is possible to name objects in MDB such that the substitution of the replacement character would create a naming conflict in PI AF. It is safest to avoid illegal characters in new MDB object names.
- Create a PI property name that ends in the *replacement suffix* (for more information, see [Replacement suffix](#)). In rare cases, it is possible to name objects in MDB such that the substitution of the replacement suffix would create a naming conflict in PI AF. It is safest to avoid using the replacement suffix in new MDB object names.

Tools for editing PI AF and PI MDB objects

PI Module Database and PI AF both have tools that enable you to directly view and edit your content.

- Tools for working in PI AF

Use PI System Explorer (PSE), which is included with PI Server. You can also download PSE from the [AVEVA Customer Portal Products page](#).

Start PSE from the PI System folder in the Windows Start menu (**Start > All Programs > PI System > PI System Explorer**). For more information, see PI Asset Framework and PI System Explorer.

Note: PSE is sometimes referred to as the AF client.

- Tools for working with the PI Module Database PI AF

Use the Module Database editor, which is included in PI SMT (choose **Operation > Module Database**).

You can also use the Module Database Builder, which is a tool for performing bulk operations on PI MDB. The Module Database Builder is available from the **Tools** menu in PI SMT.

Make PI AF content accessible in PI MDB

AF Link synchronizes PI MDB content for a Data Archive server with the PI AF content under the corresponding Data Archive element (see [About the Data Archive element](#)). If you want to access PI AF content from PI MDB-based clients, follow these steps:

Procedure

1. In PI SMT, open the AF Link tool (**Operation > AF Link**).
2. Look at the Current Status for the Data Archive server. If it says that the PI MDB to AF preparation wizard has not yet been run or has not yet been completed, then you need to run the wizard:
 - a. Click the toolbar icon  to launch the preparation wizard.
 - b. Follow the prompts in the preparation wizard. The wizard will ask you to specify a PI AF server, database, and Data Archive element.
 - c. After you complete the wizard, PI AF Link Subsystem starts the synchronization between MDB and AF.

3. Using the PI System Explorer, identify the elements that you want to access through PI MDB. Paste references to these elements under the Data Archive element. PI AF Link will create the corresponding objects in PI MDB.
-
- Note:** Under the Data Archive element, place only the content that you need to access in PI MDB. Everything you place under the Data Archive element is synchronized with PI MDB.
-
4. To see the objects in PI MDB, open the Module Database tool in SMT (**Operation > Module Database**).

Enable PI MDB

If you chose to disable PI MDB during upgrade or installation, then PI MDB is read-only. You cannot edit it. To enable PI MDB at any time after installation:

1. In PI SMT, open the AF Link tool (**Operation > AF Link**).
2. Click the toolbar icon  to launch the preparation wizard.
3. Follow the prompts in the preparation wizard. The wizard has extensive help, so consult it if you are not sure about a step.

After you complete the wizard, AF Link Subsystem starts the initial migration. When the migration is complete, PI MDB will be unlocked.

Finding and fixing synchronization problems

Problems with synchronization between PI MDB and AF come in two general categories: health status problems and security problems.

- Health Status Problems: The PI MDB to AF Synchronization tool shows the synchronization health status of each connected Data Archive server. When you show the detailed view on a particular Data Archive server, you see a more complete health status message. See [Health status](#) for an explanation of the messages and instructions for handling them.
- Security Problems: PI AF and Data Archive use different mechanisms for authentication and access permissions. Because of the differences in the two systems, some security settings cannot be synchronized. This can lead to inconsistencies in MDB and AF access permissions. See [Checking security synchronization](#) for more information.

When synchronization problems arise, users typically experience the problem as a restriction in the ability to view or edit MDB. Another common symptom is that a user cannot view or edit the same content both in MDB and AF. See [Troubleshooting synchronization](#) for common symptoms and possible causes.

Health status

The AF Link tool shows a health status message for each connected Data Archive server that runs PI Server 2010, or later.

Performance Counter Value	Message in List View	What it Means	What to Do
0	InSync	MDB and AF are in sync.	No action needed.
1	SyncInProgress	PI AF Link Subsystem is synchronizing the PI Module Database and the AF Server.	No action needed.
2	OutOfSync	The PI Module Database and the AF Server are out of sync.	Reset MDB.
3	BaseToAFLinkCommFailure	PI Base Subsystem has a communication error with PI AF Link Subsystem.	If the PI AF Link Subsystem Windows service is not running, start it. If the service is running, this error will likely resolve itself within several minutes.
4	AFLinkToBaseCommFailure	PI AF Link Subsystem has a communication error with PI Base Subsystem.	If the PI Base Subsystem Windows service is not running, start it. If the service is running, this error will likely resolve itself within several minutes.
5	AFLinkToAFCommFailure	PI AF Link Subsystem has a communication error with the AF Server.	If the PI AF server is not running, start it. If the SQL Server is not running, start it. If both are running, check the log file for other errors, such as incorrect access permissions or AF Server name.
6	CheckinFailure	There was a check-in failure when checking in changes to the AF Server. Most likely another user has the object checked out.	Do nothing. PI AF Link will retry the check-in. The error should resolve in time.
7	The MDB to AF migration wizard has not yet been completed.	You ran the PI MDB to AF preparation wizard on this Data Archive server but	Complete the preparation wizard.

Performance Counter Value	Message in List View	What it Means	What to Do
		you have not yet successfully completed it.	
8	WizardNotRun	You have not yet run the PI MDB to AF preparation wizard on this server.	Run the PI MDB to AF preparation wizard on this server.
9	TargetDatabaseDeleted	Target AF Database or Data Archive element was deleted or not found.	<p>Run the MDB to AF preparation wizard. If you do not want to do that:</p> <ul style="list-style-type: none"> Manually recreate the database or element, and then restart PI AF Link. Restore the database or element from a backup, and then restart PI AF Link.
10	DBAFRestoreInconsistent	MDB and AF Database are not restored from the same backup.	Restore MDB and the AF Database from same backup or reset MDB.
11	InsufficientPermissionsTo AF	PI AF Link Subsystem does not have the correct permissions on AF Database to continue.	<p>Make sure the PI AF Link Subsystem service has the following access permissions:</p> <ul style="list-style-type: none"> Read access on AF Databases collection and AF Elements collection on the target database. Read-write access on the target AF Database, Data Archive Element, and AF Categories collection on the target database.
12	FailedToDeleteMDBEvents	Failed to read MDB Events from update queue.	Do nothing. PI AF Link will retry and this error resolve in time.

Performance Counter Value	Message in List View	What it Means	What to Do
13	Failed to Reset MDB from AF Database	Failed to reset MDB from AF Database.	Try to reset MDB. If it continues to fail, contact Technical Support using the Customer Portal .
14		Not currently used.	
15	GenericBadState	An error exists that does not fall into any category in this table.	<ul style="list-style-type: none"> Wait 10 minutes. PI AF Link will retry. If the issue persists, check the details pane or open the Message Logs tool and look for error messages from piaflink or pibasess. If you do not find any indication of the issue, from Windows Services, restart PI AF Link Subsystem. If the issue persists, contact Technical Support using the Customer Portal.
None	PI AF Link Subsystem is not running	PI Base Subsystem is running but PI AF Link is not.	Start PI AF Link Subsystem Windows service.
None	Health status is bad for the Base subsystem	The PI AF Link Subsystem status and the PI Base Subsystem status do not match.	On the details pane, find the health status for PI Base Subsystem. Take the appropriate action for that health status message.
None	Health status is bad for PI AF Link Subsystem	The PI AF Link Subsystem status and the PI Base Subsystem status do not match.	On the details pane, find the health status for PI AF Link Subsystem. Take the appropriate action for that health status message.
None	Migration preparation is	The Data Archive server is	Upgrade the server.

Performance Counter Value	Message in List View	What it Means	What to Do
	complete, upgrade to PI Server 2010 to enable PI AF Link.	running a version earlier than PI Server 2010 and you have successfully run the preparation wizard on this Data Archive server.	

Checking security synchronization

PI AF Link synchronizes security between AF and MDB such that each PI module has the same access permissions as the corresponding AF element. Further, the access permissions on the Data Archive element in AF are the same as the database security settings for PI Modules in Database Security on the Data Archive server.

Both AF and Data Archive leverage Windows for implementing security, but the extent and mechanism of the implementations are different. Because of these implementation differences, it is possible for the security configuration in MDB to diverge from that in AF, potentially causing access problems for users.

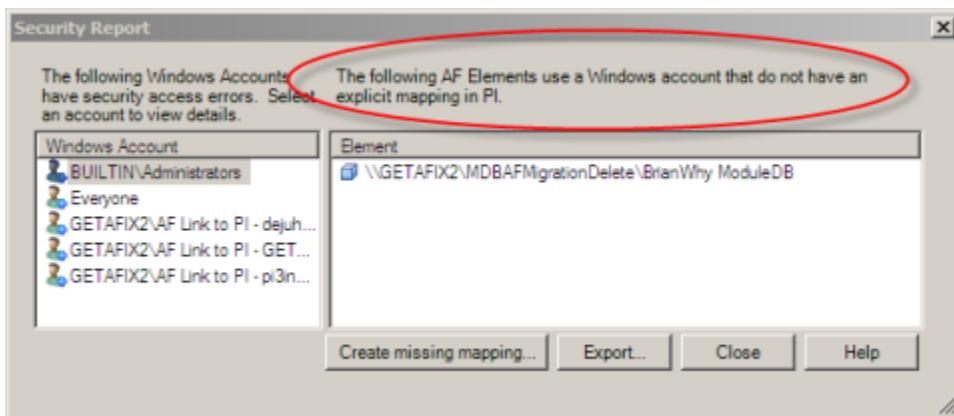
If the security settings are out of sync, then users might have different levels of access to the same content, depending on whether they are using MDB or AF clients. The MDB to AF Synchronization tool in PI SMT displays a report of any areas where the security is unsynchronized.

View the security synchronization report

You can view a report of security synchronization issues at any time.

1. In PI SMT, open the AF Link tool (**Operation > AF Link**).
2. Click the **Details** tab.
3. At the bottom of the details pane, click the **Check for Security conflicts** button. If there are problems with the security synchronization, a window appears listing them.

The report has two panes: on the left a list of Windows accounts that have security access errors; on the right the specific error for the selected Windows account. The specific error message (circled in red below) tells you what is wrong. See [Possible security conflicts](#) for more on each type of error.



Depending on the type of error, a button might appear at the bottom of the report, allowing you to fix the problem. In the example illustrated above, the fix is to create a missing mapping.

Guidelines for good security synchronization

To minimize security synchronization problems, follow these guidelines:

- The Data Archive server and PI AF server must either be in the same domain, in trusted domains, or in a trusted forest.
- Make sure the access permissions on PIModules in PI Database security are the same as the access permissions on the Data Archive element in PI AF. You can edit the access permissions on PIModules using the Database Security tool in PI SMT (**Security > Database Security**).
- Use Windows authentication on the Data Archive server for all PI MDB access. All the PI identities, users, and groups that have access to Modules must have explicit mappings. Furthermore, the Windows accounts from these mappings must be used directly in the PI AF permissions.

For example, suppose the Windows user *Bob* belongs to a group *BobGroup*, and *BobGroup* is mapped to a PI identity called *ModuleAccessIdentity*. *ModuleAccessIdentity* has access to a module on the Data Archive server. When modifying the security of the corresponding element in PI AF, you should use *BobGroup* – not *Bob* itself – because *BobGroup* is the Windows account specified in the mapping.

- Do not delete mappings that are needed by module security. If you delete a mapping that is needed by a module, then the access permissions for PI AF and PI MDB will no longer be synchronized, and you will not be able to edit the security of the affected module.
- Make sure that no users rely on PIWorld for PI MDB access. PIWorld cannot be mapped, and access permissions defined for PIWorld are not reflected to PI AF.
- Make sure that no users rely on piadmin for PI MDB access. The piadmin PI user has unrestricted read and write access to everything on the Data Archive server. Thus, we recommend that you do not map piadmin and do not use it for routine access to the Data Archive server. Reserve piadmin exclusively for the very few and rarely executed administrative tasks that no other PI identity can perform.
- In PI AF, do not use deny access for any element under the Data Archive element. PI AF allows you to explicitly deny access, but Data Archive does not. If you use deny on an element in PI AF, then everyone except piadmin will lose all access to the corresponding module.

Possible security conflicts

The following methods of handling security do not have a direct mapping between PI MDB functionality and PI AF functionality. Therefore, if you use any of the following security methods, you will need to choose how to handle this in PI AF.

Identity does not have a mapping

Cause: This means that PI MDB has access permissions for a PI identity (or PI user or PI group) that does not have a PI mapping. Every PI identity, user, or group that has PI MDB access permissions must have at least one PI mapping. If you delete the last mapping, you create a security conflict between PI MDB and PI AF.

Consequences: You cannot edit PI MDB access permissions for the affected modules, except to delete the identity that does not have mapping. You can still edit the module itself.

Fix: Create a new mapping for the relevant identity, user, or group. Alternatively, delete the PI MDB access permissions for that identity, user, or group.

Note: AF Link does not automatically pick up changes in mappings. The change does not take effect until you edit

the element in some way; this triggers AF Link to update the settings in MDB.

PI AF uses a Windows account that has no mapping

Cause: You added PI AF access permissions for a Windows account that does not have a PI mapping. This creates a security conflict between PI MDB and PI AF.

Consequences: The corresponding module can be viewed and accessed in PI MDB only by accounts that have the necessary mappings.

Fix: On the Data Archive server, add a mapping for the Windows account. Alternatively, in PI AF, remove access permissions for that account.

Note: AF Link does not automatically pick up changes in mappings. The change does not take effect until you edit the element in some way; this triggers AF Link to update the settings in PI MDB.

Element uses deny access

Cause: You specified *deny* access in PI AF for an element under a Data Archive element. Data Archive does not support deny access.

Consequences: The corresponding module can be viewed and accessed in MDB only by piadmin. If you are not piadmin, you cannot see the module.

Fix: Remove or disable the deny access.

Element security does not match module security

The PI AF element uses PI AF security options that are not reflected on the Data Archive server.

This means access permissions might not be the same for PI MDB and PI AF. If you are working with objects in both PI MDB and PI AF, this might cause a problem. To fix this problem, you can manually edit the access permissions to match, or reset PI MDB.

Missing mapped principal

This means that the PI module specifies access for a PI identity (or user or group) that is mapped to two or more Windows accounts; and one of those principals is not on the element security in PI AF.

Cause: Either of the following conditions:

- A PI AF element is added to PI AF with only one of those principals on the security descriptor.
- One of those principals was removed after PI AF and PI MDB were in synchronized state.

This problem will not arise for a one-to-one mapping between a PI identity and a Windows principal unless there are some major synchronization errors between MDB and PI AF (for example, they might be out of sync already for other reasons).

Consequences: The access permissions might not be the same for PI MDB and PI AF. This could be a problem if users are trying to work with objects in both PI MDB and PI AF.

Fixes (in order of preference):

1. Add all missing principals to the element.

2. Remove the remaining principal(s) from the element.
3. Edit the access permissions on the module to remove access for the PI identity. This will remove the remaining principals from element.

Note: AF Link does not automatically pick up changes in mappings. The change does not take effect until you edit the element in some way; this triggers AF Link to update the settings in PI MDB.

This problem will not arise for a one-to-one mapping between a PI identity and a Windows principal unless there are some major synchronization errors between MDB and PI AF (for example, they might be out of sync already for other reasons).

Troubleshooting synchronization

This section describes common problems that might occur during PI MDB to PI AF synchronization.

Cannot edit PI MDB

If you cannot edit PI MDB, then you probably have one of the following problems:

- The PI AF server is unavailable. Check your connection to the PI AF server. If AF Link cannot connect to the PI AF server, then PI MDB becomes read-only until the connection is restored.
- AF Link is not running. If AF Link is not running, MDB is read-only. Check the PI Services tool in SMT (**Operation > PI Services**).
- You are trying to make an edit that is no longer permitted in PI MDB ([PI MDB edits that are not allowed](#)).
- If you are trying to edit access permissions, then the PI MDB security might be out of sync with PI AF security. [View the security synchronization report](#) to identify the problem.
- There could be a synchronization error. Check the synchronization status in the AF Link tool in SMT (**Operation > AF Link**).
- MDB is not enabled. You can enable MDB by running the MDB to AF preparation wizard from the AF Link tool in SMT ([Enable PI MDB](#)).

Element does not appear in PI MDB

It is possible to configure security on an AF element in a way that is not compatible with MDB. If you do this, then the corresponding module can be viewed and accessed in MDB only by piadmin. If you are not piadmin, then you cannot see the module. The cause could be:

- You specified *Deny* access on an AF element for a Windows user. Data Archive does not support *Deny* access.
- None of the Windows accounts that have access permissions on an AF element have mappings on the Data Archive server.

You can generate a security report to help pinpoint the problem:

1. In PI SMT, open the AF Link tool (**Operation > AF Link**).
2. Click the **Details** tab.
3. At the bottom of the details pane, click the **Check for Security conflicts** button. If there are problems with

the security synchronization, a window appears listing them.

Cannot edit access permissions on a module

If you try to edit access permissions on a module and are not allowed to make the change, you are probably missing a required mapping. The AF Link tool in PI SMT generates a security report to help pinpoint the problem, which might be caused by any of the following situations:

- You are trying to add access for a PI user, identity, or group that does not have an associated mapping. You need the mapping in order to associate the PI identity or account with a Windows account. Create the mapping and then try the edit again.

For example, suppose a module has the following access rights:

```
PIOoperators: A(r,w) | piadmins: A(r) | PIWorld: A(r,w)
```

PIOoperators and piadmins both have associated mappings (PIWorld does not need a mapping). Suppose you have an unmapped PI identity, called PIEngineers. You cannot add access permissions for PIEngineers until you create a mapping for that identity.

- You deleted a mapping that the module required to associate access permissions with a Windows account. Now, in MDB, you can still edit the module itself, but you cannot edit the access permissions for the module, except to delete the identity associated with that mapping. Create a new mapping for the relevant identity, user, or group. Now you can edit access permissions for the module.

Note: AF Link does not automatically pick up changes in mappings. The change does not take effect until you edit the element in some way; this triggers AF Link to update the settings in MDB.

For example, in the previous example, if you deleted the mapping for piadmins, you would not be able to edit access permissions on the module, except to remove all access permissions for piadmins. To fix this, you would have to create a mapping for piadmins.

Note: In both these cases, you can continue to edit the elements directly in PI AF.

Disabled a mapping identity but PI AF still allows access

AF Link does not recognize disabled mappings or identities (or users or groups). If you want to restrict existing access, make one of the following changes:

- Edit the access permissions on the element in PI AF
- Edit the access permissions on the module in PI MDB

Module and element hierarchies look different

There are several potential causes for differences in module and element hierarchies:

- Element attributes can be hierarchical, meaning that you can nest them inside each other. Module properties are also hierarchical, however aliases are not. If you nest attributes in a way that is not compatible with aliases, then you might not be able to see all the objects in PI MDB. See [Attribute to property and alias conversion](#).
- The PI AF security settings might not be compatible with PI MDB ([Element does not appear in PI MDB](#))
- PI MDB clients and PI AF clients determine parents in a different way.

- In PI MDB, the parent list is always the current parent list; it does not take query date into account.
- In PI AF, the parent list depends on the query date.

By default, the query dates are set to retrieve the latest versions of the assets (both in PI MDB and in PI AF). With the default query dates, there will be no differences in the parent list between the two versions of a client.

- If you changed the access for the PI AF element such that Everyone has *Deny* access, the element would not appear in the PI AF element hierarchy. However the module might still be visible in PI MDB. This is because when Everyone is denied, AF Link cannot read the element and therefore cannot synchronize it to PI MDB. To correct this, remove the Deny setting on the element.

Note: You must be an Administrator to make changes to the element when Everyone is set to *Deny*.

Force a remigration

When you remigrate from PI MDB to PI AF, you overwrite everything under the Data Archive element in PI AF with the contents in PI MDB. Remigrate only if you need to change the settings for the PI AF server, the replacement characters, or the replacement suffix.

Note: A remigration is different from a reset. When you reset, you copy from PI AF to PI MDB.

1. In PI SMT, open the AF Link tool (**Operation > AF Link**).
2. Click the toolbar icon to launch the preparation wizard.
Follow the prompts in the preparation wizard. The wizard has extensive help available if you are not sure about a step.
3. Stop the PI Base subsystem server. Make sure AF Link has stopped, it should be stopped at the beginning of the MDB to AF Migration wizard.
4. Delete **pimdbafmapping.dat** from the **PI\DAT** directory.
5. Start the PI Base subsystem and AF Link server.

Reset PI MDB

When you reset PI MDB, you recreate the PI MDB from the corresponding Data Archive element in PI AF. This means that you completely overwrite the entire contents of the existing PI MDB (excepting %OSI, %OSI_MCN, & %OSI_ManagedPI) and recreate it based on the current contents of the Data Archive element in PI AF.

Note: A reset is different from a remigration. When you remigrate, you copy from PI MDB to PI AF.

1. In PI SMT, open the AF Link tool (**Operation > AF Link**).
2. Select and show the details pane for the Data Archive server you want to reset (use to toggle the details pane).
3. Click the **Details** tab.
4. Click the **Reset MDB** button at the bottom of the pane.

Identify the Data Archive element

Each Data Archive server that is synchronized with PI AF has a corresponding Data Archive element in PI AF. To determine the name of the Data Archive element, follow these steps.

1. In PI SMT, open the AF Link tool (**Operation > AF Link**).
2. Select and show the details pane for the Data Archive server you want to reset (use  to toggle the details pane).
3. Click the **Settings** tab.

The PI AF server, Database, and Data Archive elements are listed at the top of the tabbed pane.

Change the PI AF server settings

You can change the PI AF server, Database or Data Archive element to which PI MDB is synchronized. If you change these settings, you must then re-migrate the Data Archive server. To change the settings:

1. In PI SMT, open the AF Link tool (**Operation > AF Link**).
2. Click the **Settings** tab.
3. At the bottom of the Settings pane, click the button to enable the settings for editing.
A Re-migrate button appears.
4. Make your changes and then click the **Re-migrate** button.
5. Follow the on-screen instructions.

Change replacement characters or suffix

If you change the [Replacement characters](#) or the [Replacement suffix](#), you must re-migrate the Data Archive server.

1. In PI SMT, open the AF Link tool (**Operation > AF Link**).
2. Click the **Settings** tab.
3. At the bottom of the Settings pane, click the button to enable the settings for editing.
A Re-migrate button appears.
4. Make your changes and then click the **Re-migrate** button.
5. Follow the on-screen instructions.

Replacement characters

PI MDB property and alias names can contain characters that are not valid in PI AF object names. When AF Link creates objects in PI AF, it substitutes invalid characters with replacement characters in the PI AF object name. Each character that is valid in PI MDB but not valid in PI AF has an assigned replacement character.

You can change the assigned replacement characters, but you need to force a re-migration for your changes to take effect (see [Force a remigration](#)).

Default replacement characters

This table shows characters that can be used in PI MDB, but not in PI AF. It also shows the default replacement character.

MDB Character that is Invalid in AF	Default Replacement Character (to be used in AF Name)
;	,
{	(
})
[<
]	>
\	/

Replacement suffix

In PI MDB, an alias and a property in the same module can have the same name. In PI AF, aliases and properties are both represented as attributes on an element. In PI AF, no two attributes on the same element can have the same name.

If an alias and property are at the same level and have the same name, then AF Link automatically appends a replacement suffix to the property name in PI AF. This substitution is only in the PI AF name. The PI MDB name does not change.

For example, suppose a module contains a property and an alias, both named *temp*. During migration, AF Link creates the following attributes in PI AF:

- *temp* (representing the alias)
- *temp_Prop* (representing the property)

The **Substitutions** screen lists the PI AF names that will end in the replacement suffix. If PI MDB contains an alias or property name ending in the replacement suffix, you get a suffix error.

Note: If you have a tree of hierarchical properties, then only the first level of property names must be distinct from alias names in that module. For example, a property nested five levels down can have the same name as an alias on that module.

Enable tracing

If you are having severe problems with MDB and AF synchronization, a Technical Support engineer might request that you turn on tracing in the AF Link tool.

Tracing runs for a set (configurable) period of time and you can specify where to send the output, either to the PI Message log or to a separate file in the **PI\Log** directory (same directory as the PI message log).

1. In PI SMT, open the AF Link tool (**Operation > AF Link**).
2. Select and show the details pane for the Data Archive server you want to reset (use  to toggle the details pane).
3. Click the **Details** tab.
4. Click the **Enable Tracing** button at the bottom of the pane.
5. Specify the length of time (in seconds) you want the tracing to run and specify where you want the output to go.

About tracing

Tracing runs for a configurable period of time. You can specify where to send the output, either to the PI Message log or to a separate file in the **PI\Log** directory (the same directory as the PI message log).

PI MDB and PI AF object conversion details

When you create an object in MDB, PI AF Link creates a corresponding object in AF; we call this converting an MDB object to an AF object.

Similarly, when you create an object in AF, PI AF Link converts the AF object to an MDB object. This section explains the details of the object conversion.

Module to element conversion

In PI AF, PI modules are represented as PI AF elements. The AF Link tool copies top level PI modules to PI AF as child elements of the Data Archive element. The hierarchy of the PI modules is maintained in the hierarchy of the corresponding PI AF elements. See [How PI MDB objects are represented in PI AF](#) for more information.

The following table shows how the properties of the module in PI MDB are represented in PI AF. .

PI MDB: Module	PI AF: Element
Description	Copied directly from the module
Comment	Copied directly from the module
Effective Date	Copied directly from the module
Obsolete Date	Copied directly from the module
Creator	The account from which AF Link gets its permissions on the PI AF server (PI AF Link access permissions). Not copied from the module.
Creation Date	Date that AF Link created the element. Not copied from the module.
Modifier	Initially, the account from which AF Link gets its permissions on the PI AF server (PI AF Link access

PI MDB: Module	PI AF: Element
	permissions). Not copied from the module.
Modified Date	Relative to the newly-created element. Not copied from the module.
Revision number	Relative to the newly-created element. Not copied from the module.
Versions	When a module is migrated, all versions of the module are migrated to PI AF.

How batch objects are migrated

PI Units are a type of PI Module and as such they are migrated to PI AF. When AF Link migrates PI Unit Modules to PI AF, it creates a PI AF Category named PIUnit and assigns this Category to all elements that represent migrated PI Units. You can access these PIUnit Elements in PI AF, just as you access Elements representing other PI Modules.

You cannot however, access batch data in PI AF. PI Units have associated batch data that is stored in a PI Point called a PI Unit Batch storage point. This batch data is not migrated. It remains in the PI Batch Database on the Data Archive server. After the migration, all your existing batch clients based on the Module Database continue to work as usual. However, PI AF-based applications cannot access the batch data.

Note: Batch Subsystem (BSS) Units are not migrated to PI AF. If you want to migrate BSS Units, then you must first convert them to PI Units.

Alias to attribute conversion

In PI MDB, a module can have one or more aliases. Aliases are represented in PI AF as attributes. An alias has a name and a reference to a PI point. During migration, AF Link creates an attribute to represent the PI alias as follows:

Attribute Name	The attribute in PI AF has the same name as the corresponding alias in PI MDB, unless the alias name contains a character that is invalid in PI AF. In that case, the attribute name uses a replacement character instead of the invalid character (see Replacement characters).
Data reference	Attributes can have a data reference. The attribute's data reference is the same PI point referenced by the PI alias.
Attribute value type	Attributes have an associated value type. The attribute value type in PI AF is determined based on the PI point referenced by the alias. See Alias to attribute type conversion for details.

IsBatchAnchored flag	An alias has a flag called <i>IsBatchAnchored</i> . This flag is not used by any of our products, but it can be used by custom applications. If an alias uses this flag (has the flag set to true) then AF Link creates a new PI AF category, named <i>BatchAnchored</i> , and assigns the category to that attribute in PI AF.
----------------------	---

Alias to attribute type conversion

In PI MDB, an alias references a PI point. When AF Link migrates an alias into an attribute, the value type of that attribute is determined based on the PI point type.

PI Point Type	AF Attribute Value Type
Blob	Anything
Digital	Anything
Float16	Single
Float32	Single
Float64	Double
Int16	Int16
Int32	Int32
String	String
Timestamp	DateTime

Property to attribute conversion

In PI MDB, a module can have one or more associated properties. In PI AF, the properties for each module are represented as attributes on the element. A property has a name and a value. The value has a data type. Properties can be hierarchical, meaning that you can nest one property under another one.

The following table shows how properties in PI MDB are migrated to PI AF.

PI MDB property	PI AF attribute
Name	The attribute name will be the same as the name of the corresponding property unless that name conflicts with an alias name. In that case, a suffix is added to the attribute name. See Replacement suffix for more information.
Value Type	The Value Type of the attribute is determined as described in Property to attribute value type conversion .
Hierarchy	The property hierarchy, if any, is preserved as the hierarchy of the attributes in PI AF.

Property to attribute value type conversion

The following tables show how the property value data type in PI MDB is converted to the attribute value type in PI AF. This first table shows simple data type conversions.

Property to attribute value type conversions (simple types)

PI Property Data Type	AF Attribute Value Type
Boolean	Boolean
Byte	Byte
Byte Array	Byte Array
Date	DateTime
Double	Double
Double Array	Double Array
Float	Single
Float Array	Single Array
Long	Int32
Long Array	Int32 Array
Short	Int16
Short []	Int16 Array

PI Property Data Type	AF Attribute Value Type
String	String

Some property data types convert to an attribute value type of <Anything>. If the attribute value type is <Anything>, then the property type also determines the attribute value in PI AF, as shown in the following table.

<Anything> value type conversions

PI Property Data Type	AF Attribute Value Type	AF Attribute Value
Null	<Anything>	Null
Object	<Anything>	Not Supported
PIAlias	<Anything>	PIAlias
PIHeading	<Anything>	PIHeading
PIModule	<Anything>	PIModule
PIPoint	<Anything>	PIPoint
PIProperty	<Anything>	PIProperty
PIUnitBatch	<Anything>	PIUnitBatch
PIValue	<Anything>	PIValue
Server	<Anything>	PIServer

Element to module conversion

The header information in a module is saved as header information in the corresponding element:

PI AF Element	PI MDB Module
Description	Copied directly from the element.
Comment	Copied directly from the element.
Effective Date	Copied directly from the element.
Obsolete Date	Copied directly from the element.
Creator	piadmin
Creation Date	The date that AF Link created the module. Not copied from the element.
Modifier	piadmin

PI AF Element	PI MDB Module
Modified Date	Initially, the date that AF Link created the module. Not copied from the element.
Revision number	Relative to the newly-created module. Not copied from the element.

Attribute to property and alias conversion

PI AF attributes are represented either as properties or as aliases in PI MDB. In general, an attribute with a PI point as the data reference is represented as an alias and all other attributes are represented as properties.

Note: Attributes with data references that are not PI points (for example, a data reference to a formula) are represented as properties of string type with a string value stating that the data reference is not supported by PI MDB.

With nested attributes, an attribute with a data reference to a PI point might be represented as a property, rather than an alias. This is because aliases are not hierarchical, meaning they cannot be nested as children of another alias. Properties can be nested, however.

- If you nest attributes below an alias attribute, then the child attributes are not visible to PI MDB clients.
- If you nest property attributes below other property attributes, then that hierarchy is carried over to the property hierarchy in PI MDB.
- If you nest alias attributes below property attributes, then those alias attributes are represented as PI properties of string type. The string value is a message that says that the data reference is not supported by PI MDB at this level.

Data type conversion for attributes to properties

The table below shows the conversion of attribute value types in to property data types in PI MDB.

PI AF Attribute Value Type	PI MDB Property Data Type
Boolean	Boolean
Boolean Array	Object Array
Byte	Byte
Byte Array	Byte Array
DateTime	Date
DateTime Array	Object Array
Double	Double
Double Array	Double Array

PI AF Attribute Value Type	PI MDB Property Data Type
Element	<i>Not Supported</i>
Enumeration Set	<i>Not Supported</i>
File	<i>Not Supported</i>
Guid	Object Array
Guid Array	Object Array
Int16	Short
Int16 Array	Short Array
Int32	Long
Int32 Array	Long Array
Int64	<i>Not Supported</i>
Int64 Array	<i>Not Supported</i>
Single	Float
Single Array	Float Array
String	String
String Array	Object Array
<Anything>	(depends on AF attribute value; see table below)

AF attributes can have a value type <Anything>. When <Anything> is the value type, then the PI property data type is determined by the attribute value, as shown in the following table.

AF Attribute Value Type	AF Attribute Value	PI Property Data Type
<Anything>	PIAlias	PIAlias
<Anything>	PIHeading	PIHeading
<Anything>	PIModule	PIModule
<Anything>	PIPoint	PIPoint
<Anything>	PIPProperty	PIPProperty
<Anything>	PIUnitBatch	PIUnitBatch
<Anything>	PIValue	PIValue

AF Attribute Value Type	AF Attribute Value	PI Property Data Type
<Anything>	Server	Server
<Anything>	<Null>	Null

Note: The PI properties listed as Not Supported are represented in MDB as a String Type PI property with the following value: Data Reference is not supported by PI Module Database at this level.

Access permissions required by AF Link

The PI AF Link Subsystem requires certain access permissions, both on the Data Archive server and the PI AF server, in order to perform migration and synchronization. The PI MDB to AF Synchronization tool in SMT and the PI MDB to AF Preparation wizard also require certain access permissions.

PI AF Link access permissions

PI AF Link requires access permissions on both the Data Archive server and the PI AF server. The PI MDB to AF wizard automatically configures this access, so you don't need to do anything unless you change the default settings.

- On the Data Archive server, PI AF Link runs as a Windows service called *PI AF Link Subsystem*. The account that the PI AF Link service runs under on the Data Archive server requires:
 - Read and write access to the Data Archive **dat** and **log** folders
 - Membership in the AF Link to PI Windows group on the PI AF server
 - Read and execute permissions to **%PISERVER%\bin** and **%PISERVER%\bin\piaflink.exe**

Note: Do not use the same user account to make direct changes to PI AF elements. If you use the PI AF Link user to edit elements, then your changes will not be synchronized with the PI Module Database.

- On the PI AF server, PI AF Link gets its user privileges as a member of a Windows group known as the AF Link to PI Windows group. The AF Link to PI Windows group requires:
 - Read, read data, write, write data, delete and admin access on the target AF Database and the Data Archive element
 - Read, write, delete and admin access on the AF Categories collection on the target database
 - Read, write, delete and admin access on the Identities and Mappings collections on the target PI AF server
 - That the account PI AF Link uses on the Data Archive server must be a member of the AF Link to PI Windows group

Note: If the Data Archive and PI AF servers are on different, non-trusted domains, you must manually configure PI AF server access.

About the Windows account for PI AF Link

By default, PI AF Link Subsystem service runs as the NETWORK SERVICE account on the Data Archive server. The PI AF Link migration tool configures the Data Archive and PI AF servers to give the NETWORK SERVICE account

administrative access to the Data Archive element hierarchy on the AF Server. It is a good idea to configure the PI AF Link service to run on a domain account instead.

Configure PI AF Link access for cross-domain deployments

If the Data Archive server and the AF Server are on different, non-trusted domains (or if either or both are on no domain) then you need to manually configure PI AF Link's access to the AF Server.

1. Create identical user accounts on the Data Archive server and AF Server.
 - The accounts must have the same password, as well as the same user name.
 - The Data Archive account requires read and write access to the Data Archive **dat** and **log** folders.
2. On the Data Archive server, set the PI AF Link service to run as the account you just created.
3. On the AF Server, add the account to the Windows group that the PI AF to MDB Preparation Wizard automatically created during the preparation process.

The name of this group is:

AF Link to PI - MyDataServerHostname

where *MyDataServerHostname* is the host name of the Data Archive server.

For example, if the host name is MyDataServerComputer, then the Windows group on the AF Server is named:

AF Link to PI - MyDataServerComputer

Changing the AF Link account

In the default configuration, the AF Link service runs under the NETWORK SERVICE account on the Data Archive server and the NETWORK SERVICE account gets administrative access to the Data Archive element hierarchy on the PI AF server.

If you do not want to grant this access to the NETWORK SERVICE account, edit the *PI AF Link Subsystem* Windows service to use a dedicated domain account. The account you use must meet these requirements:

- Read and write access to the Data Archive **P\dat** and **P\log** directories
- Member of the AF Link to PI Windows group on the PI AF server

Note: If you have multiple AF Link services synchronizing different Module Databases to the same PI AF database, you must run each of them under a different Windows user account.

The AF Link to PI Windows group

The AF Link to PI Windows group is the Windows group that the PI MDB to AF preparation wizard creates on the AF Server during configuration. PI AF Link gets its access permissions on the AF Server through this group.

By default, the full name of the AF Link to PI Windows group on the AF Server is:

AF Link to PI - MyDataServerHostname

where *MyDataServerHostname* is the host name of the Data Archive server.

For example, if the host name is MyDataServerComputer, then the Windows group on the AF Server is named:

AF Link to PI - MyDataServerComputer

Note: The account that PI AF Link Subsystem uses on the Data Archive server must be a member of the AF Link to PI group.

If the AF Server is on the Domain Controller, then you cannot use the default group and you must specify a domain account instead.

Access permissions required to run the preparation wizard

The PI MDB to AF Preparation wizard requires certain access permissions in order to prepare for migration.

When running the preparation wizard directly on the Data Archive computer itself, you need the following access permissions:

- On the Data Archive server, you need write access on the PIBACKUP and PIModules table in Database Security (in PI SMT, choose **Security > Database Security**).

If you don't have the correct access permissions, a pop-up dialog box appears showing a -10401 (no write access) error.

Note: On PI Server versions 3.4.370 or earlier, PIBACKUP is not required, but piadmin privileges are required. If necessary, the preparation wizard prompts you for the elevated credentials.

- When running the wizard remotely, you need the access permissions listed above, and you also need to be an administrator on both the remote Data Archive server and the local client. If necessary, the preparation wizard prompts you for the elevated credentials.

Access permissions required by AF Link

Depending on the task you are performing, the AF Link tool requires that you have database security access on both PIAFLink and PIModules. To edit database security access, use the Database Security tool in SMT (**Security > Database Security**).

Error messages

This table lists error messages that are related to migration and synchronization.

Message	What it Means	What to Do
MDB Object Name exceeded maximum allowed length	You tried to create an MDB object with a name longer than 259 characters.	For Properties, the allowed length is 259 characters minus the length of the replacement suffix. For example, the default replacement suffix is _Prop, which has 5 characters. With this setting, the maximum length of a Property name is 254 characters.
Linking these modules is prohibited	In MDB, you cannot create links between objects that are under the %OSI hierarchy and objects that are	You cannot perform this action.

Message	What it Means	What to Do
	not under the %OSI hierarchy.	
New effective date is inconsistent with effective dates on existing versions or it is newer than Obsolete Date	Effective dates and obsolete dates have to be consistent with dates on other versions.	Choose a date that is consistent with other versions.
At least one PI identity on the PI Module has zero mapped Windows principals	This means that PI MDB has access permissions for a PI identity (or PI user or PI group) that does not have a PI mapping. Every PI identity, user, or group that has MDB access permissions must have at least one PI mapping. If you delete the last mapping, you create a security conflict between MDB and AF.	Create a new mapping for the relevant identity, user, or group. Alternatively, delete the MDB access permissions for that identity, user, or group.
At least one Windows principal on AF Element is not mapped to any PI identity	This means that you added AF access permissions for a Windows account that does not have a PI mapping. This creates a security conflict between MDB and AF.	On the error messages that are related to migration and synchronization server, add a mapping for the Windows account. Alternatively, in PI AF, remove access permissions for that account.
At least one Windows principal is mapped to a PI identity on the PI Module but does not exist on AF Element security	This means that the PI module specifies access for a PI identity (or user or group) that is mapped to two or more Windows accounts; and one of those principals is not on the AF element security. This can happen if: An AF element is added to AF with only one of those principals on the security descriptor or One of those principals was removed after AF and MDB were in synchronized state.	To fix (in order of preference): <ul style="list-style-type: none"> • Add the missing principal(s) to the AF element. • Remove the remaining principal(s) from the AF element. • Edit the access permissions on the module to remove access for the PI identity. This will remove the remaining principals from AF element. <p>Note: PI AF Link does not automatically pick up changes in mappings. The change does not take effect until you edit the element in some way; this triggers PI AF Link to update the settings in MDB.</p>

Message	What it Means	What to Do
Permissions for one or more Windows principals on the AF Element do not match permissions on the PI Module	This means that the AF element uses AF security options that are not reflected on the Data Archive server.	Manually edit the access permissions to match or reset the MDB.
Using default Acl for PI Module.	This means that PI AF Link is now using the default Access Control List (ACL) to control access to the Module. Two possible reasons: <ul style="list-style-type: none"> None of the Windows accounts that have access to the object in AF have PI mappings. For at least one Windows account that has access to the object in AF, you have used deny access. 	If the problem is deny access in AF, remove the deny access. If the problem is no mappings, create a mapping.
Failed to Connect to AFServer	For some reason, PI AF Link was unable to connect to the AF Server.	Check that the AF Server is up and that the Data Archive server can connect to it. Check Windows EventViewer. It has a separate entry for AF and all messages for AF are logged here.
Module not yet created in AF. Retry.	Added module but it's not yet in AF. This could happen if you added many modules and PI AF Link is trying to create them all in AF. It might take a while to process all the changes in AF.	Retry the operation. If the problem doesn't resolve itself within five minutes, contact Technical Support using the Customer Portal .
Failed to edit MDB-AF mapping table	You were running the offline piaflink mapping table command and the operation failed for some reason.	Wait a minute and retry the command. If the problem doesn't resolve itself within five minutes, contact Technical Support using the Customer Portal .
Failed to read module ids from AF	You were running the offline piaflink mapping table command and the operation failed for some reason.	Retry the command. Wait a minute and retry the command. If the problem doesn't resolve itself within five minutes, contact Technical Support using the Customer Portal .

Message	What it Means	What to Do
Error during remigration. Current PI Server Element is different from the previous run. Perform the remigration steps again.	When you did the remigration, you neglected to delete the mapping table. This old mapping table has configuration information that is no longer current.	Delete the mapping file from the PI\Dat directory (the file is pimdbafmapping.dat) and then restart PI AF Link.
AF Link subsystem is not ready to allow module edits	PI AF Link does not allow an MDB edit at this time. This could happen for any number of reasons, such as PI AF Link not running, unable to connect to AF Server, and so on. This state always occurs when you start PI Base Subsystem or PI AF Link. It will take about 15-30 seconds after the later of the two subsystems starts before module database can be edited even if there are no problems.	If PI AF Link or PI Base Subsystem just started, then wait a few seconds after both are up and running, then retry. If that is not the problem, then in the PI SMT AF Link tool, check synchronization status to find out what the problem is.
Attempted to connect to an unknown AF Server	Specified an AF Server and for some reason not found in the list of AF Systems.	Use the PI System Explorer to add it to the list: <ul style="list-style-type: none">• Click Database.• Click ...• Right-click on any system in the list.• Add the new system. See PI System Explorer help for details.
AF and MDB are out of sync	You cannot edit MDB when AF and MDB are out of sync.	In the PI SMT AF Link tool, check synchronization status to find out what the problem is.
Handshake already in progress. Will retry.	AF Link connects to the AF Server every 5 seconds (handshake), but can only do this once at a time.	Wait a while if possible. PI AF Link will re-try, so the error might go away with time. If that does not work, try to restart PI AF Link Subsystem (you can do this in Windows Services) If the problem persists, contact Technical Support using the Customer Portal .
Checkout of AF Object failed.	You are trying to edit an object in	Wait and then retry. If you still have

Message	What it Means	What to Do
	MDB but PI AF Link is not able to check out the corresponding AF Object. Someone else might have it checked out, or the same user has another version checked out, or the AF Object type is not supported for checkout. Could also be a connection or access permissions problem. If you get this message, then PI AF Link cannot determine the exact problem.	a problem, look at the suggestions for the following three errors in this table.
AF Object checked out by another user	You are trying to edit an object in MDB but PI AF Link is not able to check out the corresponding AF Object; someone else might have it checked out.	Wait until the AF object is checked in and then retry.
Another version of AF Object already checked out	You are trying to edit an object in MDB but another version of the object is already checked out.	Wait until the AF object is checked in and then retry.
AF Object type is not supported for checkout	You are trying to edit an object in MDB but PI AF Link is not able to check out the corresponding AF Object because it is not supported for checkout in AF. PI AF Link only checks out AF Elements, no other object type.	You cannot perform this type of edit in the MDB.
Required configuration is missing or incorrect	The preparation wizard says that it is complete, but PI AF Link cannot find some required configuration information.	<p>Try again in five minutes. If that doesn't work, check the message log to find out what is missing. Run the wizard again.</p> <p>Note: If you can figure out what is missing, you might be able to manually fix it.</p>
AF Database Checkin Failed	PI AF Link could not check in to the AF Database.	Do nothing. PI AF Link will retry the check in. The error should go away in time.
Complete the PI MDB to AF Preparation Wizard to initiate migration	You ran the preparation wizard but it is not complete. Either you didn't put in all the information, or you need to analyze the Data Archive server again.	Run the preparation wizard and make sure the wizard is complete. You should see a success message on the final screen.

Message	What it Means	What to Do
PI Server was moved from one machine to another.	Data Archive was moved from one machine to another.	If you intentionally moved the Data Archive server, and you want to keep the same target AF Server, then be sure to disable PI AF Link on the original Data Archive server. You can do this by disabling the PI AF Link Subsystem Windows service. If you intentionally moved the Data Archive server, but you want to change the target AF Server, then re-migrate.
MDB AF synchronization encountered an error.	This could be for any number of reasons.	Do nothing. PI AF Link will retry. The error should go away in time. If not, look in the PI message log and in the Details tab of the AF Link tool in PI SMT for more information.
MDB migration in progress. MDB cannot be edited	PI AF Link is migrating MDB to AF.	Wait until the migration is complete and try again.
Processing changes made in AF Database. Retry later.	A large number of changes were made in AF and PI AF Link is busy processing the changes for MDB.	You should retry the operation later.
Another thread is still reading AF Changes. Will retry.	An internal error.	Do nothing. PI AF Link will retry. The error should go away in time. If not, contact Technical Support using the Customer Portal .
AF Link is busy making other AF changes. Retry later.	PI AF link is busy making other changes in AF and cannot process this operation.	Retry the operation later.
MDB cannot be reset from AF. Run the migration wizard to re-migrate MDB to AF.	The system is in a state where a re-migration is not allowed.	Run the migration wizard to re-migrate MDB to AF.
Failed to add AF Element	PI AF Link tried to add an element in AF and the operation failed.	Retry the operation. If that doesn't work, look in the message log for more information.
Failed to edit AF Object	You added or edited something in MDB but PI AF Link could not create or edit the corresponding object in AF. The health goes out of sync when this happens and the	This is not an error you can fix. Retry the operation later. If that doesn't work, look in the message log for more information. The reason for this failure will contain

Message	What it Means	What to Do
	error code for it will be this error number.	information on exactly which element failed and maybe why it failed.
Does not have required privileges on AF Object	PI AF Link does not have the necessary privileges to edit the object in AF.	Restart PI AF Link. If that does not work, then make sure that the AF Link to PI Windows group on the AF Server has the necessary permissions and that the account under which PI AF Link is running is a member of that group.
Target AF Database or the PI Server Element was deleted or not found.	The Data Archive element or the database to which it belongs, was deleted; or the configuration file doesn't match.	The best solution is to run the MDB to AF preparation wizard. If you do not want to do that, you can: <ul style="list-style-type: none"> Manually recreate the database or element, then restart PI AF Link. Restore the database or element from a backup, then restart PI AF Link.
Target AF Database was not found.	The database to which the Data Archive element belongs was deleted or PI AF Link can't find it.	The best solution is to run the MDB to AF preparation wizard. If you do not want to do that, you can: <ul style="list-style-type: none"> Manually recreate the database or element, then restart PI AF Link. Restore the database or element from a backup, then restart PI AF Link.
AF Element was derived from a template. PI Properties and PI Aliases cannot be edited in MDB. Edit the attributes in AF.	You are trying to edit properties or aliases on a module, but the corresponding AF element is derived from a template.	You cannot perform this operation. If an AF element is derived from a template, then you cannot edit the properties or aliases on the corresponding module in MDB. Your only option is to remove the template from the element in AF.
MDB and AF Database are not restored from the same backup	MDB and AF Database are not restored from the same backup.	Restore MDB and the AF Database from same backup or reset MDB.
PI_MDBAFTPREP_COPYANDLOADFAI	This indicates that we were unable	Contact Technical Support using the

Message	What it Means	What to Do
LED	to successfully make a backup copy of the files that the validation uses (pimoduledb.dat, piacl.dat, piiidentity.dat, etc) and/or we were unable to load these files (possibly indicating they are corrupt).	Customer Portal .
PI_MDBAFFREP_VALIDATIONFAILED	This indicates that the validation of the module database unexpectedly quit. At this point, we don't know if the MDB is actually ready or not for migration.	Contact Technical Support using the Customer Portal .
InSync	MDB and AF are in sync.	No action needed.
SyncInProgress	PI AF Link Subsystem is currently synchronizing the PI Module Database and the AF Server.	No action needed.
OutOfSync	The PI Module Database and the AF Server are out of sync.	Reset MDB.
BaseToAF LinkCommFailure	PI Base Subsystem has a communication error with PI AF Link Subsystem.	Start the PI AF Link Subsystem Windows service, if it is not running. If PI AF Link is running, then this error should resolve itself over time, typically within minutes.
AFLinkToBaseCommFailure	PI AF Link Subsystem has a communication error with PI Base Subsystem.	Start the PI Base Subsystem Windows service, if it is not running. If the service is running, then this error should resolve itself over time, typically within minutes.
AFLinkToAFCommFailure	PI AF Link Subsystem has a communication error with the AF Server.	If the PI AF server is not running, start it. If the SQL Server is not running, start it. If both are running, check the log file for any other error, such as incorrect access permissions, wrong AF Server name, and so on.
CheckinFailure	There was a check-in failure when checking in changes to the AF Server. Most likely another user has the object checked out.	Do nothing. PI AF Link will retry the check in. The error should go away in time.
PI AF Link Subsystem is not running.	This means that PI Base Subsystem	Start the PI AF Link Subsystem

Message	What it Means	What to Do
	is running but that PI AF Link is not.	Windows service.
Health status is bad for the Base subsystem	The PI AF Link subsystem status and the PI Base Subsystem status do not match. PI Base Subsystem has bad health status.	On the details pane, find the health status for PI Base Subsystem. Take the appropriate action for that health status message.
Health status is bad for the PI AF Link subsystem	PI AF Link Subsystem status and PI Base Subsystem status do not match. PI AF Link Subsystem has bad health status.	On the details pane, find the health status for PI AF Link Subsystem. Take the appropriate action for that health status message.
PIServerMoved	Data Archive server was moved from one machine to another.	<ul style="list-style-type: none"> If you intentionally moved the Data Archive server, and you want to keep the same target AF Server, then be sure to disable PI AF Link on the original Data Archive server. You can do this by disabling the PI AF Link Subsystem Windows service. If you intentionally moved the Data Archive server, but you want to change the target AF Server, then re-migrate.
TargetDatabaseDeleted	Target AF Database or Data Archive element was deleted or not found.	<p>You have a few options:</p> <ul style="list-style-type: none"> Re-migrate MDB to AF. If it is the Data Archive element that is missing, create the element in the database and then wait for the synchronization to pick up the changes. Manually recreate the database or element, then reset MDB to match. Restore the database or element from a backup, then reset MDB to match.
DBAFRestoreInconsistent	MDB and AF Database are not restored from the same backup.	Restore MDB and the AF Database from same backup or reset MDB.

Message	What it Means	What to Do
FailedToReadMDBEvents	Failed to read MDB Events from update queue.	Do nothing. PI AF Link will retry and this error should go away with time.
Failed to Reset MDB from AF Database	Failed to Reset MDB from AF Database	Try to reset MDB again. If it continues to fail, contact Technical Support using the Customer Portal .
GenericBadState	An error exists that does not fall into any category in this table.	<p>Wait five or ten minutes if possible. PI AF Link will retry, so the error might go away with time. Otherwise:</p> <ul style="list-style-type: none"> First look for possible causes of the problem. Check the Details pane. Open the Message Logs tool and look for error messages from piaflink or pibasess. If that does not help you find the problem, try to restart PI AF Link Subsystem (you can do this in Windows Services) If the problem persists, contact Technical Support using the Customer Portal.
InsufficientPermissionsToAF	PI AF Link Subsystem does not have enough permissions on AF Database to continue.	<p>Check that the PI AF Link Subsystem service has the following access permissions:</p> <ul style="list-style-type: none"> Read access on AF Databases collection and AF Elements collection on the target database. Read-write access on the target AF Database, Data Archive Element and AF Categories collection on the target database.
This PI Server is running a version earlier than PI Server 2010. The MDB to AF migration wizard has not yet been run.	This means that you have not yet run the PI MDB to AF preparation wizard on this Data Archive server.	If you want to upgrade this Data Archive, you must first complete the preparation wizard. The wizard is included in PI SMT 2010 and later

Message	What it Means	What to Do
		versions, which you can download from the Customer Portal .
This PI Server is running a version earlier than PI Server 2010. The MDB to AF migration wizard has not yet been completed.	This means that you have run the PI MDB to AF preparation wizard on this Data Archive server but that you have not completed it.	If you want to upgrade this Data Archive, you must first complete the preparation wizard.
This PI Server is running a version earlier than PI Server 2010. Migration preparation is complete, upgrade to PI Server 2010 to enable PI AF Link.	This means that you have successfully run the preparation wizard on this Data Archive server and it is ready for upgrade.	Upgrade when you are ready.

Alarm Groups in PI SMT

Use the Alarm Groups tool to view, create, organize, and delete alarms generated by PI Alarm Subsystem. Data Archive stores historical data about these alarms in digital PI points called *alarm points*.

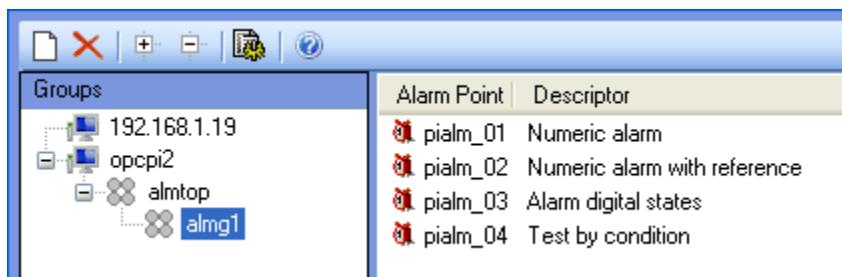
You organize alarm points by placing them in *alarm groups*. Use the Alarm Groups tool to move PI alarm points and SQC alarm points among groups.

Note: These alarms supplement, but do not replace, alarm capabilities from plant control systems.

View alarm groups and points

Each connected Data Archive server appears in the **Groups** panel, where you can navigate the list of servers and groups.

1. Expand a server icon by clicking the plus sign (+).
Alarm groups are clustered under their respective Data Archive servers.
2. Click the alarm group name to list the alarm points that belong to the selected group.



Each alarm point is listed in the **Alarm Point** list. The alarm point name and descriptor appear in columns.

Change alarm groups and points

In the **Groups** panel, you can add or delete alarm groups, or change alarm point options.

In the **Groups** panel, right-click an alarm group and select one of the following:

Select this option:	To do this:
New Group	Add a new alarm point to the selected server
Remove Group	Remove the selected alarm group
Options	Change alarm point or alarm group point source characters

Alarm groups and points status

These icons indicate the status of alarm groups and points:

Icon	Status
	Successfully connected to a Data Archive server; able to retrieve alarm groups
	Unable to connect to a Data Archive server; unable to retrieve alarm groups
	PI alarm group
	PI or PI SQC alarm point

Specify point sources

The first time you connect to a server, the Alarm Groups tool prompts you to specify point source characters corresponding to alarm points, SQC alarm points, and alarm groups on the Data Archive server. Default point source characters are supplied to match the server default characters, and for most Data Archive servers you only need to acknowledge these settings. If the point sources change on the Data Archive server, then you must change these point sources to match.

Do one of the following:

- Click the **Change alarm point sources** button
- Right-click the server and choose **Options**.

Add alarm groups to a Data Archive server

- In the Servers pane, select a server.
- In the Groups pane, select the server.
- Click **Create a New Alarm Group** .

Note: You can only add alarm groups to the primary server of a collective.

- In the Alarm Group Definition dialog box, enter a group name and optional descriptor for the new group, then click **OK**.
- Verify that the group appears in the **Groups** panel under the server icon.
See [Alarm groups and points status](#).

Organize alarm groups and points

Once created, you can reorganize the alarm-group structure and reassign groups and points to different

locations.

Note: You cannot place alarm groups on secondary nodes of a Data Archive collective. You can only add alarm groups to primary servers.

1. Use the following techniques to organize alarm groups and points:

- **Move a group.** To move an alarm group between servers or groups, drag the group and drop it onto the desired parent group or server node. You can create a multi-level hierarchy of alarm groups to represent actual plant configurations.
- **Move a point.** To move an alarm point, drag and drop the alarm point from the **Alarm Point** list to a new group in the **Groups** panel.
- **Delete a group.** To delete an alarm group, select the group in the server tree and click , or right-click the group and choose **Remove Group**.

Alarm groups reference

Goal	Right-click option	Toolbar icon
Add a new alarm to a selected server	New Group	
Delete a selected alarm group	Remove Group	
Expand all alarm groups for all server nodes	None	
Collapse all alarm groups belonging to a server node	None	
Specify point sources for alarm points and groups on a selected server when you first connect or change the point source later	Options	

Note: Right-click and toolbar options are enabled based on the item selected in the Alarm Groups window and its current status.

Archive Editor

Use the Archive Editor tool to view, edit, insert, and delete values for PI point events in a PI archive. To manage archive files, use the Archives tool.

Search for archived events

You can enter specific point attributes and conditions to search for and retrieve events and associated annotations from the PI archive.

1. In the Servers pane, select one or more Data Archive servers.
2. In the System Management Tools pane, select **Data > Archive Editor**.
3. In the right pane, from the **Server** list, select a Data Archive server.
4. In **Tagname**, enter the point name and press Enter, or click **Tag Search** and use the Tag Search dialog box to select a point. The point's values appear.
5. Use the remaining fields in the Archive Editor tool to set the values displayed. You can:
 - [Define a time, event count or event range](#)
 - [Filter search results](#)
 - [Define a boundary type](#)

The list refreshes when you select an item in a drop-down list, enter a value in a text field, or click **Get Events**

To look at values for other points without clearing the currently displayed values, click **New Tab**

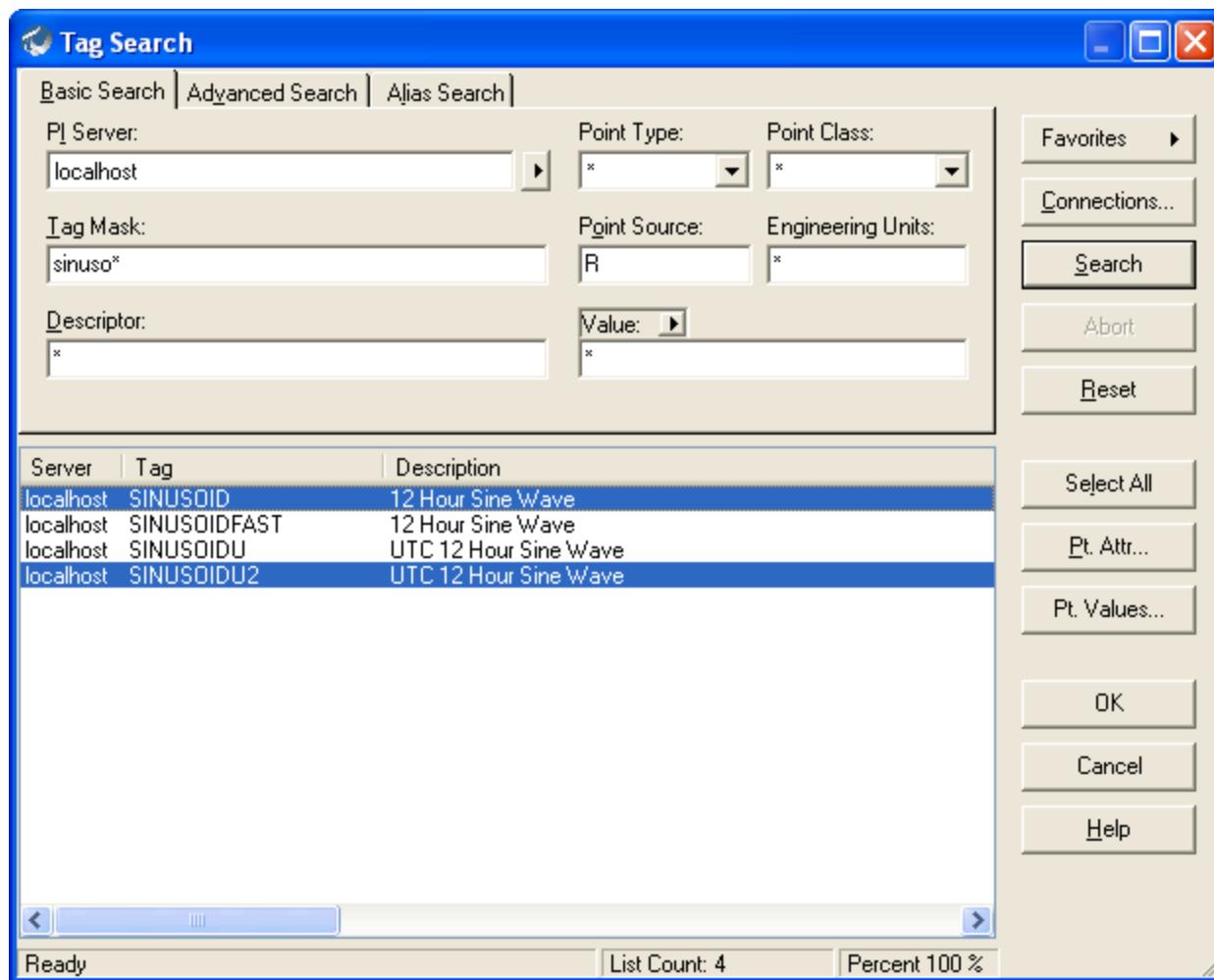
Note: When adding values to an archive, you can protect existing event values by specifying the **Merge Type**. For details, see [Protect existing archive values](#).

Use Tag Search

Click the **Tag Search** button to open the Tag Search dialog box.

The Tag Search dialog box provides three types of searches for historical or future PI points:

- **Basic Search** allows you to create a tag mask by specifying PI point attributes. The mask is used to find a list of PI points on the server with matching attributes.
- **Advanced Search** provides a query-building interface with access to more point attributes for complex searches.
- **Alias Search** provides a logical tree view of a Data Archive through the PI Module Database, which you can use to select tags by their descriptive aliases.



1. Click a tab to choose a **Basic**, **Advanced** or **Alias** search.
2. Enter the required search criteria and click **Search**.

Use * or ? as wildcard characters to search for tag names and attributes. For example, the tag mask *Tem** returns all point names that start with *Tem* while *Tem?* returns only points that start with *Tem* and end with another single character. All point mask fields are case insensitive.

You can also click **Favorites** to access previous searches.

3. Tags returned from a search appear listed in a search results panel. Select the desired tags in the results panel, and click **OK**.

Click column headers in the search results panel to sort the results. Press Ctrl+click or Shift+click to select multiple tags.

Define a time, event count or event range

You can limit the number of events displayed by defining a time range or a specific count of events.

Define a time range

To limit the number of events displayed by the search, you can define a time range by exact or relative start and end times.

1. Enter a time string in the **Start Time** field.
2. Select the **End Time** option.
3. Enter an end time string in the second field.

After a time range is set, you can scroll to change the times for the range with the arrow buttons to the right of the **End Time** and **Event count** fields.

For example, with a time range set between 6:00 AM and 8:00 AM, click the left arrow to shift the time range between 4:00 AM and 6:00 AM. Click the right arrow to shift to 8:00 AM to 10:00 AM.

Note: The latest time you can use is 10 minutes ahead of the current time. If you enter a later time, Archive Editor changes it to the current time plus 10 minutes, and shifts the other time string as needed to maintain the time range you specified.

Define an event range

To limit the number of events displayed by the search, you can define an event range with a start time and a number of events.

1. Specify a time in the **Start Time** field.
2. Select **Event count** to indicate an event count range.
3. Enter the maximum number of events to retrieve in the second field.

Use negative numbers to show events that count back in time from the start time, and positive numbers to count forward. Use the arrow buttons to change the reference time by the number of events.

Protect existing archive values

A properly selected merge type can protect existing events when you add new events to an existing archive. The merge type determines how the merge process handles multiple events with the same timestamp when merging new events into an existing archive. The Archive Editor tool offers six possible values of **Merge Type**:

- **Replace Duplicates**

Replaces an existing event with the same timestamp (if present) and sets its Substituted attribute to True. Otherwise, adds a new event.

- **Insert Duplicates**

Adds a new event, but does not affect existing events with the same timestamp (if present).

- **Error for Duplicates**

Displays an error message if there is an existing event with the same timestamp, but does not replace that event or add a new event.

- **Replace only Duplicates**

Overwrites one of the existing events with the same timestamp (if present) and sets its Substituted attribute to True. Otherwise, displays an error message.

- **Reject Duplicates Without Checking for Errors**

Adds the new event only if there is no existing event with the same timestamp. Otherwise, PI Archive Subsystem writes an error to the PI message log on the Data Archive server, or an error message may be displayed.

- **Replace Only Duplicates Without Checking for Errors**

Overwrites one of the existing events with the same timestamp (if present) and sets its Substituted attribute to True. Otherwise, PI Archive Subsystem writes an error to the PI message log on the Data Archive server, or an error message may be displayed.

Define a boundary type

Specify a boundary type to determine how the Archive Editor tool searches for data values near the start and end times of the time or event range (see [Define a time, event count or event range](#)).

- **Inside (default)**

Returns events at start and end times, if they exist, or the nearest events occurring within the range.

- **Outside**

Returns the closest events occurring immediately outside the range.

- **Interpolated**

Returns interpolated events at start and end times.

- **Auto**

Interpolated, but using **Inside** behavior for tags with step attributes set on Windows-based Data Archive servers.

Filter search results

Procedure

1. In **Filter Expression**, type an expression using Performance Equation (PE) syntax.
2. In **Show Filtered**, select a filtering option:
 - **Show Filtered Values:** Substitute **Filtered** as a placeholder for each event or block of events filtered, based on the filter expression.
 - **Remove Filtered Values:** Remove filtered events from the list, showing only events that do not match the filter expression criteria.
 - **Use Expression Times:** Return all data including and between start and end times. This is the default setting. This may be affected by how you define a boundary type.

3. Click the **Get Events** button .

Example of filter expression using Performance Equation (PE) syntax

In this example, to filter the results to show only the events where the temperature (represented by the tag **TempSensor**) is above 75°F and the equipment status (represented by the tag **EquipStatus**) is running (represented by the **value 1**), use the following filter expression:

```
TempSensor > 75 AND EquipStatus = 1
```

Procedure

1. Enter a filter expression.

In the **Filter Expression** field, type the following expression:

```
TempSensor > 75 AND EquipStatus = 1
```

2. Select a filtering option.

- **Show Filtered Values:** This replaces each event or block of events that match the filter criteria (**TempSensor > 75** and **EquipStatus = 1**) with the placeholder **Filtered**.
- **Remove Filtered Values:** This removes the events that match the filter criteria from the list, showing only the events where the temperature is 75°F or below, or the equipment status is not running.
- **Use Expression Times:** This returns all data including and between start and end times where the filter expression criteria are met.

Retrieve filtered events

Click the **Get Events** button  to apply the filter and retrieve the filtered search results based on the selected filtering option.

Explanation

- **TempSensor > 75:** This part of the expression filters events where the temperature is greater than 75°F.
- **AND:** This logical operator ensures that both conditions on either side of it must be true for an event to be included in the filtered results.
- **EquipStatus = 1:** This part of the expression filters events where the equipment status is running.

View and edit archived event values

You can view and edit events in the archived events list at the bottom of the Archive Editor tool.

- Each row contains five columns, including the event **Value** and matching **Event Time**, and three attribute columns.
- Events are listed chronologically unless the search is a negative event count (see [Define a time, event count or event range](#)).
- All events occurring within the specified range appear unless the results are filtered (see [Filter search results](#)).
- The following attributes may be flagged by edits to an archived event:
 - **Questionable**

The event value is unreliable or the circumstances under which it was recorded are suspect.

- **Annotated**

An annotation has been made to the event to include further information or commentary.

- **Substituted**

The event value has been changed from the original archived value.

Across the top, the Archive Editor tool displays these statistics:

- **Event Count**

The full count of events that match the search criteria.

- **Retrieved**

The actual number of events retrieved and displayed, excluding filtered events.

- **Row**

The number of the currently-selected row.

Note: The Archive Editor tool can appear in a host application such as PI ProcessBook in read-only mode, in which case all edit features are disabled.

Edit an archived event value

You can edit some event attributes directly in the archived events list of the Archive Editor tool:

- Right-click a value that you want to edit, then select **Edit Value**. Once you save the updated value, Data Archive flags the updated event in the Substituted column.
- Select the check box in the Questionable column to flag a questionable event.

Other attributes cannot be changed directly in the archived events list:

- The Event Time for an archived value cannot be changed. If an event is needed for a certain time, add a new event instead. (See [Add an event to the archive](#).)
- Data Archive flags an event in the Annotations column once an annotation is saved for the event. Click the **Annotations** button  to annotate an event in the PI Annotations Maintenance dialog box.

Click the **Save** button  to save event updates to the PI archive.

Refresh the list of archived events

If you change any of the parameters used to retrieve data, you may need to refresh the list of archived events in the Archive Editor tool:

- If you change **Start Time** or the **End Time** or **Event count** parameters, the archived events list does not refresh. Click the **Get Events** button  or press ENTER.
- If you change the **Server** parameter, both the **Tagname** field and the archived events list are cleared. Specify search criteria, then click the **Get Events** button  or press ENTER.

- If you change **Tagname**, **Boundary Type**, or **Show Filtered** parameters, the archived events list refreshes automatically.

Add an event to the archive

Note: Before you add a new event, protect existing archive values with a merge type that indicates how simultaneous events are managed when new values are merged into the archive. See [Protect existing archive values](#).

Procedure

1. Scroll down to the first empty row (containing the * character) and enter event attributes.
 - Enter an event value in the Value column.
 - Enter a matching timestamp in the Event Time column.
 - Select the check box in the Questionable column to denote the event as questionable.
2. Click the **Save** button  to save event updates to the PI archive.

Delete an archive event

The following procedure is useful for deleting a single archive event or a small number of archive events. To delete large amounts of data, use the AF SDK ReplaceValues methods. For more information, see the AF SDK Reference topics `AFData.ReplaceValues Method` and `PiPoint.ReplaceValues Method`.

You cannot undo a delete. Deleting large amounts of data may affect availability of an online Data Archive server. If you are deleting data on a periodic basis, investigate the root cause and consider alternatives to deleting data in bulk. Possible alternatives could be using appropriate exception and compression settings, increasing the available disk space, moving older archives to a second tier storage, and so on.

Before you begin deleting real data, we strongly recommend that you practice deleting small amounts of data on a test system.

1. Back up your data.
2. In the Servers pane, select the server from where you want to remove data.
3. In the System Management Tools pane, select **Data > Archive Editor**.
4. Right-click the value you want to delete and select **Delete**.
5. Click **Save**.

Note: There is no prompt to confirm deleting values.

Annotations in PI SMT

Use annotations to associate arbitrary information, such as text comments and other binary data, with a PI archive value.

Use the PI Annotations Editor in the PI SMT Archive Editor to view, edit, insert, and delete annotations to PI point values. Annotations can include comments, notes, supplementary values with specified data types, and files.

Every value in the snapshot or the archive may be annotated. Annotated events always bypass compression. An

annotation can be of any binary data type. The size of an annotation is controlled by the *Snapshot.AnnotationSizeLimit* tuning parameter.

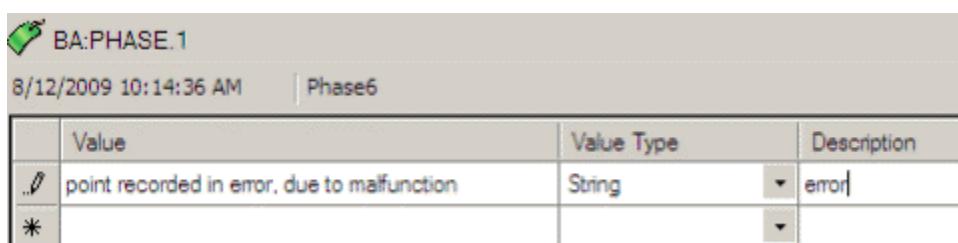
Each archive file has a single associated annotation file, with an **.ann** extension. The annotation file is created if it does not exist. It is important to store archive and annotation files together, especially when a backed up archive file is restored.

Note: Any operation on an annotation translates into an actual I/O, bypassing archive caching. Annotated events are less efficient than non-annotated events.

You can use the following modes to maintain annotations.

- **Standard/Default mode**

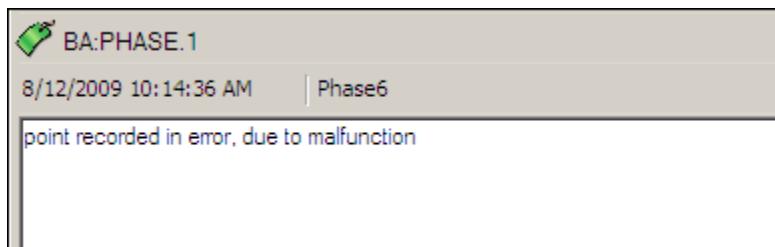
Provides a table format that can include alternate values with assigned data types. Use this mode for annotation data that is structured, read programmatically, or exported for use by another application.



	Value	Value Type	Description
...	point recorded in error, due to malfunction	String	error
*			

- **String/VARIANT mode**

Stores annotation data as an unspecified **VARIANT** data type. Use this mode for simple string annotations, annotations that do not require structured data, and to conform with legacy annotations from earlier versions of Data Archive.



Add or edit annotations

1. Select an event in the archived events list and click the **Annotations** button , or right-click the value and select **Annotations**.
2. In the PI Annotations Maintenance window, modify the following information, or enter new rows containing:
 - a. Point values or any other data that requires a specified data type in the Value column. If your annotation consists only of a string, enter it in the Value column.
 - b. Data types to match corresponding values in the Value Type column. Value Type is set automatically, and should be changed only if it is incorrect.

	Value	Value Type	Description
▶	This is a test string	String	
	1/1/1990 12:00:00 AM	PITime	
*			

Created: 10/12/2005 11:59:56 AM OSISOFT.INT\HTalvala Modified: 10/12/2005 1:22:47 PM OSISOFT.INT\HTalvala

If you need to change a Value Type, select one of the following.

- a. String (default type)
- b. Byte, Short, Long
- c. Single, Double
- d. Boolean
- e. PITime, DateTime

Other data types displayed are for internal use, and cannot be used for annotations.

- c. Related information and secondary annotations in string format in the Description column.
3. Enter as many rows as necessary and click **Save**.

Import a file to an annotation

1. Select an event in the archived events list and click the **Annotations** button , or right-click the value and select **Annotations**.
2. In the PI Annotations Maintenance window, set the Value Type to **File**, and click in the **Value** cell.
3. Click the **Import** button .
4. Select a file and click **Open**.
5. Click **Save**.

Export a file from an annotation

Once a file has been imported to an annotation, it can be exported.

1. Select an event in the archived events list and click the **Annotations** button , or right-click the value and select **Annotations**.
2. In the PI Annotations Maintenance window, select the annotation with the file and click the **Export** button , or right-click the annotation and click **Export**.
3. Navigate to the folder where you want to save the file.
4. Click **Save**.

Archives tool

The Archives tool lets you administer PI archive files and manage how Data Archive uses archive files.

To edit the data contained in the archive files, use the Archive Editor tool.

Note: The Archives tool can only display archives for Windows-based Data Archive servers.

Archives tool

The Archives tool displays a list of registered archives for each connected Data Archive server. Historical and future archives are listed under separate tabs.

To change the location where Data Archive creates archive files, whether historical or future, select the **Review and update parameters** tab.

The archive list contains columns that describe the status and properties of each archive. Toolbar functions and a context menu allow you to monitor and manage archive use. Right-click any file in the archives list to display the following actions.

Menu item	Action
Create new	Create a new archive.
Register archive	Browse to an archive file and register it.
Unregister archive	Un-register the selected archive.
Display unregistered archive	Display an un-registered archive's information.
Make writable	Make the currently selected archive writable.
Make read-only	Make the currently selected archive read-only.
Make shiftable	Make the currently selected archive shiftable. If the currently selected archive is dynamic, this option will not be enabled.
Make non-shiftable	Make the currently selected archive non-shiftable.
Force shift	Force an archive shift.
Reprocess archive	Reprocess an archive while it is online.
Backup	Backup the selected archive.
Refresh	Refresh the archive list from the selected server.
Properties	Display the archive properties for the selected archive.

Note: The archive file that you select determines which options are available; some may be enabled or disabled.

Archives toolbar

Most archive management tasks can be completed using the toolbar. Toolbar icons are enabled or disabled according to the item selected and its current state.

The toolbar contains the following command buttons:

Icon	Meaning
	Create a new archive.
	Register an archive.
	Unregister the selected archive.
	Reprocess an archive.
	Display an unregistered archive's information.
	Force an archive shift.
	Backup the selected archive.
	Display the Options dialog box. See Show or hide archive gaps .
	Export all displayed archive information to a .CSV file.
	Refresh the selected server's list of digital state sets.
	View help topics for archive management tasks.

Archive properties

Right-click an archive and choose **Properties** to see a list of the following information for an archive:

- **Archive File** name, which includes the full path and drive letter relative to the server machine.
- The **Data Archive** network name or IP address.
- The name of the **Collective**, if applicable, that the Data Archive belongs to.
- The current **Status** of the archive, either Primary, Has Data, or Empty.

Note: Each server has one primary archive where data is currently collected. Only historical archives can be primary archives; future archives used for forecast data cannot be primary archives.

- The **Size** of the archive in megabytes.
- The **Start Time** of the archive, when the first data was stored in the file.

Note: Current Time indicates an empty archive.

- The **End time** of the archive, when the last data was stored prior to an archive shift.
If the end time is Current Time and start time is not, then the archive is the primary archive. If start and end times are both Current Time, then the archive is empty.
- The **Lifetime** of the archive, or the cumulative duration over which data was stored in the file (d hh:mm:ss:0)
- The **Last Modified Time**, which indicates the last time the archive was modified.
- The most recent **Backup Time** of the archive.
- The **State** of the archive file, either Created, Initialized, Dismounted, or Mounted.
- The archive file **Type**, either Fixed or Dynamic in size.
- The **Write Flag**, which indicates if the file is Writable or Read-Only.
- The **Shift Flag**, which indicates if the file is Shiftable or Not Shiftable.
- The **Add Rate/Hour**, recording the average number of values stored per hour over the lifetime of the archive file.
- The **% Full**, indicating the amount of space already used up in the archive.

Note: Dynamic archives are always 100% full.

- The number of **Annotations** in the file, divided by the total number of stored values.
- The **Annotation File Size**, recording the size of the accompanying file used to store annotations, in kilobytes.
- For primary archives, the **Shift Prediction** indicates the estimated time the archive will fill and the server will shift to a new primary archive.
- **Primary Offset** are the start and end record numbers for primary records. The end record number is always 1/2 of the Record Count.
- **Overflow Offset** are the start and end record numbers for overflow records.
- **Record Size** is the size in bytes of one record. This is always 1024.
- **Version** is an identifier of the archive's internal architecture. This label allows Data Archive to mount and upgrade archives from older versions.
- An **Archive Number** that indicates the archive file's order of use by Data Archive.

Show or hide archive gaps

The Archives tool displays gaps in archive times with markers. You can modify this default behavior using the Options dialog box.

1. Click **Options**  on the toolbar to display the Options dialog box.
2. Do one of the following:
 - a. To hide archive gaps, clear the check box.
 - b. To show archive gaps, select the check box, then select the gap lengths you want to display:
 - a. Greater than 0 seconds apart
 - b. Greater than 1 second apart
 - c. Greater than or equal to 2 seconds apart

The default setting is greater than or equal to 2 seconds apart.

3. Click **OK**.

The archive list is automatically refreshed if an option is changed.

Archive management tasks in SMT

Data Archive stores data in archive files. You can perform most PI archive management tasks with the PI SMT Archives tool. Some tasks, however, such as performing an archive walk or backfilling data require you to use command-line utilities. For information on using command-line tools for managing archives, see *Data Archive Reference*.

Archive creation with PI SMT

Use the Archives tool in PI SMT to create historical archives. You can also use the **piarcreate** command-line utility, as described in the *Data Archive Reference Guide*.

Note: Future archives are created automatically by Data Archive when future PI points are populated with data.

Archive names

Use a naming convention for your PI archives that is valid for the underlying operating system and ensure that the file location has sufficient disk space. There are no other limitations to name PI archives.

The default archive file names are **piarch.xxx**, where **xxx** is **001**, **002**, **003**, and so on.

The associated annotation file has the same full path name as its archive file with **.ann** appended. For example, the annotation file for the **piarch.001** archive file would be **piarch.001.ann**.

PI Data Archive automatically creates archive files using default settings and configurations. These defaults are predefined values the system uses when specific configurations or options are not explicitly defined or customized by the user. For more information on the default settings and how to customize them, see [Automatic archive creation tuning parameters](#).

Create an archive

1. Select **Start > PI System Management Tools > Operation > Archives**.
2. Click the **Create a New Archive** button ().
3. In the Create a New Archive dialog box, select **Single archive**.
4. Click the browse button to change the archive path, if desired.

You can store an archive in any local or network directory that is accessible by Data Archive. Local storage with other archives provides a convenient option for managing archives.

5. If you do not want to use the chronologically numbered default name, enter a name in **Archive name**.
If the text field is yellow, then the archive name is already in use by another file, possibly an unregistered archive. You may want to cancel the procedure and use the existing archive, if empty.
6. Select a source option to create the archive:
 - **Clone primary archive fixed size:** creates an empty archive of fixed type that's based on the size of the current primary archive.

- **Create archive with a custom fixed size:** creates an empty archive with a different size (typically larger) than the current primary archive. In the accompanying field, specify the size in megabytes (MB). The size must be equal to or greater than the size of the current primary archive, up to a maximum of 2TB.
- **Create archive with fixed start and end time:** creates an empty archive to be used only for a specified time period. If you select this option, for the **Type** option, select Fixed.

Note: Start and end times cannot overlap an existing archive.

7. Click **OK**.

The Archives tool creates and registers the archive.

Create multiple archives for backfilling

Note: Use this procedure to create historical archives for backfilling. For backfilling future archives with forecast data, you do not have to create archives manually.

1. Select **Start > PI System Management Tools > Operation > Archives**.
2. Right-click an archive file from the target Data Archive server and choose **Create New**.
3. In the Create a New Archive dialog box, select **Multiple archives for backfilling**.
4. Click the browse button to change the archive path, if desired.
5. Enter a prefix for the file in **Archive name**, or accept the default prefix.
The start time and end time will be automatically appended to the archive name depending on the archives being created.
6. Define the **Maximum archive duration** for each new archive file.
7. Enter **Start time** and **End time** for the new archive files using PI time format.

Note: Start and end times must not overlap an existing archive.

8. Click **OK**.

The Archives tool registers the newly created archives automatically.

Register archives

To register an existing archive for use with a particular server, use the PI SMT Archives tool.

1. Select **Start > PI System Management Tools > Operation > Archives**.
2. Right-click an archive file from the target Data Archive server and choose **Register Archive**.
3. Browse to the archive file you want to register and click **Open**.

The list of archives is refreshed.

To register multiple archives at once, press Ctrl and click or press Shift and click to select the files and click **OK**.

Unregister archives

To move or reprocess an archive file, you must unregister the archive, make your changes, and then re-register

the file. You cannot unregister a primary archive. If the Data Archive server is not on the local machine, this task requires read and write access on the PIARADMIN database.

1. Select **Start > PI System Management Tools > Operation > Archives**.
2. Right-click an archive file from the target Data Archive server and select **Unregister Archive**.
3. Click **Yes**.

Display the header of an unregistered archive

To view the properties (such as start and end times) of an unregistered archive, you can display its header.

1. Select **Start > PI System Management Tools > Operation > Archives**.
2. In the toolbar, click **Display Unregistered Archive** .
3. Browse to the correct directory, select the unregistered archive file, and click **Open**.

PI SMT adds the unregistered archive to the list of archives.

Set an archive to writable or read only

To change the protective **Write** flag for an archive file, right-click the archive in list view and select the appropriate option.

- To make a read-only archive writable, select **Make Writable**.
- (Not recommended) To make a writable archive read only, select **Make Read-Only**.

Caution: Setting an archive to read only can result in data loss. The preferred method for preventing archive changes is to use the *EditDays* tuning parameter to set a time limit for archive changes.

Set availability of archives for shifts

To set whether an archive is available to be shifted to and therefore become the new primary archive, follow this procedure.

1. In PI SMT, select **Operation > Archives**.
2. In the list view, right-click an archive and select **Make Non-shiftable** or **Make Shiftable**.

Force an archive shift

During normal operations, you do not need to force an archive shift. However, you can force an archive shift when necessary.

Even if you are scheduling when your archive shifts occur (rather than using the default automatic shifts) you can still force an archive shift.

Note: If the following message is displayed: *Warning: This shift will clear data from the target archive. Are you sure you want to force an archive shift on *serverName* to a new archive?*, you can ignore it, if Auto Archive

Creation is enabled (Archive_AutoArchiveFileRoot is set to a value). A new archive will be generated and no data will be cleared.

Use this procedure to force a server to shift from one historical archive to another.

1. Select **Start > PI System Management Tools > Operation > Archives**.
2. Right-click the server's primary archive in the list and select **Force Shift**.

Schedule archive shifts

Users can now configure PI Archive Subsystem to automatically shift historical archives at a specific time duration rather than having PI Archive Subsystem shift based on remaining archive space, or by manually forcing a shift. The result of this is fixed time archives.

With daily duration, shifts happen at midnight every day. With weekly duration, shifts happen at the midnight of Sunday. With monthly duration, shifts happen at midnight of the first day of every month. In regions where daylight savings time occurs at midnight, for the Spring transition, the shift time will be at 1:00:00 daylight saving time. For the Fall transition, the shift time will be at 0:00:00 standard time. Natural shifting is just allowing the shift to occur automatically when the archive is full.

When the *Archive_AutoArchiveFileSize* tuning parameter is set, the new primary archive's size will be based on this value. When it is not set, PI Archive Subsystem calculates the new primary archive's size based on past data rate and the shift duration, so that the archive can accommodate the expected amount of data.

In collective environments, scheduled shift can be configured on the primary server which applies to all members. Secondary members will shift based on the primary server's time zone. As in earlier versions, future archives are still considered monthly archives. In a collective, future archives' start and end times are now based on the primary's time zone.

Note: If you are upgrading an existing PI Collective, you will need to upgrade all the secondary members before upgrading the primary server. For more information see [Scheduled archive shifts in collective manager](#).

Configure scheduled shifts

1. Open **SMT**
2. Click **Operation**
3. Click **Archives**
4. In the main panel, with the **Historic** tab selected, click **Review and update parameters**
5. Select the desired **shift duration** in the Shift Schedule list
6. Click **Save**

Note: If Auto create size (MB) is set, and shift type is changed from natural shift to scheduled shift, the size setting will be automatically removed by SMT. Auto create size (MB) is the same as the *Archive_AutoArchiveFileSize* tuning parameter. When it is not set, PI Archive Subsystem will automatically calculate the archive size based on past data rate and the shift duration. If you still want to configure the size in the Auto create size (MB) box, fill in the desired size and click **save**.

Force shifts

When scheduled shift is configured, if a force shift is issued within 10 minutes of the next scheduled shift time, the shift will happen, the new primary archive's start time will be set to the next scheduled shift time, and the next scheduled shift will be skipped.

PowerShell support for scheduled shift

To get the current settings for your scheduled shift in Powershell, use the cmdlet **Get-PIArchiveSetSettings**.

In the example below, 420_cp1 is the collective name. You can use either the target server name or collective name (name in KST.)

```
PS C:\WINDOWS\system32> Get-PIDataArchiveConnectionConfiguration 420_cp1 |  
Connect-PIDataArchive | Get-PIArchiveSetSettings
```

Sample output:

```
SetName      : 0_Historical  
ArchiveSize   : 128  
Description   : Classical Historical Set 0  
ShiftFrequency : Natural  
TimeZone      : Pacific Standard Time  
FutureLimit    : 00:10:00
```

To set the shift schedule, use the command **Set-PIArchiveSetSettings**.

```
PS C:\WINDOWS\system32> Get-PIDataArchiveConnectionConfiguration 420_cp1 |  
Connect-PIDataArchive | Set-PIArchiveSetSettings -Name 0_Historical -ShiftFrequency Monthly
```

Manage archive gaps

Note: Use this procedure for historical archives only, not for future archives.

An archive gap is a range of time when no archive file is registered. If an event is sent to the archive and no archive file is registered within the appropriate time range, the event is discarded and an error is logged. If data retrieval is attempted for a time range that overlaps with a gap, the returned data includes a digital state Arc Offline that indicates the beginning of the gap. This prevents values from being interpolated when data is missing.

Generally, PI archive files account for all of history with no gaps or overlaps. If an unintentional gap between archive files occurs, you can use PI SMT or the **pidiag** tool to fix the gap. Until the gap has been fixed, data cannot be collected and stored in the time range in the gap.

1. Select **Start > PI System Management Tools > Operation > Archives**.

All the archives registered on the selected Data Archive server are listed. Any archive gaps are labeled and highlighted in red.

2. Right-click the line displaying the archive gap and select **Create New**.

The Create New Archive dialog box appears. The dialog box is already populated with the correct start and end times to fill the archive gap.

3. Click **OK**.

The new archive is created and registered, and an archive gap no longer appears in the archive list.

Reprocess an archive

Using the Archives tool you can reprocess archives while they are online. Reprocessing archives repairs corrupt archives, and for archives that are not corrupt, reprocessing can potentially recover disk space or improve the speed of certain transactions.

Note: You have the option to automatically delete archive files, after they have been successfully reprocessed. To set this behavior as the default, in **Operation > Archives** click the **Options** icon  in the toolbar and select the check box **Automatically delete input archive after reprocessing**. Even when this check box is selected, you can still choose to reprocess without deleting the original archive file.

1. In PI SMT, choose **Operation > Archives**. If you have any corrupt archive files, PI SMT prompts you to reprocess them.
2. Select the archives you want to reprocess and click **Reprocess Selected**.

You can initiate reprocessing for multiple archives, but they will be processed sequentially. The status of the reprocessing operation, and the list of archives queued for reprocessing, will be shown in the Online reprocessing pane, in the lower half of the window.

Note: If you do not have any corrupt archives, you can initiate reprocessing by right-clicking any archive and choosing **Reprocess archive**.

3. Monitor the progress of the reprocessing job in the Online reprocessing pane.
From this pane, you can view the queue of archives to be reprocessed and cancel active and pending jobs. For example, to stop the current reprocessing job, right click it and choose **Cancel**.
4. Check the **Status** column to see when the reprocessing job finishes and to verify that it completes without error.

For more information about reprocessing archives online, see [Archive reprocessing](#).

Enable auto-dynamic archive conversion

Automatic conversion of full fixed archives to dynamic archives is enabled by default for PI Server 3.4.375 and later. However, if you upgrade from an earlier version of PI Server, this parameter may not be enabled.

Beginning with PI Server PR1 SP1a (3.4.375.59), auto-dynamic archives preserve their shift flag and turn into fixed size archives again if they become primary archives.

Use this procedure to enable the automatic conversion of full fixed archives to dynamic archives, for versions prior to 3.4.375:

1. Select **Start > PI System Management Tools > Operation > Tuning Parameters** and click the **Archive** tab.
2. Select the **Archive_Enable_AutoDynamic** parameter and set the **Value** to 1.

Data Archive Version	Auto-dynamic Archive Conversion Behavior
Prior to 3.4.375	If you upgrade from a version of PI Server prior to 3.4.375 to a version prior to 2012, the Archive_Enable_AutoDynamic parameter might not be enabled.
PI Server PR1 SP1a (3.4.375.59)	Beginning with PI Server PR1 SP1a (3.4.375.59), auto-dynamic archives preserve their shift flag and turn into fixed size archives again if they become primary archives. The Archive_Enable_AutoDynamic tuning parameter is enabled by default.
PI Server 2012	The Archive_Enable_AutoDynamic tuning parameter is deprecated. Automatic conversion of full fixed archives to dynamic archives is default behavior.

Export a list of archives to a file

1. Select **Start > PI System Management Tools > Operation > Archives**.
2. Select an archive from the target Data Archive server from the list.
3. Click **Export Archive List**  and select a file type.
4. In the Save Archive List As window, select a location to store the archive file.
5. Click **Save**.

Create a .bat registration file

1. Open **SMT**
2. Select **Operation**
3. Select **Archives**
4. Select **Registration BAT File**

The screenshot shows the AVEVA PI System Management Tools Archives interface. The left sidebar contains navigation links for Servers, System Management Tools (Alarms, Batch, Data, Interfaces, IT Points, Operation, Archives, Backups, Licensing, Message Logs), and Help. The main area displays a list of archive entries for the 'INCINERATE' server. The 'INCINERATE' entry is selected, and a context menu is open, showing options: 'Comma Separated (.csv)' and 'Registration BAT File'. The table lists 18 archive entries with the following columns: #, Start Time, End Time, Duration, and Size (MB). The first entry is highlighted.

#	Start Time	End Time	Duration	Size (MB)
0	9/1/2017 10:03:30 AM	Current Time	91d 07:15:00	4096
1	8/28/2017 12:00:00 AM	9/1/2017 10:03:30 AM	04d 10:03:30	0
2	8/21/2017 12:00:00 AM	8/28/2017 12:00:00 AM	07d 00:00:00	0
3	8/14/2017 12:00:00 AM	8/21/2017 12:00:00 AM	07d 00:00:00	0
4	8/7/2017 12:00:00 AM	8/14/2017 12:00:00 AM	07d 00:00:00	0
5	7/31/2017 12:00:00 AM	8/7/2017 12:00:00 AM	07d 00:00:00	0
6	7/24/2017 12:00:00 AM	7/31/2017 12:00:00 AM	07d 00:00:00	1
7	7/17/2017 12:00:00 AM	7/24/2017 12:00:00 AM	07d 00:00:00	1
8	7/10/2017 12:00:00 AM	7/17/2017 12:00:00 AM	07d 00:00:00	0
9	7/3/2017 12:00:00 AM	7/10/2017 12:00:00 AM	07d 00:00:00	0
10	6/26/2017 12:00:00 AM	7/3/2017 12:00:00 AM	07d 00:00:00	0
11	6/19/2017 12:00:00 AM	6/26/2017 12:00:00 AM	07d 00:00:00	0
12	6/12/2017 12:00:00 AM	6/19/2017 12:00:00 AM	07d 00:00:00	0
13	6/5/2017 12:00:00 AM	6/12/2017 12:00:00 AM	07d 00:00:00	0
14	5/29/2017 12:00:00 AM	6/5/2017 12:00:00 AM	07d 00:00:00	0
15	5/22/2017 12:00:00 AM	5/29/2017 12:00:00 AM	07d 00:00:00	0
16	5/15/2017 12:00:00 AM	5/22/2017 12:00:00 AM	07d 00:00:00	0
17	5/8/2017 12:00:00 AM	5/15/2017 12:00:00 AM	07d 00:00:00	0
18	5/1/2017 12:00:00 AM	5/8/2017 12:00:00 AM	07d 00:00:00	0

AutoPointSync List

The AutoPointSync List tool displays PI interfaces that write to selected Data Archive servers and are configured to use PI Auto Point Sync (PI APS), a powerful tool for maintenance of the AVEVA™ PI System™. PI APS automatically synchronizes the PI points an interface or COM Connector uses with the tag definitions on the DCS. It also locates points on the DCS that do not yet exist in Data Archive and creates those points.

The AutoPointSync List tool derives its list of interfaces from a server's PI Module Database, a hierarchical information store that organizes an enterprise's information streams into logical sub-areas, or modules.

Use this tool to enable or disable synchronization of selected interfaces.

AutoPointSync List window

The AutoPointSync List window lists the PI interfaces configured to use PI Auto Point Sync and write to the selected Data Archive servers. You can also use the window to change the current synchronization status.

Interfaces are listed in rows with the following column values:

- **Interface**

The name of interface configured to use PI APS.

- **Point Source**

The point source used by the interface.

- **ID**

The ID number assigned to points belonging to this interface.

- **Server**

The host Data Archive for the interface.

- **APS Node**

The computer where the interface points are synchronized by PI APS.

- **Sync Status**

The sync status for the interface, either **Enabled** or **Disabled**.

- **Sync Period**

The frequency of synchronization for the interface.

- **Create Rule**

The creation rule used for synchronization, either **Create automatically**, **Store as Available Tags**, or **Discard new Points**.

- **Edit Rule**

The editing rule used for synchronization, either **Edit automatically**, **Store to PIConfig file**, or **Discard edits**.

- **Delete Rule**

The deletion rule used for synchronization, either **Turn scan off**, **Delete PI Points automatically**, **Do nothing**, **Remove from interface by changing PS and/or ID**, or **Store to PIConfig file**.

- **Debug Level**

The APS debug level set for the interface.

Click columns to sort interface listings in ascending or descending order. Click the Refresh button  to refresh the display.

Note: For selected non-Windows servers, messages are written to the Session Record pane of the PI SMT host.

Synchronization status

Synchronization is the process that compares PI points for an interface with data source tags and either changes Data Archive to resolve any differences or logs the differences.

The **Sync Status** column in the AutoPointSync List window indicates the synchronization status of an interface:

- **Enabled**

Synchronization is enabled for the interface.

- **Disabled**

Synchronization is currently disabled for the interface.

You can activate or deactivate synchronization for each interface included on the list:

- To activate the synchronization status for an interface, right-click the interface and select **Enable Sync**.
- To deactivate the synchronization status for an interface, right-click the interface and select **Disable Sync**.

Export the list of interfaces that use PI Auto Point Sync

You can export the list of interfaces show in the AutoPointSync List window to a CSV file.

1. Right-click the list and then click **Export** .
2. Select the location to save the exported list.
3. Click **Save** and verify that the file exists in the location you selected.

Backups

The primary purpose of the Backups tool is to view the backup history of a Data Archive. You can also use the Backups tool to run unscheduled, or on-demand, backups for troubleshooting, data mining, or testing. On-demand backups do not interfere with your regularly scheduled backups.

Note: Do not use on-demand backups to substitute regularly scheduled Data Archive backups.

View backup history of a Data Archive server

1. Open PI SMT.
2. In the Servers pane, select the server you want to check.
3. In the System Management Tools pane, select **Operation > Backups**.
4. In the **PI Server** drop-down list, select the server you want to examine. The list includes all the servers selected in the Servers pane.

The **Backup History** table shows the backup history for the server selected in the **PI Server** list. By default, the table lists the last 100 backups and shows a subset of these available data:

Column	Data
Index	Number to represent the order in which the backups occurred.
Start Time	Time the backup started.
Status	Status code and the status code description for each backup.
Files Copied	Number of files copied during the backup.
Size	Total size of the backup.
Duration	Time it took to complete the backup.
File Copy Failures	Number of files for which the backup failed.
Total Files	Number of files selected for backup.
Type	Copy, Incremental, Differential, Full, and Numarch/Cutoff. See Data Archive backup types .
VSS	True for VSS backups; False for non-VSS backups.
Component Mode	True for component mode backups; False if not.
Third Party	True if a third-party backup application was used to

Column	Data
	back up Data Archive; False if the PI Backup Subsystem was used.
Initialization Duration	Time elapsed before the backup began after the backup request.

Note: Right-click a column heading to see a complete list of columns you can display.

Data Archive backup types

The **Backup History** table might show the following backup types:

- **Copy**

The backup type for unscheduled backups, that is, those run with the **Backups** tool.

- **Incremental**

The backup type for the regularly scheduled Data Archive backups.

- **Differential**

A backup type if you are using third-party backup software to back up Data Archive.

- **Full**

A backup type if you are using third-party backup software to back up Data Archive.

- **Numarch/Cutoff**

The backup type for regularly scheduled backups that were configured on PI Server versions 3.4.370 or 3.4.375.

Change the number of backups shown in the Backup History table

To change the number of backups shown in the **Backup History** table, set the *Backup_MaxHistory* tuning parameter.

1. Open PI SMT.
2. Under Collectives and Servers, select the server you want to check.
3. Under System Management Tools, select **Operation > Tuning Parameters**.
4. In the Tuning Parameters tool, click the Backup tab.
5. Double-click *Backup_MaxHistory* and set the parameter value.

View backup information summary

The **Summary** fields below the **Backup History** table contain information about the selected backup.

Select a backup in the **Backup History** table to view information about the backup in the **Summary** fields below the table:

- **Status**

The status code of the backup: either **Success** or an error code for the error that occurred.

- **Method**

Information about the backup method:

- **Type**

Backup types include incremental, copy, full, differential, or Numarch/Cutoff (see [Data Archive backup types](#)).

- **VSS/non-VSS**

True for VSS backups; False for non-VSS backups.

- **Component Mode**

Meaningful only for third-party backups. Some third-party backup applications cannot do component mode backups and do not provide information to Data Archive about the success or failure of the backup.

- **Third party**

Appears if PI Backup Subsystem is not the application used to perform the backup.

- **Start Time**

The time the backup started.

- **Duration**

The time taken for the backup to complete.

View backup details

The Backups tool provides access to details about Data Archive backups.

Double-click a backup in the **Backup History** table to open the Backup Details window.

The Backup Details window shows a summary of backup details and a list of backed up files.

Backup details summary

Use the **Summary** tab in the Backup Details window to view details about the selected backup. At the top left of the tab, the **Index** field shows the index number of the backup you are currently viewing.

The index number represents the order in which the backup occurred, relative to the other backups in the list. Use these options to see details for a different backup:

- Click the arrows on the **Index** field
- Enter an index number directly in the **Index** field
- Use the **Previous** and **Next** arrows at the top of the window

The **Summary** tab displays all the information that is included in the **Summary** area under the main **Backup History** table. See [View backup information summary](#) for more on these fields. The **Summary** tab also provides:

- The following backup statistics for each subsystem and archive included in the backup:
 - **Freeze Start Time**

The time at which all subsystems entered a frozen state; if the backup is successful, the last backup time of archive files is updated with this time stamp.

- **Freeze Duration Time**

Maximum amount of time that the databases could have been frozen (to writes) for a subsystem that was backed up.

- **Freeze Transition**

Time spent for all subsystems to go from an unfrozen to frozen state.

- **Initialization Duration**

Time elapsed between the issue of the backup request and the start of the backup.

- A list of the subsystems that were available for backup at the time the backup was executed.

Backed up file list

The **Backed Up File List** tab in the Backup Details window displays summary information and a list of files that were backed up. For aborted or otherwise unsuccessful backups, any files that were not backed up are also listed in red.

The summary information consists of:

- **Total Files**

Number of files selected for backup.

- **File Copy Failures**

Number of files that should have been backed up and were not.

- **Files Backed Up**

Number of files that were backed up.

- **Unchanged Files for Incremental Backup**

Number of files that were not backed up because they were unchanged since the last backup. This field appears only if the selected backup was an incremental backup. See [Data Archive backup types](#).

The table at the bottom of the tab shows lists the backed-up files:

Column	Description
File Name	The name of the backup file
Source Directory	The full pathname of the directory where the backup file is located
Report	The action taken or the reason the backup failed
Destination	The name of the directory to which this file was backed up
Status	A status code: <i>success</i> for a successful backup or an error for an unsuccessful backup
Size	The size of the file in KB

Column	Description
Component Description	The name of the component to which the file belongs

You can:

- Click a column heading to sort the list by that column.
- Right-click a column heading to select the columns you want to display.

Troubleshooting Data Archive backups

A backup is a copy of data that you can use in case your original data is damaged or lost. Backups provide a means of recovery after unintended configuration changes (such as an accidental point deletion) and database corruption. Generally, the best way to recover data is to restore a backup that was taken prior to when the issue happened.

View Data Archive backup logs and messages

Check the following log files for error messages:

- **Backup script log**

The backup script log is written to the target directory of the backup with a name of the form **pibackup_dd-mmm-yy_hh.mm.ss.txt**. For example:

```
pibackup_5-Aug-05_14.16.22.txt.
```

- **PI message log**

To display all error messages between the start and end time of a backup, use a command of the form:

```
pigetmsg -sl E -st starttime -et endtime
```

If any errors occur during a backup, the output of this command is automatically dumped to the backup script log.

To display all messages related to backup, use a command of the following form:

```
pigetmsg -src pibackup -st starttime -et endtime
```

To display only those messages from the PI Backup subsystem itself, use a command of the following form:

```
pigetmsg -pr pibackup -st starttime -et endtime
```

- **Windows application event log**

Look for messages from VSS and COM+ event sources.

- **Windows system event log**

Look for messages from VOLSNAP and NTFS event sources.

Common issues with backups

Backups may fail for the following reasons:

- Sometimes the system can get into a state where the **ole32.dll** becomes unregistered. If **ole32.dll** is unregistered, backups do not work. To resolve this issue, re-register **ole32.dll** with the following command:

```
regsvr32 ole32.dll
```

- Backups can fail if either of the following services is disabled:

- Microsoft Software Shadow Copy Provider
- Volume Shadow Copy
- COM+Event System

- Typically, the Volume Shadow Copy service should not be running. It is started on demand whenever it is needed. If the service is running, it may be stuck in a bad state. To stop the service, enter the following command:

```
net stop "Volume Shadow Copy"
```

If this command does not work, use Windows Task Manager to end the **VSSVC.exe** process.

- Backups of Data Archive have been known to fail when the OfmProvider from St. Bernard software is installed on the Data Archive computer. To determine if this software is installed, run the **vssadmin list providers** command and look for *OfmProvider* in the output.
- All archives to be backed up must be on the Data Archive computer. If the archive to be backed up is on a remote drive, such as a UNC path, the backup will fail unless the file is marked read only, and the pibackup service is configured to run as an account that has appropriate permissions.
- Backups require at least one NTFS partition on the machine where Data Archive is installed.

Data Archive backup failure due to offline subsystem

When a subsystem registers for backup, the subsystem must remain online during the next backup or else the backup will fail with the following error:

Backup start failed, Status: [16896] RPC Resolver offline for a subsystem to which the backup subsystem is communicating. Find -10733 error in message log to identify problematic RPC

Two messages will appear in the Data Archive message log with the **-10733** error similar to the following:

```
E 19-Oct-09 13:54:57 pibackup (5059)
>> Callback failed for <pibatch> for the IDENTIFY event. Error [10733]
PINET: RPC Resolver is Off-Line.
E 19-Oct-09 13:54:57 pibackup (5061)
>> Error [10733] PINET: RPC Resolver is Off-Line., failed to send the IDENTIFY backup event
to pibatch
```

To fix the problem, you can either start the subsystem that is not running, or you can do the following, if the subsystem was purposefully stopped:

1. Run the command **piartool -backup -query** and make note of the subsystems that are currently registered for backup.
2. Restart PI Backup Subsystem.
3. Wait for the previously registered subsystems to register for backup again, with the exception of the problematic subsystem. Subsystems may take up to 5 minutes to re-register for backup after the backup subsystem has been restarted.

Perform an on-demand Data Archive backup

To back up Data Archive, users must have read and write access permissions in the PIBACKUP table. Use the PI SMT Database Security tool to provide this access to users.

Use the Backups tool in SMT to run on-demand backups of Data Archive for testing and troubleshooting. On-demand backups run in component mode.

Note: Do not use the Backups tool as a substitute for scheduled backups. Your daily backups should be set up as a Microsoft Windows scheduled task.

1. In the Backups window, select the data server from the **PI Server** list.
2. Click **Backup Now** .
3. In the server Backup window, select the Data Archive components to back up.
As you make a selection, the right side of the window shows the list of files that will be backed up.
4. In the **Backup Location** field, enter or browse to the target directory path.

You can specify a UNC or a local path on the server that you are backing up.

Note: If you are not running PI SMT on the same Data Archive server that you are backing up, then you cannot use the browse button unless you have Windows administrator access to that server.

5. Click **Backup**.

The Backup History window shows the backup details, including a backup's progress and whether it was successful or aborted.

Export Data Archive backup history

You can use the Backups tool to export a backup history report to an XML file.

1. Click the **Export Backup Reports** button to open the Save Backup History As window.
2. Browse to the save location.
3. Enter the file name.
4. Click **Save**.

Batch Custom Names

PI Batch has applicability across many industries. However, the terminology used in each industry varies. In some cases, generic batch terminology may be inadequate. Use the Batch Custom Names tool to customize batch-related terms that appear in the PI BatchView application, both at run time and build time.

PI BatchView lets you view batch data in Windows clients. It can be added as a plug-in to PI ProcessBook, PI DataLink, and any 32-bit program that can contain ActiveX controls, such as Microsoft Access and Microsoft Visual Basic.

Custom name sets

Batch custom names are organized in sets. Each user-defined set contains one or more custom names that apply to a single concept or theme. Every set has the same range of batch-related terms that can be customized. You have the option to customize terms within this range. For details, see [Custom name examples](#).

Note: Custom names apply only to specific batch-related client tools. They do not apply to the Data Archive, PI SDK, or any of the COM interfaces for the PI BatchView objects.

You can view, create, edit, and delete custom name sets with the Batch Custom Names tool. You can also export and import lists of custom name sets.

View custom name sets

In the PI SMT System Management pane, click **Batch > Batch Custom Names** to open the Batch Custom Names tool.

Use the Batch Custom Names tool to view and edit custom name sets. The pane on the left side of the tool lists the servers selected in the Collectives and Servers pane.

1. In the pane on the left side of the Batch Custom Names tool, expand the server to view custom name sets defined on that server.
2. Select a custom name set to view or edit the customized names in that set.
3. Use the toolbar to add, delete, copy, paste, import, or export sets.

Custom name set details

Details about the selected custom name set, including the name of the set, the editable terms, and definitions for the terms, are displayed on the right side of the Batch Custom Names tool, in the **Custom Names Definition** pane.

The pane includes three main sections: Properties, Custom Names, and Description.

Note: You must save changes made in the **Properties** and **Custom Names** sections to have changes take effect. If you select another tool without saving changes, PI SMT will prompt to save your changes. If you select a different custom name set, your changes are automatically saved.

Properties

The top section of the Batch Custom Names tool shows two properties about the selected name set:

- **Custom Names Set**

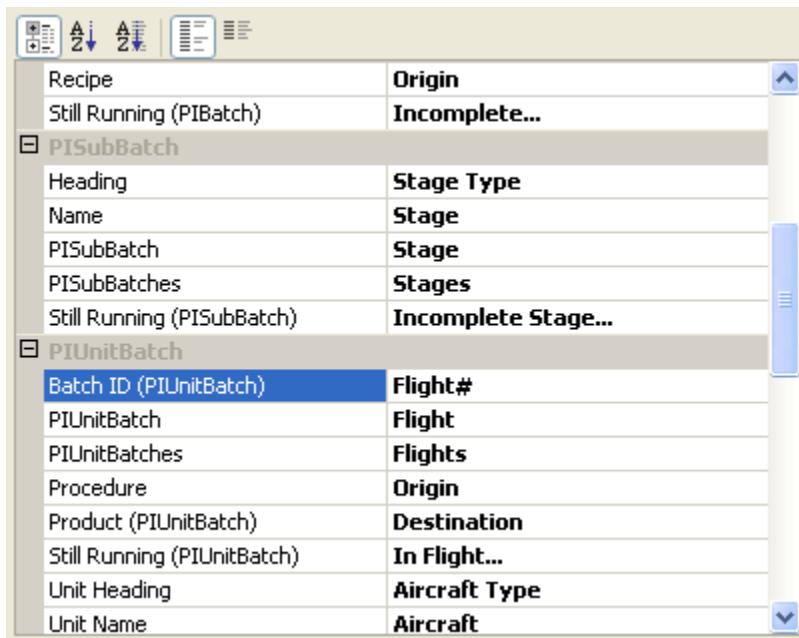
The name of the custom name set. If you change the name, the tool updates the name in the list of servers and name sets.

- **Status**

The number of custom names defined and the total number of batch terms that are re-nameable. This status also indicates if changes are unsaved.

Batch terms of a custom name set

You can view and edit batch terms used in PI BatchView and their corresponding custom names:



The screenshot shows a table with two columns. The left column lists batch terms, and the right column lists their corresponding custom names. The table is organized into sections: General, PIBatch, PISubBatch, PIUnitBatch, and Results. The PIUnitBatch section is currently active, indicated by a blue background. The first row of the PIUnitBatch section is highlighted in blue, showing 'Batch ID (PIUnitBatch)' and 'Flight#'. Other rows include 'PIUnitBatch' (Flight), 'PIUnitBatches' (Flights), 'Procedure' (Origin), 'Product (PIUnitBatch)' (Destination), 'Still Running (PIUnitBatch)' (In Flight...), 'Unit Heading' (Aircraft Type), and 'Unit Name' (Aircraft). The top of the table has icons for sorting and filtering.

Batch Term	Custom Name
Recipe	Origin
Still Running (PIBatch)	Incomplete...
PISubBatch	
Heading	Stage Type
Name	Stage
PISubBatch	Stage
PISubBatches	Stages
Still Running (PISubBatch)	Incomplete Stage...
PIUnitBatch	
Batch ID (PIUnitBatch)	Flight#
PIUnitBatch	Flight
PIUnitBatches	Flights
Procedure	Origin
Product (PIUnitBatch)	Destination
Still Running (PIUnitBatch)	In Flight...
Unit Heading	Aircraft Type
Unit Name	Aircraft

The left column lists the batch terms that can be renamed and is read-only. The right column contains a name that can be edited. If you do enter a custom name, the name in the left column is used. To enter custom names, enter or paste a new name in the corresponding field. Batch terms with blank custom names are considered undefined.

You can organize the list in one of three ways: categorized, sorted by batch term, and sorted by custom name:

Icon	Description
	Categorize the list into General , PIBatch , PISubBatch , PIUnitBatch , Results and Search Criteria table columns. Some terms are identical in two categories but have different meanings. For example, both PIBatches and PIUnitBatches have Batch ID properties that are separate entities. Some terms are repeated in

Icon	Description
	two or more categories because, although they have the same meaning, they are represented differently in different contexts. For example, for a custom name set called airline , you can apply the term Still running to a PIBatch (Trip) as Incomplete and to a PIUnitBatch (Flight) as in Flight .
	Sort the list by batch term to find the exact term if it is known and the category is not known. In this view, terms with identical names are appended with the category they belong to. For example, Batch ID would appear in the list twice; once as Batch ID (PIBatch) and once as Batch ID (PIUnitBatch) .
	Sort the list by custom name to find a custom name if you do not know the corresponding batch term.
	Show every batch term that can be renamed, whether or not a custom name has been defined. This mode is the best mode to use when entering new custom names.
	Show only those batch terms with defined custom names. This is useful when there are only a few custom names defined. The defined terms can be seen or edited without the clutter of the undefined terms.

Note: The options you select are saved with each custom name set; reloaded name sets retain previously selected options. Therefore, each custom name set may have different settings.

Description

The bottom of the Batch Custom Names tool shows a description of the currently selected combination of batch term and custom name set:

Batch Term:	Batch ID (PIUnitBatch)
Custom Name:	Flight#
Category:	PIUnitBatch
Description:	The mask-type of the Attributes component that represents a PIUnitBatch's Batch ID.

This description provides a context within which the custom name is used. You can use the description to ensure that the correct custom name is used. For example, if you select the batch term **Active**, the description indicates that it is the activity state of the Time Range component.

Note: You can copy text from this box; select the text to be copied and press Ctrl+C.

Custom name examples

The following simple examples show how you might use custom names.

Change "Unit" to "Reactor"

Company X is a traditional batch-oriented production company. Most of the batch terms used in PI BatchView apply to Company X's processes, but it would be helpful to change a few of the terms. For example, Company X would like to use Reactor instead of Unit. They can use custom names to change the label for every instance of the term "Unit" and its counterparts so that these terms appear as "Reactor." Such terms include: Unit, Units, Unit Name, and Unit Heading.

Change "Still Starting" to "Still Running"

Company Y is a power generation company that does not do traditional batch processing. PI BatchView is still useful to a part of the company because they monitor the startup process of each turbine. Treating each startup as a PIUnitBatch allows them to compare them with previous startups. Traditional batch terms that appear in PI BatchView are not very helpful. They can use custom names to rename several of their terms to more appropriate nomenclature. For example, they would like to use Still Starting in place of Still Running in the results table or Turbine in place of Unit Name in the attributes search component.

Change Product to Stadium

Organization Z runs an annual, company-wide tennis tournament and uses PI BatchView to compare games, matches, and so on. This is an example where not only do they not want to see the PI BatchView standard batch terms, but they are using properties for things that have completely different meanings from the original batch term. For example, they chose to use the Product field of the PIUnitBatch to store the name of the stadium hosting each set; therefore, using the word Product in the user interface does not make much sense. Following are views of the PI BatchView QuickSearch window without and with terms customized for a tennis tournament:

PI BatchView QuickSearch [PIUnitBatches]

Search Parameters																																				
Find:	All	PIUnitBatches	Toaster	<input type="button" value="..."/>																																
Include:	<input type="radio"/> Running	<input type="radio"/> Completed	<input checked="" type="radio"/> Both																																	
Batch ID	*																																			
Product	*																																			
Unit Name	*																																			
Time Range and Duration:																																				
Active	Between:	26-Oct-05	and	+1d																																
Any Length			and																																	
<table border="1"> <thead> <tr> <th>Batch ID</th> <th>Unit N...</th> <th>Product</th> <th>Recipe</th> <th>Start Time</th> <th>End Time</th> </tr> </thead> <tbody> <tr> <td>Pierce vs Conn...</td> <td>Court A</td> <td>TennisLand</td> <td>Connors</td> <td>26/10/2005 8:02:20 AM</td> <td>26/10/2005 8:30:15 A</td> </tr> <tr> <td>Brozowski vs S...</td> <td>Court D</td> <td>TennisLand</td> <td>Stokes</td> <td>26/10/2005 8:05:00 AM</td> <td>26/10/2005 8:42:13 A</td> </tr> <tr> <td>Chmelik vs Singh</td> <td>Court D</td> <td>TennisLand</td> <td>Chmelik</td> <td>26/10/2005 9:15:12 AM</td> <td>26/10/2005 9:52:44 A</td> </tr> <tr> <td>Santoni vs Jan...</td> <td>Court C</td> <td>TennisLand</td> <td></td> <td>26/10/2005 9:41:28 AM</td> <td>Still Running</td> </tr> </tbody> </table>							Batch ID	Unit N...	Product	Recipe	Start Time	End Time	Pierce vs Conn...	Court A	TennisLand	Connors	26/10/2005 8:02:20 AM	26/10/2005 8:30:15 A	Brozowski vs S...	Court D	TennisLand	Stokes	26/10/2005 8:05:00 AM	26/10/2005 8:42:13 A	Chmelik vs Singh	Court D	TennisLand	Chmelik	26/10/2005 9:15:12 AM	26/10/2005 9:52:44 A	Santoni vs Jan...	Court C	TennisLand		26/10/2005 9:41:28 AM	Still Running
Batch ID	Unit N...	Product	Recipe	Start Time	End Time																															
Pierce vs Conn...	Court A	TennisLand	Connors	26/10/2005 8:02:20 AM	26/10/2005 8:30:15 A																															
Brozowski vs S...	Court D	TennisLand	Stokes	26/10/2005 8:05:00 AM	26/10/2005 8:42:13 A																															
Chmelik vs Singh	Court D	TennisLand	Chmelik	26/10/2005 9:15:12 AM	26/10/2005 9:52:44 A																															
Santoni vs Jan...	Court C	TennisLand		26/10/2005 9:41:28 AM	Still Running																															
<input type="button" value="Options..."/> <input type="button" value="About..."/> <input type="button" value="Close"/> <input type="button" value="Help"/>																																				
Search complete (4 results found)																																				

PI BatchView QuickSearch [Sets]

Search Parameters																																				
Find:	All	Sets	Toaster	<input type="button" value="..."/>																																
Include:	<input type="radio"/> Playing	<input type="radio"/> Finished	<input checked="" type="radio"/> Both																																	
Players	*																																			
Stadium	*																																			
Court#	*																																			
Time Range and Duration:																																				
Playing	Between:	26-Oct-05	and	+1d																																
Any Length			and																																	
<table border="1"> <thead> <tr> <th>Players</th> <th>Court#</th> <th>Stadium</th> <th>Winner</th> <th>Start Time</th> <th>End Time</th> </tr> </thead> <tbody> <tr> <td>Pierce vs Conn...</td> <td>Court A</td> <td>TennisLand</td> <td>Connors</td> <td>26/10/2005 8:02:20 AM</td> <td>26/10/2005 8:30:15 A</td> </tr> <tr> <td>Brozowski vs S...</td> <td>Court D</td> <td>TennisLand</td> <td>Stokes</td> <td>26/10/2005 8:05:00 AM</td> <td>26/10/2005 8:42:13 A</td> </tr> <tr> <td>Chmelik vs Singh</td> <td>Court D</td> <td>TennisLand</td> <td>Chmelik</td> <td>26/10/2005 9:15:12 AM</td> <td>26/10/2005 9:52:44 A</td> </tr> <tr> <td>Santoni vs Jan...</td> <td>Court C</td> <td>TennisLand</td> <td></td> <td>26/10/2005 9:41:28 AM</td> <td>In Play</td> </tr> </tbody> </table>							Players	Court#	Stadium	Winner	Start Time	End Time	Pierce vs Conn...	Court A	TennisLand	Connors	26/10/2005 8:02:20 AM	26/10/2005 8:30:15 A	Brozowski vs S...	Court D	TennisLand	Stokes	26/10/2005 8:05:00 AM	26/10/2005 8:42:13 A	Chmelik vs Singh	Court D	TennisLand	Chmelik	26/10/2005 9:15:12 AM	26/10/2005 9:52:44 A	Santoni vs Jan...	Court C	TennisLand		26/10/2005 9:41:28 AM	In Play
Players	Court#	Stadium	Winner	Start Time	End Time																															
Pierce vs Conn...	Court A	TennisLand	Connors	26/10/2005 8:02:20 AM	26/10/2005 8:30:15 A																															
Brozowski vs S...	Court D	TennisLand	Stokes	26/10/2005 8:05:00 AM	26/10/2005 8:42:13 A																															
Chmelik vs Singh	Court D	TennisLand	Chmelik	26/10/2005 9:15:12 AM	26/10/2005 9:52:44 A																															
Santoni vs Jan...	Court C	TennisLand		26/10/2005 9:41:28 AM	In Play																															
<input type="button" value="Options..."/> <input type="button" value="About..."/> <input type="button" value="Close"/> <input type="button" value="Help"/>																																				
Search complete (4 results found)																																				

Create a new custom name set

1. In the pane on the left side of the Batch Custom Names tool, select a server.
2. Click **New Set** .
3. In the **Custom Names Set** field, change the set's name to something useful.
4. Select a standard batch term, type the term's corresponding custom name, and press Enter.
5. Click **Save Set** .

Edit batch terms of a custom name set

1. In the pane on the left side of the Batch Custom Names tool, expand the server to view custom name sets defined on that server.
2. Select the custom name set to be edited.
3. Select the standard batch term you want to edit, type the batch term's corresponding custom name, and press Enter.
4. Click **Save Set** .

Rename an existing custom name set

1. In the pane on the left side of the Batch Custom Names tool, expand the server to view custom name sets defined on that server.
2. Select the custom name set to be edited.
3. In the **Custom Names Set** field, type a new name.
4. Click **Save Set** .

Delete an existing custom name set

1. In the pane on the left side of the Batch Custom Names tool, expand the server to view custom name sets defined on that server.
2. Select the custom name set to be deleted.
3. Click **Delete Set** .
4. Confirm the deletion.

Export existing custom name sets

1. In the pane on the left side of the Batch Custom Names tool, expand the server to view custom name sets defined on that server.
2. Select the custom name set to be exported.

Tip: Select a server rather than a set to export that server's entire collection of custom name sets.

3. Choose an appropriate folder, type the name of the file to be created, and click **Save**.

Import existing custom name sets

1. In the pane on the left side of the Batch Custom Names tool, select a server.
2. Click **Import** .
3. Find the CSV file to be imported and click **Open**.

All of the custom name sets in the file will be imported to the selected server.

Revert a custom name set

You can undo any changes you made to a custom name set since you last saved the set.

1. Select the custom name set to be reverted.
2. Click **Revert** .

The set will be reverted back to its state before changes were made since the last save.

Show only customized batch terms

1. In the left pane of the Batch Custom Names tool, select a server.
2. In the right pane under the server name, select a set.
3. In the Custom Names Set: pane, click **Show Only Batch Terms with Custom Names Defined** .

Sort the batch terms

Use the sort buttons to sort the list of terms:

- **Arrange by Category**  sorts the batch terms by batch category.
- **Sort by Batch Term**  sorts the batch terms by the batch-term name.
- **Sort by Custom Name**  sorts the batch terms by the customize name for the batch term.

Find information about a batch term or custom name

1. In the pane on the left side of the Batch Custom Names tool, expand the server to view custom name sets defined on that server.
2. Select a custom name set that contains the batch term.
3. In the list of batch terms, select the batch term.

4. Read the description in the area below the list of terms.

Custom names toolbar

The main toolbar provides the basis for most of the actions that you can apply to a custom name set. Each action applies to the selected custom name sets. If you select a Data Archive server, most actions apply to all of that server's sets.

Button	Goal
	Create a new custom name set and add it to the selected Data Archive server.
	Save a selected custom name set.
	Revert changes made to a selected set.
	Delete a selected set from the list and Data Archive. If a Data Archive server is selected, all custom name sets from that server are deleted.
	Remove a selected custom name set and copy it to the clipboard. If a Data Archive server is selected, all of the server's sets are removed to the clipboard.
	Copy a selected custom name set to the clipboard. If a Data Archive server is selected, all of the server's sets are placed in the clipboard. Supported formats are: Text in ANSI or Unicode format, with the set name and all custom names separated by a CR or LF. Custom Clipboard Format, a binary representation which can be used only to paste the set back into the list. Module Database Format, which can be pasted directly into the PI Module Database tree.
	Paste a copied set into the list under the selected Data Archive or Data Archive of the selected custom name set.
	Export a selected custom name set to a CSV file. If a Data Archive server is selected, all sets are exported. The format of an exported file is: CustomNameSetName1,Category,BatchTerm1,CustomerName... CustomNameSetName1,Category,BatchTermN,CustomerName CustomNameSetName2,Category,BatchTerm1,CustomerName... CustomNameSetNameN,Category,BatchTermN,CustomerName

Button	Goal
	Import custom name values from a file to a selected Data Archive server. The file can be any text file with format compatible to the CSV export format.
	Launch online help.

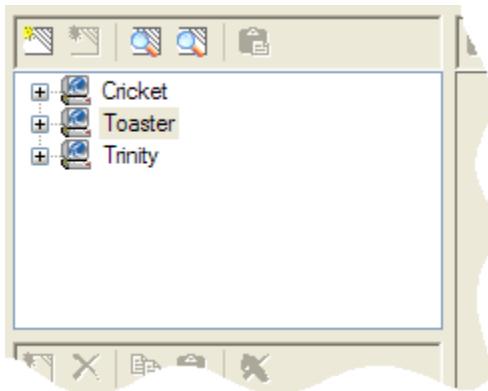
Batch Database

Use the Batch Database tool to view, create, edit, and delete batch items from the PI Batch Database. Batch items include PIBatches (including PIProperties), PIUnitBatches, and PISubBatches.

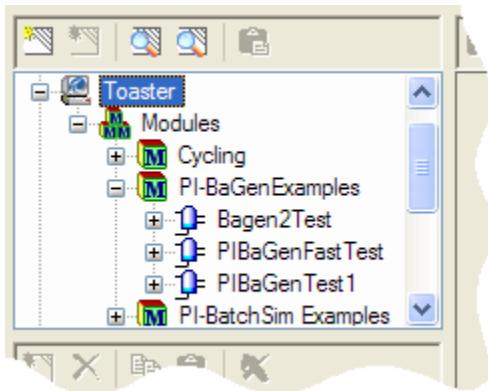
Note: This utility does not support creating or editing batch items from PI Batch Subsystem.

Module Database tree

You can view PIBatches (including PIProperties), PIUnitBatches, and PISubBatches in the Module Database tree, located at the top-left of the Batch Database tool. It lists the servers that are selected in the PI SMT Servers pane.



To view the hierarchy of PIModules for a selected server, click to open the branch. PIAliases and PIProperties are not shown in this tree and the PIModules are read-only.



Use this view to create new batch items and search for existing batch items. The results of a search are displayed in the search results list from which you can view, edit, and delete existing batch items.

Creation of PI Batch Database items

You can create a new PI Batch or PI UnitBatch.

Create a new PIBatch

1. In the Module Database tree, select the icon of the server for which you want to create a new PIBatch.
2. Click **New PIBatch** .
3. In the New PIBatch window, enter the basic properties of a PIBatch.
 - **Start Time** is the only required field. All other fields can be left blank.
 - If **End Time** is blank, the PIBatch will be set to a "still running" state; a PIBatch is considered to still be running if it has no end time.
 - **Server** pre-populates with the name of the server selected or the server of any selected PI module or PI unit in the [Module Database tree](#), that is, the server for which the PIBatch is to be created.
 - **Unique ID** shows <*Undefined Unique ID*> because the Unique ID of a PIBatch is assigned at the time the PIBatch is created (when you click **OK**).
4. Click **OK**.

Note: A PIBatch may have a reference to one or more PIUnitBatches. A PIBatch may also have one or more hierarchical PI properties. You can assign both of these properties after the PIBatch is created. For details, see [Link or unlink a PIUnitBatch to a PIBatch](#) and [PIBatch details](#).

Create a new PIUnitBatch

1. In the Module Database tree, select the icon of the server for which you want to create a new PIUnitBatch.
2. Click **New PIUnitBatch** . Alternatively, right-click a **PIUnit**  and then click **New PIUnitBatch**.
3. In the New PIUnitBatch window, enter the basic properties of a PIUnitBatch.

All available fields default to blank except for:

- **Start Time**, which defaults to * (current time)
- The **PIUnit** and **Server** fields are pre-populated, based on the unit that is selected in the Module Database tree, that is, the PIUnit for which the PIUnitBatch will be created.

Start Time is the only required field of this window, as indicated by the label's bold text.

All other fields can be left blank.

If **End Time** is blank, the PIUnitBatch is set to a still running state; a PIUnitBatch is considered to still be running if it has no end time.

4. Click the **Browse** button  on the right side of the **PIBatch** field to give a PIUnitBatch a reference to an existing PIBatch. For more information about assigning a PIBatch reference to a PIUnitBatch, see [Link or unlink a PIUnitBatch to a PIBatch](#).

The **Unique ID** field shows <*Undefined Unique ID*> because the unique ID of a PIUnitBatch is assigned at the time the PIUnitBatch is created; that is, after **OK** is clicked.

Note: A PIUnitBatch may contain one or more PISubBatches. See [Create a new PISubBatch](#) for details.

Searches for PI Batch Database items

You can search for a PIBatch or PIUnitBatch.

Search for PI Batches

Search for PI Batches on the selected server or the server of any selected PIModule or PIUnit.

1. Click **Search for PI Batch**  or right-click the PIModule or PIUnit icon and then click **Search for PIBatch**.
2. In the Search window, enter the criteria to limit your search.

Note: The **PI Server** field is automatically set to the server selected at the time you opened the window. The window will open with the same search criteria as was specified for the previous search.

3. Click **OK** to search the database.

The batches that match the specified criteria appear in the search results list.

Search for PIUnitBatches

Search for PIUnitBatches on the selected server or the server of any selected PIUnit .

1. Click **Search for PIUnitBatch**  button, or right-click the server or PIUnit  and then click **Search for PIUnitBatches**.
2. Enter search criteria in the Search window.

When the window opens, the **PI Server** and **PIUnit** fields are pre-populated, depending on how you opened the window:

- If you select **Search for PIUnitBatches** while a server icon is selected, the **PI Server** field will contain the name of the selected server and the **PIUnit** field will contain the default mask of *.
 - If you select **Search for PIUnitBatches** while a PIUnit icon is selected, the **PI Server** field will be blank and the **PIUnit** field will contain the fully qualified path of the selected PIUnit.
3. Use one of two different methods to indicate which PIUnits to search against:
 - Enter a mask to the **PIUnit** field. For example, type R-4* to search all PIUnits that begin with R-4. When you click **Search**, the search is executed on the selected server for all PIUnits that have names that match the entered PIUnit mask.
 - Enter the fully qualified path to a specific PIUnit in the **PIUnit** field to search that PIUnit. This takes the form of \\Server\PIModule\PIModule\.... The window automatically recognizes the \\ and expects the **PIUnit** field to contain a path to a PIUnit. At the same time, the window clears and disables the **PI Server** field because the server is provided by the PIUnit path.
 4. Click **Search** to execute the search.

If the search succeeds, the window closes and the results are displayed in the [Search results list](#) in the Batch Database tool.

Results from PI Batch Subsystem batches

A batch of the PI Batch Subsystem is analogous to and corresponds to the PIUnitBatch of the PI Batch Database. These Batch Subsystem Batches cannot be edited using the Batch Database tool. You must use the **piconfig** utility to edit these batches.

If your search results include PI Batch Subsystem batches, they will be represented as disabled PIUnitBatches;

the PIUnitBatch title at the top of the panel will be appended with "(BSS Batch)", and all fields will be read-only, indicated by a dimmed state. Although you cannot make changes to these batches, you can select and copy the text in these fields. Furthermore, you can copy the batch as a whole to the clipboard, and use the **Paste** button  to copy it to another PIUnit (See [Paste PI_batches](#)). The copy will result in a valid PIUnitBatch.

Copy batch items

You can use the **Copy** command to copy PIBatch, PIUnitBatch, and PISubBatch items to the clipboard.

1. In the search results pane, select a batch item.
2. Click **Copy**  to copy the selected batch item to the clipboard.

Use the **Paste** command to create a new batch item from the copy. The **Paste** command is available if the task is appropriate for the type of batch item on the clipboard.

Paste PI_batches

You can paste PI_batches if the clipboard contains a PIBatch item and if a server, PIModule, or PIUnit node is selected in the [Module Database tree](#).

Note: The paste function is intended to be equivalent to the ability to create a new PIBatch based on the PIBatch in the clipboard, because it is not possible to make an exact copy of a PIBatch; a new PIBatch will always be a unique ID and will not include any references to PIUnitBatches.

1. Click the **Paste** button  on the Module Database tree toolbar.
2. In the New PIBatch window, edit the **Batch ID, Product, Recipe, Start Time** and **End Time** fields.
Note: The New PIBatch window opens with most of the fields populated with the values from the PIBatch in the clipboard. The **PI Server** field contains the server that was selected in the Module Database tree.
3. Click **OK**.

Paste PIUnitBatches

You can paste PIUnitBatches if the clipboard contains a PIUnitBatch item and if a server, PIModule, or PIUnit node is selected in the [Module Database tree](#).

1. Click the **Paste** button  on the Module Database tree toolbar.
2. Use the New PIUnitBatch window to edit the **Batch ID, Product, Recipe, Start Time**, and **End Time** fields.
The New PIBatch window opens with most of the fields populated with the values from the PIBatch in the clipboard. The **PI Server** field contains the server that was selected in the Module Database tree.
3. Click **OK**. If the PIUnitBatch being copied contains any PISubBatches, you are prompted whether you want to copy the PISubBatches as well:
 - Click **No**, to create only the PIUnitBatch.
 - Click **Yes**, to create the PIUnitBatch and a copy of *all* of the PISubBatches.

Note: PI Batch Subsystem Batches can be copied and pasted as described above. The result of the paste is a

PIUnitBatch, not a PI Batch Subsystem batch.

Paste PISubBatches

You can paste PISubBatches if the clipboard contains a PISubBatch item and you select a PIUnitBatch or a PISubBatch in the search results list.

Note: The paste function is intended to be equivalent to the ability to create a new PISubBatch based on the PISubBatch in the clipboard, because it is not possible to make an exact copy of a PISubBatch; a new PISubBatch will always have a unique ID.

1. Select a PISubBatch item and a PIUnitBatch or a PISubBatch in the search results list.
 2. Click the **Paste**  button on the toolbar above the search results list which is located in the bottom-left of the Batch Database tool.
 3. Remove the **New PISubBatch** to edit the **Name**, **PIHeading**, **Start Time**, and **End Time** fields.
- The New PISubBatch window appears with most of the fields already populated with the values from the PIUnitBatch in the clipboard. The **PI Server** field will indicate the server that was selected in the Module Database tree and the **PIUnit** field will indicate the PIUnit that the new PISubBatch will be created under (even if the PIUnit is not the immediate parent).
4. Click **OK** to create the new PISubBatch.

If the PISubBatch being copied contains any PISubBatches, you are asked whether you want to also copy the PISubBatches:

- Click **No** to create only the PISubBatch.
- Click **Yes**, to create the PISubBatch and a copy of all of the PISubBatches.

Note: You can copy a PISubBatch (including child PISubBatches) as a child of itself; this does not pose a recursion problem.

Search results list

The search results list is located at the bottom-left of the Batch Database tool. This list shows the results of PIBatch searches and PIUnitBatch searches, as well as newly created PI_batches, PIUnitBatches, and PISubBatches. It has a toolbar located above it that applies only to the list.

When the Batch Database tool is loaded, this list is empty. Because of the hierarchical nature of batch items, this list is presented as a tree. When you search for batch items, the results are posted in the list under a different results list (tree parent node) for each search.

The search results list has the following format:

Search n: BatchItemType (count)

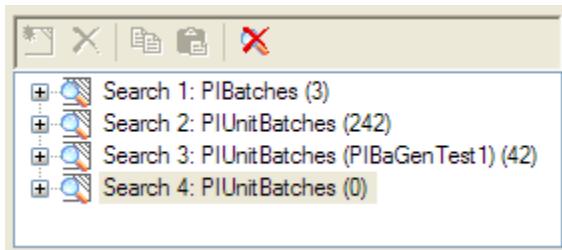
where:

- *n* is a sequential number that usually represents the number of searches executed this session.
- *BatchItemType* is the type of batch item searched for and returned.
- *count* is the number of batch items actually returned.

If a search was done against a specific PIUnit, the results list will also have the name of the PIUnit inserted before

the count:

Searchn: BatchItemType (PIUnitName)(count)



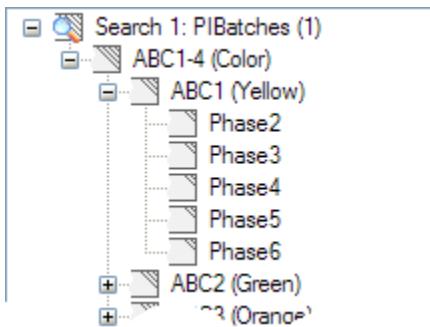
View and edit batch items from search results

From the search results list, you can view batch items found in that search and access and edit details about those items.

1. In the search results list, expand any node to see all of the batch items from that search, including each item's hierarchy (collapsed).

Batch items in the list are displayed in the following formats:

PIBatch:	BatchID (Product)
PIUnitBatch:	BatchID (Product)
PISubBatch:	Name



2. Click any batch item in this list to view details about that item in the panel on the right side of the Batch Database tool.
3. Edit details about the batch item.
The toolbar located above the panel contains options available to the selected batch item.
4. Click **Save Changes** to save your edits.
5. If you edit a batch item without saving your changes and then select a different batch item in the results list, PI SMT prompts you to save the changes:
 - Click **Yes** to save your changes and add view the selected batch item to the display.
 - Click **No** to cancel your changes for the newly selected item.
 - Click **Cancel** so that the batch item in the panel remains intact (unsaved) and the selected item in the results list does not change.

PIBatch details

If you select a PIBatch in the search results list, the right panel shows the properties of that PIBatch.

The screenshot shows a window titled "PIBatch" with the following properties:

- Batch ID:** ABC1-4
- Product:** Color
- Recipe:** RGB Blend
- Start Time:** 14/03/2005 6:23:56 AM
- End Time:** Still Running
- Server:** Toaster
- Unique ID:** 883833D4-3283-4E35-814F-A7099F7B5017
- PIUnitBatches:** ABC1, ABC2, ABC3, ABC4
- PIProperties:** A tree view showing properties for "ABC1-4 (Color)" including:
 - Color (Int32[]) = 0, 11, 22, 33, 44, ...
 - Authors (DBNull) =
 - Weight (Single) = 198

PIBatch properties

Property	Type	Description
<i>BatchID</i>	String	Usually represents a batch ID but can be used for anything.
<i>Product</i>	String	Usually represents a Product but can be used for anything. This plug-in supports only character strings.
<i>Recipe</i>	String	Usually represents a Recipe but can be used for anything. This plug-in supports only character strings.
<i>StartTime</i>	String	Represents the start time of the PIBatch. It can take the form of any Windows or PI time string, such as

Property	Type	Description
		* , T, Y, or Monday, and time arithmetic. It cannot be left blank.
<i>EndTime</i>	String	Represents the end time of the PIBatch. It can take the form of any Windows or PI time string, such as *, T, Y, or Monday, and time arithmetic. A blank field or the string <i>Still Running</i> indicates a running PIBatch.
<i>Server</i>	Read-only string	The server that the PIBatch belongs to. To move a PIBatch from one server to another, copy and paste it to the new server and delete the original. See Copy batch items .
<i>UniqueID</i>	Read-only string	The unique identifier of the PIBatch. This property is globally unique; that is, no other PIBatch on any other Data Archive server has the same ID. It is assigned at the time the PIBatch is created and cannot change.
<i>PIUnitBatches</i>	Read-only string	A short list of the PIUnitBatches that are linked to this PIBatch. To see the PIUnitBatches in detail, use the search results list and explore the PIBatch's hierarchy. PIUnitBatches can be linked to a PIBatch. See Link or unlink a PIUnitBatch to a PIBatch for details.
<i>PIProperties</i>	Tree	The PIProperties hierarchy of the PIBatch. The first node in the tree represents the PIBatch itself. A summary of each PIProperty is shown in the tree in the format <i>PIPropertyName (DataType) = Value</i> . If the data is an array, the first five elements are displayed. Use the buttons to the right of the tree to add, edit, and delete PIProperties from this hierarchy.

PIBatch edits

You can edit every PIBatch property field that is not read-only, or disabled, with the exception of copy and paste. You can select and copy every field to the clipboard, except **PIProperties**. See [PIBatch properties](#).

When you change the value of a field, the text of the field turns bold and blue. Multiple fields can be in this state:



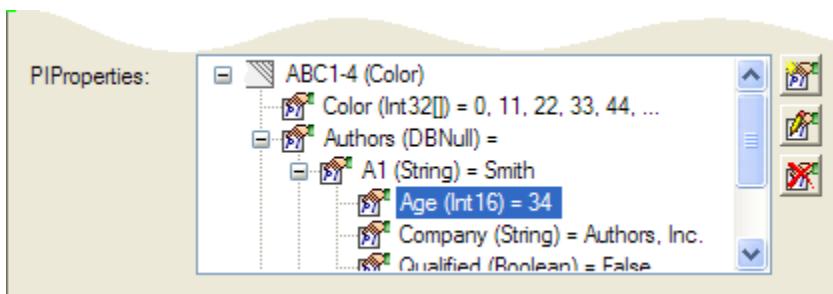
When the text of a field is bold and blue, the panel is changed, but not saved. To save the changes to the PIBatch, click **Save Changes** in the toolbar above the edit panel, or press Enter on your keyboard. When the new PIBatch values are saved to Data Archive, the text returns to black and appears in regular typeface. If the PIBatch could not be saved successfully, the panel reports the error and remains in its changed state.

If you want to cancel your unsaved changes, you can click **Revert** in the toolbar above the edit panel, or press Esc on your keyboard, to revert the changed fields back to the previous values.

Note: Changes made to PIProperties tree are effective immediately and cannot be reverted. You will not see bold blue text in the tree or any of its items.

PIProperty updates

The PIProperties of the PIBatch are displayed in a tree at the bottom of the **PIBatch** panel:



Note: Changes take effect immediately when you use the buttons to the right of the **PIProperties** field.

Use the buttons to the right of the tree to create, change, and delete PIProperties.

Icon	Name	Description
	New PIProperty	Opens the PIProperties window with all fields blank except the Property Name field, which has a value of <i>New property</i> . If you click OK , a new PIProperty is created for the PIBatch as a child of the node in

		the selected PIProperties tree.
	Edit PIProperty	Opens the PIProperties window with the editable fields filled in for the selected PIProperty. Because the node text does not always provide all of the details of a PIProperty, it is also useful to view the details. Click Cancel to ensure that no edits are made.
	Delete PIProperty	Prompts for confirmation that you want to delete the selected PIProperty and its entire list of child PIProperties when applicable. Once a PIProperty is deleted, you can't recover it.

PIUnitBatch details

If you select a PIUnitBatch in the search results list, the right panel shows the properties of that PIUnitBatch.

PIUnitBatch

Batch ID:	ABC1
Product:	Yellow
PIUnit:	PIBaGenFastTest
Procedure:	Procedure3
Start Time:	3/04/2005 12:03:07 PM
End Time:	3/04/2005 12:09:07 PM
Server:	Toaster
Unique ID:	63396C51-D626-4D81-B87E-676C7883B0DD
PISubBatches	Phase2, Phase3, Phase4, Phase5, Phase6
PIBatch	ABC1-4 (Color) ... 

PIUnitBatch properties

BatchID	This is a string that usually represents a batch ID but can be used for anything.
Product	This field is a string that usually represents a product but can be used for anything. The server is capable of storing almost any data in this field; however, this plug-in currently supports only character strings.
PIUnit	Read-only. This is the PIUnit to which the PIUnitBatch belongs. The field contains only the name of the PIUnit. It cannot be changed because a PIUnitBatch cannot be moved from one PIUnit to another. An equivalent action would be to copy the PIUnitBatch to the clipboard (using the Copy button  in the toolbar above the edit panel) and paste it to another PIUnit and then delete the original PIUnitBatch. See Copy batch items .
Procedure	This is a string that usually represents a procedure name of a PIUnitBatch but can be used for anything.
StartTime	This is a string that represents the start time of the PIUnitBatch. It can take the form of any Windows or PI time string (including *, T, Y, Monday, etc. and time arithmetic). It cannot be left blank (indicated by the bold label).
EndTime	This is a string that represents the end time of the PIUnitBatch. It can take the form of any Windows or PI time string (including *, T, Y, Monday, etc. and time arithmetic). A blank field (or the string <i>still running</i>) indicates a running PIBatch.
Server	Read-only. This is the Data Archive server that the PIUnitBatch belongs to. To move a PIBatch from one server to another, it must be copied to a PIUnit on the target server. See the PIUnit field above.
UniqueId	Read-only. This uniquely identifies the PIUnitBatch. It is globally unique (that is, no other PIUnitBatch on any other Data Archive server will have the same ID). It is assigned at the time the PIUnitBatch is created and cannot ever change. If this PIUnitBatch is copied, the copy will have a different ID, regardless of the PIUnit or server it is copied to.
PISubBatches	Read-only. This is a brief list of the PISubBatches that

	belong to this PIUnitBatch (one level only). To see these in detail, use the search results list and explore the PIUnitBatch hierarchy.
PIBatch	Read-only. This is the PIBatch that this PIUnitBatch is linked to. If the box is empty then the PIUnitBatch is not linked to a PIBatch. Although this field is disabled and cannot be edited directly, the link can be changed or removed using the buttons to the right of the box. See Link or unlink a PIUnitBatch to a PIBatch .

PIUnitBatch edits

You can edit every PIUnitBatch property field that is not read-only, or disabled, with these exceptions:

- You can edit the **PIBatch** field, although it is disabled, using the buttons to the right of the field.
- You can select and copy every field, except **PIProperties**, to the clipboard. See [Copy batch items](#).
- You cannot edit PI Batch Subsystem batches. See [Results from PI Batch Subsystem batches](#).

When you change the value of a field, the text of the field turns bold and blue. Multiple fields can be in this state:

PIUnitBatch	
Batch ID:	ABC2
Product:	Dark Yellow
PIUnit:	PIBaGenFastTest

When the text of a field is blue, the panel is changed, but not saved. To save the changes to the PIUnitBatch, click the **Save Changes** button in the toolbar above the panel, or press Enter on your keyboard. When the new PIUnitBatch values are saved to Data Archive, the text returns to black. If the PIUnitBatch could not be saved successfully, the panel reports the error and remains in its changed state.

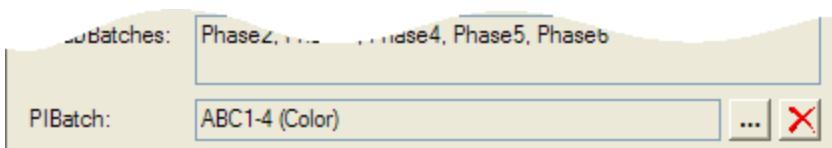
If you want to cancel your unsaved changes, you can click **Revert** in the toolbar above the panel, or press Esc on your keyboard, to revert the changed fields back to the previous values.

Link or unlink a PIUnitBatch to a PIBatch

You can link a PIUnitBatch to exactly one PIBatch; that PIUnitBatch becomes a child of the PIBatch. Unlike a PISubBatch, a PIUnitBatch can exist independently; therefore linking a PIUnitBatch to a PIBatch is optional.

1. In the search results list, click a PIUnitBatch to view details about that PIUnitBatch on the right side of the Batch Database tool.

The **PIBatch** field shows the PIBatch linked to the selected PIUnitBatch.



2. Use the buttons to the right of the **PIBatch** field to add or remove a link:

To:	Do this:
Link to a PIBatch	<ul style="list-style-type: none"> a. Click the Link to a PIBatch button to the right of the PIBatch field. b. Use the Search for PI_batches window to search for the PIBatch you want to link to. c. Select the PIBatch that you want to link to, and click Select.
Unlink from a PIBatch	Click Remove a PIBatch Link to remove a link between the PIUnitBatch and a PIBatch. If there is no link, this button is disabled.

PISubBatch details

If you select a PISubBatch in the search results, the right panel shows the properties of that PISubBatch.

PISubBatch

Name:	Phase2
PIHeading:	Operation
PIUnit:	PIBaGenFastTest
Start Time:	3/04/2005 12:04:07 PM
End Time:	3/04/2005 12:05:37 PM
Server:	Toaster
Unique ID:	2324AE41-36BF-4FD8-A7FD-FE70077A8F8E
PISubBatches	
PIUnitBatch	ABC1 (Yellow)

The panel contains fields for all of the properties of a **PISubBatch**. Each field can be selected and copied (to the clipboard), even the disabled (read-only) fields.

PISubBatch properties

Name	This is a string that usually represents the name of the PISubBatch but can be used for anything.
PIHeading	This field is a PIHeading that usually represents the PISubBatch type (for example, Phase). It is in the form of a drop-down menu which contains all of the PIHeadings from all of the PIHeadingSets from the server referenced in the Server field below.
PIUnit	Read-only. This is the PIUnit to which the PISubBatch belongs. The field contains only the name of the PIUnit. It is read-only because a PISubBatch is not directly owned by the PIUnit, it is owned by its parent (which could be a PISubBatch or a PIUnitBatch), which is ultimately owned by a PIUnitBatch. The PIUnit is here to provide context to the PISubBatch. To see the full hierarchy of this PISubBatch, see the search results list at the bottom-left of the plug-in.
StartTime	This is a string that represents the start time of the PISubBatch. It can take the form of any Windows or PI time string (including *, T, Y, Monday, etc. and time arithmetic). It cannot be left blank (indicated by the bold label).
EndTime	This is a string that represents the end time of the PISubBatch. It can take the form of any Windows or PI time string (including *, T, Y, Monday, and so on. and time arithmetic). A blank field (or the string <i>still running</i>) indicates a running PIBatch.
Server	Read-only. This is the Data Archive server that the PISubBatch belongs to.
UniqueId	Read-only. Uniquely identifies the PISubBatch. It is globally unique (that is, no other PISubBatch on any other Data Archive will have the same ID). It is assigned at the time the PISubBatch is created and cannot ever change. If this PISubBatch is copied, the copy will have a different ID, regardless of the PIUnitBatch it is copied to.
PISubBatches	Read-only. This is a brief list of the PISubBatches that belong to this PISubBatch (one level only). To see these in detail, use the search results list and explore the PISubBatch hierarchy.

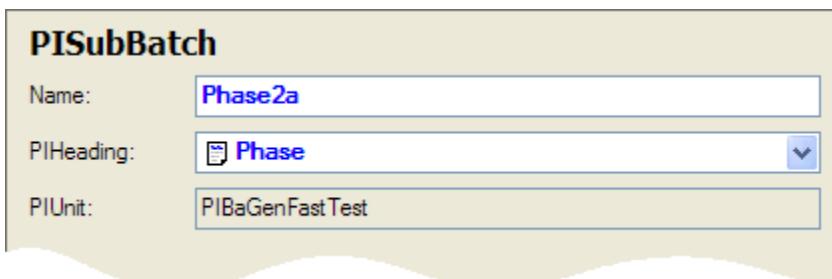
PIBatch

Read-only. This is the PIUnitBatch that this PISubBatch belongs to. Although this PISubBatch may be the child of a PISubBatch, the PIUnitBatch that ultimately owns this PISubBatch is provided here for context. To see the full hierarchy of this PISubBatch, see the search results list at the bottom-left of the Batch Database tool.

PISubBatch edits

You can select and copy all properties of a **PISubBatch**, to the clipboard, including those in the read-only, or grayed fields. See [Copy batch items](#).

When you change the value of a field, the text of the field turns bold and blue. Multiple fields can be in this state:



When the text of a field is bold and blue, the panel is changed, but not saved. To save the changes to the PISubBatch, click **Save Changes** or press Enter on your keyboard. When the new PISubBatch values are saved to Data Archive, the text returns to black and a regular typeface. If the PISubBatch could not be saved successfully, the panel reports the error and remains in its changed state.

If you want to cancel your unsaved changes, you can click **Revert** , or press Esc on your keyboard, to revert the changed fields back to the previous values.

Create a new PISubBatch

1. In the search results list, select the parent PIUnitBatch or PISubBatch .
2. Click **New PISubBatch**.
3. In the New PISubBatch window, set the basic properties of the new PISubBatch.
 - **Start Time** is the only required field of this window (indicated by the label's bold text). All other fields can be left blank.
 - If **End Time** is blank, the PISubBatch is set to a still-running state; a PISubBatch is considered to still be running if it has no end time.
 - The **PIUnit** and **Server** fields pre-populate based on the PIUnit under which the PISubBatch is to be created; the PIUnit will not necessarily be the immediate parent of the new PISubBatch.
 - The **Unique ID** field shows *Undefined Unique ID* because the unique ID of a PISubBatch is assigned at the time the PISubBatch is created.
4. Click **OK** to create the PISubBatch.

Note: A PISubBatch may contain one or more PISubBatches but creating these cannot be accomplished from this window. This must be done after the PISubBatch is created.

Delete batch items

When you delete a batch item, you delete it permanently.

1. Select a batch item (PIBatch, PIUnitBatch, or PISubBatch) in the search results list.
2. Click the **Delete** button  on the toolbar above the search results list, or select a batch item in the list and press the Delete key.
3. Click **Yes** to verify that you want to delete the batch item.

Note: When you delete a PIUnitBatch or PISubBatch, any child PISubBatches that it has will automatically be deleted at the same time. When you delete a PIBatch, it does not delete its child PIUnitBatches. The reason for this is because a PISubBatch is wholly owned by its PIUnitBatch/PISubBatch parent, whereas a PIBatch has only references to PIUnitBatches; it does not own them.

Remove search results

Periodically remove searches that you no longer need to maintain a manageable search results list.

1. Select a node in the search results list.
2. Click **Remove search results**  or press the Delete key.

Note: Removing results from the list does not delete the batch items that were under that results list, it merely removes the references to the batch items.

Batch Generator

Generation of PI Batches with the PIBaGen (PI Batch Generator) Interface requires a **PIUnit**, some setup information, and PI point data. The setup information determines which points are retrieved from the PI archive; **PIUnits** are specialized modules that are objects, or units, that represent processing equipment capable of making a batch; the **PIUnit** provides a context, such as active.

Use the Batch Generator tool to configure the PIBaGen (PI Batch Generator) Interface to the Data Archive and create and configure **PIUnits**, **PIUnitBatches**, **PISubBatches**, **PIBatches**, PI points, and other information that allows the PIBaGen Interface to gather and store the batch data on Data Archive. The Batch Generator tool also provides a view of PI Module Databases on multiple Data Archive servers.

When the Batch Generator tool is connected to a secondary server in a Data Archive collective, you can view but not edit configuration information. To make configuration changes, the Batch Generator tool must be connected to the primary server of a Data Archive collective.

Note: To migrate from Batch Generator Interface 1.x version to PI Batch Generator Interface 2.x, use the [Migration Note](#) that is provided with the Batch Generator tool.

Migration Note

The configuration required for PI Batch Generator Interface 2.x is slightly different from the configuration required for Batch Generator Interface 1.x version. For this reason, a migration utility is provided with the Batch Generator tool. This utility can be launched from the toolbar menu of the Batch Generator tool. After selecting the PIUnits to be migrated, the migration utility converts the configuration so that the batch data is generated on those PIUnits just like PIBaGen 1.x was generating batch data with one exception. The recovery done by PIBaGen 1.x interface was only for PIBatches and PIUnitBatches. However, when the PIUnit is migrated, the recovery option is set so that PIBatches, PIUnitBatches and PISubBatches are recovered.

View PIUnits and PI Module Databases

Select the Data Archive server from the list of Data Archive servers on the top left side of the PI SMT host.

Configure PIUnits

Select the Data Archive node in either the **Registered Units only** view or **MDB View** tab. The **Configuration for** page appears on the right side.

PIUnit configuration involves PIUnitBatch Configuration, PIBatch Configuration and PISubBatch Configuration. The PIUnitBatch Configuration means setting up the PI points and the attributes necessary to create PIUnitBatches. The PI points should be specified for Active Point, Unit Batch ID, Product Name and Procedure Name. The other PIUnitBatch attributes that can be configured are Active Point behavior, PIBatch/PIUnitBatch/PISubBatch recovery option, Recovery time, Evaluation Delay for PIUnitBatches, Merge Consecutive and PIUnit-specific Debug messages. These attributes will take default values if they are not specified. The entire configuration information is saved as PIAliases and PIProperties in the Configuration module Name sub-module

under the PIUnit.

Most of the PIUnit configuration properties concern the start and stop times of such batches.

A PIUnit is monitored by the PIBaGen interface only if the PIUnit is *Registered* (see [Register or Unregister PIUnits](#)). A PIUnit can be Registered or Unregistered at any time using the Batch Generator tool. This is similar to turning the **Scan** option **ON** or **OFF** for a traditional PI point.

Configure the PIBaGen interface

1. Select the **Interface** tab and use the fields in the **Interface** tab to configure the PIBaGen Interface.
2. Click **Save**  to save any configuration changes and make them effective. Configuration changes are saved in the PI Module Database.

Note: Complete and save the data in the Configuration Module Name field before completing other fields in the Interface tab; if there is no module name, no other information can be saved.

Configuration Module Name

Enter a name into the **Configuration Module Name** field that is unique to the PIBaGen interface. Without a unique name, none of the PIUnits can be configured for PIBaGen interface. Typically, this name is either *PIBaGen* or *PI-BaGen*.

Caution: Complete and save the data in the **Configuration Module Name** field before completing other fields in the **Interface** tab; if there is no module name, no other information can be saved.

Interface Debug Messages

Use the **Interface Debug Messages** option when troubleshooting:

- Turn this option **ON** to print additional interface-specific messages useful for debugging.
- Turn this option **OFF** to send standard error messages to log file.

Event Analysis Delay Time

Enter a time in seconds that the interface should wait before analyzing an event. The time is applied to the moment the interface is notified about the event and not to the event timestamp itself.

This delay is particularly useful if the same event triggers creation of **PIUnitBatch** and calculation of **BatchID** and so on. If two points, such as a point that triggers a batch to become active and another that indicates the batch name, trigger at the same time, but the batch name point requires a Performance Equation, the batch active point will trigger the batch to begin its process after the event analysis delay - typically 10 to 15 seconds.

PIBaGen Status Tag Name

The **PIBaGen Status Tag** displayed in this field is the name of the tag to which the interface writes update values. This point is created by the Batch Generator tool or the interface if it does not exist.

Retry Timeout

Use this option to set the time in minutes that the interface will retry certain failed calls to the Data Archive server. After this timeout, the interface prints an error message and continues. The timeout applies to the following error messages for this version of the interface:

- If **No Data** value is received when the interface is looking for tag values
- Server error number is **-15003**, Server error description is **Generic Item Not Found**--when editing batch data
- Server error description is **End Time Access Failed**--while closing any of the batch objects
- Server error description is **Batch ProcedureName access failed**--when setting the Procedure Name
- Server error description is **Insert failed**--when inserting a PIUnitBatch in a PIBatch

All of the above errors typically occur when the Data Archive server is not accessible or if it is in a backup state. The errors are resolved with time but to avoid being stuck in an infinite number of retries, the timeout parameter can be used.

Select **No Timeout** if you do not want to use a timeout. This is the default setting.

Maximum Events in Event Pipe

Use this option to set the maximum number of events an event pipe can hold. The interface creates two event pipes, one for all snapshot updates for all active points and another for Module Database updates. This setting applies to both those event pipes. The default value is set by the PI SDK and is 10,000 events for PI SDK version 1.3.3.304.

Maximum Stop Time

Use this option to set the time the interface waits after a second interrupt message (Ctrl+C) before the interface forces an exit. This parameter is used when the interface is run in the interactive mode. The default value is **120 seconds**.

Maximum Events in Event Pipe, **Maximum Stop Time** and **Retry Timeout** can be set either through the **Batch Generator** tool, which stores configuration data in the Module Database, or as a command line parameter. During startup, the interface uses whichever value is the non-default value. A message in the **pipc.log** file indicates which of the two values is being used. If both values are non-default, the value stored in Module Database takes precedence. While the interface is running, any changes made to the parameters through the Batch Generator tool are picked up by the interface without the need to restart the interface.

Configure PIUnitBatches

To configure a **PIUnit** to generate batch data, first configure a **PIUnitBatch** to specify all the necessary information to generate **PISubBatches** on the **PIUnit**.

1. Select the **PIUnit** to be configured and use the fields in the **PIUnitBatches** tab to configure the **PIUnitBatch**.
2. Click **Save**  to save any configuration changes and make them effective. Configuration changes are saved in the PI Module Database.

Active Point (Required)

You must specify an **Active Point** to start generating **PIUnitBatches**. An **Active Point** is the PI point that determines the start and end of a **PIUnitBatch**. Use a PI point of type **integer**, **digital**, or **string**. If this attribute is not specified or the PI point does not exist, the **PIUnitBatches**, and therefore the **PIBatches** and **PISubBatches**, are not generated.

ActivePoint Behavior

Use the **ActivePoint Behavior** option to determine how the transitions in Active Point values are interpreted. There are three possible ways the start and end of a PISubBatch can be determined: **Pulse**, **Step**, and **Include zeroth state (Continuous)**. The default value is **Step**.

- **Step:** Step type behavior of **Active Point** will cause a **PISubBatch** to start if there is a non-zero state and a **PIUnitBatch** is not currently running. If a **PIUnitBatch** is currently running when the non-zero state occurs, the **PIUnitBatch** will stop and a new **PISubBatch** will be started.
- **Pulse:** Pulse type behavior of **Active Point** means that transitions from zeroth state (or value **0** for integer type or the string that indicates zeroth state) to any other state to start a **PISubBatch** and transition from any other state to zeroth state to end an existing **PISubBatch**. All other transitions are ignored. Zeroth state is what you assign a value to - mostly strings, either a digital state or integer point are used. For example, **0,1,2**.
- **Continuous:** Continuous type behavior of **Active Point** means that all transitions result in ending the current **PISubBatch** and starting a new **PISubBatch**. This option is available only when Step is selected.
- **Strings Indicating ZerothState:** For string type active points, if **Step** or **Pulse** option is selected, the string or strings indicating the zeroth state must also be specified. These strings are case-sensitive and they should be separated by a comma. For example if the strings **Inactive,Off** and an empty string indicate a zeroth state, then this parameter should be **Inactive, ,Off**. The space between the two commas is interpreted as an empty string. Double quotes are not required. If this string is not specified, then a Pulse or Step for a string type active point is equivalent to a Continuous type.
- For string type active points, if **Step** or **Pulse** option is selected, any string change to the string will cause the **PIUnitBatch** to stop and start new **PIUnitBatch**. These strings are case-sensitive and they should be separated by a comma. For example, if the strings **Inactive, Off** and an empty string indicate a zeroth state, then this parameter should be **Inactive, ,Off**. The space between the two commas is interpreted as an empty string. Double quotes are not required. If this string is not specified, then a **Pulse** or **Step** for a string type active point is equivalent to a **Continuous** type.

Unit Batch ID Point (Optional)

Use the **Unit Batch ID Point** to specify which PI point value is used to determine the Unit Batch ID for the **PIUnitBatches** generated on the selected **PIUnit**. The Unit Batch ID does not have to be unique. If you do not specify a valid PI point for this attribute, the Unit Batch ID for a **PIUnitBatch** generated would be empty.

Product Name Point (Optional)

Use the **Product Name Point** to specify which PI point value is used to determine the Product name for the

PIUnitBatches generated on the selected PIUnit. The Product name does not have to be unique. If you do not specify a valid PI point for this attribute, the Product name for a PIUnitBatch generated would be empty.

Procedure Name Point (Optional)

Use the **Procedure Name Point** to specify which PI point value is used to determine the Procedure name for the PIUnitBatches generated on the selected PIUnit. The Procedure name does not have to be unique. If you do not specify a valid PI point for this attribute, the Procedure name for a PIUnitBatch generated would be empty.

Evaluation Delay

Use the **Evaluation Delay** to specify the time that the interface should wait before it evaluates the values from the PI points specified above. The PIUnitBatch starts at the time the **Active Point** indicates the start of the PIUnitBatch. However, other attributes of PIUnitBatch - **Unit Batch ID**, **Product Name**, **Procedure Name**, **PIBatch**, **PIBatch Product Name**, and **PIBatch Recipe Name** are evaluated after waiting for the number of seconds specified in **Evaluation Delay**.

If a PIUnitBatch end event is received while the interface is waiting for the evaluation delay to elapse, the PIUnitBatch properties are evaluated at the end event time and the PIUnitBatch is started and stopped. If the option **Evaluate at the end of each UnitBatch** is selected, then the attributes are evaluated at the end of the PIUnitBatch and the **Evaluation Delay** value specified is ignored. The default value for **Evaluation Delay** is **0** seconds.

Recovery Options

Use this option to indicate whether the **PIUnitBatches**, **PIBatches**, and/or **PISubBatches** are recovered for each **PIUnit** in case the interface is shut down and restarted.

- The default option **Recover all PIBatch objects** recovers **PIBatches**, **PIUnitBatches**, and **PISubBatches** on the selected **PIUnit**.
- The **Recover only PIBatches and PIUnitBatches** option recovers only the **PIBatches** and **PIUnitBatches** on the selected **PIUnit**. The **PISubBatches** are not recovered.
- The **Do not recover anything** option will not recover any **PIUnitBatches** and **PISubBatches** on the selected **PIUnit**. During recovery, if there is a running **PIUnitBatch**, the recovery starts from the start time of that running **PIUnitBatch** irrespective of the **Recovery Time** specified. If there is no running **PIUnitBatch**, the interface searches for **PIUnitBatches** on the **PIUnit** up to **Recovery Time** into the past. If the search finds **PIUnitBatches** within that period, recovery is done from the end time of the last **PIUnitBatch**. If there are no **PIUnitBatches** in the search results, then recovery is done for **Recovery Time** into the past. This option helps in reducing the time it takes to recover if the recovery for that **PIUnit** is deemed not important.

Note: If **PISubBatches** are recovered, the **PISubBatches** on the running **PIUnitBatch** are removed and recreated based on the archive events. Since the interface depends on snapshot events during real-time operation and looks at archive events during recovery, it is possible that some of the **SubBatches** that are generated during real-time operation might be missing after recovery. This condition occurs only under certain circumstances. If there is a running **PIUnitBatch** at the time of recovery and if that **PIUnitBatch** has **PISubBatches** (running or not), those **PISubBatches** are removed during recovery. If the start events for those **SubBatches** were not archived, then those **PISubBatches** will be missing after recovery. This occurs only for

that one **PIUnitBatch**. This situation can be avoided by setting the compression parameters on the Active Point for the **SubBatches** such that all the start and stop events for the **PISubBatches** are archived.

Recovery Time

Recovery Time is the time into the past starting from current time that the interface should attempt to recover events for each PIUnit.

When the interface is shutdown for a certain period of time and restarted, it attempts to recover batches occurred during the shutdownperiod. The recovery is done by checking the archivedvalues for the Active Points. Recovery Time is the time that the interface looks back into the past from the restart time of the interface for the archived events. Some batch data may not be generated if the **Recovery Time** is shorter than the shutdown period.

During recovery, if there is a running PIUnitBatch, the recovery starts from the start time of that running PIUnitBatch irrespective of the **Recovery Time** specified.

If there is no running PIUnitBatch, the interface searches for PIUnitBatches on the PIUnit up to **Recovery Time** into the past. If the search finds PIUnitBatches within that period, recovery is done from the end time of the last PIUnitBatch.

If there are no PIUnitBatches in the search results, then recovery is done for **Recovery Time** into the past. For example, if the **Recovery Time** is 4 days, and if the shutdown period is less than 4 days and the last PIUnitBatch on a PIUnit was 2 days ago, then the recovery for that PIUnit is done only for the past 2 days. In this case, the recovery for each PIUnit depends on when the last PIUnitBatch started.

If the shutdown period is 7 days, then PIBaGen will recover batch data only for the past 4 days. There will be no batch recovery for the first 3 days of the shutdown period. This option helps in reducing the time it takes to recover if the shutdown time is long and recovery during the entire shutdown time is not critical. The default recovery time is 4 days.

PIUnit Debug Messages

Use the **PIUnit Debug Messages** option when troubleshooting:

- Turn this option **ON** to print additional interface-specific messages useful for debugging.
- Turn this option **OFF** to send standard error messages to log file.

PI SubBatch Configuration

The **PISubBatch Configuration** specifies the necessary PI points and attributes to generate PISubBatches. It involves creating the SubBatch hierarchy. The important nomenclature to understand the PISubBatch configuration is defined here.

- **SubBatch**

Represents one item in the SubBatch Hierarchy. Each SubBatch has its own **SubBatch Active Point** which acts as a trigger for PISubBatches. Each SubBatch does not necessarily represent only one PISubBatch. The number of PISubBatches will depend on the **SubBatch Active Point** and the **Active Point Behavior** chosen for that SubBatch.

- **SubBatch Title**

The name given to represent a SubBatch in the hierarchy. Two SubBatches at the same level cannot have the same SubBatch Title.

- **SubBatch Hierarchy**

The tree-like structure to represent all the SubBatches. Each SubBatch can have only one parent SubBatch but can have many child SubBatches.

- **SubBatch Level**

Represents the depth in the hierarchy where the SubBatch belongs. The first Level is the topmost level. SubBatches at the first level do not have a parent SubBatch.

Each SubBatch defined in the hierarchy requires a SubBatch Title and SubBatch Active Point. Active Point Behavior option should also be specified. The name for the PISubBatch can be simply the SubBatch Title or the value of the active point itself or a separate PI point can be specified whose value at the start time of the PISubBatch is used. PISubBatch Configuration also involves setting an optional PIHeading set for the entire SubBatch hierarchy.

1. Select the PIUnit to be configured and select the **SubBatches** tab.
2. Select the **PIUnit** to be configured and use the fields in the **PIUnitBatches** tab to configure **PIUnitBatches**.
3. To save and apply any configuration changes, click **Save** (). Configuration changes are saved in the PI Module Database.

Add New SubBatch

1. Select the parent level and click the **New** () button to define a new SubBatch.
2. Enter the SubBatch Title.

Delete SubBatch

1. Select the SubBatch to be deleted and click the **Delete** () button. This SubBatch and its children are deleted.
2. It is important to save these changes to the PI Module Database.

Refresh SubBatch Hierarchy

The **Refresh** button on the SubBatch tab toolbar refreshes the SubBatch Hierarchy. If there were any changes made to the SubBatch Hierarchy, you are prompted to save the changes.

SubBatch Active Point (Required)

SubBatch Active Point is the PI point that determines the start and end of a PISubBatch. The **SubBatch Active Point** can be set either by typing a PI point name or by using the tag search button. The PI point can be of integer, digital, or string type. It is required to specify a SubBatch Active Point to start generating PISubBatches. If this attribute is not specified or the PI point does not exist, then the PISubBatch and its child PISubBatches are

not generated. Along with the SubBatch Active Point, the **Active Point Behavior** option for the selected SubBatch determines how PISubBatches are generated under a PIUnitBatch.

A PISubBatch starts only if the parent PISubBatch is running. The first levels of PISubBatches start only if the PIUnitBatch is running. A PISubBatch ends if either the **SubBatch Active Point** indicates the end or if the PIUnitBatch ends and does not necessarily end if the parent PISubBatch ends.

ActivePoint Behavior

Use the **ActivePoint Behavior** option to determine how the transitions in Active Point values are interpreted. There are three possible ways the start and end of a PISubBatch can be determined: **Pulse**, **Step**, and **Include zeroth state (Continuous)**. The default value is **Step**.

- **Step:** Step type behavior of **Active Point** will cause a **PISubBatch** to start if there is a non-zero state and a **PIUnitBatch** is not currently running. If a **PIUnitBatch** is currently running when the non-zero state occurs, the **PIUnitBatch** will stop and a new **PISubBatch** will be started.
- **Pulse:** Pulse type behavior of **Active Point** means that transitions from zeroth state (or value **0** for integer type or the string that indicates zeroth state) to any other state to start a **PISubBatch** and transition from any other state to zeroth state to end an existing **PISubBatch**. All other transitions are ignored. Zeroth state is what you assign a value to - mostly strings, either a digital state or integer point are used. For example, **0,1,2**.
- **Continuous:** Continuous type behavior of **Active Point** means that all transitions result in ending the current **PISubBatch** and starting a new **PISubBatch**. This option is available only when Step is selected.
- **Strings Indicating ZerothState:** For string type active points, if **Step** or **Pulse** option is selected, the string or strings indicating the zeroth state must also be specified. These strings are case-sensitive and they should be separated by a comma. For example if the strings **Inactive,Off** and an empty string indicate a zeroth state, then this parameter should be **Inactive, ,Off**. The space between the two commas is interpreted as an empty string. Double quotes are not required. If this string is not specified, then a Pulse or Step for a string type active point is equivalent to a Continuous type.
- For string type active points, if **Step** or **Pulse** option is selected, any string change to the string will cause the **PIUnitBatch** to stop and start new **PIUnitBatch**. These strings are case-sensitive and they should be separated by a comma. For example, if the strings **Inactive, Off** and an empty string indicate a zeroth state, then this parameter should be **Inactive, ,Off**. The space between the two commas is interpreted as an empty string. Double quotes are not required. If this string is not specified, then a **Pulse** or **Step** for a string type active point is equivalent to a **Continuous** type.

SubBatch Name

When a PISubBatch is added, the name of the PISubBatch is obtained either from a PI point that is configured in the **SubBatch Name** section or it is the same as the SubBatch Title. A PISubBatch cannot be created without a name. Therefore, if a separate PI point is specified then it should be a valid PI point and that PI point should have good values at the start time of the PISubBatch.

- **Use ActivePoint Value:** Selecting this option will result in using the value of the ActivePoint for the SubBatch at the start time of the SubBatch as the name of the new PISubBatch.
- **Use SubBatch Title:** Selecting this option will result in using the SubBatch Title (the name used to represent this SubBatch in the SubBatch hierarchy) as the name of the new PISubBatch.

- **Use This PI Point Value:** Selecting this option will result in using the value of the specified PI point for the SubBatch at the start time of the SubBatch as the name of the new PISubBatch.

PIHeading Set

A **PIHeadingSet** can be selected from the available PIHeading Sets in the **PIHeading Set** combo box.

A **PIHeading Set** is optional and provides a collective name for each level of PISubBatches defined in the SubBatch Hierarchy. For example, according to the S88 standards, the first level of PISubBatches is called **Operations** and the second level is called **Phases**. To provide flexibility in naming these levels, PIHeading Sets are used.

A **PIHeading Set** consists of PIHeadings and a level corresponding to the PIHeading. There can be only one PIHeading at each level. Therefore, if a PIHeading Set consisting of PIHeadings called **Operations**, **Phases**, and **Steps** at levels 1, 2, and 3 respectively is specified for a particular PIUnit, it means that all the PISubBatches defined at the first level are Operations, all the PISubBatches defined at the second level (it does not matter which PISubBatch is their parent PISubBatch) are Phases, and all the PISubBatches defined at third level are Steps. If there is no **PIHeading Set** specified or if there is no corresponding PIHeading for any level, this attribute of the **PIHeading Set** is left empty.

PIBatch Configuration

The *PIBatch configuration* includes specifying PIBatch Index point, PIBatch Product Name point, PIBatch Recipe Name point, and PIBatch Search Time.

1. Select the **PIUnit** to be configured and use the **PIBatches** tab.

Note: The **PIBatch Configuration** tab is available only if the **Active Point** for the PIUnitBatches is an existing PI point.

2. To save and apply any configuration changes, click **Save** (). Configuration changes are saved in the PI Module Database.

PIBatch Index Point (Optional)

The **PIBatch Index Point** specifies the PI point whose value determines the name of the PIBatch to which the PIUnitBatch generated would belong. A valid PI point for this attribute serves as the source for the PIBatch name. The point does not act as **Active Point** for the PIBatches, in other words, transitions in this PI point will not trigger start and stop of PIBatches. A PIBatch is started only when a PIUnitBatch is started and ends when all the running PIUnitBatches in that PIBatch end.

If a new PIUnitBatch is added after the PIBatch ends, the end time is updated with the new PIUnitBatch end time. When a PIUnitBatch starts, the value of the **PIBatch Index Point** is used as name of the PIBatch as one of the search criteria for PIBatches.

If there are no matching PIBatches within the **PIBatch Search Time**, a new PIBatch is created. If a valid PI point is not specified for this attribute, the PIUnitBatch does not belong to any PIBatch. See [PIBatch Search Time](#) for more details on how PIBatches are generated.

PIBatch Search Time

The **PIBatch Search Time** is the time into the past and future (during recovery) from the unit batch start time that the interface searches the PI Batch Database for PIBatches when a **PIUnitBatch** starts.

If there is a value for the PIBatch Index point at the PIUnitBatch start time (plus evaluation delay), then the interface will first search the PI Batch Database for a PIBatch with the PIBatch name same as the PIBatch Index point value, PIBatch product name and PIBatch recipe name within the time period (PIUnitBatch Start Time + Evaluation Delay – PIBatch Search Time) and (PIUnitBatch start time + Evaluation Delay + PIBatch Search Time).

If a PIBatch with the search criteria is found in the PI Batch Database, then the PIUnitBatch will be added to the PIUnitBatches collection in that PIBatch or else a new PIBatch will be created. If the search results in more than one PIBatch, then the following rules are used to select the PIBatch.

- If there is *only one running PIBatch* in the search period, then that PIBatch is selected irrespective of the start time of the PIBatch.
- If there is *more than one running PIBatch*, then the one that has the start time between (PIUnitbatch start time + Evaluation delay - PIBatchSearchTime) and (PIUnitbatch start time + Evaluation delay + PIBatchSearchTime) is selected. If there is more than one running PIBatch in this range, then the one with the latest start time is selected.
- If there is *no running PIBatch* but the search results in PIBatches, then the one that has start time between (PIUnitbatch start time + Evaluation delay - PIBatchSearchTime) and (PIUnitbatch start time + Evaluation delay + PIBatchSearchTime) is selected. If there is more than one PIBatch with start time in this range, the one with the latest start time is selected.
- If there is *no running PIBatch* but the search results in PIBatches and there is no PIBatch that has start time between (PIUnitbatch start time + Evaluation delay - PIBatchSearchTime) and (PIUnitbatch start time + Evaluation delay + PIBatchSearchTime) then the one that has the latest start time is selected.
- If *no PIBatch* is found with the search criteria, then a new PIBatch is added.

The **PIBatch Search Time** can be seen as the approximate duration of the PIBatch to which the PIUnitBatches in this PIUnit belong. The default value is 4 hrs. It is recommended that this value be slightly greater than one batch duration and less than twice the duration of a typical batch. If the search time is large and if the PIBatch IDs are not unique, then there is a possibility of having multiple results during the search and the PIUnitBatch may not end up in the right PIBatch.

PIBatch Product Point (Optional)

The **PIBatch Product Point** specifies the PI Point whose value determines the PIBatch Product name. This could be different from the Product name of **PIUnitBatches**. It is possible to have intermediate products on the PIUnitBatches and therefore separate names for **PIUnitBatches** and PIBatches is necessary. If a valid PI point for this attribute is not specified, the PIBatch Product name would be empty.

PIBatch Recipe Point (Optional)

The **PIBatch Recipe Point** specifies the PI point whose value determines the PIBatch Recipe name. This refers to the Recipe used to make the batch. If a valid PI point for this attribute is not specified, the PIBatch Product name would be empty.

Save Configuration

Any changes made to the configuration should be saved before they can be effective. The **Save**  button saves the changes in the configuration in the PI Module Database.

Register or Unregister PIUnits

The interface will not monitor a PIUnit and generate batch data if the PIUnit is not registered. Registering a PIUnit is like turning the SCAN option **ON** for a PI point. A PIUnit can be registered or unregistered at any time.

Select the PIUnit to be registered and click the **Register**  button. To un-register a PIUnit, select the PIUnit and click the **Unregister**  button.

Migration

To launch the migration utility, select a server node and click  on the toolbar. A new window called Migration Chart is displayed if there are any PI points belonging to PIBaGen 1.x on that Data Archive server and if they are not yet migrated. If the userint1 attribute for a PI point belonging to PIBaGen 1.x is set to 1, it is considered that the corresponding PIUnit is already migrated and is not loaded into the Migration Chart. The migration utility sets the userint1 to 1 during the migration process. Only those PIUnits that do not have the **Error**  icon next to the configuration point can be migrated. If there is an error sign, the **Migration Status** column explains why the PIUnit cannot be migrated. See [Migration errors in PI SMT](#) for common errors that arise before and during migration. While the selected PIUnits are migrated, the migration utility follows these steps for each PIUnit:

- Rename the configuration module used by PIBaGen 1.x to *PIBaGen 1.x Configuration-Migrated*.
- If the renaming process is successful, the PIUnit is considered to be migrated and userint1 for the PIBaGen 1.x PI point is set to 1.
- A new module is then added under the PIUnit with the name as *Configuration Module Name*.
- All the PIUnitBatch and PISubBatch configuration is then copied to the newly added Configuration Module.
- If PISubBatches are configured in PIBaGen 1.x, then each PIAlias in each PIModule under **SubBatchSupport** module represents one SubBatch in PIBaGen 2.x configuration. Therefore, one PIModule with the name format as *H:P* is added, where *H* stands for the PIHeading name and *P* stands for the PI point name (data source for that PIAlias). The same PI point is also used for deriving the PISubBatch name.
- The *ActivePoint Behavior* is set to *Continuous* because the *Step* behavior for SubBatches in PIBaGen 1.x is now behaved like the new *Continuous* option.
- The PIUnit is registered if the corresponding PIBaGen 1.x PI point had the Scan attribute set to 1. In such a case, the PI point Scan attribute is set to zero after registering the PIUnit.
- If there are any errors during the migration process, the corresponding row is highlighted and the error messages are shown at the bottom of the Migration Chart and are also sent to the log file. If the migration is successful, the **Migration Status** column shows *Migration Completed*.

Migration errors in PI SMT

This section provides a list of common errors that occur during migration and what needs to be done to resolve those errors.

Error Message	Description	Action
Cannot Migrate: Error Retrieving the PIUnit	The PIUnit cannot be found at the specified location.	Check the path in the ExDesc attribute for the PIPoint
Cannot Migrate: Invalid Module Path	Path in ExDesc does not start with \\	Modify the ExDesc for the corresponding PIPoint
Cannot Migrate: PIModule is not a PIUnit	PIModule does not have IsPIUnit set to true	IsPIUnit should be set to true for the interface to generate batch data
Cannot Migrate: Configuration module not found	PIUnit is probably not configured for PIBaGen 1.x	Configure this PIUnit for PIBaGen 2.x. There is no need to migrate this PIUnit
Cannot Migrate: Already has SubBatch Hierarchy	PIUnit has second level SubBatches already defined.	This PIUnit is probably migrated. Use the Batch Generator Plug-in 2.x to make configuration changes.
Cannot Migrate: Does not contain PIBaGen1.x configuration	This PIUnit has SubBatch configuration that is not compatible with PIBaGen 1.x	This PIUnit is probably migrated. Use the Batch Generator Plug-in 2.x to make configuration changes.
Migration completed but with errors: See below for details	There were errors while migrating.	Check the Error Messages box at the bottom of the Migration Chart
Error Registering the Unit	PIUnit was migrated but not registered	Register using the Batch Generator Plug-in
Error changing attributes for configuration point	Error setting either the Scan to zero or userint1 to 1 for the configuration point.	Use DataLink or Point Builder to change those two parameters for that PIPoint.
Error renaming <i>configmodulename</i> configuration module to PIBaGen 1.x Configuration- Migrated	Error renaming the PIBaGen 1.x configuration module.	Check if a PIModule named PIBaGen 1.x Configuration-Migrated doesn't exist already. If it did, probably the PIUnit is already migrated. Change the userint1 attribute to 1 for the corresponding PIPoint.

Add PIModules and PIUnits

1. From the Batch Generator tool, you can add PIModules and PIUnits, but you cannot delete them.
2. Under System Management Tools, click **Batch > Batch Generator**.
3. Click the **MDB View** tab.
4. Do any of the following:
 - To add a PIModule, select a PIModule node or Data Archive server node and click the **New Module**  button on the toolbar.
 - To make the PIModule a PIUnit, right-click the module node and select **Convert to PIUnit**.
 - To convert a PIUnit back to PIModule, right-click the module and select **Convert to PIModule**.

Refresh MDB and Registered Units Only view

To refresh the current view, click **Refresh** .

If you made changes to any of the PIUnits, you are prompted to save the changes.

Right-click menu options

Right-click options are enabled and disabled depending on selected node. There are two right-click menus.

Right-click on the icons in the **MDB View** and **Registered Units Only** tabs to get these options:

Text	Description
Add New PIModule	Add a new PIModule under the selected node.
Register	Register the selected PIUnit.
Unregister	Unregister the selected PIUnit.
Refresh	Refresh the current view.
Convert to PIUnit	Convert the selected PIModule to a PIUnit.
Convert to PIModule	Convert the selected PIUnit to a PIModule.
Save	Save the changes in the configuration.
Migrate PIBaGen1.x to 2.x	Launch migration utility for PIBaGen 1.x PIUnits.

Right-click on the **SubBatch Hierarchy** icons in the **PISubBatches** tab to get these options:

Text	Description
Add New SubBatch	Add a new SubBatch under the selected SubBatch node.
Delete	Delete the selected SubBatch and its children.
Refresh	Refresh the SubBatch configuration.
Save	Save the changes in the configuration.

Current Values

Use the Current Values tool to view data for all requested tags, respective servers, and Data Archive collectives, event values and timestamps, and the point descriptor and engineering units.

Current values of future PI points

PI Server 2015 and later version support future PI points to store predicted data past current time. For both historical and future PI points, current value is extrapolated to the end-of-stream value when the step option is set to '1' and interpolated when the step option is set to '0'.

Display current values

The Current Values tool displays the current value for any point on any connected Data Archive server, and can display new values as they are received by the server.

Note: These values are derived from the PI Snapshot. For more information about the PI Snapshot, see the System Management Guide.

1. Click the **Tag Search** button  in the Current Values tool to display the Tag Search dialog box, or right-click the data and select **Tag Search**.
2. Enter **Tag Mask** field information and click **Search** to search for matching points.
3. Select the desired points in the list and click **OK**. Ctrl-click or Shift-click to select multiple points and move them to the Current Values tool.
4. Right-click on column headings and select or de-select the column names to customize the display.

Update in real-time or freeze values

You can customize the display to update the values as they come into Data Archive, or capture a display that reflects current values.

1. Click the **Start** button  in the Current Values tool to retrieve continuously updated event values for listed tags.
2. Click the **Stop** button  in the Current Values tool to freeze the display with current event values. Tag data is not updated to reflect current values in Data Archive.

Customize the display

To further customize the points list in the Current Values tool, remove one or more points.

1. To remove a single point, select a point in the Current Values tool and click , or right-click the data and

select **Remove**.

2. To remove all listed points, select a group of points in the Current Values tool and click , or right-click the data and select **Remove All**.

Export current values data

To export data to a CSV file, select the data in the Current Values tool and click , or right-click the data and select **Export List**.

Refresh the current values

To refresh the data in the Current Values tool, select the data and click , or right-click the data and select **Refresh**.

Current Values quick reference

Toolbar items and right-click options may be enabled or disabled based on the status of a selected item.

Goal	Right-click Option	Toolbar Icon
Search for PI tags to add to the Current Values window	Tag Search	
Remove a selected tag from the Current Values window	Remove	
Remove all tags from the Current Values window	Remove All	
Continuously update listed tags with new event data	Start	
Freeze updates to listed tags, retaining currently-displayed data	Stop	
Export data for listed tags to a CSV file	Export	
Refresh event values for all listed tags to display current data	Refresh	
Help	Help	

Database Security

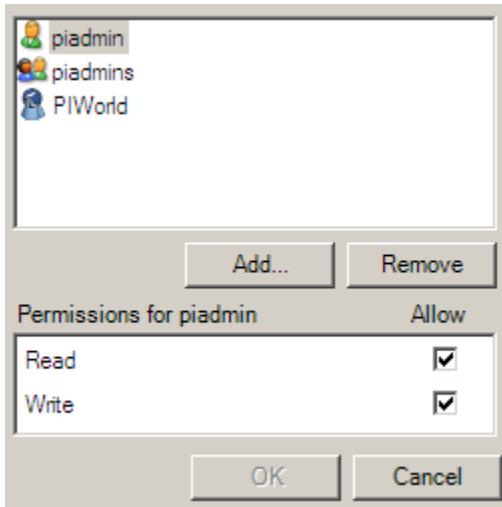
The Database Security tool controls read and write access to Data Archive administrative functions, such as the ability to create and edit PI points, run backups, create new PI identities, and so on. Permissions for specific points and modules are configured on the objects themselves. (Default access settings for points are determined by the Database Security tool.)

Edit database security settings

To edit the access permissions for an entry in the list, double-click it.

The Properties dialog box appears.

- For Data Archive version 3.4.380 and later, the dialog box shows access permissions like this:



- For Data Archive versions earlier than 3.4.380, the dialog box shows access permissions like this:



Database security for PI Server 2010 and later versions

PI Server 2010 (Data Archive 3.4.380) and later versions of Data Archive provide a more flexible access

permissions model than the owner/group/world model of previous versions.

Updated access permission model

Each entry in the Database Security editor has an Access Control List (ACL) string that defines the access permissions for that entry (all access permissions on the Data Archive server are defined by an ACL). The ACL lists each identity (or user or group) for which access permissions are set and what level of access that identity has. For example, the ACL for an entry might look like this:

```
Identity1:A(r,w) | Identity2:A(r,w) | Identity3:A(r) | IdentityN:A(r,w)
```

Access permissions for each PI identity are separated by a pipe (|) symbol. Each entry consists of the PI identity name, then a colon (:) followed by the *access specifier*. The access specifier is defined in the format: A(*r,w*). The A in this notation stands for *Allow* and "r,w" indicates the allowed access rights – read and write, in this example.

The possible levels of access are *read* and *write*. The possible access rights string can be "r", "w", "r,w" or "" (null). Note that there is no level for *deny*, as there is in Windows.

Users that belong to more than one Windows group might be mapped to multiple PI identities, PI users, or PI groups. In this case, they get the cumulative access permissions for all the associated PI identities, users, and groups. In addition, unless PIWorld is disabled, all users get the access permissions for PIWorld.

Set access permissions with the Database Security tool

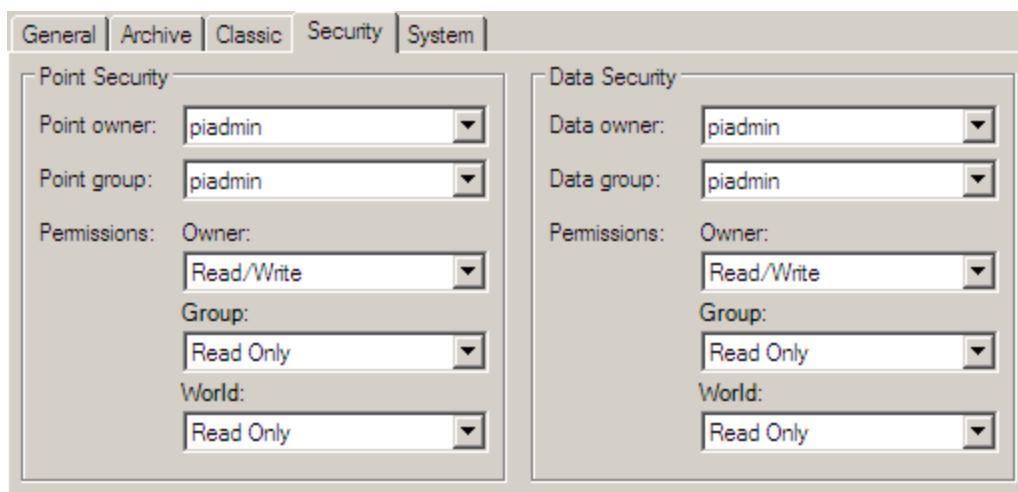
1. Under **Collectives and Servers**, select the server.
2. Under **System Management Tools**, select **Security > Database Security** to open the Database Security tool.
3. Double-click the table that you are interested in to open its security dialog box.
4. Select the PI identity, PI user, or PI group that you want to set access permissions for. If the PI identity, user, or group does not appear in the list, click **Add** to add it.
5. Select the appropriate check boxes to assign read and/or write access to that PI identity, user, or group.
6. Click **OK** to save the changes.

Set access permissions for versions earlier than PI Server 2010

Versions earlier than PI Server 2010 (Data Archive 3.4.380) use the owner/group/world model for access permissions. Each object can have an owner, which must be a PI user and a group, which must be a PI group. You can set access permissions for the owner, access permissions for the group, and the access for everyone else (called world access).

Perform the following steps to set access permissions for a point

1. Select the point in Point Builder, then click the **Security** tab.



2. Use the drop-down lists to choose an owner and a group and to specify access permissions for the owner and group. The possible levels of access are *read* and *write*. The possible access rights string can be "*r*", "*w*", "*r,w*" or "" (null). Note that there is no level for *deny*, as there is in Windows.

Set default access for new PI points

On Data Archive version 3.4.380 and later, all new points are created with the same access points that are defined for the Point Database. The Point Database is represented as PIPOINT in the Database Security tool. If you are creating many new points at once, first set the access permissions on PIPOINT as you would like them to be for the new points. After you create the new points, you can change the settings back, if necessary.

Export database security settings

To review the access permissions for the Database Security editor, export them into a file by clicking **Export** . The file is a list of comma-separated values that you can open in a spreadsheet. The contents of the file look different depending on the version of the server.

The file lists:

- The server
- The owner and group
- An access string specifying the permissions for owner, group, and world
- The access permission model (Data Archive version 3.4.380 and later)
- A description of the entry (Data Archive version 3.4.380 and later)

See also [Updated access permission model](#).

For example:

```
PIPOINT,MyPIDataArchiveServer,,piadmin,piadmins,o:rw g:rw w:r,piadmin: A(r,w) | piadmins: A(r,w) | PIWorld: A(r),Controls top-level access to Points, Point Classes, and Attribute Sets
```

The older owner/group/world access information is included for backward-compatibility with older versions of SMT and other administrative tools. These tools cannot interpret the new ACLs.

The owner and group might be listed as *PIUserIncompatible* and *PIGroupIncompatible*. These identities are displayed as the owner and group when the server cannot translate new-model access permissions into the owner/group/world model. For more information, see [PIUserIncompatible user and PIGroupIncompatible group](#).

PIUserIncompatible user and PIGroupIncompatible group

Old-model administrative tools (Point Builder, SMT, and so on) expect access permissions in the owner/group/world model. Data Archive (versions 3.4.380 and later) uses a different access permissions model. When old administrative tools try to view or modify access permissions on a new Data Archive server, Data Archive attempts to determine an owner and group.

If Data Archive cannot determine an appropriate owner or group, then it defines the owner as the user **PIUserIncompatible** and the group as **PIGroupIncompatible**. On Data Archive version 3.4.380 and later, **PIUserIncompatible** and **PIGroupIncompatible** are provided for this purpose. They do not appear in the list of users and groups by default. To show them, click the **Options** button.

If the access permissions for the object meet the following format, then Data Archive can determine an owner and group for the object:

- One (and only one) PI user
- One (and only one) PI group
- PIWorld identity

If any of these conditions is not met, Data Archive cannot determine an owner and group and it uses the **PIUserIncompatible** user and **PIGroupIncompatible** group instead. Therefore, these names are displayed as the owner and group in older tools. You can change the names to something more descriptive, if you want.

Digital States

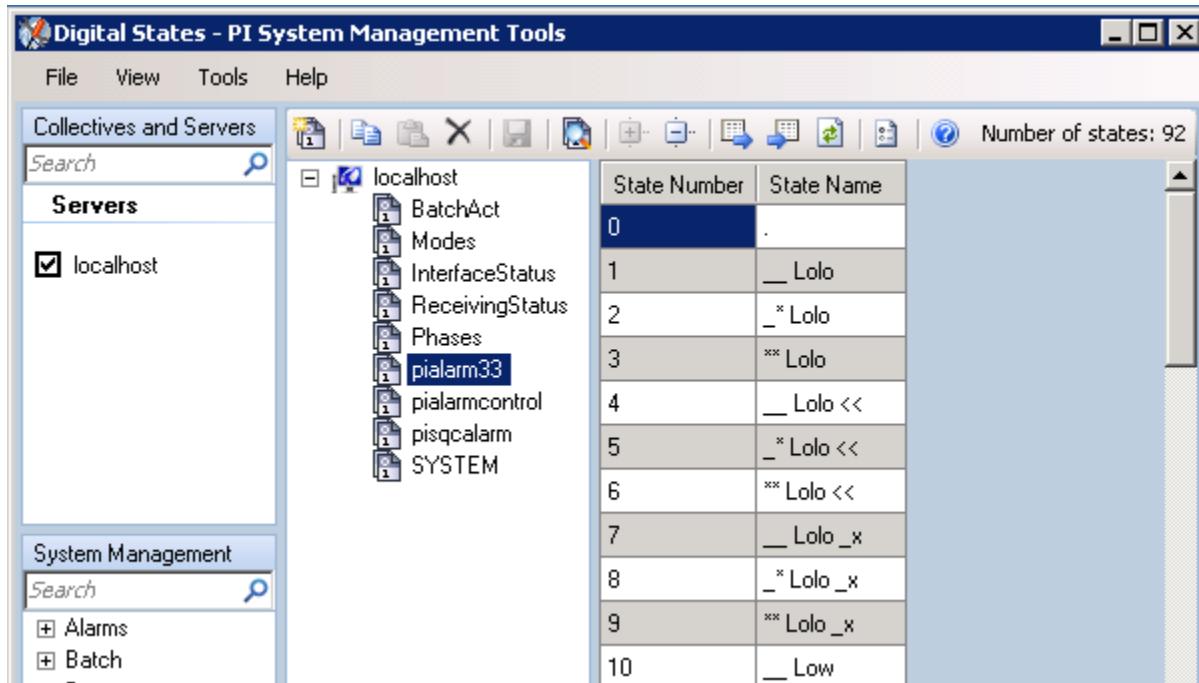
Use the Digital States tool to view, create, edit, and delete digital states on a Data Archive server. Digital states translate numeric PI point values into string data and allow you to associate PI points with a discrete status or state. For example, you can create PI points to monitor the temperature of a tank and configure those points to show a state of Red for temperatures exceeding 120 degrees, Green for temperatures below 88 degrees, and Yellow for temperatures between 88 and 120 degrees.

The points that reflect digital states are known as digital points. They are organized into groups known as digital state sets.

The Digital States tool displays digital state sets for each connected Data Archive. Connected servers and collectives are listed in the pane to the right of the Data Archive list and list associated digital state sets under each respective server nodes. The types of digital state sets are: system, built-in, and custom.

View digital state set properties

1. In the **Collectives and Servers** pane, select and connect to the server that contains the digital states.
2. Click the + sign to expand the selected server to display a list of associated digital state sets.
3. Select the name of a digital state set to display the set's properties.



System digital state set

The System digital state set is the default set for Data Archive. It contains over 300 pre-defined digital states that may apply to any point. Examples are Point Created (Pt Created), I/O Timeout, No Data and Archive Offline, Over

Range and Under Range. Use the Digital States tool to verify that the System digital state set is up to date.

We recommend that you keep changes to the digital state set to a minimum. Add additional digital states when directed to do so by documentation, or by a Technical Support Engineer. You can also translate the states into another language without changing their meaning.

Note: Digital points should use a custom digital state, not the System digital state set. You do not need to include system states in custom digital states because the system states contained in the System State Set are used where needed by Data Archive.

Note: To see the defined digital state sets in SMT, open SMT and select **Points > Digital States > SYSTEM**.

Note: You may also use piconfig using the commands: @tabl pistate, @ostr *, @ends. If you use piconfig, it does not tell you which digital set uses what state (a single state can belong to many digital sets). Also note that digital sets are not the same as digital states. If you want to find out the list of digital sets in the system using piconfig, use the following commands: @tabl pids, @ostr *, @ends.

Built-in digital state sets

Built-in digital state sets are digital state sets that the Data Archive server contains for its built-in features such as the PI Alarm Subsystem, SQC Alarm, and System Digital States.

Other built-in states are included as examples and for use by the default PI points:

- BatchAct
- InterfaceStatus
- Modes
- Phases

Custom digital state sets

Custom digital state sets are digital state sets that you can create to contain the digital states to meet the needs of your organization.

For example, a digital state set called ValveStates may contain the two possible states of a valve: OPEN and CLOSED. ValveStates is an example of a custom digital state set that you create and then apply digital points to the states OPEN and CLOSED.

Digital state set status icons

The Digital States tool uses these icons to display the status of servers and digital state sets:

Status	Icon
The server is connected and digital state sets are accessible.	
The server is disconnected and digital state sets cannot be retrieved.	

Status	Icon
A digital state set on a connected server.	

Search for digital states and digital sets

You can search for up to three states, and use the asterisk character (*) as a wildcard. Wildcard searches may take several moments on remote servers. Any matches are displayed in the Search dialog box, with the set name in the **Digital Sets** list, and the associated states in the **Digital States** list.

1. Click **Search** , or right-click and choose **Search** from the context menu to display the Search dialog box.
2. Enter characters from one or more digital states or digital state sets as search criteria.
3. Choose **Retrieve sets that have the specified states in consecutive order** to display only digital sets that contain specified states in the order listed, with Digital State 1 first, followed by State 2, and then State 3.
4. Click **Search**.
5. Click **Find Next** to continue searching.
6. Double-click on a set name to close the Search dialog box and select the set in the Digital States tool.
7. Click **Close**.

Verify that System digital state set is up to date

There are times you may need to verify that your digital state sets are up-to-date. For example, when you upgrade your Data Archive, new system digital states are not added, so that any modifications you made to the system digital set are not overwritten. Only on a new install of the current version of Data Archive is the System digital set guaranteed to have all the appropriate states.

1. Right-click the digital state set icon in the tree control and click **Check System State Set**.
2. Review the message:
 - If the digital state set is up to date, a System Set Check Complete dialog box appears indicating that the set is up to date.
 - If the digital state set is not up to date, the Check Results dialog box appears. Click **Update states** to update the selected states.

Create custom digital states

First, add a digital state set and add digital states that contain the digital states Red, Green, and Yellow. Then, create digital points that will use those digital states.

Add a digital state set using PI SMT

1. Select the desired Data Archive node in the tree control and click the **Add** button .

2. Select the **New Set** icon and enter a name for the digital state set.

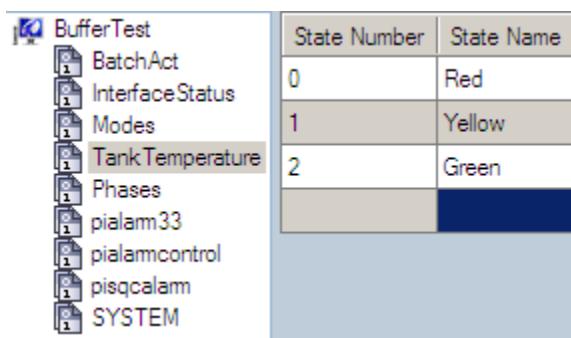
Names that are used for digital states will be truncated to 79 characters and are not case-sensitive. Digital set names cannot have the prefix "DIGSET_". The *DIGSET_* prefix is reserved for deleted digital state sets.

3. Press **Enter**.

4. Click **Save** .

Add digital states to a set

When you add a digital state to a digital state set, a numeric digital code is assigned by Data Archive, according to the position of the digital state string in the Digital State table. The first value is 0, the second is 1; the third is 2, and so on. The digital code is the value used, typically by an interface, to write the digital state value; that value is translated to the state when it is read by a client such as ProcessBook. This is called *digcode* in PI API.



State Number	State Name
0	Red
1	Yellow
2	Green

To add digital states to a digital state set:

1. Select the digital state set in the Digital States tool.
2. Enter a name for the digital state.
3. Press Enter.

Create digital points

To create digital points, set the Point type attribute to Digital and set the Digital set attribute to a digital state set. The digital state set and its digital states must exist before you can configure digital PI points to use those digital states.

For example, the following steps explain how to create points that represent the states of a valve (CLOSED and OPEN) and points that represent the states of a switch (ON and OFF).

1. Create a digital state set named ValvePosition with two states: CLOSED and OPEN.
2. Use the PI SMT Point Builder tool to configure the point or points that will use the CLOSED and OPEN states:
 - a. Create a point with Point type set to Digital.
 - b. Set Digital set to the ValvePosition.
 - c. Configure other point attributes and settings as needed.
3. Create points that use the states ON and OFF.
4. Create a digital state set named SwitchPosition with two states: ON and OFF.

5. Use the PI SMT Point Builder tool to configure the point or points that will use the ON and OFF states:
 - a. Create a point with Point type set to Digital.
 - b. Set Digital set to SwitchPosition.
 - c. Configure other point attributes and settings as needed.

Digital state set edits

Changes to digital states affect previously archived values. For example, if you change the digital state Valve Position from CLOSED to SHUT, all values retrieved from the archive are reported with the SHUT string instead of the previously archived CLOSED string. You can change this default behavior so that when you edit digital states you update only new digital state sets.

Update only new digital state sets

You can change the default behavior that automatically updates archived digital state sets, by using the DigitalSet point attribute. This attribute specifies the name of the digital state associated with the point.

If you want only new values received by the archive to use the updated digital state, you can add a new state set and change the DigitalSet point attribute to use this new state set. For example, you can create a new state set named *Old Valve Position* and configure events of 0 that were recorded before you change the DigitalSet point attribute to use the *CLOSED* digital state. Then, you can configure *Valve Position* to use *SHUT* for events of 0 recorded after the point attribute edit.

Edit digital states in a set or add new states to a set

1. Select the state set in the tree control.
The associated states appear as a table in the State Set window.
2. Click in a cell in the State Name column to edit an existing state.
3. Click in the next empty cell in the State Name column and enter a name for the state.
4. Click **Save** .

Delete a digital state

Note: Although you can delete digital state sets, we recommend that you do not delete a digital state set that has been in use. Instead, edit the set or add a digital state set whenever possible.

To delete a digital state, right-click the state in the state set table it belongs to and select **Delete State**.

Delete a digital state set

Caution: We recommend that you do not delete a digital state set that has been in use, because archived events associated with the deleted set are no longer visible to PI SDK-based clients, such as ProcessBook 3.x, DataLink 3.x, and PI SMT. Instead, edit the set or create a new digital state set whenever possible. Users who try to

retrieve archive values associated with a deleted digital set, whose clients run PI SDK 1.3.6.353 and later, will see:

? StateID: ## | Offset: ##

Note: Previously archived events can be displayed correctly in PI API-based clients. You can continue to see the archived events using apisnap, ProcessBook 2.x and DataLink 2.x. Internally, Data Archive does not remove deleted digital state sets; it renames the deleted digital state set to *DIGSET_##*, where ## is the set number, and they can be accessed only through **piconfig**.

If you do delete a digital state set, reconfigure points and events that use the deleted digital state set. For more information, see [Reconfigure points that use a deleted digital state set](#) and [Reconfigure events that use a deleted digital state set](#).

The easiest way to delete a digital state set is to use the Digital States tool in PI SMT.

1. In PI SMT, select **Points > Digital States**.

Digital state sets for the server you are connected to are shown.

2. Right-click the digital state set that you want to delete and click **Delete Set** .
3. Click **OK**.

Reconfigure points that use a deleted digital state set

If you have digital points associated with the deleted digital state set, you should reconfigure the points to use another valid digital set. You must reconfigure the points, even if you create a new digital state of the same name.

Note: A deleted digital state set remains associated with its original digital state set number. No other set can reuse this number. Thus, even if you create a new digital set with the exact same name, it is a set that is unique from the deleted set.

Reconfigure events that use a deleted digital state set

After you reconfigure the points to use a valid digital state set, you must reprocess the archives that formerly used the deleted digital state set, to use a new valid digital set. For details on how to reprocess archives, see the *Data Archive Administration guide*.

Note: You must reprocess the archives, even if you create a new digital state set that uses the same name as the deleted state set. If you do not reprocess the archives, the events that were archived before the digital state was deleted remain associated with the deleted set and will not be visible.

Copy a digital state set

There are times you might want to copy a digital state to another Data Archive server. If, for example, you are upgrading your hardware and want to transfer data from an existing Data Archive server to a new server, you will need to copy over your digital state sets in addition to your digital points.

1. Select the state set icon in the tree control and click the **Copy** button  or press Ctrl+C. You can also right-click the node and choose **Copy Set** from the context menu.
2. Select the Data Archive node where you want to copy the set and click the **Paste** button  or press Ctrl+V.

You can also right-click the node and choose **Paste Set**.

You might want to rename the digital state set, because by default it will be named the same as the set you copied.

Rename a digital state set

1. Select the state set node in the digital state set list and press F2, or right-click and select **Rename Set**.
2. Enter a new name for the digital state set and press Enter.

Export or import digital state sets

You can export digital state sets to or import digital state sets from the following file formats.

- Column-based CSV, where the Data Archive server name, digital state set name, and each individual state are represented by separate columns. For example:

```
server name,state set,state1,state2...
rex,CHIPALARM,No Alarm,C alarm,B alarm,A alarm,D alarm
rex,CHIPMODE,Off,On
```

- Row-based CSV, where the server name, digital state set name, and each individual state are represented by separate rows, with the state index comma-separated to the left of the state. For example:

```
* PI Server rex
CHIPALARM
0,No Alarm
1,C alarm
2,B alarm
3,A alarm
4,D alarm

rex
CHIPMODE
0,Off
1,On
```

Export digital state sets

- Select the digital state set node in the tree control and click  , or right-click and choose **Export to File**. Specify the filename and folder location and click **Save**.
- To export all digital state sets on a specified server, select the Data Archive node instead of an individual state set.

Import digital state sets

1. Click the **Import** button  or right-click a Data Archive server in the tree control and choose **Import from File** from the context menu.

2. Click the **Open** button  to browse to the CSV-formatted file containing the digital states information. The contents of file are displayed in the **File Preview**.
Check the sets to verify that the syntax is correct, and edit the files if necessary.
3. Choose an **Overwrite Option** for existing sets with duplicate names:
 - **Do not overwrite existing sets** aborts the import of a state set if another already exists with the same name.
 - **Import set with a different name if set already exists** renames a new state set if another already exists with the same name.
 - **Prompt before overwriting existing sets** prompts you to overwrite the old state set if a new set has the same name.
 - **Automatically overwrite existing sets** automatically overwrites old state sets if new sets have the same names.
4. Select the Data Archive server where you want to import the state sets, if it is not already selected, and click **Create Sets**.
5. Import the sets to different servers if necessary, or click **Close** to close the dialog box.

Digital States quick reference

Goal	Right-click option	Toolbar Icon
Add a new digital state set to a selected server.	Add Set	
Copy details of a selected state set to the clipboard.	Copy Set	
Create a new state set using details previously copied to the clipboard, under a selected server.	Paste Set	
Delete selected state sets from a server.	Delete Set	
Save changes to a selected state set.	N/A	
Search state sets on connected Data Archive servers. You can include the wildcard character * in searches.	Search	
Expand the list of state sets under a server.	Expand All	
Collapse the list of state sets under a server.	Collapse All	

Goal	Right-click option	Toolbar Icon
Export a selected digital state set, or all digital state sets on a selected server to a CSV file.	Export to File	
Import one or more state sets from a CSV file to a selected server.	Import from File	
Refresh the list of state sets for all connected servers.	Refresh	
Rename a digital state set.	Rename Set	N/A
Check system state set.	Check System State Set	N/A
Select to sort digital state sets alphabetically or by state set ID.	N/A	
Launch online help.	N/A	

Firewall

Use the Firewall tool to create, view, and edit a PI Firewall. The PI Firewall is a security feature that allows the PI Network Manager to control access to Data Archive at the IP network address level. System administrators can use the PI Firewall to allow or deny specific computers to connect.

You can use the PI Firewall to temporarily deny all access to a Data Archive server. For example, you might want to use the PI Firewall when you upgrade a primary Data Archive server within a collective. In this case, you can ensure that users and applications will connect to a secondary member within the collective, not the primary, while you upgrade and test the Data Archive server.

Note: In the default Data Archive configuration, the PI Firewall allows connections from all possible client addresses. Create a PI Firewall to control access to a specific computer.

For more information, see the Knowledge article [Questions and answers on the PI Firewall table](#).

PI Firewall database

PI Firewalls allow you to block connections from certain addresses. If a PI Firewall exists, it takes precedence over a trust; that is, before an application can connect to Data Archive, it must first pass the PI Firewall before it attempts a trust logon. Data Archive recognizes modifications to PI Firewalls within 15 minutes, or upon restarting Data Archive, whichever comes first.

When you add a PI Firewall you are adding an entry to the PI Firewall database, which is maintained by the PI Net Manager subsystem. The process manages all connections to Data Archive, including subsystem connections and TCP/IP applications. When Data Archive receives an incoming connection, **pinetmgr** first checks the PI Firewall Database for partial or complete IP host names or addresses.

Use the PI Firewall to allow or disallow connections from designated workstations or subnets. If there is no entry in the PI Firewall database that would allow an incoming connection, the connection will be terminated immediately.

Requests from the local host, that is, the same computer on which **pinetmgr** is running, are always allowed. So, in effect, the firewall does not apply to the local host.

PI Firewall connection protocol

New connection host names and IP addresses are checked against PI Firewalls in the following order:

1. If the connection originates from the local host, the connection is always accepted.
2. The PI Firewall database is searched for an exact match of IP address entry or host name entry. The search stops when Data Archive finds an entry and processes the connection according to that entry.
3. The PI Firewall database is searched for a wildcard match, for example, the connecting address of 192.168.168.22 matches a host mask of 192.168.*.*. A matching **DISALLOW** has precedence over an **ALLOW**.

Here is an example of the data entered into the PI Firewall tool for two PI Firewalls. In this example, only hosts within the 192.168.168.0 subnet are allowed connections. 192.168.168.67 is not allowed to connect even

though it matches the first host mask.

Host/Mask	Value
192.168.168.*	ALLOW
192.168.168.67	DISALLOW

Create a PI Firewall

You can use IP addresses, IP address masks, or host names to limit connections with a PI Firewall. You can restrict connections only from those from the subnets defined for your users. To allow users to create PI Firewalls for Data Archive, use the Database Security tool to grant write access for the PITUNING entry to the appropriate PI identities or PI users. Data Archive recognizes modifications to PI Firewalls within 15 minutes or upon restarting Data Archive, whichever comes first.

1. Click , or right-click in the Firewall tool and click **New**.
2. Use the Firewall Details dialog box to:
 - Select or enter the name of the Data Archive server on which you will create the PI Firewall in the **Server** field.
 - Enter the IP address, IP address mask, or host name of the computer you want to deny access to in the **Host** field.
 - Select **Disallow** in the **Value** field to deny access to the computer you entered; select **Allow** to allow to the selected computer.
3. Select or enter the name of the Data Archive server on which you will create the PI Firewall in the **Server** field.
4. Click **OK**.

Use an IP address mask

To enter an IP address mask with the Firewall tool:

- Use the entire IP address, such as: **192.168.149.55**
- Use a portion of an IP address and asterisks; an asterisk in an IP address field matches any number.

If you use a portion of an IP address, such as **192.168.177.***, the PI Firewall denies or allows access for all computers with IP addresses that begin with **192.168.177**.

Note: If the address mask is already used for a PI Firewall, the existing entry of the same name is overwritten.

View PI Firewalls

Existing PI Firewalls appear in the upper right pane of the Firewall tool:

- **Server** is the Data Archive server on which the PI Firewall is stored.
- **Collective** is the name of the Data Archive collective of which the Data Archive server is a member, if applicable.
- **Host** contains the IP address, IP address mask, or host name of the computer you want to deny access to in the **Host** field.
- **Value** indicates whether the PI Firewall allows or disallows access to the host machine.

Edit a PI Firewall

Note: After you edit the PI Firewall, it can take as long as 15 minutes for your changes to take effect.

Double-click the name of a PI Firewall in the Firewall tool, or select the name of a PI Firewall and click .

Save a list of PI Firewalls in XML format

1. Select and the names of the PI Firewall that you want to save in a list in the Firewall tool and click  or right-click and select **Export**.
2. Navigate to the directory where you want to save the list.
3. Click **Save**.

Delete a PI Firewall

Select the name of the PI Firewall in the Firewall tool and click .

PI Firewall quick reference

Right-click and toolbar options are enabled based on the item selected in the **Alarm Groups** window and its current status.

Goal	Right-click option	Toolbar icon
Create a PI Firewall	New	
Delete a selected PI Firewall	Delete	
View or edit details for a selected PI Firewall	Properties	
Export a list of PI Firewall data to an XML file	Export	
Refresh the list of state sets for all connected servers	Refresh	

Goal	Right-click option	Toolbar icon
Launch online help	N/A	

Identities, Users, and Groups

Use the Identities, Users, & Groups tool (along with the Mappings & Trusts tool) to manage authentication and access permissions on the Data Archive server. If you are using Windows for Data Archive authentication, then you work exclusively with PI identities. If you are managing individual accounts on the Data Archive server, then you work with PI users and PI groups.

About PI identities, PI users, and PI groups

PI identities, PI users, and PI groups are used for Data Archive security. Computer security includes:

- **Authentication**

Who is the user, and how do we confirm that users are really who they say they are?

- **Authorization**

Once we know who the user is, what is that user allowed to do? Authorization is synonymous with Data Archive access permissions.

Data Archive access permissions (authorization) can be defined for PI identities, PI users, and PI groups. The differences arise in the authentication configuration.

You can use these authentication methods for Data Archive:

- Windows authentication
- PI trusts (only with PI API 1.6.8 or earlier)
- PI user account logins (only with PI API 1.6.8 or earlier)

The following table shows which components you can use for which types of authentication. See also [Data Archive authentication methods](#).

Component	Windows authentication?	Trusts?	Explicit user logins on Data Archive? (insecure authentication method)
PI identities	Yes	No, if using PI API 2016 for Windows Integrated Security. Yes, if using PI API 1.6.8 or earlier.	No
PI users	Yes	Yes	Yes
PI groups	Yes	Yes	No

When configuring Data Archive authentication, we recommend that you use PI identities and authenticate through Windows security. PI identities do not imply individual user accounts or groups on the Data Archive

server.

PI users and PI groups are legacy components. When you use them for Windows authentication, you might create some confusion about the role of the PI user or the PI group on the Data Archive server. Are these components being used to manage actual PI user accounts, are they used only for trusts and mappings, or are they used for both?

If you are creating a new component to use in a mapping or a trust, use a PI identity to avoid confusion. If Windows authentication is not possible and you need to authenticate your users directly on the Data Archive server, then you must use PI users and PI groups.

About access permissions

Each PI identity, PI user, and PI group can be granted access permissions to objects on the Data Archive server (point data, modules, backup permissions, and so on). When a Windows user, PI interface, or other client application is authenticated and mapped to a PI identity (or user or group), then the Windows user or client application gets all the access permissions for the corresponding PI identity, user, or group.

Users that belong to more than one Windows group might be mapped to multiple PI identities, PI users, or PI groups. In this case, they get the cumulative access permissions for all the associated PI identities, users, and groups. In addition all users get the access permissions for PIWorld.

Note: SDK Applications connecting to a Data Archive server configured for Windows authentication (through SDK 1.3.6 and later) are authenticated through Windows, if possible. If Windows authentication succeeds, the access permissions are defined by that mapping, and not by a trust. See [Windows authentication versus SDK trusts](#).

Built-in PI identities, users, and groups

Data Archive includes several built-in PI identities, users, and groups. The most important are:

- piadmin — A PI user with super-user access
- piadmins — A PI group with administrative access
- PIWorld — A PI identity with general access

PI users, groups, and identities	Description
piadmin user	A PI user with super privileges. The piadmin user has complete read/write access to all Data Archive resources. You cannot modify the access permissions for piadmin. In most cases, do not map piadmin to any AD group or user. At most, map piadmin to a small group of administrators. Though you cannot delete the piadmin user, you can disable it to varying degrees.
piadmins group	A PI group intended to represent Data Archive administrators. Use piadmins for all routine administrative tasks.

PI users, groups, and identities	Description
	<p>This pre-configured group has read and write access to all Data Archive resources and default points. You can map piadmins to the AD group that represents your Data Archive system administrators and you can adjust the piadmins access permissions to meet your needs. You cannot delete the piadmins group.</p>
piusers group	<p>A built-in PI group with no pre-configured access permissions.</p>
PIOperators, PIEngineers, and PISupervisors identities	<p>Sample identities that have no pre-configured access permissions. You can configure or delete these PI identities.</p>
PIWorld identity	<p>A PI identity with default access permissions for read-only access to most PI resources. The PIWorld identity represents the "everyone" concept of Windows; it specifies the rights of non-explicit users or groups. By default, PIWorld is granted read access to most Data Archive databases and objects. All authenticated Data Archive users are given at least PIWorld privileges.</p> <p>You can rename and change the access permissions of the PIWorld identity. You cannot delete PIWorld. You cannot map PIWorld to an AD group or use PIWorld in a trust.</p>

Note: There is also a hidden user and a hidden group: PIUserIncompatible and PIGroupIncompatible. Data Archive uses them to display an owner and group in older administrative tools that do not support Windows authentication. They do not appear in the list of identities by default. To show them, use the **Options** button.

PIUserIncompatible user and PIGroupIncompatible group

Old-model administrative tools (Point Builder, SMT, and so on) expect access permissions in the owner/group/world model. Data Archive (versions 3.4.380 and later) uses a different access permissions model. When old administrative tools try to view or modify access permissions on a new Data Archive server, Data Archive attempts to determine an owner and group.

If Data Archive cannot determine an appropriate owner or group, then it defines the owner as the user PIUserIncompatible and the group as PIGroupIncompatible. On Data Archive version 3.4.380 and later, PIUserIncompatible and PIGroupIncompatible are provided for this purpose. They do not appear in the list of users and groups by default. To show them, click the **Options** button.

If the access permissions for the object meet the following format, then Data Archive can determine an owner and group for the object:

- One (and only one) PI user

- One (and only one) PI group
- PIWorld identity

If any of these conditions is not met, Data Archive cannot determine an owner and group and it uses the PIUserIncompatible user and PIGroupIncompatible group instead. Therefore, these names are displayed as the owner and group in older tools. You can change the names to something more descriptive, if you want.

View in single list or three tabs

If you are connected to one or more Data Archive servers of version 3.4.380 or later, then you can choose to view all the PI identities, PI users, and PI groups in a single list, instead of in individual tabs.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Click the **Option** button .
- The Options dialog box appears.
4. Choose the option to **Show all PI Identities in a single view**.

Note: If an earlier version of Data Archive is connected, the **Users** and **Groups** tabs still appear.

Configure Data Archive authentication

Data Archive authentication can be configured through Windows authentication, PI Trusts, or Data Archive user accounts.

Data Archive authentication methods

There are three methods you can use to configure authentication on the Data Archive server:

- Windows authentication

You can set up Data Archive to automatically grant access to authenticated Windows users. To do this, you create mappings between PI identities (or PI groups or PI users) and Windows groups or users (AD principals or local Windows security). This is the recommended method.

- PI trusts

PI trusts work by comparing the connection credentials of a connecting application to criteria you specify when you create the trust. For applications that cannot use Windows authentication, PI trusts are the recommended authentication method. See [Create a PI trust](#).

- User accounts on the Data Archive server

Each PI user account has an associated password. Nest PI users inside PI groups to manage access permissions.

Note: This method is not recommended. Recently upgraded servers might use Windows for authentication but still have some individual PI user accounts and PI groups. However, for the best security, retire individual PI user accounts as soon as possible.

Set up Windows authentication

1. Identify or create a PI identity, user, or group that has appropriate access permissions.
2. Create a mapping between the Windows user or group and the PI identity, user, or group.

The mapped Windows accounts now have the Data Archive access permissions defined on the PI identity, user, or group.

Note: Ideally, use Windows Active Directory (AD) groups for authentication. It is also possible to use local Windows groups and users, but this requires additional configuration. For information, see [PI mappings](#), or refer to the Security Configuration Guide on the [Customer Portal](#).

Manage PI identities

The **PI Identities** tab lists all the PI identities from each of the selected servers.

PI identities

PI identities are the link between Windows authentication and Data Archive authorization (access permissions). Each PI identity represents a set of access permissions on the Data Archive server. For example, one PI identity might be allowed to create points, while another PI identity could be allowed to read point data but not create new points.

When a mapping exists between a PI identity and a Windows group, all the users in the Windows group are automatically authenticated on the Data Archive server and granted the access permissions defined for that PI identity.

Create a PI identity

When you create a new identity, the identity name is required. Note the following restrictions on identity names:

- The name must be unique.
- The name cannot include the vertical pipe (|) character or the colon (:) character.
- The name cannot be a positive integer, although it can contain numbers. For example, the name 407 is not valid, but the name Admins407 is valid.
- The name is not case sensitive.

1. Under the Servers panel (or if you have a collective deployment, Collectives and Servers), select a server.
2. Under System Management Tools, select **Security > Identities, Users, & Groups**.
3. Select the **PI Identities** tab.
4. Click the **New Identity** button  to open the **New Identity** dialog box, where you can create and configure a new PI identity.
5. In **Identity**, type in a name for the new identity.

This field is required. If you try to create an identity with an invalid name, an error message appears and the identity is not created. Note that you can change an identity name any time after creation.

6. Select the appropriate server from the drop-down **Server** list.

This list is populated from the selected servers under Servers (or if you have a collective deployment, Collectives and Servers). Only PI Server versions 3.4.380 and later appear in the list.

7. Optional: Enter a brief description in **Description**.

8. At the bottom of the dialog box, select the **Identity cannot be deleted** check box.

This prevents the identity from being accidentally deleted. In order to delete this identity, you must first edit the identity and clear this check box.

9. Click **Create**.

The new PI identity appears in the **PI Identities** tab.

PI identity configuration options

When you create or edit a PI identity, you can select any of the following options:

- **Identity cannot be deleted**

When selected, you cannot delete the PI identity. If you later want to delete the identity, you must first edit it to clear this option.

- **Identity cannot be used in a Mapping**

When selected, you cannot use the PI identity in a mapping. If you later want to use the identity in a mapping, you must first edit the identity to clear this option.

- **Identity cannot be used in a Trust**

When selected, you cannot use the PI identity in a trust. If you later want to use the identity in a trust, you must first edit the identity to clear this option.

- **Identity is disabled**

When selected, you cannot use the identity for authentication of any kind. The access permissions are retained so that if you re-enable the identity you do not need to reconfigure.

Delete a PI identity

Caution: When you delete a PI identity, Data Archive automatically deletes any PI mappings or PI trusts that reference it. Access permissions that depend on a deleted identity are automatically reset on next access attempt. Before deleting, consider how that change will impact the rest of your security configuration. In most cases, renaming or disabling PI identities is preferable to deleting them.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Select the **PI Identities** tab.
4. Select the PI identity.
5. To check the impact of the deletion, disable the account and make sure no access problems arise.
6. Click the **Delete** button .

Disable a PI identity

When you disable a PI identity, you also disable all PI trusts and PI mappings based on that identity. This means that Data Archive can no longer authenticate using that identity. Any users who have already been authenticated on the Data Archive server will retain their access until they log off.

Data Archive does not delete access permissions for a disabled identity. If you re-enable the identity, then you do not need to reconfigure the access permissions.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Select the **PI Identities** tab.
4. Double-click the PI identity.
5. The Properties dialog box appears.
6. Click to check the **Identity is disabled** check box.
7. Click **OK**.

Create a PI identity mapping

When you create a mapping between a PI identity and a Windows group, Data Archive automatically authenticates all members of that Windows group *as that PI identity*. Ideally, you should use Windows Active Directory groups for mappings, but you can use any AD principal, local Windows group, or local Windows user.

You can also create a mapping between a PI identity and an AVEVA Identity Manager role using claims-based authentication via OpenID Connect (OIDC). See [Map a role or client ID to a PI identity using OIDC](#) for instructions.

You can create mappings between individual Windows users and PI identities, but it is not recommended. Mappings based on individual users will prevent you from managing your Data Archive security access by manipulating group memberships.

1. Under **Servers**, select the Data Archive server.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Select the **PI Identities** tab.
4. Double-click the identity that you want to map in order to open its Properties dialog box.
5. Click the **Mappings & Trusts** tab.

The top portion shows all existing mappings for this PI identity. The bottom portion shows all existing PI trusts for this PI identity.

6. Click **Add** under the mappings to open the Add New Mapping dialog box. (There is also an **Add** button under the trusts; be sure to click the right button.)

Note: The **Add** button is disabled if the selected PI identity is disabled or not usable in a mapping.

7. In **Windows Account**, enter the account you want to map to. This can be an AD principal or a local Windows group or user. To select the account either:
 - Click the browse button  to browse for the account.
 - Type in the account name. If you choose to type in the account name, click the resolve SID button  to verify that this is a valid account. If the account is valid, a SID appears in the field. Otherwise, a dialog

box with an error message opens.

8. In **Description**, enter a description of the mapping, if desired. There are no restrictions on the contents of this field.
9. Click **OK** to create the new mapping.

Define a PI trust against a PI identity

When you define a PI trust against a PI identity, Data Archive authenticates all applications that use that PI trust as that PI identity.

You can define trusts against a PI identity using the Identities, Users, & Groups tool or the Mappings & Trust tool. If you use the Identities, Users, & Groups tool, you cannot use the Add Trust Wizard. To use the wizard, use the [Mappings and trusts](#) tool to create the trust.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Select the **PI Identities** tab.
4. Double-click the identity against which you want to define the trust. That identity's Properties dialog box opens.
5. Click the **Mappings & Trusts** tab. The top portion shows all existing mappings for this PI identity. The bottom portion shows all existing PI trusts for this PI identity.
6. Click **Add** under the trusts to open the Add New Trust dialog box.

(There is also an **Add** button under the mappings; be sure to click the correct button.)

Note: The **Add** button is disabled if the selected PI identity is disabled or not usable in a trust.

7. Enter the appropriate information for the PI trust. At a minimum, you need to specify a name for the trust. You can also specify optional information. The more specific you make the trust, the more secure it will be. See [Manage trusts](#) for more information.

You can enter the following optional trust specifications:

- **IP Information:** Specify which computer to trust. Enter either a network path or an IP address and netmask:
 - **Network path:** Fully-qualified domain name of the trusted computer, such as an interface node (for example, *my_laptop.my_company.com*).
 - **IP Address:** The IP address of the trusted computer, such as the interface node.
 - **Net Mask:** The netmask of the trusted computer (for example, 255.255.255.255).
- **Windows Account Information:** For SDK trusts only, specify a trusted Windows domain and account:
 - **Domain:** Windows domain of the user who runs the trusted application (for example: *osi*).
 - **Account:** Windows user name of the user who runs the trusted application (for example: *my_account*).
- **Application Information:** Specify the **Name** of a trusted application. Enter the name differently for API and SDK connections.
 - **API connection:** Enter an identifier called an application process name, or procname. This is a four-character string with an E appended (for example, *PIPeE* for the Perfmon Interface).
 - **SDK connection:** Enter the full name of the connecting application, including the extension, but not

- the path (for example: **PI-ICU.exe**).
8. Click **OK** to create the new trust.

Manage PI users

The **PI Users** tab lists all the PI users from each of the selected servers.

Create a new PI user

Note the following restrictions on user group names:

- Names must be unique. They cannot match the name of an existing PI user, PI group, or PI identity.
- Names cannot include the vertical pipe (|) character or the colon (:) character.
- Names cannot be a positive integer, although they can contain numbers. For example, the name 329 is not valid, but the name *Admins329* is valid.
- Names are not case sensitive.

You can change a PI user or group name any time after creation.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Click the **PI Users** tab.
4. Click the **New** button  to open the New User dialog box.
5. In **Username**, type the a name of the new PI user. This field is required.
6. Select the appropriate server from the drop-down **Server** list. This list shows the servers selected under **Collectives and Servers**.
7. Optionally, enter a brief description in **Description**. There are no restrictions on the contents of this field.
8. In **Password**, enter a password for the PI user.
9. Click **Create**. The new PI user now appears in the **PI Users** tab.

Change a PI user password

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Select the **PI Users** tab. Select the user.
4. Click the **Change Password** button  in the tool bar.
5. Type in the old password and the new password. Assuming you have administrator permission for PI users, you can use an exclamation mark (!) as the old password.

Delete a PI user account

If you delete a PI user, Data Archive automatically deletes any PI mappings or PI trusts that reference that user. Before deleting a user, consider how that change will impact the rest of your security configuration. To simulate this, you can disable the user. If the change causes problems, you can then restore the user.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Select the **PI Users** tab.
4. Select the PI user.
5. Click the **Delete** button .

Disable a PI user account

When you disable a PI user, you also disable any trusts and mappings for that user. Data Archive does not delete access permissions for a disabled PI user. If you re-enable the account, then you do not need to reconfigure the access permissions.

Note: You can disable a PI user account only on Data Archive versions 3.4.380 or later; earlier versions do not have this feature.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Select the **PI Users** tab.
4. Double-click the user name to open that PI user's Properties dialog box.
5. On the **General** tab, select the **User is disabled** check box.
6. Click **OK**.

To enable a previously disabled PI user account, perform the same steps, but *clear* the **User is disabled** check box.

Disable explicit logins for a user account

When you disable explicit logins for a PI user, then users cannot access Data Archive by typing in that user name and password. You can still use the user in mappings and trusts. Because these passwords are not as secure as trusts or Windows authentication (mappings), it is a good idea to disable explicit logins where possible.

Note: You can disable explicit logins only on Data Archive versions 3.4.380 or later; earlier versions do not have this feature.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Select the **PI Users** tab.
4. Double-click the username to open that PI user's Properties dialog box.
5. On the **General** tab, select the **User cannot be used for an explicit login** check box.

6. Click **OK**.

To re-enable explicit logins for a PI user account, clear the check box.

Import Windows users

The **Import Windows Users** feature was designed to make it easier to facilitate SDK trusts based on Windows accounts such as dollar-sign (\$) trusts (see [PI SDK trusts](#) for information about trusts, and see [Windows account information \(SDK only\)](#) for information about Windows accounts). This feature is not needed on Data Archive servers that support Windows authentication. Instead, use PI mappings.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Click the **Import Users** button .

The Import Windows Domain Users dialog box appears.

4. Specify a Windows domain from which to import the users
5. Specify the Data Archive server to which the users are imported.
6. Click the **List Users** button to list all the Windows users on the selected domain.
7. Set the initial password for each user, you can choose either:
 - The PI User name (each initial password is the same as the newly-created PI User account name)
 - A string of your choosing (the initial password is the same for all users)

Create a PI user mapping

When you create a mapping between a PI user and a Windows group, all members of that Windows group are automatically authenticated on the Data Archive server *as that PI user*. Ideally you should use Windows Active Directory groups for the mapping, but you can use any AD principal, local Windows group, or local Windows user.

Note: You can create mappings between individual Windows users and PI users, but it is not recommended. Mappings based on individual users will prevent you from managing your Data Archive security access by manipulating group memberships.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Select the **PI Users** tab.
4. Double-click the PI user that you want to map to open the user's Properties dialog box.
5. Click the **Mappings & Trusts** tab. The top portion shows all existing mappings for this PI user. The bottom portion shows all existing PI trusts for this PI user.
6. Click **Add** under the mappings to open the Add New Mapping dialog box. (There is also an **Add** button under the trusts; be sure to click the right button.)

Note: The **Add** button is disabled if the selected PI user is disabled or not usable in a mapping.

7. In **Windows Account**, enter the account you want to map to. This can be an AD principal or a local Windows group or user. To select the account either:

- Click the browse button  to browse for the account.
 - Type in the account name. If you choose to type in the account name, click the resolve SID button  to verify that this is a valid account. If the account is valid, a SID appears in the field. Otherwise, a dialog box with an error message opens.
8. In **Description**, enter a description of the mapping, if desired. There are no restrictions on the contents of this field.
 9. Click **OK** to create the new mapping.

Define a PI trust against a PI user

When you define a PI trust against a PI user, Data Archive authenticates all applications that use that PI trust as that PI user.

You can define trusts against a PI user using the Identities, Users, & Groups tool or the Mappings & Trusts tool. If you use the Identities, Users, & Groups tool, you cannot use the Add Trust Wizard. To use the wizard, use the Mappings and Trusts tool to create the trust.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Select the **PI Users** tab.
4. Double-click the user against which you want to define the trust. That user's Properties dialog box opens.
5. Click the **Mappings & Trusts** tab. The top portion shows all existing mappings for this PI user. The bottom portion shows all existing PI trusts for this PI user.
6. Click **Add** under the trusts to open the Add New Trust dialog box. (There is also an **Add** button under the mappings; be sure to click the correct button.)

Note: The **Add** button is disabled if the selected PI user is disabled or not usable in a trust.

7. Enter the appropriate information for the PI trust. At a minimum, you need to specify a name for the trust. You can also specify optional information. The more specific you make the trust, the more secure it will be. See [Manage trusts](#) for more information.

You can enter the following optional trust specifications:

- **IP Information:** Specify which computer to trust. Enter either a network path or an IP address and netmask:
 - **Network path:** Fully-qualified domain name of the trusted computer, such as an interface node (for example, *my_laptop.my_company.com*).
 - **IP Address:** The IP address of the trusted computer, such as the interface node.
 - **Net Mask:** The netmask of the trusted computer (for example, 255.255.255.255).
- **Windows Account Information:** For SDK trusts only, specify a trusted Windows domain and account:
 - **Domain:** Windows domain of the user who runs the trusted application (for example: *osi*).
 - **Account:** Windows user name of the user who runs the trusted application (for example: *my_account*).
- **Application Information:** Specify the **Name** of a trusted application. Enter the name differently for API and SDK connections.
 - **API connection:** Enter an identifier called an application process name, or procname. This is a four-

- character string with an E appended (for example, *PipeE* for the Perfmon Interface).
- SDK connection: Enter the full name of the connecting application, including the extension, but not the path (for example: **PI-ICU.exe**).
8. Click **OK** to create the new trust.

Export a PI user list to file

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Select the **PI Users** tab.
4. Click the **Export List** button .
5. The Save As dialog box appears.
6. Type in a path to where you want to save the list of PI users.
The file format is XML.

Add a PI user to a PI group

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Select the **PI Users** tab.
4. Double-click the user name to open that user's Properties dialog box.
5. Click the **PI Group Memberships** tab. The tab shows which groups the user belongs to.
6. Click **Add** to open the Select Groups dialog box.
7. Select the desired PI group from the list of available PI groups.
8. Click **Add Group**.
9. Click **OK** to add the group to the user's list of groups.
10. Click **OK** to save the changes.

Remove a PI user from a PI group

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Select the **PI Users** tab.
4. Double-click the user name to open that user's Properties dialog box.
5. Click the **PI Group Memberships** tab. The tab shows which groups the user belongs to.
6. Select the group you want to remove from the list.
7. Click **Remove**.
8. Click **OK** to save the change.

Manage PI Groups

The **PI Groups** tab lists all PI groups from each of the selected servers.

Create a new PI group

Note the following restrictions on user group names:

- Names must be unique. They cannot match the name of an existing PI user, PI group, or PI identity.
- Names cannot include the vertical pipe (|) character or the colon (:) character.
- Names cannot be a positive integer, although they can contain numbers. For example, the name 329 is not valid, but the name Admins329 is valid.
- Names are not case sensitive.

You can change a PI group name any time after creation.

- Under **Collectives and Servers**, select the server or collective.
- In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
- Click the **PI Groups** tab.
- Click the **New** button  to open the New Group dialog box.
- In **Group**, type the name for the new PI group. This field is required.
- In **Server**, select the appropriate server. This list shows the servers selected under **Collectives and Servers**.
- In **Description**, enter a brief description, if desired. There are no restrictions on the contents of this field.
- Click **Create** to create the new PI group.

Manage group memberships

Each PI user can belong to any number of PI groups. When you add a PI user to a PI group, that user gains all the access permissions associated with that PI group. The PI user gets access to every PI resource available to all of these PI groups.

For example, suppose a PI user belongs to two PI groups, Group1 and Group2. Group1 has read/write access to the data for a point called test_point1. Group2 has read/write access to the data for a point called test_point2. This PI user has read/write access to the data for both test_point1 and test_point2.

- Under **Collectives and Servers**, select the server or collective.
- In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
- Select the **PI Groups** tab.
- Double-click the PI group to open that group's Properties dialog box.
- Click the **PI User Members** tab.
- Use the **Add** and **Remove** buttons to add and remove PI users from the group.
- Click **OK** to save the changes.

Delete a PI group

If you delete a PI group, Data Archive automatically deletes any PI mappings or PI trusts that reference that group. Before deleting a group, consider how that change will impact the rest of your security configuration. To simulate this, disable the group first. If the change causes problems, you can then restore the group.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Select the **PI Groups** tab.
4. Select the PI group.
5. Click the **Delete** button .

Disable a PI group

When you disable a PI group, you also disable any trusts and mappings for it. This means that Data Archive can no longer authenticate using that group. Any users who have already been authenticated on the Data Archive server will retain their access until they log off.

Data Archive does not delete access permissions for a disabled PI group. If you re-enable the group, then you do not need to reconfigure the access permissions.

Note: You can disable a PI group only on Data Archive versions 3.4.380 or later; earlier versions do not have this feature.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Select the **PI Groups** tab.
4. Double-click the PI group.
The Properties dialog box appears.
5. On the **General** tab, click the check box to disable the group.
6. Click **OK**.

Create a PI group mapping

When you create a mapping between PI group and a Windows group, Data Archive automatically authenticates all members of that Windows group *as that PI group*. Ideally you should use Windows Active Directory groups for mappings, but you can use any AD principal, local Windows group, or local Windows user.

Note: You can create mappings between individual Windows users and PI groups, but it is not recommended. Mappings based on individual users will prevent you from managing your Data Archive security access by manipulating group memberships.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Select the **PI Groups** tab.

4. Double-click the PI group that you want in a mapping to open that group's Properties dialog box.
5. Click the **Mappings & Trusts** tab. The top portion shows all existing mappings for this PI group. The bottom portion shows all existing PI trusts for this PI group.
6. Click **Add** under the mappings to open the Add New Mapping dialog box. (There is also an **Add** button under the trusts; be sure to click the right button.)

Note: The **Add** button is disabled if the selected PI group is disabled or not usable in a mapping.

7. In **Windows Account**, enter the account you want to map to. This can be an AD principal or a local Windows group or user. To select the account either:
 - Click the browse button  to browse for the account.
 - Type in the account name. If you choose to type in the account name, click the resolve SID button  to verify that this is a valid account. If the account is valid, a SID appears in the field. Otherwise, a dialog box with an error message opens.
8. In **Description**, enter a description of the mapping, if desired. There are no restrictions on the contents of this field.
9. Click **OK** to create the new mapping.

Define a PI trust against a PI group

When you define a PI trust against a PI group, Data Archive authenticates all applications that use that PI trust as that PI group.

You can define trusts against a PI group using the Identities, Users, & Groups tool or the Mappings & Trust tool. If you use the Identities, Users, & Groups tool, you cannot use the Add Trust Wizard. To use the wizard, use the Mappings & Trusts tool to create the trust.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
3. Select the **PI Groups** tab.
4. Double-click the group against which you want to define the trust. That group's Properties dialog box opens.
5. Click the **Mappings & Trusts** tab. The top portion shows all existing mappings for this PI group. The bottom portion shows all existing PI trusts for this PI group.
6. Click **Add** under the trusts to open the Add New Trust dialog box.

(There is also an **Add** button under the mappings; be sure to click the correct button.)

Note: The **Add** button is disabled if the selected PI group is disabled or not usable in a trust.

7. Enter the appropriate information for the PI trust.

At a minimum, you need to specify a name for the trust. You can also specify optional information. The more specific you make the trust, the more secure it will be. See [Manage trusts](#) for more information.

You can enter the following optional trust specifications:

- **IP Information:** Specify which computer to trust. Enter either a network path or an IP address and netmask:
 - **Network path:** Fully-qualified domain name of the trusted computer, such as an interface node (for example, *my_laptop.my_company.com*).

- **IP Address:** The IP address of the trusted computer, such as the interface node.
 - **Net Mask:** The netmask of the trusted computer (for example, 255.255.255.255).
 - **Windows Account Information:** For SDK trusts only, specify a trusted Windows domain and account:
 - **Domain:** Windows domain of the user who runs the trusted application (for example: *osi*).
 - **Account:** Windows user name of the user who runs the trusted application (for example: *my_account*).
 - **Application Information:** Specify the **Name** of a trusted application. Enter the name differently for API and SDK connections:
 - API connection: Enter an identifier called an application process name, or procname. This is a four-character string with an E appended (for example, *PIPeE* for the Perfmon Interface).
 - SDK connection: Enter the full name of the connecting application, including the extension, but not the path (for example: **PI-ICU.exe**).
8. Click **OK** to create the new trust.

Export a PI group list to file

Export the currently listed PI group list to a file.

1. Under **Collectives and Servers**, select the server or collective.
 2. In the **System Management Tools** pane, select **Security > Identities, Users, & Groups**.
 3. Select the **PI Groups** tab.
 4. Click **Export List** button .
- A Save As dialog box appears.
5. Specify where you want to save the list of PI groups.

The file format is XML.

Interface List

Use the Interface List tool to view, start, and stop PI interfaces on connected Data Archive servers that are managed through the PI Interface Configuration Utility (ICU). The list of interfaces is read from the PI Module Database, where PI ICU stores interface configuration information.

View interface information

Interfaces that are configured to run as a service can be started and stopped from the **Interface** list, provided your security settings allow you to start or stop the service on the machine the interface is running on.

You can also rename the interface service, change its startup type, or export a list of the interface services to a comma-separated value (CSV) file.

Note: Interfaces can only be displayed for Windows-based Data Archive servers.

Interfaces are listed in a window with columns that display the following information:

- **Interface** name
- Optional **User Set Name** configured in ICU
- Host **PI Server**
- **Interface Node** computer that runs the interface
- **Point Source** used by the interface
- **ID** number assigned to points belonging to the interface
- **Interface Type**
- Current **Status** of the interface service (**Running**, **Stopped**, **Not a service**, **Unknown**, **Not checked**)
- **Startup Type** used for the interface (**Auto**, **Manual**, **Disabled**)
- **Service Display Name**, if the interface is installed as a service
- **Interface Version**
- **APS Connector**, if Auto Point Synchronization is used
- **APS Node** that runs PI APS for synchronization
- **Description** of the interface entered in ICU

You can click individual column headings to sort the list view.

Change the status of an interface service

The **Status** column indicates whether:

- An interface is accessible
- An interface runs as a Windows service and is presently *Running* or *Stopped*
 - If the interface **Status** is listed as *Stopped*, select the interface service and click the **Start** button ➤ or

- right-click the interface service and choose **Start** from the menu.
- If the interface **Status** is listed as *Running*, select the interface service and click the **Start** button ➤ or right-click the interface service and choose **Start** from the menu.

Rename an interface service

1. Right-click the interface service and choose **Rename**.
2. Change the name of the service in the dialog box and click **OK**.

Assign a Windows startup type to an interface service

If the interface is installed as a Windows service, you can assign one of three standard Windows startup types, as indicated in the **Startup Type** column.

Right-click the interface service and:

- Choose **Auto** to start the service automatically on start.
- Choose **Manual** to start the service manually or only when needed.
- Choose **Disabled** to shut down the service so that it does not start.

Export a list of interfaces

To export a list of the interface services to a comma-separated value (CSV) file:

1. Click ➔, or right-click and select **Export**.
2. Browse to the directory where you want to save the file.
3. Click **Save**.

Interface tool quick reference

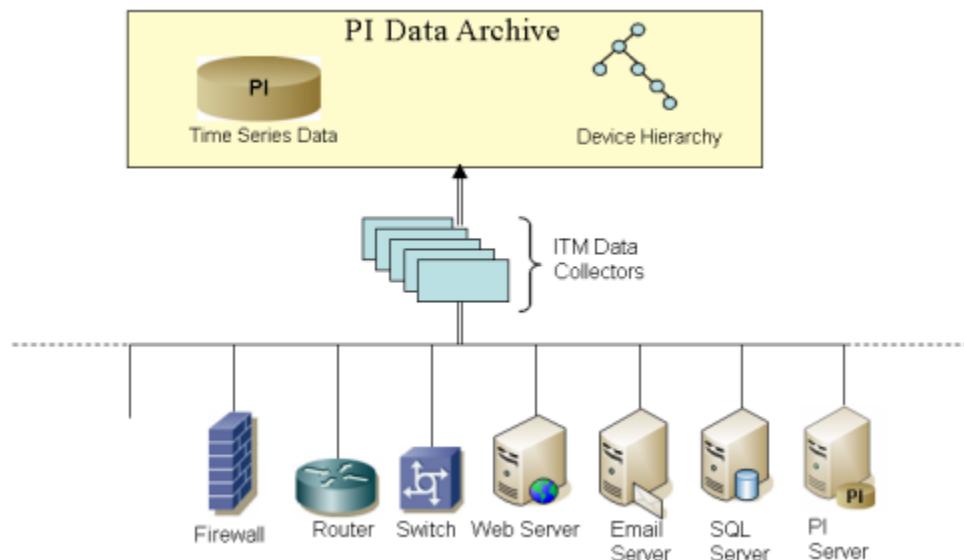
Note: Right-click and toolbar options are enabled based on the item selected in the **Alarm Groups** window and its current status.

Goal	Right-click Option	Toolbar Option
Start the selected interface service	Start	➤
Stop the selected interface service	Stop	■
Export the displayed list of interfaces to a CSV file	Export	➔
Set the interface service to start automatically on boot	Automatic	N/A

Goal	Right-click Option	Toolbar Option
Set the selected interface service to start only when required or by user startup	Manual	N/A
Prevent the selected interface service from starting	Disabled	N/A
Change the Service Display Name of the selected interface	Rename Service	N/A
Refresh the list of interfaces from all selected servers	Refresh	
Launch online Help	Help	

IT Organizer

The IT Organizer tool helps you manage your IT Monitor deployments. In a typical IT Monitor deployment, the System Manager builds many tags belonging to many different interfaces, sometimes on multiple Data Archive servers. The following diagram illustrates some of the management difficulties in a distributed data collection environment:



The IT Organizer tool helps you group your tags by monitored device. This allows you to act on all the tags belonging to one device. Some of the operations you can perform on a device level are:

- Change the IP address for all the tags (if the device's IP address changes)
- Change the PI point attributes of many tags at once
- Show how many tags are not collecting data per device
- Add an entire device to the Module Database

You can also assign each group of tags a role that describes the type of device or application you are monitoring. Once you have assembled and labeled the device groups, you can organize them in a device tree.

Configuring the IT Organizer

Before you begin collecting points with the IT Organizer tool, you must connect to a Data Archive configuration node and set up your PI Interface information.

If you plan on adding content, you should also identify device roles and add icons for device tree building.

Data Archive configuration node

The configuration node is the Data Archive server that stores all of your interface, monitored device, and

organizational metadata in MDB. The configuration node should be the Data Archive server where you collect all of your data, or it can be a different Data Archive server in your network.

You must connect to a Data Archive server before you select the configuration node.

The configuration node stores the configuration information in its Module Database. The monitored device list file stores a history of the devices collected. The interface list file stores information about the PI interfaces in your deployment.

Connect to a Data Archive configuration node

The first time you attempt to start IT Organizer, you get an error message that you have no configuration node selected.

1. Under **System Management Tools**, click **IT Points > IT Organizer**.
2. Click **Setup > Configuration Node**.
3. Select a Data Archive.

This creates a branch in the MDB node that stores all of the configuration information.

Set up your PI Interface information

The IT Organizer tool needs to know which interfaces you are working with and also which points are associated with which interfaces. PI interfaces collect the data from your monitored device. You can select and save the list of PI interfaces that can be scanned for points in the IT Organizer.

1. Under **System Management Tools**, click **IT Points > IT Organizer**.
2. Click **Setup > Interfaces**.

The Store your PI Interface Information dialog box appears.

3. Select one or more of the **Available Servers**.

The interfaces for those servers that were configured with PI ICU appear on the right.

4. Select the interfaces with points you want to organize.

In addition to identifying which interface instances are collecting data, you must also label each interface by its interface type (SNMP, PerformanceMonitor, Ping, or TCPResponse). The IT Organizer tool needs to associate each point with an interface type in order to extract the monitored device name properly.

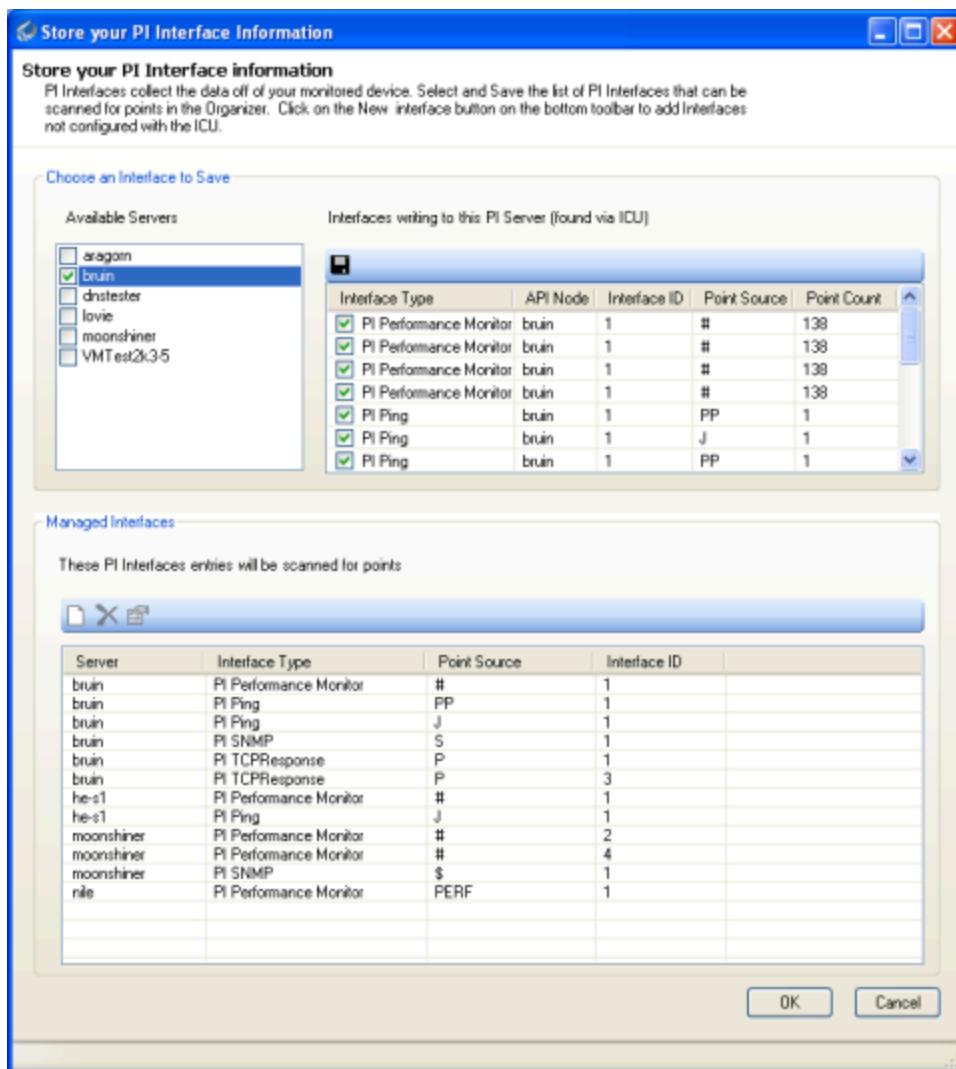
5. Click **Save** to add the selected interfaces to the **Managed Interfaces** list.

You can also add interfaces manually to the Managed Interfaces list by clicking **New** ().

Note: We recommend that you use PI ICU. Manually configured interfaces are prone to error and show only the scan class numbers rather than actual scan times in IT Organizer.

6. Click **OK**.

Repeat these steps when you add or remove an interface.



Assign a role to a device

The extensibility of the AVEVA™ PI Server™ and the protocols used to collect network and application data (SNMP, Perfmon, and so on) are intended to be as generic as possible. In other words, an interface is not built to collect data from one type of device or application, but rather for one type of protocol (the data collectors are protocol specific, not equipment specific). This means that you may collect performance information from a multitude of device types. These device and application types include but are not limited to:

- Data Archive servers
- Windows 2003 machines
- Windows XP machines
- Cisco routers
- Cisco switches
- SQL Servers
- Exchange Servers

A role designates a context and corresponds to a point template used to create the data collection tags on the Data Archive server. For example, you can assign the Data Archive role to a computer to indicate that you are monitoring Data Archive performance.

A device can have multiple roles. Roles are associated with the device and are used by IT Overview to build displays.

The server must be on the device list, before you can assign a role to it.

1. Select the device or devices.



2. Select a role from the **Roles** () list.
3. Select the role type.

For example, select all Data Archive servers and assign them the **AVEVA Data Archive 3.4** role. You should see the role added to the role view as seen below:

Device Name	Last Update Date	Percent Reporting	Total Points	Comment
69.236.191.111	12/12/2006 5:57:36 PM	0%	0	
BRUIN (2)	12/27/2006 5:22:12 PM	99%	141	
HE-S1	12/12/2006 5:57:47 PM	0%	171	
HE-S1.OSITEST.DEV (1)	12/14/2006 7:19:41 AM	0%	161	
LOVIE	12/14/2006 7:19:44 AM	0%	429	
LOVIE.OSISOFT.INT	12/27/2006 5:22:09 PM	0%	1	
MOONSHINER (1)	12/14/2006 7:19:45 AM	100%	380	
RTPMPS	12/12/2006 5:57:36 PM	100%	1	

Device Name	Role Name
BRUIN	OSisoft PI Interface Performance
BRUIN	OSisoft PI Server 3.4

From this window you can also delete a device from the list, clear all devices from the list, and refresh the point count.

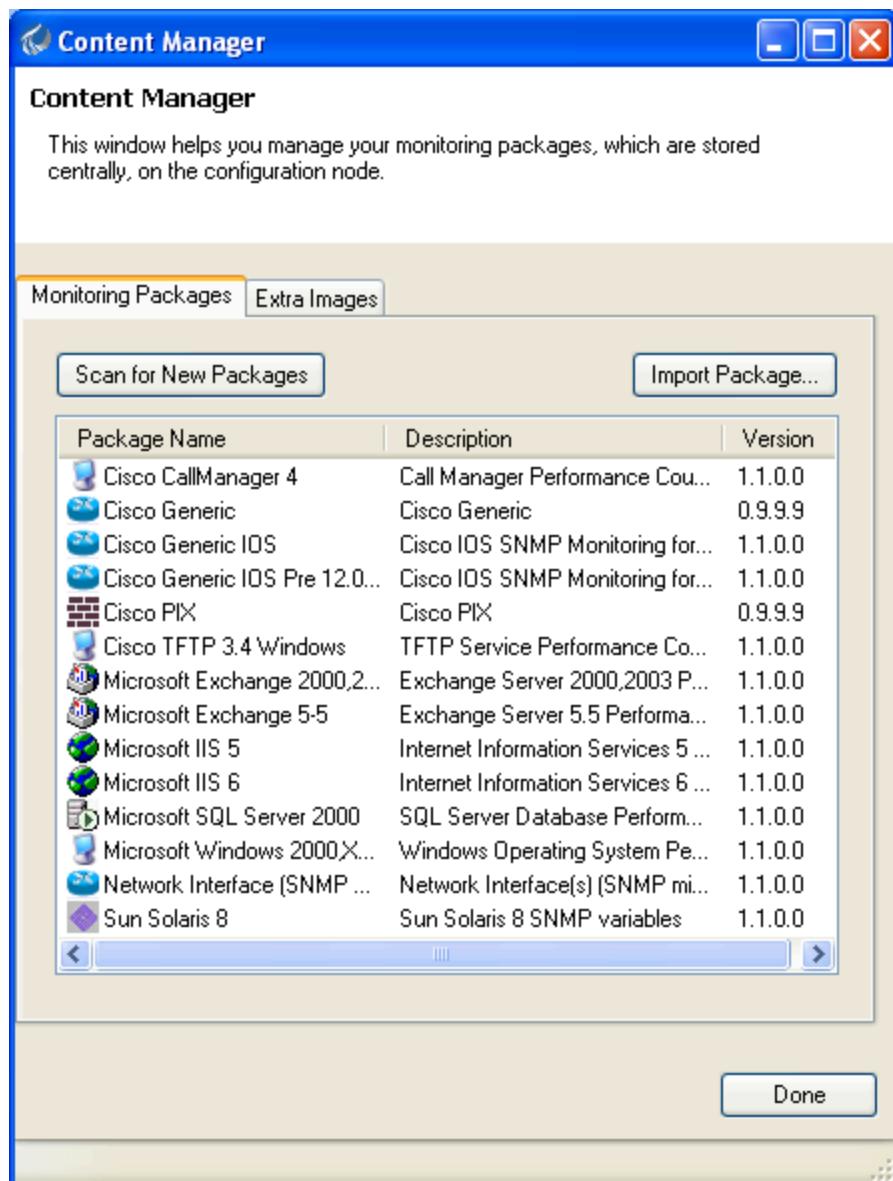
Add device roles

After you organize your points into the monitored device list, you can assign a role to your device. Roles are packaged in .pkg files which store a tree node image which you can place in your navigation tree, as well as a display transformation file, which is used with the IT Overview ProcessBook Add-In.

The IT Organizer tool will install the following packages by default:

- Cisco CallManager 4.pkg
- Cisco PIX.pkg
- Cisco TFTP 3.4 Windows.pkg
- Cisco Generic IOS Pre 12.0(3)T.pkg
- Cisco Generic IOS.pkg
- Microsoft Exchange 5-5.pkg
- Microsoft Exchange 2000,2003.pkg
- Microsoft IIS 5.pkg
- Microsoft IIS 6.pkg
- Sun Solaris 8.pkg
- Network Interface (SNMP mib-2).pkg
- PI Interface Performance.pkg
- PI Server 3.4.pkg
- Microsoft SQL Server 2000.pkg

1. Click **Setup** and select **Packages and Images**.
2. In the Content Manager, select the **MonitoringPackages** tab.
3. To import a role package, click **Import Package**, and load the desired .pkg file. This package file is stored on the configuration node of the server, so any users that choose this configuration node can see the roles.



Add images to the icon list

The IT Organizer tool provides a configurable list of icons that you can use to help identify your monitored devices and applications in the device tree. This step is not required but suggested if you intend to use the device tree.

1. Click **Setup** and select **Packages and Images**.
2. In the Content Manager dialog box, select the **Extra Images** tab.
3. Click **Import Images** to load an image from your directory to the configuration node.

All users that select that configuration node will be able to view these icons in their navigation tree.

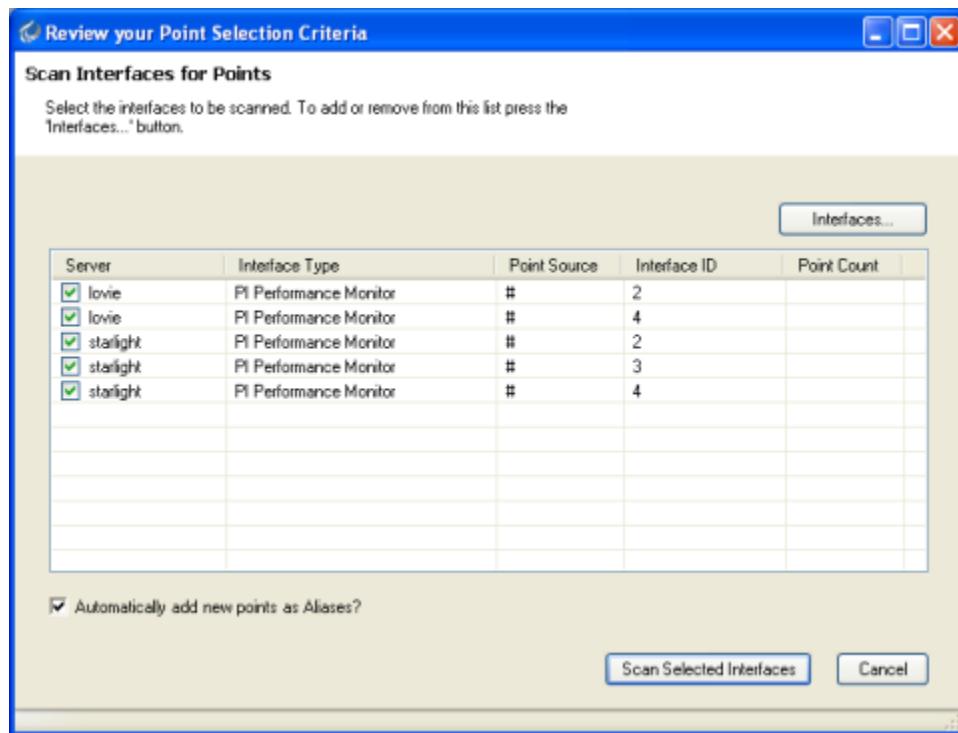
Managing devices and tags

This section explains how to work with the monitored device list to manage your monitored devices and tags.

Update points

This section describes how to see which interfaces are selected and to add or remove interfaces from the list.

1. In the main panel of IT Organizer, click **Update List**. This opens the Review your Point Selection Criteria window.
2. Click **Interfaces** to open the Store your PI Interface Information window. There you can choose other interfaces and add them to the update the list **Managed Interfaces**.
3. Select the **Automatically add new points as Aliases**. This creates aliases that IT Overview requires for any point to be monitored without any further user action. This is the default for this option.
4. Select interfaces and click **Scan Selected Interfaces**. This scans each Data Archive server associated with **Managed Interfaces** for points whose point source and interface ID match the interfaces that you selected. When the scan finishes, you see the devices from which the interface is gathering performance data listed in your **Monitored Device List** on the main pane of IT Organizer.



Monitored device list

The monitored device list shows all the monitored devices that IT Organizer has detected on your Data Archive servers.

Monitored Device List (16 Devices)				
Configuration Node: moonshiner				
Delete Clear Refresh Assign Update List				
Device Name	Last Update Date	Percent Reporting	Total Points	Comment
192.168.0.1	12/14/2006 7:19:46 AM	0%	61	
69.236.191.111	12/12/2006 5:57:36 PM	0%	0	
BRUIN	12/27/2006 5:22:12 PM	95%	147	
HE-S1	12/12/2006 5:57:47 PM	0%	171	
HE-S1.OSITEST.DEV (1)	12/14/2006 7:19:41 AM	0%	161	
LOVIE	12/14/2006 7:19:44 AM	0%	429	
LOVIE.OSISOFT.INT	12/27/2006 5:22:09 PM	0%	1	
MOONSHINER (1)	12/14/2006 7:19:45 AM	100%	380	
RTPMPS	12/12/2006 5:57:36 PM	100%	1	
TACKLEBERRY	12/14/2006 7:19:46 AM	0%	84	
TESTTESTESTE (6)	12/14/2006 7:19:39 AM	0%	31	

You can use the toolbar buttons to manage your device list:

- **Delete** () : Remove a device from the device list.
- **Clear** () : Clear the entire device list.
- **Refresh** () : Refresh the **Percent Reporting** column for each of the monitored devices.
- **Assign** () : Assign a role to one more devices.

For each found device, IT Organizer displays the following information:

- **Device Name**
- **Last Update Date**: The last time the tag list for this device was updated.
- **Percent Reporting**: The percentage of tags for this device which do not have a digital state from the system digital state set as a current value. Such states, including INTF_SHUT, BAD VALUE or PT_CREATED, indicate that data is not being properly collected for that point.
- **Total Points**: The total number of points discovered for this monitored device.
- **Comment**: A comment you can enter for this device.

Modify a monitored device

To make changes for a particular device, right-click the list entry for that device and choose one of the following options:

- **Assign Role**: Create a role for this device so that it can be represented in the device tree.
- **Add Comment**: Add a comment for this discovered device.
- **Change IP Address**: Change the IP address in every tag in this device's tag list.
- **Resolve IP**: If the device is listed by its IP address but you would prefer to browse by a DNS host name, choose this action to attempt to resolve the host name.
- **Refresh Percent Reporting Column**: Refresh the tag list for this device. When you invoke this operation, the servers will again be mined for points, but only points monitoring this device will be refreshed. To refresh the entire tag list, press the **Update Device List** button.
- **Show Points**: Show the points for this device in the point list below.

- **Show Bad Points:** Show all digital-state points that have current values in the system digital state set.
- **Synchronize Aliases to Points:** Rebuild the alias structure for this monitored device

Note: If you make any administrative changes that require changes to the tags (for example, you change the IP address or change SNMP values), you will need to have write permissions to that tag.

View point and device role details

To get more detailed information about a particular device, double-click that device in the monitored device list. Detailed information about the device appears in the device details, located below the monitored device list. The device details pane has two tabs: **Points** and **Roles** that show different views.

View point details

1. Click the **Point View** tab to see a list of all the points that are monitoring the device that you selected in the monitored device list.

The Point Details View: 1) Search options, 2) Point Details

Point Name	Server	Interface Type	Snapshot Value	Snapshot Time	Last Archive Value	Last Archive Time	Status	Sampling Rate
BRUIN_Login4Disk_Used/sec	BRUIN	PI TCPResponse	31.5262	1/4/2007 3:29 PM	31.5262	1/4/2007 2:29 PM	Good	60
BRUIN_Memory_Pages_In/sec	BRUIN	PI TCPResponse	Scan Off	12/26/2006 10:39 AM	4.600037	12/26/2006 10:22 AM	Bad	60
BRUIN_R_Archive_Subsystems_Acquired_Calls/sec	BRUIN	PI TCPResponse	0	1/4/2007 3:29 PM	0	1/4/2007 2:29 PM	Good	60
BRUIN_R_Archive_Subsystems_Acquired_Events/sec	BRUIN	PI TCPResponse	0.2820007	1/4/2007 3:27 PM	0.2820213	1/4/2007 2:27 PM	Good	60
BRUIN_R_Archive_Subsystems_Cache_Flush_Operations/sec	BRUIN	PI TCPResponse	0	1/4/2007 3:24 PM	0	1/4/2007 2:23 PM	Good	60
BRUIN_R_Archive_Subsystems_Cache_Record_Count	BRUIN	PI TCPResponse	0	1/4/2007 3:27 PM	0	1/4/2007 10:26 AM	Good	60
BRUIN_R_Archive_Subsystems_Cache_Record_Disk_Reads/sec	BRUIN	PI TCPResponse	0	1/4/2007 3:25 PM	0	1/4/2007 9:25 AM	Good	60
BRUIN_R_Archive_Subsystems_Cache_Record_Memory_Reads/sec	BRUIN	PI TCPResponse	0	1/4/2007 3:27 PM	0	1/4/2007 9:27 AM	Good	60
BRUIN_R_Archive_Subsystems_Config_Verify_Calls/sec	BRUIN	PI TCPResponse	0	1/4/2007 3:27 PM	0	1/4/2007 9:16 AM	Good	60
BRUIN_R_Archive_Subsystems_Events_Fetch/sec	BRUIN	PI TCPResponse	0	1/4/2007 3:27 PM	0	1/4/2007 9:16 AM	Good	60
BRUIN_R_Archive_Subsystems_Graph_Fragment_Calls/sec	BRUIN	PI TCPResponse	0	1/4/2007 3:27 PM	0	1/4/2007 9:16 AM	Good	60
BRUIN_R_Archive_Subsystems_Import_Verify_Calls/sec	BRUIN	PI TCPResponse	0	1/4/2007 3:29 PM	0	1/4/2007 2:29 PM	Good	60
BRUIN_R_Archive_Subsystems_Out_of_Order_Events/sec	BRUIN	PI TCPResponse	0	1/4/2007 3:29 PM	0	1/4/2007 2:29 PM	Good	60

The **Point List** displays the following point properties:

- **Server:** The server where this point is stored.
 - **Interface type:** The type of PI interface collecting data for this point
 - **Snapshot value:** The last value sent to Data Archive
 - **Snapshot time:** The last time a value was sent to Data Archive for this point
 - **Last archive value:** The last value sent to Data Archive
 - **Last archive time:** The last time a value committed to Data Archive for this point (last point that passed compression)
 - **Status:** A indicator showing if the point is in a bad (System) state
 - **Sampling rate:** The rate at which the interface collects information from the device. The sampling rate is only available if you have registered your interface using the ICU. Otherwise, the scan class for the point will be displayed.
2. To narrow your view to a subset of the point, you can enter a string to reduce your view.
 3. To make changes to the listing for one or more points, right-click the list entry for that point and choose one of the following options from the resulting pop-up menu:
 - **Trend:** View a recent trend of this data
 - **Delete:** Delete point(s) from Data Archive (may require user authentication)
 - **Refresh Selected Points:** Get the most recent point information for the selected points

- **Properties:** See the attributes for tag(s)

View device role details

Click the **Roles** tab to see the list of roles for the device that you selected in the **Monitored Device List**.

Points	Roles	Details for BRUIN
Device Name	Role Name	
 BRUIN	Microsoft IIS 5	
 BRUIN	OSIsoft PI Interface Performance	
 BRUIN	OSIsoft PI Server 3.4	
 BRUIN	Microsoft SQL Server 2000	

The **Device Role** view displays the following role attributes:

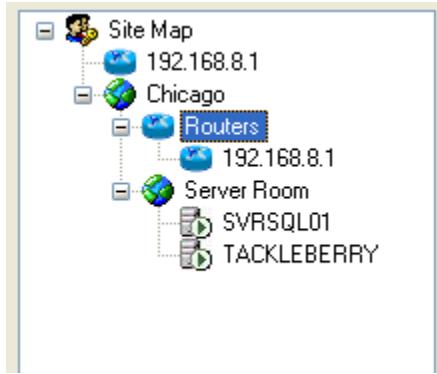
- **Device name:** The name of the device
 - **Role name:** The name of the role you have assigned to this device

You can perform the following actions on each device role, via the pop-up menu:

- **Add Role:** Add a new role to this device's role collection
 - **Delete Role:** Remove the role from the device's role collection

Manage the navigation tree

The navigation tree is a hierarchical representation of the devices in your organization.



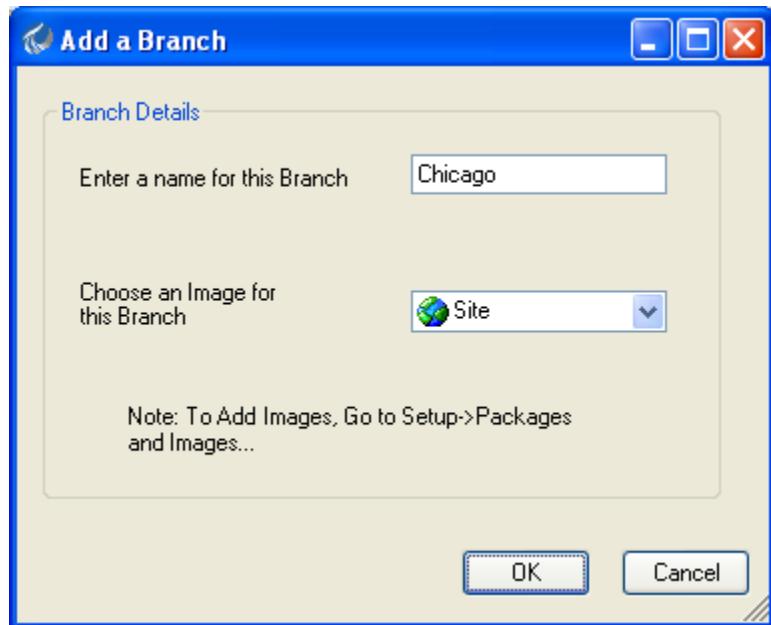
Add a group to the navigation tree

You can add two kinds of items to your device tree: a group, which is just an entry meant to describe a collection of child devices (for example, Chicago, SQL Servers, or CRM Deployment), or an actual device (for example, a device which has a SQL and Exchange Role).

Note: To add a device or group to your device tree, you must have sufficient permissions to create Modules on your Configuration Node.

1. To add a group, right-click on the tree location where you want to add your placeholder, and select **Add Branch** from the resulting pop-up menu.

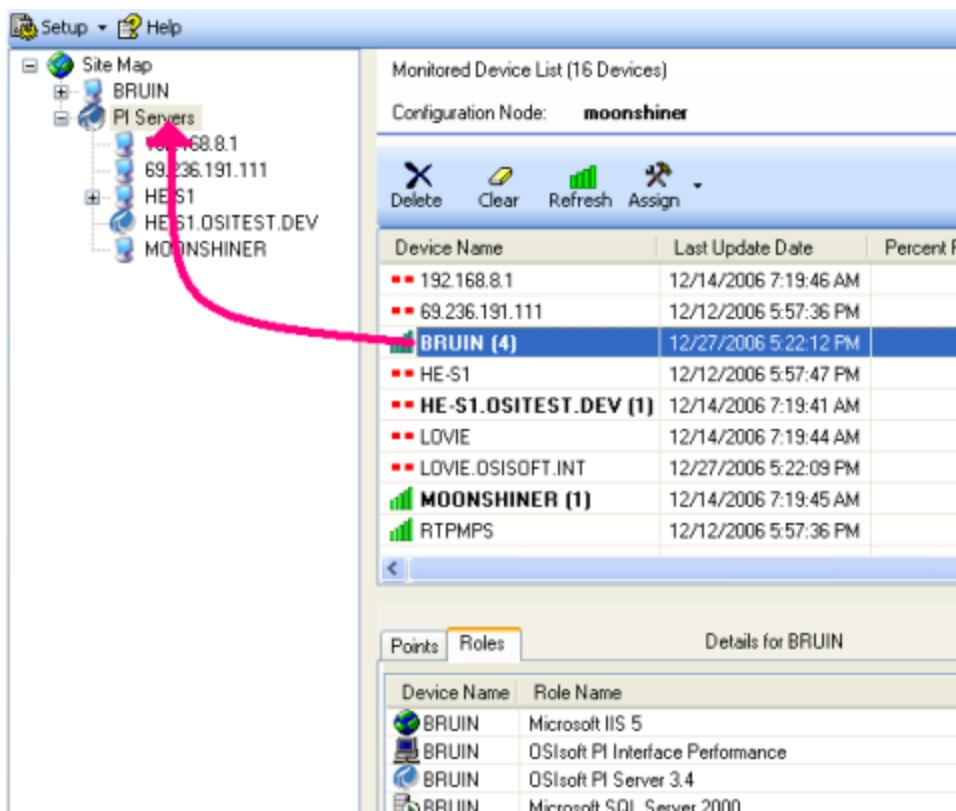
The Add a Branch dialog box appears.



2. Name your placeholder and select a registered icon to represent it on the device tree.

Add a device to the navigation tree

To add devices to your navigation list, drag a device from the **Monitored Device List** to the navigation tree:



- In addition, you can perform the following actions on the device tree by right-clicking a tree node:
 - Add a Placeholder:** Add a placeholder at this node
 - Change Tree Image:** Change the image used for this node in the navigation tree
 - Show Points for Device:** Show the points associated with this device role
 - Delete Site Map Entry:** Remove this entry from this list
- Auto Refresh:** If this option is selected, the **Site Map** will be scanned for changes every 5 seconds.

When you drag a device from the **Device List** to the **Site Map**, or when you add a Role to the device, the points for that device are added to the Module Database (MDB) as aliases. These aliases create a context which could be used in automatic display building.

Device identification

Organizing points into different device buckets is done in a couple of steps. Device buckets are collections of points which are all monitoring the same device, regardless of interface type. For example, a Dell Server may have PerfMon, SNMP, and Ping points. Organizing the points based on device rather than interface type allows the user to see all the points monitoring a particular device. This device-centric view is the standard method for representing the IT assets rather than the point-centric view.

The IT Organizer accomplishes this task in a number of steps. The IT Organizer is a plug-in which may run on a number of client machines. Because of this distributed architecture, we require a centralized data store to keep track of our configuration files. The sections that follow describe some of these configuration files, along with partial schemas.

Interface definition file

An interface definition file describes how to determine which device a point is monitoring. For example, PI SNMP points have the *HOST=<device_name>* string in the instrumenttag property. The file includes the PI Point attribute which stores the device name, as well as two regular expressions to determine what part of that attribute is the device name and how to reconstruct that attribute if the device name is changed. Here is the XML:

```
<CinterfaceDefinitionEntry>
<m_sInterfaceName>pisnmp</m_sInterfaceName>
<m_sDeviceAttribute>instrumenttag</m_sDeviceAttribute>
<m_sDeviceRegexp>^HOST=([^\;]+)$</m_sDeviceRegexp>
<m_sReplaceRegexp>^(HOST=)[^\;]+$</m_sReplaceRegexp>
<m_iReplaceAfterIndex>1</m_iReplaceAfterIndex>
<m_sLocation1>interface instance</m_sLocation1>
<m_sLocation2>time normalization</m_sLocation2>
<m_sLocation3>group SNMP requests?</m_sLocation3>
<m_sLocation4>scan class number</m_sLocation4>
<m_sLocation5>point level debugging</m_sLocation5>
</CinterfaceDefinitionEntry>
```

Note that in addition to the point attribute and the device determining regular expression, the file also contains descriptions of the location codes, so point attributes can be displayed more clearly to the user. This file is compiled with the **ITOrganizer.dll** file.

Interface instances

In order to sort points into device buckets, you must set up your PI Interface information to indicate which points you would like to sort.

The interfaces dialog asks you to select which points (that is, points that share the same PI Server/Location1/Point Source combinations) to examine for sorting. You might choose to search all selected Data Archives for interfaces registered via the ICU or enter an Interface ID/Point Source combination manually. The resultant configuration information is stored in the Module Database as an XML file. A snippet of this file is shown here:

```
<CinterfaceEntry>
<m_sInterfaceName>piperfmon</m_sInterfaceName>
<m_sInterfaceType>piperfmon</m_sInterfaceType>
<m_sInterfaceID>1</m_sInterfaceID>
<m_sPointSource>#</m_sPointSource>
<m_sServerName>dusty</m_sServerName>
<alPointIndices />
<alScanClasses>
<string>00:00:05</string>
<string>00:01:00</string>
</alScanClasses>
</CinterfaceEntry>
```

Note that for interfaces which are registered with the ICU (as shown here) the scan classes are stored as well. This information will be displayed to the users on a per point basis in the point organizer.

The configuration files mentioned above are enough for the IT Organizer to get started; the IT Organizer uses the SDK's GetPoints queries to find all the points on a server that fulfill the interface instance and point source criteria stored by the point selection wizard. The queries look like this:

```
PISDK.PointList plCurrent = oServer.GetPoints("pointsource = '" + sPointSource + "' and
```

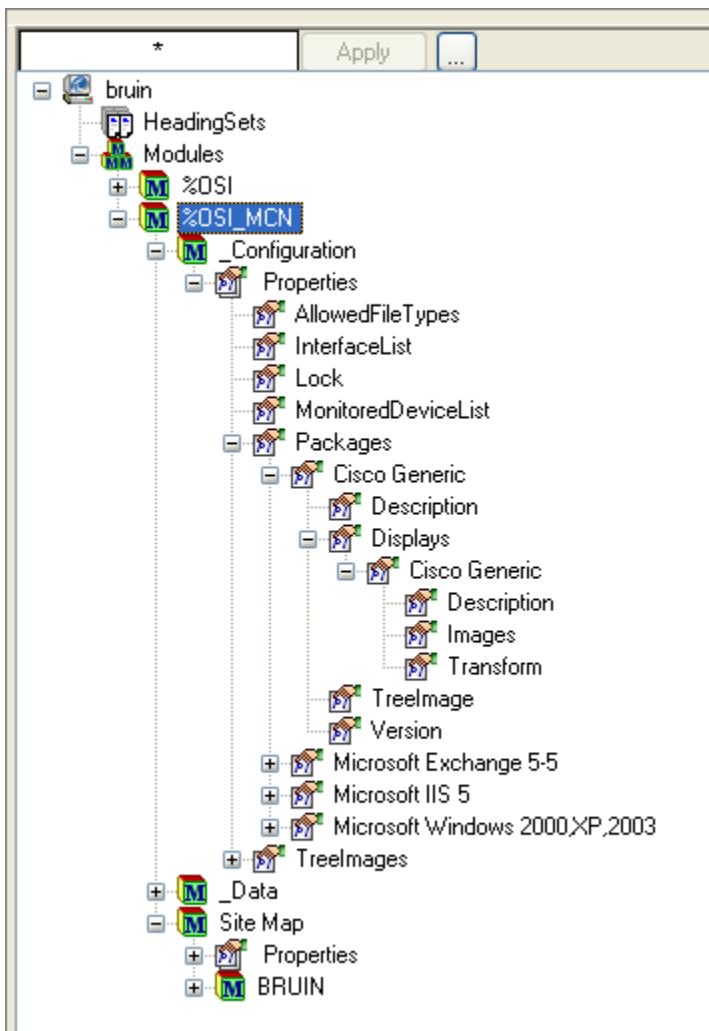
```
location1 = " + sLocation1",null);
```

The resultant point list is then iterated through. Each point is examined using the regular expression for that interface type, and is thrown into its device bucket.

Clear configuration node settings

Use this procedure to completely clear out the settings of your configuration nodes.

1. In PI SMT, open the Module Database Editor.
2. Select and delete the **%OSI_MCN** node on your Data Archive configuration node.
3. Remove the configuration node settings file, **ConfigurationNodes.xml**. located in **\$PIPC\ITOrganizer**.



Licensing

The Licensing tool displays license activation status, availability, and restrictions for connected Data Archive servers. The most common task in the Licensing manager tool is to check your point count and module count limitations. If you exceed these limitations, Data Archive does not allow you to create new points.

To work with licenses, in the PI System Management Tools pane, expand **Operation** and click **Licensing**.

View point and module statistics

1. To see how many points you have left:
 - a. Choose **count** from the drop-down menu in the toolbar.
 - b. Under **Resources** double-click to expand the `pibasess.MaxAggregatePointModuleCount` item.
2. This shows usage information for the combined point and module count:
 - a. **Total** is the total number of points and modules you are licensed for. You are always licensed for the combined number of points and modules. This combination of points and modules together is called a data stream.
 - b. **Amount Used** is the number of data streams (points and modules together) that you are using.
3. To see how many points you are using:
 - a. Double-click to expand `pibasess.MaxPointCount` and look at **Amount Used**.
Ignore the **Total** count. This statistic is meaningless for points because your license counts data streams, rather points.
4. To see how many modules you are using:
 - a. Double-click to expand `pibasess.MaxModuleCount` and look at **Amount Used**.
Ignore the **Total** count. This statistic is meaningless for modules because your license counts data streams, rather than modules.

Note: Some applications create modules rather than points. It is best to monitor the statistics for the data streams (`pibasess.MaxAggregatePointModuleCount`), rather than only looking at points or modules used.

Monitor point and module count together

You might get an error message saying that you cannot create a new point because your data stream limitations are exceeded, even though the Licensing tool says you still have points left. This is a common point of confusion that occurs because what you are actually licensed for is the number of data streams, not the number of points. Data streams are points and modules.

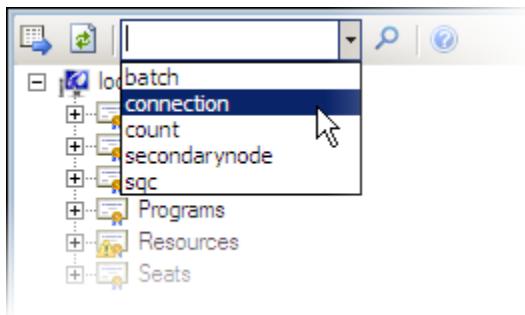
For example, suppose you think that you are licensed for 5000 points. If you look at your point count (`pibasess.MaxPointCount`) and see that **Amount Used** is 4000, you might assume that you have 1000 points left. However, your license counts *data streams*, rather than points. That means you also need to know how many modules you are using (`pibasess.MaxModuleCount`).

It is best to monitor your point and module count together (`pibasess.MaxAggregatePointModuleCount`). **Total**

shows you how many data streams (points and modules together) you are licensed for, **Amount Used** shows you how many data streams you are using, and **Amount Left** shows you how many data streams you have left.

View licensing information

Use the drop-down menu in the toolbar to select common license views, such as point/module count (**count**), number of secondary nodes in a Data Archive collective or number of allowed connections.



To find a specific item that does not appear in the drop-down menu, type a search term in the text field. License Manager icons indicate the status of your license.

License icons

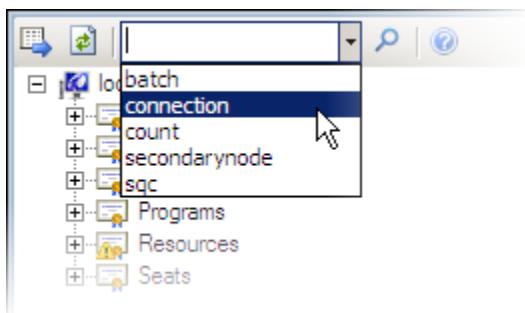
This icon indicates that there is a license for this application.

This icon indicates that there is no license to run this application.

This icon is a warning that indicates you are either close to the maximum limit of points/connections allowed, or in the case of a demo product, you are close to expiration.

View connection limitations

To see connection limitations, choose **connection** from the drop-down menu in the toolbar.



The Licensing tool displays information about each server that is selected in the Collectives and Servers pane. The information is grouped under two headings: **Programs** and **Resources**.

- **Programs**

Under **Programs** you can check the maximum number of anonymous connections allowed (`pinetMgr.AnonymousConnectionsAllowed`). An anonymous connection is a connection from something that has not identified itself with a known GUID (Globally Unique Identifier). Products are assigned a GUID that is stored in the license activation file. However, some custom applications might not have a GUID.

- **Resources**

Under **Resources** you can check limitations on API (`pinetMgr.MaxAPIConnections`) and SDK connections (`pinetMgr.MaxSDKConnections`). For example, if you reach the maximum number of SDK connections, you get a message that looks something like this:

```
>> License Error: [-12212] Maximum licensed SDK Application connections exceeded.  
Connection refused. Maximum allowed: 5. Process name: (calling process name) ID:  
(process ID)
```

View licenses for Data Archive collective nodes

For Data Archive collectives, you typically have licensing limitations on the number of secondary nodes you can have. If you try to configure more secondary servers than you are licensed for, you get an error message that looks something like this:

```
0 pilicmgr 7-Jun-08 09:10:56  
>> Failed to get a license for the secondary server - Aborting ... [-12206] usage exceeded  
licensed amount
```

To see how many secondary servers you are licensed for:

1. In **Servers** pane under **Collective**, select the checkbox next to the primary server.
2. In the **System Management Tools** pane, double-click to expand **Operation** and click **Licensing**.
Data for the primary server displays in the right pane.
3. Choose **secondarynode** from the drop-down menu in the toolbar located on top of the right pane.
4. Expand the primary server, and under **Resources**, double-click to expand the `pilicmgr.MaxSecondaryNodeCount` item.

Total is the number of secondary nodes you are licensed for and **Amount Used** is the number you are currently using. **Amount Left** is the number you have left.

Check licensing for PI BatchView

Under **Seats**, click to expand the `BatchView.Default` node.

Total indicates the number of seats you are licensed for. **End Time** indicates the time at which this license expires, if any.

Check whether you are licensed to use PIBatch

To see whether you are licensed to use PIBatch:

1. Choose **batch** from the drop-down menu in the toolbar.

2. Under **Programs**, find pibatch.Default. A value of **Connection Allowed** indicates that you are allowed to use PIBatch.
3. Click to expand the pibatch.Default node.

End Time indicates the time at which this license expires, if any.

Check licensing for SQC

1. Choose **sqc** from the drop-down menu in the toolbar.
2. Under **Programs**, find pialarm.RTSQCAllowed.

A value of **Connection Allowed** indicates that you are allowed to use SQC.

3. Click to expand the pialarm.RTSQCAllowed node.

End Time indicates the time at which this license expires, if any.

4. To check your licensing for SQC client connections, look under **Seats**.
5. Click to expand the **SQC.Default** node.

Total indicates the number of seats you are licensed for. **End Time** indicates the time at which this license expires, if any.

Mappings and trusts

Use the PI SMT Mappings & Trusts tool to create and edit PI mappings and trusts on the Data Archive server.

Before Data Archive 2016 R2, trusts were typically used to authenticate PI interfaces, while on the Data Archive server, mappings were used to authenticate Windows users via single sign-on. With PI API 2016 for Windows Integrated Security, Windows authentication extends to PI interfaces.

Note: PI API 2016 for Windows Integrated Security extends Windows authentication to API-based client applications. If you choose to install PI API 2016 for Windows Integrated Security, you can use only Windows Integrated Security for authentication. Both trusts and explicit logins will fail. For more information, see [Overview of Data Archive security](#).

Manage mappings

You manage PI mappings from the PI SMT Mappings & Trusts tool. The **Mappings** tab shows you all the mappings defined for the selected Data Archive server. The **Mappings** tab does not appear unless you are connected to at least one Data Archive server version 3.4.380 or later. Earlier versions do not support mappings.

PI mappings

PI mappings create an association between a PI identity, user or group to one of the following:

1. A Windows user or group (such as an AD group)
2. An Identity Management service role

A role is a group of users with similar job functions and access permissions. Roles are stored and managed by the Identity Management service. Role members that are mapped to a PI identity inherit the same access permissions as the PI identity.

Note: The AVEVA Identity Manager is the provided identity service for PI Server 2023.

OpenID Connect (OIDC) can be used to map a role to a PI identity. OIDC provides a secure way to map roles to PI identities through claims, and then grant members of the role the same level of access to PI Server resources.

To view or edit the mappings for a particular PI identity, PI user, or PI group, use the [Identities, Users, and Groups](#) tool.

Windows side of the mapping

You can map a PI identity to an Active Directory user or group or to a local user or group.

Your PI mappings can use the following Windows resources:

- Any Active Directory (AD) Principal (user or group)
- Any local Windows user or group

If you use local Windows security, then the Windows user accounts on Data Archive must exactly match the Windows user accounts on each client workstation. The account names, and also the passwords must be identical on the server and all client machines. When a password changes for a Windows user account, you must make that change on Data Archive and all participating client machines.

Local Windows authentication is not as secure as AD authentication. If at all possible, use Active Directory (AD) rather than local Windows security. Local Windows authentication is still far more secure than individual PI user accounts.

Data Archive side of the mapping

On the Data Archive side of the mapping, you can use a PI identity, a PI user, or a PI group. It is best to use a PI identity, for consistency. PI users and PI groups imply direct management of users on Data Archive. If you are no longer managing users on Data Archive, this could cause confusion for other Data Archive administrators. If you use PI identities only for mappings or trusts and PI users/groups only for PI account management, then other administrators can easily interpret your security configuration.

Note: If you need to use a PI user account in a mapping, then disable explicit logins for that account. PI user passwords are not secure.

For more information about configuring security, see [Overview of Data Archive security](#).

Create a mapping in PI SMT

Procedure

1. In the **Collectives and Servers** pane, select the server or collective.
 2. In the **System Management Tools** pane, select **Security**, then select **Mappings and Trusts**.
 3. Select the **Mappings** tab.
 4. In the toolbar, click the **New** button to open the **Add New Mapping** dialog box.
 5. Optional: Select the **Windows** option if OpenID Connect is enabled.
 6. In **Windows Account**, enter an AD principal or a local Windows group or user. To select the account either:
 - Click the browse button to browse for the account.
 - Type the account name, then click the resolve SID button to verify that this is a valid account. If the account is valid, an SID appears in the field. Otherwise, a dialog box with an error message opens.
 7. In **Description**, enter a description of the mapping. There are no restrictions on the contents of this field, and the field is optional.
 8. In **Server**, choose the server where you want to create the mapping.
This drop-down list contains all the Data Archive servers that are selected under Collectives and Servers and that are version 3.4.380 or later versions. Earlier versions of Data Archive do not support mappings.
 9. In **PI Identity**, enter a PI identity, group, or user.
 10. To choose, click the browse button and open the **Select PI Identity, PI Group, or PI User** dialog box. Make a choice in **Type** to filter the choices. In the list, choose either a PI identity, a PI group, or a PI user, and then click **OK**.
- We recommend choosing a PI identity. PI users and PI groups represent an older security model that involves managing individual user accounts on Data Archive. PI user accounts and passwords are not as secure as

Windows accounts.

11. Click **OK** to create the mapping.

Map a role or client ID to a PI identity using OIDC

You can map an AVEVA Identity Manager (AIM) role or client ID to a PI identity using Open ID Connect (OIDC) instead of Windows authentication. This feature enables you to use claims-based authentication to map a role or client ID to a PI identity in PI System Management Tools (SMT).

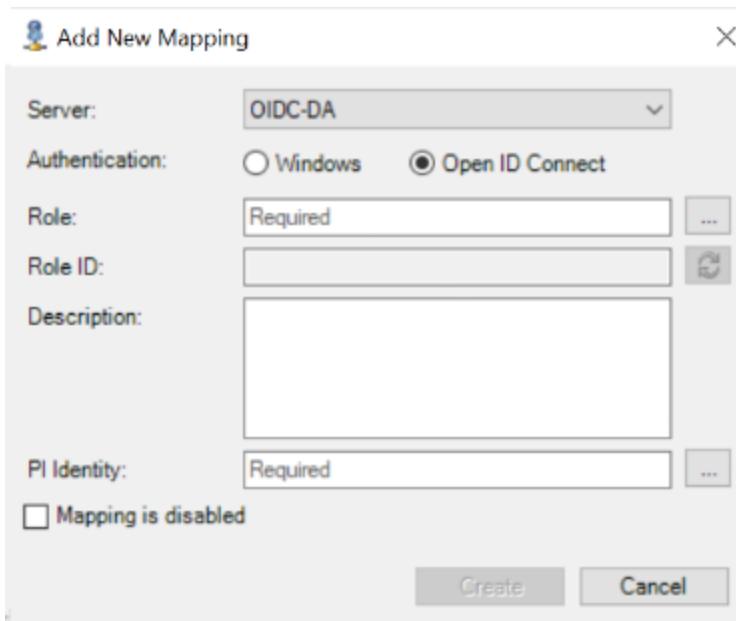
The basic process for mapping a role or client ID to a PI identity is very similar to mapping a Windows user or group to a PI identity. The only difference is selecting the OIDC authentication method during the process. In PI SMT, claims-based authentication using OIDC is restricted to the creation of PI Identity mappings. All other PI SMT plug-ins and tools use Windows authentication.

Prerequisites:

- Your AIM Server account must be a member of the Application Access (AA) administrator group on the AIM server. If you try to sign on to the AIM server during this procedure using an account that is not a member of the AA administrators group, you cannot complete the procedure.
- Before you can map a role to a PI identity, Data Archive and the AIM server must be configured. See [Data Archive installation and upgrade](#).

To create a new mapping from a role or client ID to a PI identity using OIDC, complete these steps:

1. Open the PI SMT Administrator window.
2. In the **System Management Tools** pane, select **Security**, then select **Mappings and Trusts**.
A list of the current mappings is displayed in the **Mapping** tab.
3. Do one of the following:
 - Right-click in the list view of the **Mappings** tab.
 - Select the **Add New Mapping** icon (directly above the Mapping tab).
The **Add New Mapping** dialog opens.
4. Select the **Open ID Connect** option.



5. Retrieve roles or client IDs from the AIM server for mapping, and choose them in the **Select an OIDC Mapping** dialog. Once this step is complete, proceed to the **Add New Mapping dialog** to finalize the mapping process.

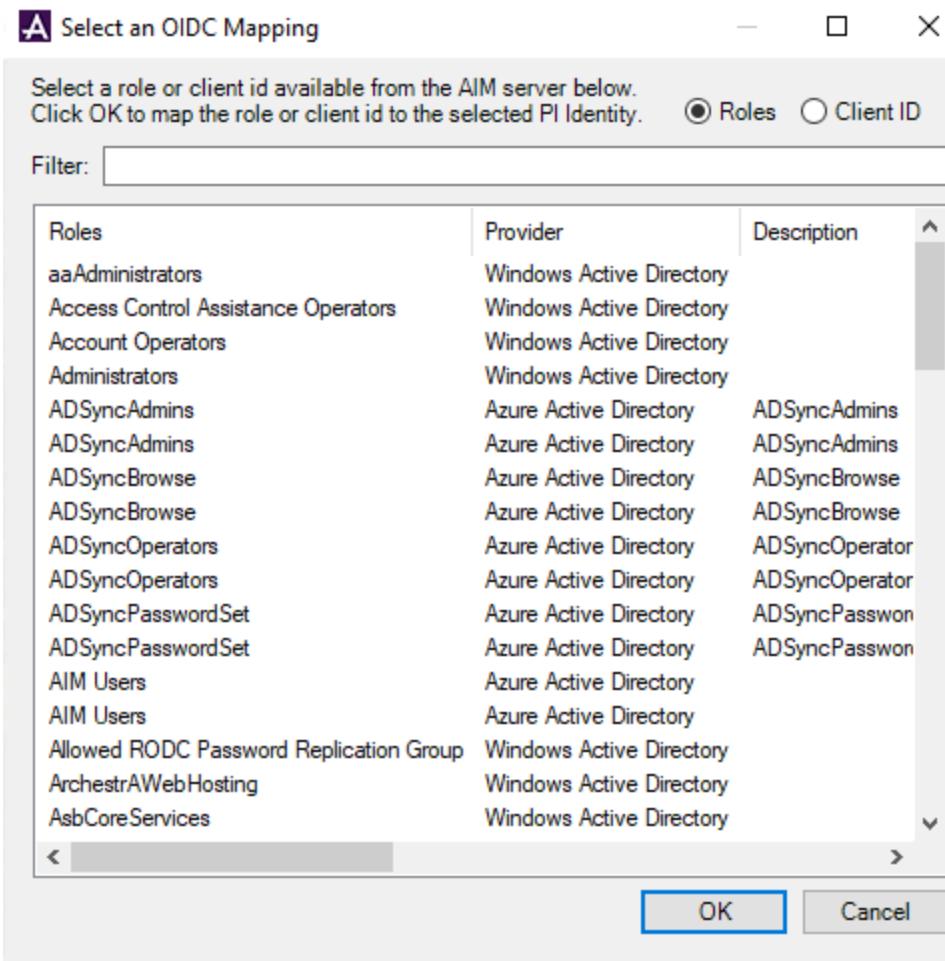
- In the **Add New Mapping** dialog, select next to the **Role** text box, or **Client Name** text box. A connection message displays to notify you that PI SMT is attempting to connect with the AIM server.
- Once the connection is made, enter your credentials in the sign on page, and click **Sign in**. The browser displays a message that sign on was successful.

Note: If the connection is not made in a reasonable amount of time, click **Cancel** to stop the connection request and try again.

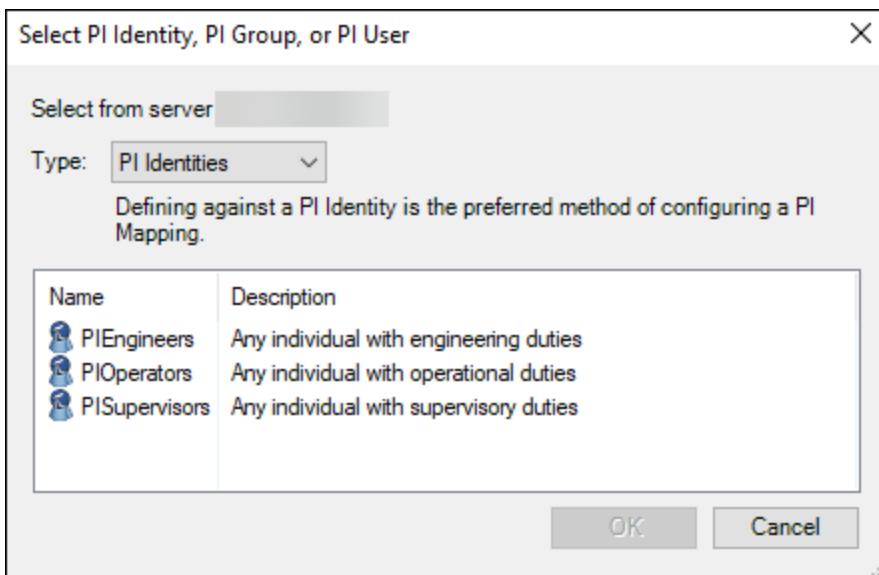
The **Select an OIDC Mapping** dialog opens populated with the list of configured roles or client IDs.

- Select the mapping type, and use the **Filter** text box to search for specific roles or client IDs. Double-click your selection, and then click **OK**. This dialog lists only the current roles or client IDs that you have access to that can be mapped as a PI identity.

Tip: Select **Role** mapping when you want to manage access by groups, and choose **Client ID** mapping for pinpointing access to specific users.



6. Return to the **Add New Mapping** dialog which now shows your selections either in the **Role** text box, or in the **Client name** text box.
7. In the **Description** text box, enter a description for the new PI identity mapping.
8. In the **Add New Mapping** dialog, select next to the PI Identity text box. This action prompts a new window to open, displaying the **Select PI Identity, PI Group, or PI User** dialog. Double-click on your choice, and then click **OK** to confirm your selection.



9. Return to the **Add New Mapping** dialog, click **Create**.

The new PI Identity mapping is automatically added to the **Mappings** tab in PI SMT.

Manage trusts

You manage PI trusts from the PI SMT Mappings & Trusts tool.

Alternatively, you can manage trusts from the Identities, Users, & Groups tool. Open the Properties dialog box for an identity, a user, or a group and then click the **Mappings & Trusts** tab. Here you can create, edit, or delete PI trusts that are defined against that identity, user, or group.

PI trusts

A PI trust compares the connection credentials of a connecting application to criteria specified in the trust. Each PI trust is defined against a PI identity, a PI group, or a PI user. The trust gives the connecting application the same access permissions as the associated identity, group, or user.

Before Data Archive 2016 R2, trusts were typically used to authenticate PI interfaces. Windows authentication extends to PI interfaces through PI API 2016 for Windows Integrated Security.

Using PI trusts or explicit logins for authentication is not recommended. For a more secure environment, OpenID Connect (OIDC) or Windows Integrated Security (WIS) is recommended.

Note: PI API 2016 for Windows Integrated Security extends Windows authentication to API-based client applications. If you choose to install PI API 2016 for Windows Integrated Security, you can use only Windows Integrated Security for authentication. Both trusts and explicit logins will fail. For more information, see [Overview of Data Archive security](#).

PI trust authentication process

If you understand how the Data Archive server authenticates PI trusts, it will be easier to understand how to configure the PI trust:

1. When an application attempts to connect, it sends some connection information to the Data Archive server. The connection information includes the application name, and some information about the computer that the application is running on.
The connection information is different depending on whether it is a PI API connection or a PI SDK connection. You need to know the type of connection in order to configure the PI trust. See [Connection types](#).
2. The Data Archive server compares the connection information sent by the application to each PI trust defined on the Data Archive server. Each PI trust is defined by a trust record in the trust table. Each field in the PI trust record is compared to the corresponding field in the connection credentials. You can leave some fields blank when you define the PI trust. Blank fields are not compared to the connection credentials. Every field that is not blank in the trust record must exactly match the passed credentials. Otherwise, the authorization is not granted. The more information you enter in the PI trust definition, the more difficult it is for an interface or client application to match the trust.
3. The Data Archive server compares connection credentials to each trust record. If only one record matches exactly, that record is used to grant login. If more than one record contains matching fields, then the record that matches most closely is used.
4. If a match is found, the connection is granted the same access permissions as the PI user, group, or identity defined in the trust.

Default trusts

Data Archive includes default trusts that guarantee access to all applications running on the local machine. These default trusts are automatically recreated every time the system starts, to ensure that they are always configured in their default state. Different versions of Data Archive have different default trusts. The following list includes default trusts for PI Server 3.4.375 and later.

Trust name	Description	Data Archive version
<i>!Proxy_127!</i>	Allows access for local AVEVA™ PI System™ applications.	All versions
<i>!PIServerName_FQDN!</i>	Used for communication between primary and secondary servers in a Data Archive collective.	Not needed for Data Archive versions 3.4.380 and later.

Data Archive versions 3.4.380 and later do not need the FQDN trust and it is not included in new installations. However, that trust is not removed if Data Archive is upgraded from an earlier version.

PI SDK trusts

PI SDK 1.3.6 and later supports Windows authentication, which is more secure than PI trusts. On Data Archive versions that support Windows authentication (3.4.380 and later) you should use Windows authentication (PI identity and PI mapping) instead of a PI trust. If you create a trust, application users might still be authenticated through Windows and not the trust (see [Windows authentication versus SDK trusts](#)).

Windows authentication versus SDK trusts

If a Windows user running an SDK application has access to the Data Archive server through Windows authentication (PI mappings and PI identities), then that user will be authenticated through Windows, rather than through the trust. This is because newer versions of the SDK try Windows authentication first.

This means that their access permissions will be dictated through the mappings, rather than the trust. It is best to retire SDK trusts wherever possible, and rely on the Windows authentication instead.

Configure SDK authentication protocols in SMT

When a PI SDK application attempts to connect to the Data Archive server, it tries all available authentication methods until it succeeds. You can configure the order in which it tries the authentication methods. The possible methods are:

- Windows Security. We recommend you use this method wherever possible.
If a valid PI mapping exists for the Windows user (or for any group to which the user belongs) then the user is authenticated as the PI identity, PI user, or PI group defined for that mapping.
- PI Trust.
If the connection request matches an existing PI trust, then the user is authenticated as the PI identity, PI user, or PI group defined for that trust.
- A default PI user account. We do not recommend this method.

The first method that succeeds defines the access permissions granted to the connecting application. After an authentication method succeeds, no other methods are tried. By default the SDK (1.3.6 and later) tries to authenticate first through Windows.

You can use PI SMT to configure the methods the SDK tries first:

1. Select **File > Connections** to open PI Connection Manager.
2. Select **Tools > Options** to open the Connection Option dialog box.
3. Under **Specify Authentication Procedure**, specify which protocols to allow and in which order to try them in **Protocol order**.

Note: You can also access PI Connection Manager from the About PI-SDK application. Select **File > Connections**.

Create a PI trust

To create a PI trust, you need to know connection types (API or SDK). The type of connection determines what information you can use to define the trust. At a minimum, for a new PI trust you have to specify the following:

- The name of the trust (must be unique on the Data Archive server)
- The name of the PI identity, PI group, or PI user against which this trust is defined (this determines the access permissions granted to the connecting application)
- One optional specification from the following list:
 - Application name

- IP information as a network path or IP address and netmask for the connecting computer
- Windows account information (SDK only) (only for SDK applications)

See also [Connection types](#).

Trust wizard and Advanced Trust dialog box

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Security > Mappings and Trusts**.
3. Click the drop-down menu next to the **New** button and choose **Wizard**.
The Add Trust wizard appears.
4. Follow the wizard prompts to enter the connection type and the configuration information valid for that connection type.

Note: You can alternatively choose the **Advanced** option. This option opens the Add New Trust dialog box, where you can specify configuration information for all types of PI trusts. It is possible to enter information that is not valid for the connection type, therefore if you are not sure, use the wizard.

Connection types

When you configure a PI trust, you need to know the type of connection the trust will be used for. There are two different Data Archive connection types. Each PI interface and client application is configured to use one or the other. (There are also a few interfaces that use both.) The two mechanisms are:

- PI API Connection: Most PI interfaces use the PI API to connect to the Data Archive server.
- PI SDK Connection: Most client applications use the PI SDK to connect to the Data Archive server.

The PI API and PI SDK connections both support Windows authentication, which is the most secure authentication method available for the Data Archive server. We recommend that you always use Windows authentication when possible.

For more information about configuring security, see [Overview of Data Archive security](#).

Application name

A PI trust can require a specific application name. When you specify an application name in a trust, you have to use the appropriate format for the connection type:

- Applications that connect through the API Connecting PI API applications send an identifier called an application process name, or procname. This is a four-character string with an E appended. For example, the procname for the Perfmon Interface is: PIPeE

Note: PI API versions before 1.6.0 always send a five-character string: 4 characters plus a capital E. For PI API versions 1.6.0 and later, the name can be up to 8 characters, without a trailing capital E.

- For applications that connect through the SDK, use the full name of the connecting application, including the extension, but not the path. For example, the application name for the ICU is: **PI-ICU.exe**

If you are running the same PI interface on another Data Archive server, then you can use PI SMT to determine

the correct application name. Select **Operation > Network Manager Statistics**. Find the interface in the list. The application name is listed in the **Name** field.

See also: [Connection types](#)

IP information as a network path or IP address and netmask

You can define a trust based on the IP information of the computer that runs the PI interface or client application that you want to trust. To find the necessary information, you can run **pidiag -host** on the interface (or client) computer. This returns the connection credentials as retrieved from the local operating system.

You can specify IP information as a network path or IP address and netmask:

- **Network Path.** The fully-qualified domain name. For example, **my_laptop.my_company.com**
- **IP Address.** IP address of the interface computer (for static IP address only).
- **NetMask.** If you specify an IP address, you must also explicitly provide a netmask value. Failure to do so will generate an error. If you are requiring an exact match on an IP address, specify the netmask as **255.255.255.255**. If you are specifying a Class C subnet, specify the netmask as **255.255.255.0** and the fourth field of the IP address as **0**.

Note: When applications run on machines with multiple network cards, it is unpredictable which credentials are sent to the Data Archive server for the trust authorization. AVEVA recommends that you either avoid such configurations, or create a PI trust for every IP address on the machine where the application runs.

Windows account information (SDK only)

For SDK connections only, you can specify Windows account information as part of the PI trust.

- **Windows Domain:** the Windows domain of the user who is running the application.
- **Windows Account:** the Windows user name of the user who is running the application.

It is possible to use a dollar sign (\$) for the user name for a particular domain; then users on that domain can connect through this trust to existing PI users of the same name as the Windows user running the application. This is called a dollar-sign trust. To configure a dollar-sign trust, you typically first import your Windows usernames and passwords as PI users. See [Import Windows users](#). (The dollar sign is a valid value for the domain as well as the user name.)

Note: A PI mapping serves the same purpose as a trust based on OS user name and Windows domain membership. With Data Archive version 3.4.380 and later, you should use PI mappings instead of dollar-sign trusts.

Remove a PI trust

1. Under **Collectives and Servers**, select the server or collective.
2. In the System Management Tools pane, select **Security > Mappings and Trusts**.
3. Select the Trust tab.
4. Select the trust you want to delete and click the **Delete** button .

Edit a PI trust

1. Under **Collectives and Servers**, select the server or collective.
 2. In the System Management Tools pane, select **Security > Mappings and Trusts**.
 3. Select the **Trust** tab.
 4. Select the trust you want to edit and click the **Properties** button .
- The Trust Properties dialog box appears.
5. Edit the properties and click **OK**.

Copy a PI trust

Caution: The PI trust name must be unique on the Data Archive server. This means that when you copy a PI trust you have to change either the name of the trust or the name of the Data Archive server.

1. Under **Collectives and Servers**, select the server or collective.
 2. In the System Management Tools pane, select **Security > Mappings and Trusts**.
 3. Select the **Trust** tab.
 4. Select the trust and click the **Copy** button .
5. The Copy Trust dialog box appears.
- To copy the trust to another Data Archive server, select the server from the **Server Name** menu. Edit the trust, if desired.
 - To copy the trust on the same Data Archive server, edit the trust as desired and then change the name of the trust.

Export trusts to a file

1. Under **Collectives and Servers**, select the server or collective from the left pane.
 2. In the System Management Tools pane, select **Security > Mappings and Trusts**.
 3. Select the **Trust** tab.
 4. Click the **Export** button  (located at the bottom-right or in a dropdown).
The Save As dialog box appears.
5. In the Save As dialog box:
- Type the file path manually, if you know exactly where you want to save the file.
 - Click **Browse** to navigate to the folder where you want to save the file.
6. Click **OK** to save the file.
The file format is comma-separated values (CSV).

Import trusts from another Data Archive

To import trusts from another Data Archive, you first need to export the trust on the originating Data Archive

server and save the export file. Then import the trust file on the desired Data Archive server:

1. Under **Collectives and Servers**, select the server or collective.
2. In the System Management Tools pane, select **Security > Mappings and Trusts**.
3. Select the **Trust** tab.
4. Click the **Import** button .

Message Logs tool

The Message Logs tool is a centralized diagnostic tool for viewing logs, errors, and other messages from Data Archive, client PCs running PI SDK, and other PI applications.

Search for messages

First select a Server or Servers, and then choose a time range for which you want to see messages.

You can narrow the search results using a selection of filters.

Message time range

The Message Logs tool shows messages by time range. By default, the Message Logs tool is set to display all messages over the past five minutes. You can change the default time range by changing the setting in the Options dialog box.

You can change the current time range by changing the values in the **Start** field and **End** field.

These fields can take either a relative time or an absolute time. So, for example, to get messages for the last five minutes, you could set the start time to five minutes ago and the end time to now:

```
Start: *-5m
End: *
```

The interval between the start and end time specifies a range. The Message Logs tool uses this range to group messages so that you can use the forward and back arrows (▶◀) to see the next or previous set of messages. The above example specifies a five minute range. If you click the back arrow, you get all the messages between five and ten minutes ago. If you click the back arrow again, you get all the messages between ten and fifteen minutes ago, and so on.

Note: ▶◀ are enabled only when the **Server & SDK Logs** tab is selected, and at least one Data Archive server is selected.

Refresh message list automatically

Select **Refresh Automatically** for **Server and SDK Logs** for the selected servers to be refreshed at the rate configured on the Options dialog box.

Filter messages

When you choose a time range, the Message Logs tool shows all the messages that occurred in that time range. Depending on the time range, this might be a long list. You can filter the list to show only the messages you are interested in.

Filter by source program

Choose a source program from the drop-down **Program** menu.

This menu is populated with the source programs for all currently displayed messages.

Filter by message details

To search for text from the message description, type the text in the **Message** field.

The filter can include wildcards either in the beginning or at the end. The wildcard character is asterisk (*). For example, to search for all messages that begin with the text *User query failed*, you would use:

User query failed*

Message severity levels

Severity indicates whether the message is informational or requires immediate attention; the severity levels are, in order of most severe to least severe: Critical, Error, Warning, Information, and Debug. You can choose to display messages filtered by severity.

When you select a level to display, you are selecting to view all messages of the selected severity or greater. For example, to see all messages, select **Debug**.

Messages displayed for each severity selection

Selected severity level	Displayed messages
Critical	Critical only
Error	Critical, and Error,
Warning	Critical, Error, and Warning
Information	Critical, Error, Warning, and Information
Debug	Critical, Error, Warning, Information, and Debug

Severity levels

Severity level	Description
Critical	Loss of system functionality; requires immediate attention
Error	Action failed
Warning	An anomaly has occurred that does not impact the user
Information	Action succeeded
Debug	Debug/Tracing message

Display advanced filter options

If you know more detailed information about the message or messages you want to find, click the **Advanced** button in the filter options section.

- ID: the full message ID number
- Count: the maximum number of messages you want to display in the Message Logs tool
- Message source (see [View message source fields](#))
- Process identity (see [View identifying information about a process](#))
- Originating identity (see [View identifying information about the origin of a message](#))

Note: When any advanced settings are specified, the **Advanced** button appears highlighted in blue.

[View message source fields](#)

Source1, **Source2** and **Source3** fields can be used according to the user's discretion. For example, in the default Data Archive configuration, PI Net Manager uses the **Source2** field to denote connection information messages as a subset of its other messages.

The Message Logs tool does not show source fields by default.

1. To see the source fields, right-click anywhere in the column headings.
A menu of available column headings appears.
2. Check the source fields that you want to display in the Message Logs tool.

[View identifying information about a process](#)

Select the **Identity** tab in the Message Logs tool to view details about the process identity for the selected message.

- **Host**
Name of the host computer that prompted the message
- **OSUser**
Authenticated user or application that sent the message to the log
- **PI User**
Name of the PI identity or PI user that sent the message to the log
- **ID**
Message ID

[View identifying information about the origin of a message](#)

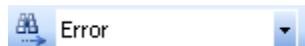
Select the **Identity** tab in the Message Logs tool to view details about the originating identity for the selected message.

- **Host**

- Name of the server from which the message originated
- **OSUser**
 - Authenticated user or application that sent the message to the log
- **PI User**
 - Name of the PI identity or PI user that sent the message to the log

Find messages displayed in Message Logs tool

To search for an item in the list of messages currently displayed in the Message Logs tool, enter text into this field in the toolbar:



- Click or press **Enter** to find the next message that matches the search criteria.
- Click to redisplay the messages that fit the current criteria.

Find messages by file properties

On the **PIPC logs** tab only, you can filter messages by file properties.

Enter one or more of these file properties:

- **Name** and location of the message file, for example, **C:\Program Files\pipc\dat\pipc0000.log**.
- **File start time**
- **File end time**

View message logs

Use the Message Logs tool to conduct advanced searches for, and then view log files from different locations.

Select the tab for the logs you want to search. For each tab, you can use filtering criteria to limit the messages displayed.

View Data Archive & PI SDK Log details

1. Select the **Server & SDK Logs** tab to view messages retrieved from the PI SDK and Data Archive log files.

Message details are displayed both in a tabular format, and in the **Message Details** tab at the bottom of the active pane. Details include:

- **Server**
 - The source of the message
- **Collective**
 - For the server, if applicable

- **ID**

The message ID, if applicable

Note: All messages created prior to the introduction of the PI Message Database have an ID **0**. In most cases, the higher the ID number, the greater the message severity.

- **Time**

The time when the message was sent

- **Severity**

Indicates the seriousness of the message. See [Message severity levels](#) for definitions of the severity levels used in the default Data Archive configuration.

- **Message Source**

Shows:

- The **Program** that generated the message
- [View message source fields](#)

Server	Collective	ID	Time	Program	Severity	Source1
XPISERVER		0	6/26/2009 3:13:20 PM	SMTHost.exe	Information	
XPISERVER		0	6/26/2009 3:13:20 PM	SMTHost.exe	Information	
XPISERVER		0	6/26/2009 3:13:15 PM	SMTHost.exe	Information	
XPISERVER		6079	6/26/2009 3:12:45 PM	pibassess	Information	Point Table
XPISERVER		0	6/26/2009 3:12:33 PM	SMTHost.exe	Information	
XPISERVER		0	6/26/2009 3:12:30 PM	SMTHost.exe	Information	
XPISERVER		7080	6/26/2009 3:12:07 PM	pinetmigr	Information	Connection Information
XPISERVER		7082	6/26/2009 3:12:07 PM	pinetmigr	Information	
XPISERVER		0	6/26/2009 3:12:07 PM	SMTHost.exe	Information	

Message:

```
PI-MDB> Error: MDBTree ERROR: The target server database failed to load.[-12218] Not licensed to use Module Database .Class = HeadingSets, Method=Constructor
```

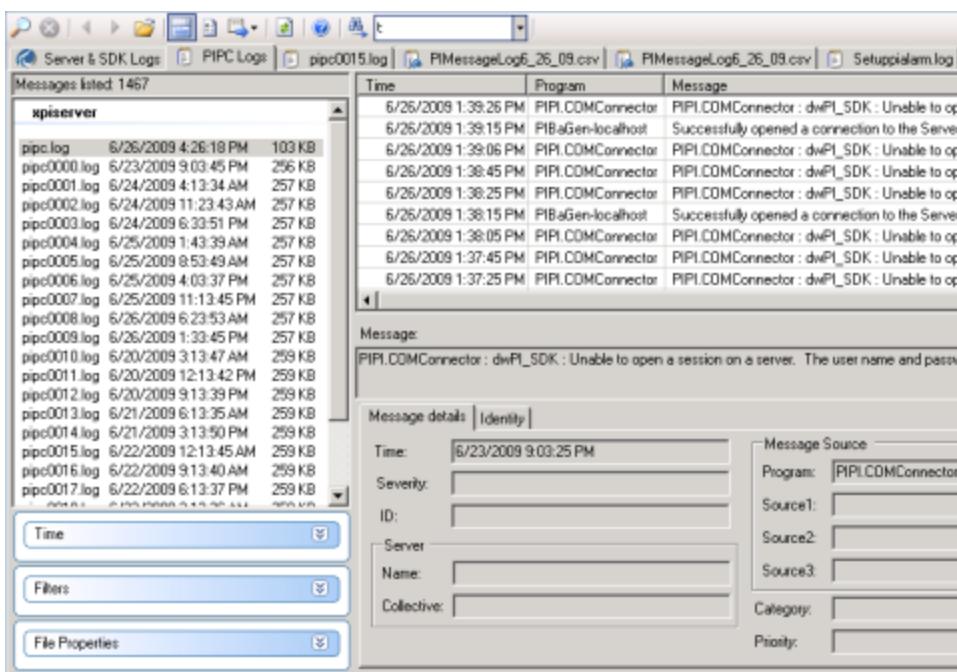
Message details | **Identity**

Originating Identity	Process Identity
Host: <input type="text"/>	Host: <input type="text"/>
OS User: <input type="text"/>	OS User: PISECTEST.LOCAL\dbobowman
PI User: <input type="text"/>	PI User: <input type="text"/>
	ID: 4952

View PIPC log details

1. To view a list of the PIPC log files on the local machine, select the **PIPC Logs** tab.
2. Under **Messages listed**, select a log file to display the messages in that log file.

You can also right-click a log file and select **Display** to show that log file's messages in the table, or select **Open with Notepad** to open the file in Notepad.



- In the table, view a list of messages in the selected file, ordered by the **Time** that the message was sent and the **Program** that generated the message.

By default:

- The oldest timestamp in the log file is the start time of the display, and the newest timestamp in the log file is the end time of the display.
- All text filters are set to *, to show everything without filtering. However, if you specify a filter before opening the file, the tool applies that filter to the displayed messages.

- Select a message in the table to view information about the message in the details pane.

Note: Alternatively, you can double-click the message to open the Message Detail dialog box, which shows the same information as the details pane.

Details include:

- The **Message** box, which shows the text in the message.
- The **Message details** tab, which shows information about the time and source of the message, as well as the Data Archive server, if appropriate.
- The **Identity** tab, which shows information about the originating identity and the process identity.

View other log files

You can open additional log files in separate tabs, provided they are recorded in LOG, TXT, CSV, or XML formats generated by **PIPC.LOG**, **PIGetMsg.exe**, or exported from the Message Logs tool.

To open a log file, click the **Open File** button , browse to the desired log file, and click **OK**.

You can open files with the following formats.

- LOG files are present on client nodes in the default location (..\PIPC\Dat subfolder), or the user specified location, if a **pipc.ini** file is found on the local computer

- DAT files belong to the Data Archive or PI SDK and are present on server nodes (..\PI\Log subfolder)
- TXT, XML and CSV formats generated by **PIPC.LOG**, **pigetmsg**, or exported from the Message Logs tool.

Set Message Logs options

To specify default values for the Message Logs tool and filter options, click the **Options** button . Use the Options dialog box to set:

- **Initial Time Range**

Enter the default start and end times for the Message Logs window period in PI time format. The Message Logs window displays only messages occurring within the time range. New values are validated automatically, and an invalid time format causes the field to turn yellow.

- **Initial Message Retrieval**

Select to load messages based on the default **Initial Time Range** when the Message Logs tool is first opened.

- **Refresh Display Automatically Rate**

Specify the rate at which messages are refreshed on the **Server & SDK Logs** tab when **Refresh Automatically** is selected.

- **Display Options**

Specify where you want options displayed, **Options on left** or **Options on top**.

Export messages

These file formats are supported for export:

- XML (XML)
- Text (TXT)
- Comma-Separated Value (CSV)

Click  or right-click in the list of messages, and then:

- To export the entire list of messages displayed, select **Export All**.
- To export only selected messages, select **Export Selected**.

Message Logs quick reference

Goal	Right-click option	Toolbar icon
Find and retrieve messages.	N/A	
Cancel message search.	N/A	
Move the time frame for message	N/A	

Goal	Right-click option	Toolbar icon
display back one period and refresh the view. The time frame is adjusted by the total time period currently displayed, where the original start time becomes the new end time.		
Move the time frame for message display forward one period and refresh the view. The time frame is adjusted by the total time period currently displayed, where the original end time becomes the new start time.	N/A	
Open a log file.	N/A	
Set default values for the Message Logs tool and filter options.	N/A	
Export the currently highlighted messages to a file of one of the following formats: <ul style="list-style-type: none"> • XML • Text • Comma-Separated Value (CSV) 	Export	
Export the contents of the Message Logs tool to a file.	Export All	
Refresh the Message Logs active pane based on current settings	Refresh	
Launch online help	N/A	
Find the next instance of the text in the adjacent field within the Message Logs tool.	N/A	
Display a PIPC log file.	Display	N/A
Open a PIPC log file in Notepad.	Open with Notepad	N/A
Toggle the Message detail tabs on and off.	N/A	

Goal	Right-click option	Toolbar icon
Copy selected messages to the clipboard.	Copy	N/A
Highlight all messages.	Select All	N/A

Module Database Editor

The Module Database Editor is a tool for working with the content in the Module Database on a Data Archive server.

About the Module Database

Each Data Archive includes a PI Module Database (MDB). The MDB stores information in a hierarchy that you can use to organize and classify resources according to your organization's needs. For example, you can store and view equipment type, office location, building names, and so on. You can also use the Module Database as a central registry for several PI applications' settings, including PI ICU.

Note: With PI Server 2010 and later, you cannot make changes to MDB unless it is synchronized with PI AF. See the MDB to AF Synchronization tool in PI SMT ([Operation > MDB to AF Synchronization](#)) for more information.

Query Module Databases

The PI Module Database retains version histories for all objects. Using the Module Database tool, you can display the PI Module Database hierarchy and object values for any selected time.

In the box above the Module Database tree view, enter a date and time and click **Apply**, or click  to display the Query Date Select dialog box, where you can enter a date and time.

All times are based on the computer where the Module Database tool runs.

The default is the current date and time. The minimum acceptable date is January 1, 1970 (GMT).

Note: The query retrieves Module Database data from all connected servers.

View the Module Database tree

The Module Database tree has the following structure:

- The root level shows the connected Data Archive servers. Each Data Archive server contains one Module Database.
- Each Data Archive server may have two types of branches: PI heading sets and PI modules. Each PI module can be associated with a PI heading set and a corresponding PI heading.

Note: Heading set and heading objects are not migrated to AF, and AF elements are not assigned a PI heading even if the corresponding PI module references a PI heading.

If you don't see all expected objects, right-click in a blank space around the Module Database tree and click **Show**, then click additional data types you want to display (for example, **Properties** or **Headings**).

Heading Sets

The HeadingSets branch contains PI heading sets, which contain PI headings. PI headings are used to promote common naming conventions for modules, for example, equipment. They can also be associated with sub-batches in the PI Batch Database.

For example, a PI heading object may be used to categorize a sub-batch, such as a Phase, or a position in a plant layout, such as a Line.

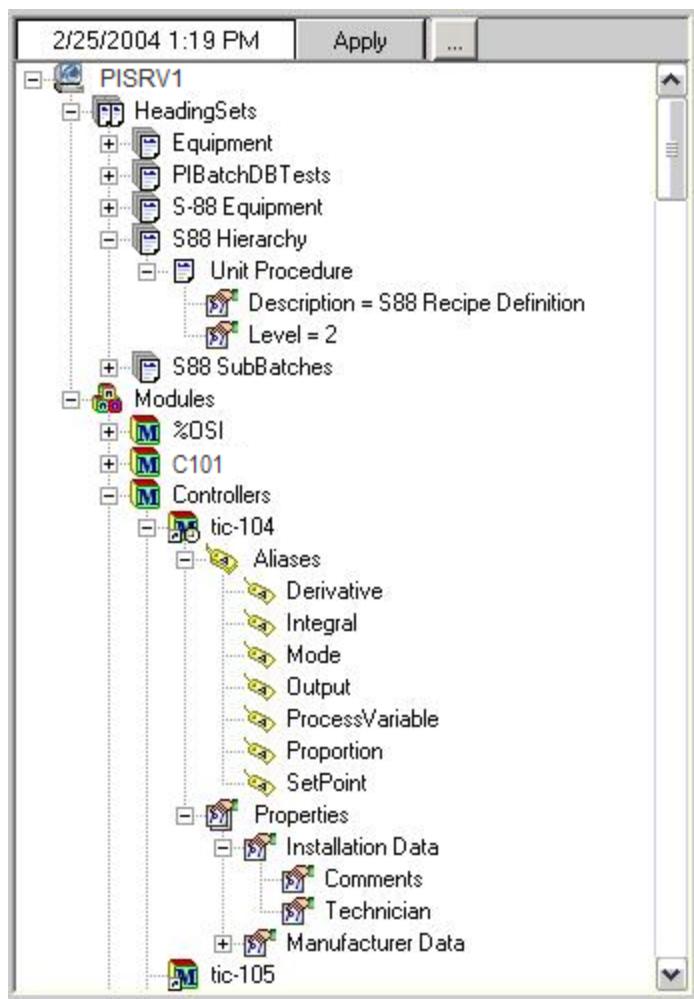
Each PI heading has two attributes: Description and Level. Description is a text string and Level is an integer. PI headings are for descriptive or informational purposes only, and no hierarchical rules are enforced.

Modules

The Modules branch contains top level PI modules. PI modules can represent various types of information. For example, a module could be used to group PI points that are related to a particular control loop, or PI points used to calculate yield efficiency. Each PI module may contain:

- Child modules 
- PIUnits , a special type of PI module specifically used for batch tracking
- Optional Aliases, , used to associate common names with PI points
- Optional Properties , which store information related to a module

Modules and Properties can be expanded to show individual items. Below is an example of a Module Database tree containing heading sets, headings, modules, aliases, and properties.



Navigate the Module Database tree

You can navigate the Module Database tree much the way you navigate files and folders in Windows Explorer.

- To expand an object, click the + next to it.
- To collapse an object, click the – next to it.
- Right-click an object to display the options available for that object.

You can update the view by right-clicking in a blank space around the Module Database tree and then clicking **Refresh** or **Reset**. **Refresh** updates the view as currently shown, including any expanded objects. **Reset** collapses the tree to the top level (servers) and updates the view.

Keyboard Commands

As an alternative to the mouse, you can use the following keyboard commands to work with the Module Database tree.

Function	Key	Description
Expand	Right Arrow	Expands the current object or moves down and selects the first child object.
Collapse	Left Arrow	Collapses the current object or moves up and selects the parent object.
Move Up	Up Arrow	Moves cursor up to the previous sibling or parent object and selects the object.
Move Down	Down Arrow	Moves cursor down to next sibling or child object and selects the object.
Page down	Page Down	Selects the last visible object.
Page Up	Page Up	Selects the first visible object.
Go Home	Home	Selects the first and uppermost object and makes it visible.
Go to the End	End	Selects the last object and makes it visible.
Select/clear check boxes	Spacebar	When check boxes are displayed, toggles the check box for the selected object between selected and cleared.
Invoke menu	Alt F10, Menu key	Invokes the context menu for the selected object.
General menu	Alt M	Invokes the general menu.
Multi-select	Shift+arrow (Ctrl+arrow)	Selects/unselects objects while moving up/down the tree (arrow keys only).
Copy	Ctrl+C	Copies selected objects.
Paste	Ctrl+V	Pastes from the clipboard.
Add/Insert	Insert	Adds or inserts a new sub-object.
Delete	Delete	Deletes the selected object or objects.

Function	Key	Description
Refresh	F5	Refreshes the display.

Edit module hierarchy

You can reorganize and edit the entire module database structure displayed in the Module Database tool.

Note: To modify a module's hierarchy, you need Read/Write access permissions to the parent module you are moving from, the module you are moving, and the module to which you move it. See [Module Database security permissions](#) for more information.

These rules apply under the following circumstances.

For all copying

- When a copied entity represents a collection of modules, PIUnits, aliases or properties, all items in the collection are copied to the clipboard.
- To paste a copied entity below it, right-click the entity in the target database.
- Pasting an entity with a duplicate name prompts you to confirm the replacement of the existing entity.
- Pasting a heading into a set with a heading of the same level number prompts you to confirm the reassignment to a different level.

For copying data within a Data Archive server

- Module links are maintained in all cases. Pasting a linked module creates a new copy of the module.
- Hierarchical relationships are maintained. You cannot make a parent a child of any of its children.

For copying data between servers

- Modules can be copied, but you cannot insert or link modules between servers.
- Module links are maintained within branches that are copied.
- Heading sets and headings are automatically created on the target server if headings of copied modules do not exist. If local headings with the same name already exist on the target server, they are used even if their heading level is different.

For drag-and-drop copying and moving

- To copy an object and its value without the tree structure, drag the object to a new location.
- To copy the entire tree branch structure, press and hold CTRL and drag. To copy values as well, press CTRL+ALT.
- To move the tree branch structure, press and hold SHIFT and drag. To copy values as well, press SHIFT+ALT.
- To insert and link a module, Press and hold ALT and drag.
- To show options for copying and moving, right-click and drag.

Context menu commands

To edit the module hierarchy, right-click a module in the tree control and select one of the following options.

Note: The options displayed depend on which node is selected in the module database tree. For most modules, you can also double-click a module to edit attributes specific to the module.

Right-click Option	Result
New	Add a new module database entity below the selected item
Copy	Copy the selected item and its associated values, but not child items
Paste	Paste the copied module database item from the clipboard below the selected tree item
Paste Values	Paste only the item and associated values (the item's effective dates), and not child items
Paste Hierarchy	Paste the item and child items
Paste Value Hierarchy	Paste the item, child items, and associated values (effective dates)
Insert Link	Insert a link to the copied item below the selected item
Delete	Delete the item. If the item is linked, all instances are deleted from the module database.
Delete Link	Delete a selected link to an item, leaving other linked instances in the module database
View All Parents	View all parent relationships for a selected linked item
Make Root Node	Create a link to the item making it the root node for the current server
Refresh	Refresh the tree control with updated values
Edit	Add or modify Module Database attributes
Edit Values	Add or edit module values

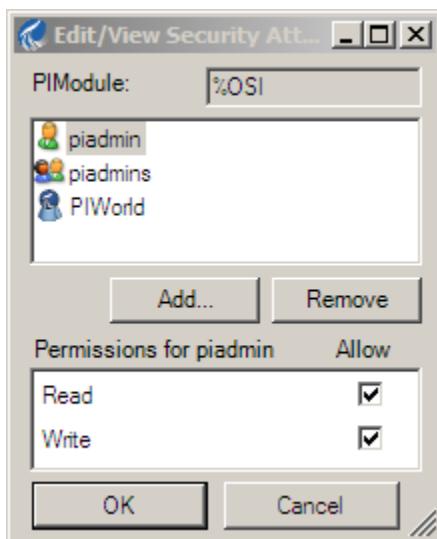
Set security attributes

Different operations on the Module Database require different access permissions. See [Module Database security permissions](#). This topic describes how to view or modify access permissions for module database objects

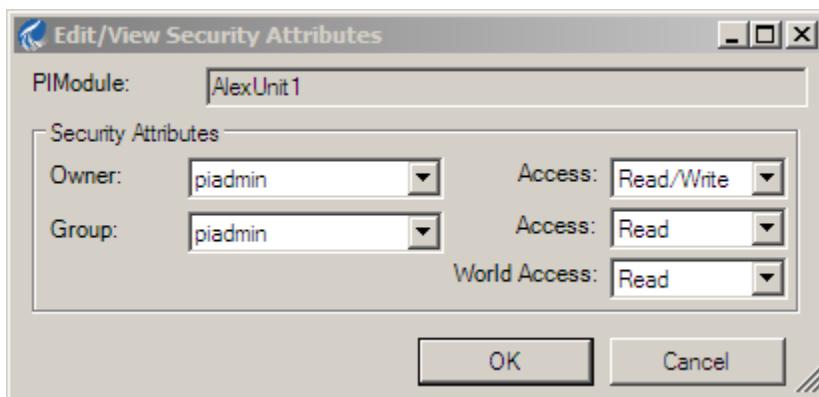
(modules, Plunits, heading sets, and headings).

Note: For all Data Archive servers, the possible levels of access are *read* and *write*. The possible access rights string can be "r", "w", "r,w" or "" (null). Note that there is no level for *deny*, as there is in Windows.

1. Right-click the object in the tree, and then click **Edit** to open the Edit/View dialog box for the selected object.
2. Next to the **Access Rights** field, click  to open the Edit/View Security Attributes dialog box. This dialog box is different for different versions of Data Archive:
 - For Data Archive versions 3.4.380 and later it shows the list of identities/users/groups that have defined access permissions for the selected object. Select an identity, user, or group to see its access permissions. You can add or remove identities, users, and groups using the **Add** and **Remove** buttons.



- PI Server versions earlier than 3.4.380 use the owner/group/world model for access permissions. Each object can have an owner, which must be a PI user and a group, which must be a PI group. You can set access permissions for the owner, access permissions for the group, and the access for everyone else (called world access). You can also change the owner or group assigned to that object.



Module Database security permissions

Permission type	Requirements
Data Archive	Required access permissions vary by task (see the following table)
File System	None required.
Registry Access	None required.
Service Control Manager	None required.

The following table lists Data Archive access permissions required for module-related tasks.

Task	Database security permissions	Other permissions
Modules: Create	PIModules (r,w)	parent module (r*,w)
Modules: Delete	PIModules (r,w)	parent module (r,w) module (rw)
Modules: Edit	PIModules (r)	module (r,w)
Modules: Edit – Link / Unlink	PIModules (r)	new parent module (r*,w) module (r*,w)
Modules: Edit – Add / Remove alias	PIModules (r) PIPOINT(r)	module (r*,w) PtSecurity (r)
Modules: Edit – Add / Remove heading	PIModules (r) PIHeadingSets (r) If you have PIModules (r), then PIHeadingSets (r) is automatically set.	module (r*,w) heading (r) heading set (r)
Modules: View	PIModules (r)	module (r*)

* module (r) also assumes (r) for all modules along the hierarchy path above it.

Add or modify Module Database attributes

Use the tree in the Module Database tool to add or modify individual nodes and their attributes including:

- Modules
- Aliases
- PIUnits
- Properties

- Headings
- Heading Sets

Users who add or modify PI Module Databases and Module Database objects require read/write permissions to the Database Security entries **PIHeadingSets** and **PIModules** (see [Set security attributes](#)). To update these Database Security entries, use the Database Security tool in PI System Management Tools.

Add or edit module attributes

Module attributes include a description, heading assignment, comment, and version information.

1. Do one of the following:
 - To add a module, right-click the parent module in the tree and then click **New > Module**. (To add a PIUnit module, you can either select the **Is PIUnit?** check box for the new module, or right-click the parent module and then click **New > PIUnit**. See [Add or edit PIUnit attributes](#).)
 - To edit a module, right-click the module in the tree and then click **Edit**.
2. Use the Edit/View PIModule Attributes dialog box to update the module.

Many fields in the Edit/View PIModule Attributes dialog box are read-only. You can edit the following fields:

- **Module Name**
Name for the module (displayed in the tree); applies to all versions (values) of the module.
- **Description**
Optional descriptive text about the module; applies to all versions (values) of the module.
- **Comment**
Optional comment about the selected version (or value) of the module.
- **Effective and Obsolete Dates**
Enter dates, or click the calendar icons to select dates and times. These dates apply to the selected version (or value) of the module.
- **Heading Sets**
Optional. To associate a heading with this module, on the Heading Sets menu, select a heading set to display a list of headings, including the following:
 - **Level**
Level number for this heading (integer)
 - **Heading**
Name of the heading
 - **Description**
Optional description of the headingThen click a heading in the list to set the **Current Heading** for the module.
- **Security Attributes**
Set user access permissions for the module. (See [Set security attributes](#).)

3. Click **OK**.

Add or edit module values

1. Double-click the module, or right-click the module and click **Edit Values**.
2. In the Module Values dialog box, you can edit any of the following by double-clicking the current value, or add new objects by right-clicking in a column and then clicking **New**:
 - Module Values and effective dates (each Effective Date in the Values column represents a version of the module)
 - Associated Sub-Modules by name and effective date
 - Associated Properties by name and value, with sub-properties displayed below the parent and denoted by a dash (-) representing each hierarchical level below the parent
 - Associated Aliases by name and tag name

You can also drag and drop modules, sub-modules, properties, or aliases from the Module Database tree to the Module Values Maintenance dialog box.

Delete module values

To delete an object in the Module Values Maintenance dialog box, do one of the following:

- Right-click the object and select **Delete**.
- Select the object and press Delete.

Add or edit alias attributes

Use aliases to associate common names with PI points. For example, you might have a group of PI points used to create a variable for a process, but the points are named according to a complex naming convention. To simplify the use of this point, you can create an alias within the Module Database named *ProcessVariable*.

1. Do one of the following:
 - To add an alias, right-click a module and then click **New > Alias**, or right-click an **Aliases** object and then click **New**.
 - To edit an alias, double-click the alias in the tree, or right-click the alias and then click **Edit**.
2. Use the Edit/View PIAlias Attributes dialog box to configure a new alias, or update an existing alias. You can set the following values:
 - **Alias Name**
Enter a name for the PI alias.
 - **Data Archive**
Enter or select the name of a connected Data Archive server that contains the point the alias refers to.
 - **Tag Name**
Enter or search for the name of the tag associated with the target PI point.
3. Click **OK**.

Add or edit PIUnit attributes

You can use the Module Database tool to add and edit some properties of PIUnits. These specialized modules are objects, or units, that represent processing equipment capable of making a batch. Most of the PIUnit configuration properties concern the start and stop times of such batches.

PIUnits are the modules where PIBatch objects are created. To create and maintain these objects, which record and track the process for producing material in batches, use the PI Batch Generator tool in PI System Management Tools.

1. Do one of the following:
 - To add a PIUnit, right-click the module under which you want to add the PIUnit and then click **New > PIUnit**. (Or you can create a module, then select the **Is PIUnit?** check box.)
 - To edit a PIUnit, right-click the PIUnit and then click **Edit**.
2. Use the Edit/View PIModule Attributes dialog box to configure a new PIUnit, or update an existing PIUnit. See [Add or edit module attributes](#) for details.
3. Click **OK**.

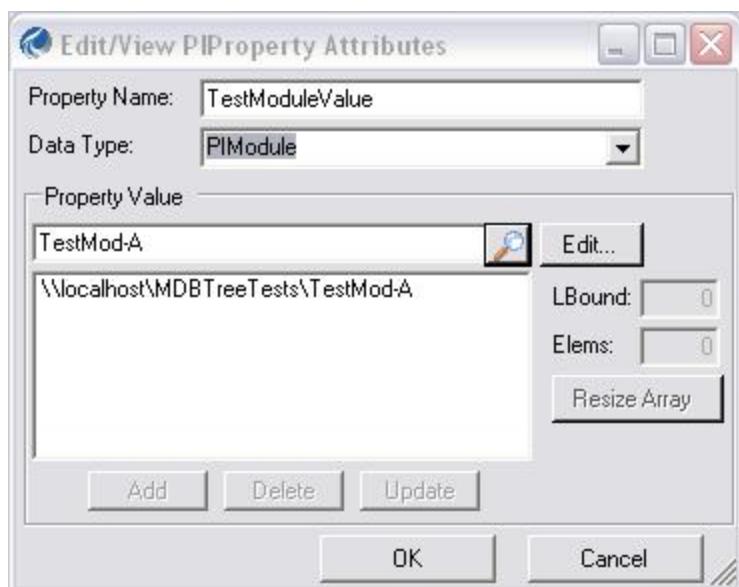
Add or edit property attributes

Property attributes store information related to a module, including its name, data type and a value. PI properties may store values for most of the data types supported by Data Archive, such as string, numeric, date, and so on. They may also store references to other module database items or file objects.

For example, you can use PI property attributes to store the serial number of a valve, the date it was installed, the technician who installed it, and its maintenance record.

Note: The Module Database editor provides read-only support for PI properties defined as file objects.

1. Do one of the following:
 - To add a property, right-click the module it will belong to and then click **New > Property**.
 - To edit a property, double-click the property, or right-click it and then click **Edit**.



2. Select an existing **Data Type** for the property value, or enter one directly in the field. Array types are denoted by square brackets `[]` following the data type name.
3. Enter a value for the property using the provided tools (not all tools are available for all data types):
 - a. Enter a value in the field for property types that take a single value.

For information about array types, see the next step.

- a. Click **Browse**  to select an entity from the module database as a property value for PIModule, PIProperty, PIAlias and PIHeading data types.
- b. Click **Edit** to update attributes of the referenced property.
- c. Use the controls to the right of the value panel to size an array for array data types and then use the value panel to enter array values separately.
- d. Click **OK** to save property values as entered.

Note: For objects and other non-displayable data types, the data type name appears in the value field.

4. Array controls (LBound, Elems) are enabled when you select an array data type, which is shown in the list followed by square brackets `[]`. For array data types only:
 - a. Set **LBound** and **Elems** values to set the lower boundary and number of elements, respectively.
 - b. Click **Resize Array** to create the array in the value panel.
 - c. To define the contents of the array, select a placeholder in the panel, then enter a value in the field and click **Update**. Use the **Add** and **Delete** buttons to add and remove selected array values or placeholders.
5. Click **OK**.

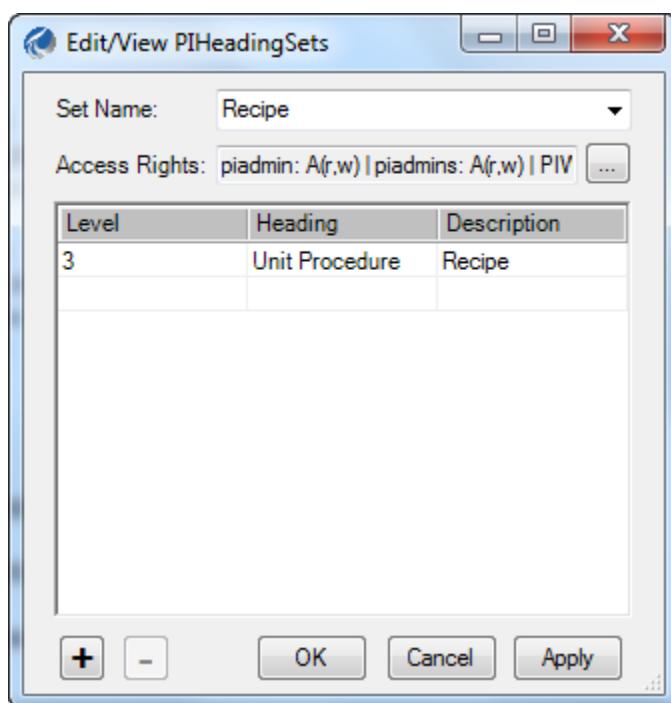
Add or edit heading set attributes

Heading sets are used to organize headings in the Module Database hierarchy.

Heading Set attributes include the set name and access rights, as well as individual headings and their description and level attributes.

1. Do one of the following:

- To add a Heading Set, right-click the HeadingSets object and then click **New**.
 - To edit a Heading Set, double-click the Heading Set in the tree control.
2. In the **Set Name** list, select an available Heading Set to edit.
3. If needed, set user access permissions for the heading set. (See [Set security attributes](#).)
4. Select and edit headings in the Heading panel as follows:
- To add a heading, edit the first empty row or click the + button.
 - Click in a cell to edit its contents.
 - Click column headings to resize or sort the list of headings.
 - Click the - button to delete the selected heading.



5. Click **OK**.

Delete a heading set

To delete a Heading Set, do one of the following:

- Right-click the Heading Set and select **Delete**.
- Select the Heading Set and press Delete.

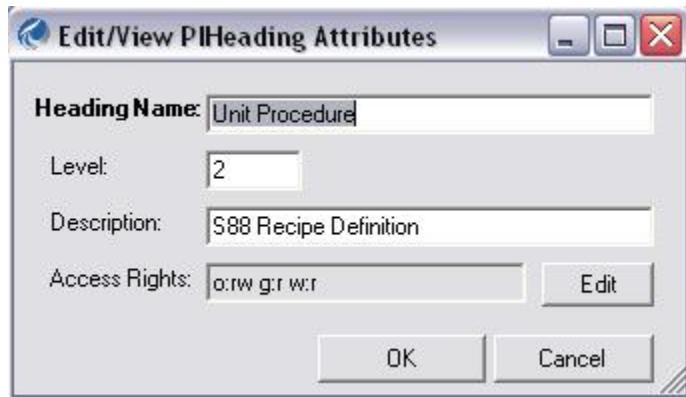
Add or edit heading attributes

A PIHeading is usually referenced by a module or a sub-batch in the Batch Database. For example, a heading may be used to name a type of sub-batch, such as a Phase, or a position in a plant layout, such as a Line.

Heading attributes include the heading level and name, which must be unique within a heading set, as well as the heading level, description, and access rights.

1. Do one of the following:

- To add a heading, right-click the heading set it will belong to and then click **New**.
- To edit a heading, double-click the heading, or right-click the heading and then click **Edit**.



Note: You can also add or edit headings while editing a heading set. See [Add or edit heading set attributes](#) for details.

2. Use the Edit/View PIHeading Attributes dialog box to update the heading.

- **Heading Name**

Enter or edit the name of the heading node.

- **Levels**

Specify the hierarchical level of the heading node.

- **Description**

Enter a contextual description for the heading.

- **Access Rights**

Click **Edit** to update user access permissions to the heading. See [Add or edit heading set attributes](#) for details.

3. Click **OK**.

Network Manager Statistics

Use the Network Manager Statistics tool to view and export statistical information about PI applications and services on each connected Data Archive server.

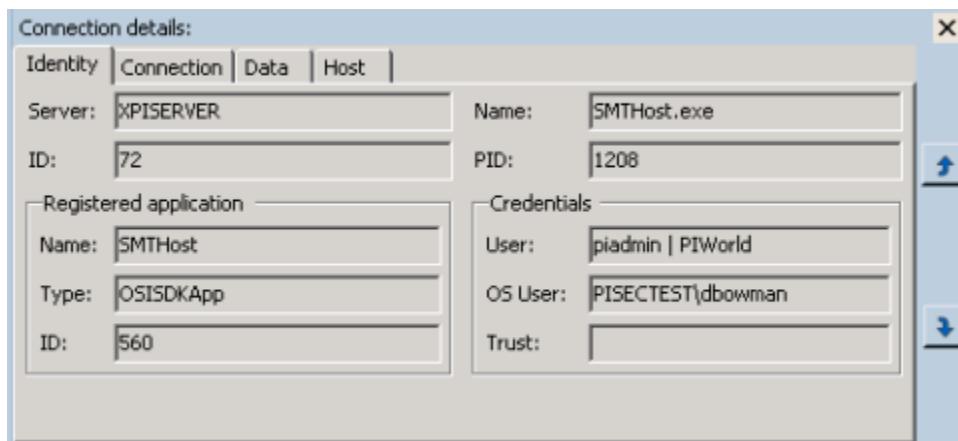
The available statistics are generated by the PI Network Manager Subsystem, which monitors the connections of computers, and are stored in PI Server Network Manager Statistics Table. The statistical columns listed by the Network Manager Statistics tool depend on the connected Data Archive server version and operating system.

View connection details

The Network Manager Statistics tool allows you to view connection details in a tabbed pane or within the Network Manager Statistics list. Fields in the Connection details pane contain values that correspond to information that is available from the parent Data Archive server. If a field is blank, that information is not available from the server.

1. Under Collectives and Servers, select a server.
2. Select an item in the list to view statistics in the Connection details pane.

To open or close the Connection details pane, click .



Statistics in PI SMT

The following statistics may appear in the **Network Manager Statistics** list depending on the version numbers of connected servers and the availability of the information from the parent Data Archive server:

Field	Description
ID	Connection ID; this is the primary key
APICount	Number of active API connections

Field	Description
BytesRecv	Bytes received by the connection
BytesSent	Bytes sent by the connection
ConStatus	Connection status
ConTime	Time connection was established
ElapsedTime	The number of seconds that the connection was active
IsStandAlone	True if the server is in standalone mode; False if not
IsTCPListenerOpen	True if the server is accepting requests through TCP, False if not
LastCall	The timestamp for the time that the connection made its last call to the server
MsgRec	The number of messages received by the connection
MsgSent	The number of messages sent by the connection
Name	Connection name
NetType	Connection network type; WIN32 named pipes, UNIX, or TCP/IP
NumConnections	The number of SDK and API connections to the Network Manager
OSBuild	Operating system build
OSSysName	Operating system name
OSUser	Operating system user
OSVersion	Operating system version
PeerAddress	IP Address of connecting machine
PeerName	Host name of connecting machine; available in versions earlier than PI Server 2010 (3.4.380 and earlier)
PeerPort	The port being used by the remote process for the connection to the PI Network Manager; this information is especially useful when troubleshooting firewalls and network routing

Field	Description
PID	The ID of the process that made the connection; this applies to all entries except pinetmgr
PINetMgrVersion	Version of PI Network Manager
PIPPath	Data Archive directory on the server; this item is the same for all connections
ProtocolVersion	PI Protocol version of connecting application
RecvErrors	Number of receive errors on the connection
RegAppID	The public ID for the connecting application
RegAppName	Registered Application name
RegAppType	Registered Application type
SDKCount	Number of active SDK connections
SendErrors	Number of send errors on the connection
Server	Name of the connected Data Archive server
Trust	Name of the trust in use by the connection
User	Identities or PI users with which the connection is logged on

Set display refresh rate

You can select an option to automatically refresh the Network Manager Statistics display. The connection details will refresh at the rate set in the Options dialog box. The toolbar also includes an option to maintain the data display, choose **Refresh Display Automatically** to continuously refresh the display. The default refresh rate is 5 seconds.

1. Click the **Options** button  on the toolbar.
2. Change the value in the **Refresh rate** field. You can set the rate from **1 second** to **60 seconds**.

Set Network Manager Statistics list display

You can populate the Network Manager Statistics list with PI Services or Data Archive Applications only. The default setting populates the list with connection details for both PI Services and Data Archive Applications.

To change the contents of the Network Manager Statistics list:

1. Click the **Options** button  on the toolbar.
2. Select or de-select:
 - **Display PI Services**
 - **Display PI Server Applications**

Export statistics to a file

To export the **Network Manager Statistics** list to a CSV file:

1. Right-click on the list and select **Export**.
2. Select the location to save the **Network Manager Statistics** list.
3. Click **Save**.

Performance Counters tool

Use the Performance Counters tool to build PI points for the PI Performance Monitor (PerfMon) Interface.

Performance counters can provide important insights into a number of performance management problems, such as memory, disk, and process management. Use the Performance Counters tool to select performance counters, associate them with PI points, and then build and configure or edit those PI points on a Data Archive server.

Select a Performance Monitor interface

To change point attributes other than *Point Source*, *Program instance*, and *Scan class number* for Performance Monitor points, see [Edit point attributes](#).

1. Click the Tag Settings tab.
2. Select the PI Performance Monitor interface for which you are building points.
3. Enter or update the **PI Interface Identification** information.

If you have configured your PerfMon Interface with the PI Interface Configuration Utility (ICU), you should see the interface and its *Point Source*, *Program instance*, and *Scan Class number* attributes.

If you have not used the ICU to configure the Performance Monitor Interface, manually enter the configuration information for *Point Source*, *Program instance*, and *Scan Class number*.

Attribute	Description	Default
<i>Point Source</i>	Because Data Archive stores data collected from a variety of sources, the Point Source field distinguishes where the data arrived from. It is a one-character value; for example, \$. This character must match the one specified in the PI Performance Monitor's startup command file.	#
<i>Program instance</i> [Location 1]	You can run multiple copies of the full version of PI Performance Monitor on the same machine. The Program instance field associates a tag with a particular copy of PI Performance Monitor. Program Instance is a positive integer. It should match the id= parameter used in the interface startup command file.	1

Attribute	Description	Default
<i>Scan Class number</i> <i>[Location 4]</i>	Corresponds to a scan class set that is created at interface startup.	1

4. Optional: Enter a prefix for the point names in the **Point Name Prefix** text box.

This option allows you to add a prefix string to all point names that you create. If you change the Point Name Prefix while points are in the **Build Tags** tab, the tag names of those points change accordingly.

Note: When an existing server is selected, you cannot change the point source or interface instance, and the Performance Counters tool selects a scan rate from the available scan rates configured for the interface.

Select counters to build points and view details

1. Click the **Build Tags** tab.
2. Select the performance counters to build points for the Performance Monitor Interface.
3. Click **Apply Counters to Instances**.
4. View details about the selected counters in the active pane to the right.

Build?	Use to select or deselect tags based on the selected performance counter.
Point Name	Name of the point to be built.
Performance Counter	Performance Counter path of the counter to be monitored.
Value	Current value of the Performance Counter.
Type	Data type of the counter .

To perform a variety of tasks related to clearing, checking, and highlighting entries, right-click a performance counter in the **Tag Preview** pane and select the appropriate task from the drop-down list.

Build Performance Monitor points on the Data Archive server

1. On the **Build Tags** tab in the **Build Tags** section, select **Create tags on PI Server**.
2. Open the drop-down menu to view and select an available server.
3. Click **Create Tags**.

Note: If a point with the same tag name already exists on the server, an alert appears. Click **Edit** to choose a different tag name or click **Skip** to choose not to create the point.

Use the **PI Builder** application to build Performance Monitor points if you want to perform complicated tag manipulations, such as renaming or making other point attribute changes.

Build Performance Monitor points with PI Builder

1. On the **Build Tags** tab, in the **Build Tags** section, select **Write tags to CSV file**.
2. Assign a filename and location for the file.
3. Import this file into MS Excel, adjust as necessary, and export to the Data Archive server using PI Builder.

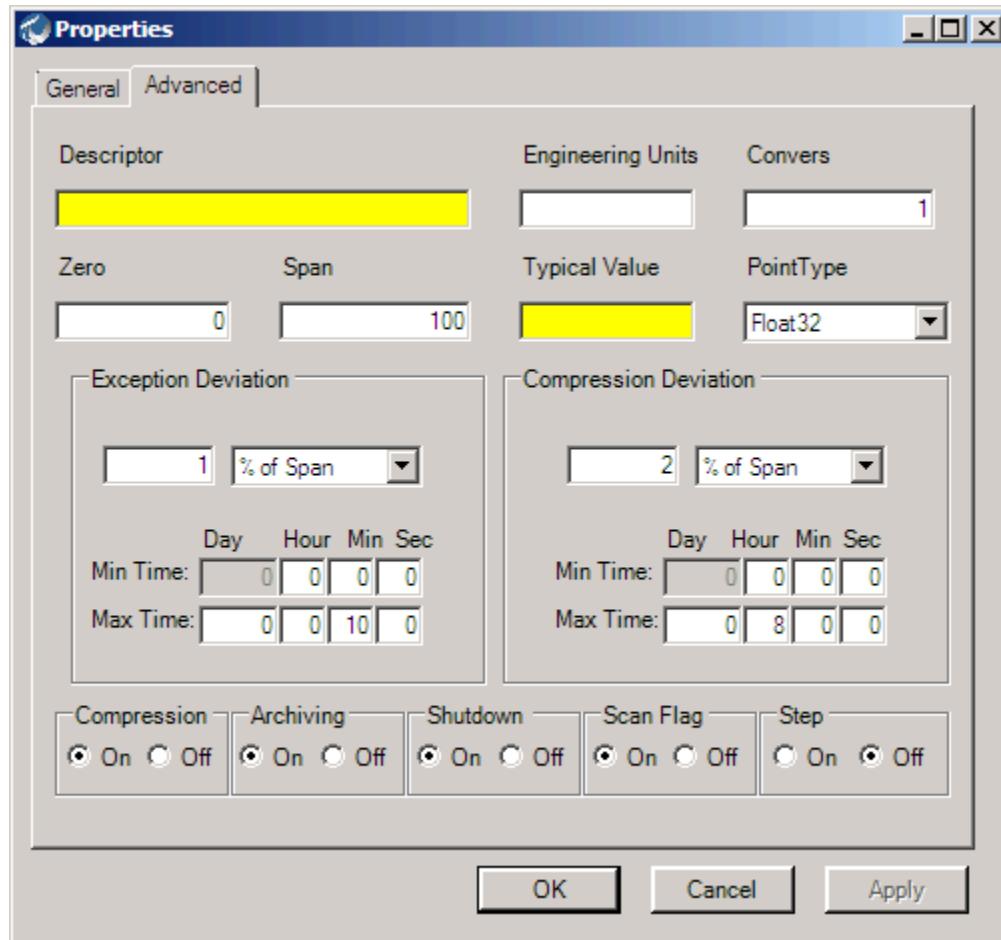
Note: For issues with tag names, see [Long tag names](#).

Edit point attributes

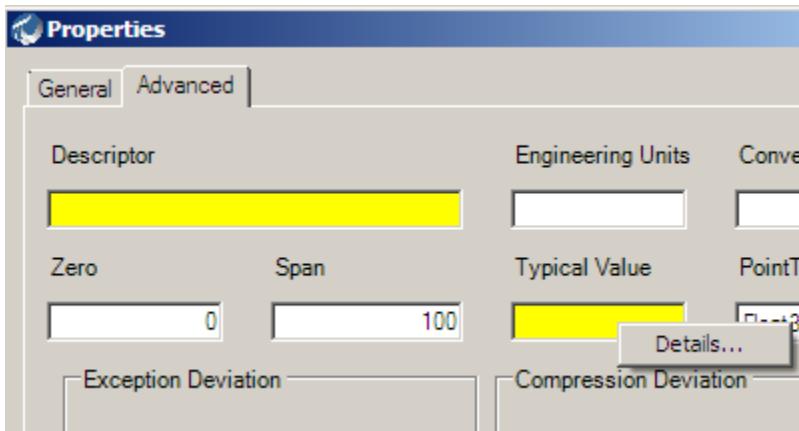
To change point attributes other than Interface ID and Point Source (Locations 1 and 4) for Performance Monitor points:

1. Select the point or points you want to edit in the pane on the right side of the **Build Tags** tab, and then right-click and select **Properties**.
All points, except string and digital state points, are displayed on the **General** tab in the Properties dialog box.
2. Click the **Advanced** tab to view all of the point attributes.

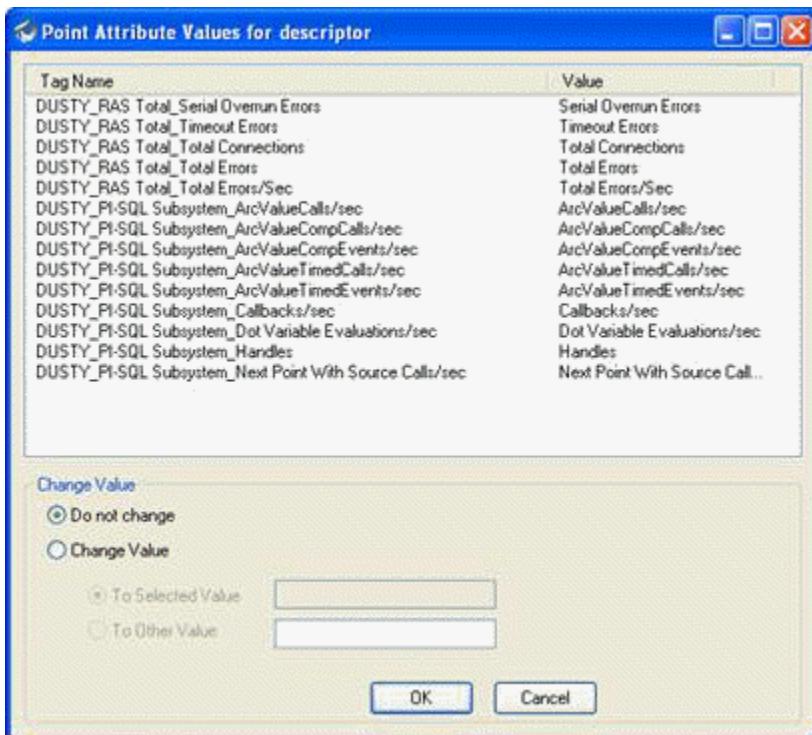
The point attributes that are different among the selected tags are highlighted.



3. To see the value of a selected point attribute for each of the points, right-click the attribute, such as **Typical Value**, and select **Details**.



4. Optional: To change the values displayed in the Point Attribute Values for <attribute> dialog box, select **Change Value** and change the values in one of the following ways:



- Select **To Selected Value**. Select one of the attribute values displayed. The selected value is applied to all the tags in the list.
- Enter a new attribute value in **To Other Value**. The value you entered is applied to all the tags in the list.

Note: To leave the values unchanged, select **Do not change**.

5. Click **OK** to close the dialog box and enter your change into the **Advanced** tab of the Properties dialog box.
6. Click **Apply** or **OK** in the Properties dialog box to propagate the change to the selected points.

Long tag names

Because the Performance Counters tool builds points automatically, it uses a rigid naming scheme, which sometimes yields long tag names. These tag names may be problematic for some versions of the Performance Monitor Interface and some client tools. These tag length limitations include:

- PI DataLink 2.x (80)
- PI Performance Monitor Interface versions before 1.2.0.5 (80)
- PI Performance Equations (73). This is the tag length limit for tags on which event-based processing is based.

If necessary, you can use the Performance Counters tool to rename the Performance Monitor points.

Rename Performance Monitor points

1. Select the points in the **Tag Preview** pane and select **Rename Highlighted Tag Names**.
2. Make changes in the Rename Tags dialog box.

Create a Performance Monitor points template

You can either load an existing template or create your own. To create a Performance Monitor points template:

1. Click the **Create** button in the Template Management pane.
2. Select the Performance Counter you want to use as a basis for the template in the **Template Builder**.
3. Use the Template Builder to configure these point attributes:
 - **Descriptor**
The descriptor field is optional.
 - **Engineering Units (Eng Units)**
Optional engineering units to describe the default unit of measurement for the point.
 - **Typical value**
Enter a reasonable sample value for the point. For a numeric tag, a typical value must be greater than or equal to the Zero, and less than or equal to the Zero plus the Span.
 - **Zero**
Enter the lowest possible value for the point, as the bottom of the range used for scaling float16 values and trends. Recorded values lower than the Zero value are recorded in Data Archive with the digital state Under Range.
A Zero value is required for numeric data type points, but not for non-numeric points. The instrument zero is a logical zero value, and certain interfaces require that Zero and Span values match the instrument system range. See your interface documentation for details.
 - **Span**
Enter the maximum difference between the top and bottom of the point range, as a positive value. The Span value is used for scaling float16 values and trends and is required only for numeric data type points. Recorded values above the sum of the zero and Span values are recorded in Data Archive with the digital

state Over Range.

- **Scan**

Enable Scan if an interface that supplies data to the point requires a scan flag. Interfaces that require a scan flag do not update points when the flag is disabled. See your interface documentation for Scan attribute requirements.

- **Archiving**

Enable Archiving to store point values in Data Archive.

- **Compression**

See [Compression deviation](#).

- **Exception**

See [Exception deviation](#).

- **Digital Set**

A digital set on the Data Archive server. This field is active only if Digital Point? is selected.

- **Digital States**

A digital state on the Data Archive server. This field is active only if Digital Point? is selected.

- **Display Digits**

An optional number of display digits to determine the display precision of point values.

- **Convers**

- **Step**

See [Step](#).

- **Shutdown**

See [Shutdown](#).

4. Use one of two options to save your changes:

- To save changes to the template and apply the changes to the selected points, click **Save**.
- To apply changes to the selected points, but not save the changes to the template, click **Apply**.

Deviation

Exception and compression deviation algorithms can be used to filter erroneous and redundant data before it is recorded in the archive.

Note: By default, future PI points have *excmin*, *excmax* and *excdev* turned off (set to 0).

Each algorithm is based on three specifications:

- Deviation in point value
- Minimum duration of time
- Maximum duration of time
- These specification are used to determine which points are recorded to the snapshot from the buffer (Exception) and to the archive from the snapshot (Compression).

Note: Exception deviation and compression deviation filter externally-generated events for archiving. Under no circumstances does this cause Data Archive to generate events.

Compression deviation

Once events are sent to the PI Snapshot subsystem, a compression algorithm can further filter data and reduce storage to only significant values as they are moved into the archive. An event is recorded:

- After a specified minimum duration of time since the previous event, if it exceeds a specified deviation in value from that event.
- After a specified maximum duration of time since the previous event.
- When activated, compression reporting filters events and stores only periodic values (including duplicates), unless an event represents a significant change in the short-term trend of values.

To turn off compression and archive every event that passes exception reporting, disable the *Compressing* attribute.

Compression Deviation — Enter the deviation in value required to record an event, either as a number of engineering units, or as a percentage of the point's *Span* value.

For most flows, pressures, and levels, use a deviation specification of 1 or 2 percent of *Span*. For temperatures, the deviation should usually be 1 or 2 degrees.

Min Time — Enter the minimum time that must elapse after an event before a compressed value can be recorded. The minimum time should be set to 0 if exception reporting is activated for the same point.

Max Time — Enter the maximum time that can elapse after an event before automatically recording the next event as a compressed value. The recommended maximum time is one work shift (for example, 8 hours). If this value is too low, the compression effects are too limited to save significant archive space. If this value is too high, useful data may be lost. Events that reach the Data Archive server in asynchronous order bypass the compression calculation and are automatically recorded to the archive.

The compression specifications consist of a deviation (*CompDev*), a minimum time (*CompMin*), and a maximum time (*CompMax*).

Events are also archived if the elapsed time is greater than the maximum time. Duplicate values will be archived if the elapsed time exceeds *CompMax*. Under no circumstances does this cause Data Archive to generate events; it only filters events that are externally generated.

The most important compression specification is the deviation, *CompDev*. For non-numeric tags, *CompDev* and *CompDevPercent* are ignored. They will be displayed by applications as zero.

Note: *CompDev* specifies the compression deviation in engineering units; *CompDevPercent* specifies the compression deviation in percent of *Span*. If you change one, the other is automatically changed to be compatible. If you try to change both at once, *CompDevPercent* takes precedence.

Exception deviation

Exception reporting is used to define the precision of a data stream, and the amount of deviation that constitutes a significant change. Most interface programs can execute an exception-reporting algorithm to determine when to send a point value to the PI Snapshot subsystem. An exception is an event that occurs either:

- After a specified minimum duration of time since the previous event, while exceeding a specified deviation in value from that event.
- After a specified maximum duration of time since the previous event.

This means that when activated, exception reporting filters events and stores only periodic values, including duplicates, unless an event represents a significant change in the short-term trend of values. An exception event, both timestamp and value, is sent with the previous event to the Snapshot.

- **Exception Deviation**

The deviation in value required to store an event, either as a number of engineering units, or as a percentage of the point's Span value. The exception deviation should be less than the compression deviation by at least a factor of 2, and is ignored for digital, string and BLOB data type points.

- **Min Time**

The minimum time that must elapse after an event before an exception value can be stored.

- **Max Time**

The maximum time that can elapse after an event before automatically storing the next event as an exception value.

Set the minimum and maximum time values to *0* to turn off exception reporting.

Step

Enable the Step attribute to treat archived point values as discrete units that would appear stepped on a chart. An archived value is assumed constant until the next archived value, and adjacent archived values are not interpolated. For example:

- at 12:00:00, the value 101.0 is archived
- at 12:01:00, the value 102.0 is archived
- a request for the value at 12:00:30 returns 101.0

Disable Step to treat archived point values as a continuous signal, and linearly interpolate adjacent values. For example:

- at 12:00:00 the value 101.0 is archived
- at 12:01:00 the value 102.0 is archived
- a request for the value at 12:00:30 returns 101.5

Note: By default, future PI points have their step attribute turned on (set to 1). This is because most future PI Points store discontinuous signals (for example, a series of discrete predictions) for which linear interpolation cannot be assumed. With the step attribute turned on, trends show a staircase pattern. Future PI points, by default, also have compression, excmin, excmax and excdev turned off (set to 0).

The Step attribute also affects compression calculations. When enabled, a linear change of value greater or equal to the compression deviation allows the point to pass compression.

The Step attribute is relevant only to numeric points. In general, data from continuous signals such as thermocouples and flow meters should be archived without stepping. Data from discrete measurements such as lab data and batch charge weight should be stepped.

To access the Step attribute setting:

1. **Open PI SMT.**

2. **Navigate** to Points > Point Builder.
3. **Search** for and **select** the point for which you would like to check the step attribute.
4. In the Archive tab, you can **view/change** the step attribute.

Shutdown

Use the Shutdown attribute to determine whether shutdown events are written. The timestamp of Shutdown events normally represents the actual shutdown time of the Data Archive server as recorded by the Snapshot Subsystem.

Beginning with PI Server PR1 SP1, shutdown events for most points are disabled. Unless you configure points to receive shutdown events, only test points such as sinusoid and sinusoidu will receive shutdown events.

- Enable Shutdown to record server shutdown events as point values with the timestamp of the server shutdown. If data is collected on the Data Archive server, shutdown events are helpful to clearly indicate gaps in data collection.
- Disable Shutdown when point data is collected on a distributed collection nodes, as buffering services collect, manage and retain the data until the server is up and running again. Disabled Shutdown PI points have a configurable attribute to determine whether shutdown events are written.

Note: The default behavior of Data Archive is to write the SHUTDOWN digital state to all PI points when Data Archive is started. The timestamp that is used for the SHUTDOWN events is usually updated every 15 minutes, which means that the timestamp for the SHUTDOWN events will be accurate to within 15 minutes in the event of a power failure.

Compression

Enable compression to apply compression algorithms and save only event values that indicate a change in point value. When disabled, all values collected and sent to the snapshot, including redundant values, are saved in the PI archive.

Note: By default, future PI points have *compression* turned off (set to 0).

Enable compression for all real-time points. Digital, BLOB and string data type values pass through compression only when the value changes. Compression is typically turned off for points that collect sample data, such as lab data, or other sparse data streams.

Load a Performance Counters template

1. Click the **Load** button in the **Template Management** pane.
2. Navigate to the folder that contains the Performance Monitor points template.
3. Select the template you want to load and select **Open**.
4. Select **Apply** to apply the template in the **Current Template(s)** field to the points selected in the active pane of the **Build Tags** tab.

To remove the selected template from the **Template Management** pane, click the **Unload** button.

Replace updated tag names

Use the Lookup Table to replace tag names that have been updated. This table stores strings for the New tag names listed and correlates them with the original tag names.

- Every time you build a point the Lookup Table or Substitution Table is scanned.
- If any of the strings in the lookup table exist in the **Original Name** field for the point you are creating, they are replaced with the New Name for that string.
- To configure the entries of the table, click **See Table** on the **Tag Settings** tab.
- If you do not want to use the Lookup Table, uncheck the box **Use tag name substitution table?** on the **Tag Settings** tab.

Performance Equations

Use the Performance Equations tool to create and maintain PI Performance Equation (PE) calculations. These calculations allow you to use PI System data to program complex calculations such as the sum of two flows, the difference between two temperatures, and the simple mass balance on a process unit.

PE calculations can be used in other calculations, graphed in trend displays, and included in reports. These real-time calculations are generated by the Performance Equations Subsystem.

You can also use the Performance Equations tool to build and maintain the PI points you use to store data from the results of PE calculations. PI points that use PE calculations are known as *PE points*.

Note: Future PI points cannot be used as output, input, or trigger tags in Performance Equations.

Viewing performance equations

Use the Performance Equations tool to retrieve a list of performance equations and view information about:

- The Data Archive server that stores the Performance Equation (PE)
- The point source for the PI point that uses the PE
- The name of the PI point that uses the PE
- the PE point calculation logic, shown in the **Extended Descriptor** field

Note: By default, Data Archive assigns *C* as the **Point Source** for PE Points.

Server	PS	Point	Extended Descriptor
cosen	C	94_bsw	Options["exp:100, (10^94 gridx,1)+5"] sq:100-150 2*func("sinusoid",0.6/10)
cosen	C	94_GRDIDX1	TimeEq("PE_Source","10n","1") if badval("BaGen_Campaign,1") then 0 else 'BaGen_Campaign,1' + 1
cosen	C	a PE tag	if badval("BaGen_PIBatchIndex,1") then 0 else BaGen_PIBatchIndex,1 + 1
cosen	C	BaGen_Campaign,1	(Beer_Pdrn/Beer_Pdrn_Plan) (Beer_Production/Brew_Prod_Plan) (Cleveland_Prof) test('1') sinusoid
cosen	C	BaGen_PIBatchIndex,1	if badval(DailyBatchStep) then 0 else 'DailyBatchStep' + 1
cosen	C	Beer_Pdrn_vs_Plan	(Dist_Product_Inv/Dist_Product_Inv_Plan) (Houston_Prof)
cosen	C	Brew_Prod_vs_Plan	event = manual_input_int32; if manual_input_int32=0 then prevval(manual_input_int32,"1") NextVal("manual","30z")
cosen	C	Cleveland_vs_Prof	if '1401' > 180 or ('PhaseDischargeR401' = "Active" AND '1401.Active' = "Active") then 3 (1401.active="Inactive" then 0 else 1) if '1402' > 180 or ('PhaseDischargeR402' = "Active" AND '1402.Active' = "Active") then 3 (1402.active="Inactive" then 0 else 1) if '1501' > 180 or ('PhaseDischargeR501' = "Active" AND '1501.Active' = "Active") then 3 (1501.active="Inactive" then 0 else 1)
cosen	C	DailyBatchStep	TimeEq("PTC.TLB7680F_PV","1H","1") if '5150.U400.HS>CV4092' then digitext('5150.U400.HS>CV4092') else digitext('5150.U400.HS>CV4092')
cosen	C	Dist_Product_Inv_vs_Plan	event=DailyBatchStep, day("") daysec("")
cosen	C	Houston_vs_Prof	
cosen	C	Maximum_manual_input_int32_pe	
cosen	C	maxvaltest	
cosen	C	OperationR401	
cosen	C	OperationR402	
cosen	C	OperationR501	
cosen	C	P7C-TLB7680F_1H_PsAlarm.PV	
cosen	C	PE Test ?	
cosen	C	PE_Day	
cosen	C	pe_Daysec	

You can select and edit PE points in this list.

Create a PE point

1. Select the Performance Equations tool.
2. Enter point attributes on the **General** tab. See [Enter general point attributes](#).
3. Define a PE calculation on the **Calculation** tab. See [Define a PE calculation](#)
4. Schedule the PE calculation on the **Scheduling** tab. See [Schedule a PE calculation](#).

Enter general point attributes

Note: If you have used the Interface Configuration Utility (ICU) to configure the interfaces that use the Data Archive you are using, the Performance Equations tool will populate the **Point source** field. By default, this value is C. You can change the point source, provided the PE points you use are not dependent on PE points with Point source C. The **Point class** field is also populated. All Performance Equations use the classic point class.

Edit the following fields in the **General** tab of a PE point:

- **Name**

Enter a point name.

Note: The name text appears blue if the point does not exist on the selected Data Archive server. The text appears black if the point exists, or a server is not selected, preventing validation.

- **PI Server**

Select a Data Archive from the list of connected servers in the **Collectives and Servers** pane.

- **Descriptor**

Enter an optional description of the point to describe the underlying intention or performance equation used.

- **Point Class**

For PE points, the point class is a read-only field set to **classic**.

- **Point Source**

Select the correct point source.

The point source should correspond to the Performance Equation Scheduler instance that the point belongs to.

Multiple instances of the PE Scheduler can be created, registered with PI Interface Configuration Utility (PI ICU), and read from the Module Database. If no instances are registered with PI ICU, the default point source used by the Scheduler is C.

- **Point Type**

Select the point type for the point. A Performance Equation can use Point Types Digital, Float64, Float32, Int32, Int16 and String.

- **Digital Set**

Choose a digital set from the Data Archive. This menu is activated only if the selected Point Type is Digital. The System digital set is otherwise used.

- **Engineering Units (Eng Units)**

Optional: Enter engineering units to describe the default unit of measurement for the point.

- **Display Digits**

Enter an optional number of display digits to determine the display precision of point values.

- **Extended Descriptor (Exdesc)**

For PE points, this field is read-only and displays the PE point calculation logic.

The screenshot shows the PI System Management Tools interface for defining a Performance Equation (PE). The 'General' tab is active. The 'Name' field is set to '2timesSinusoid'. The 'Descriptor' field contains the text 'Multiply sinusoid times two'. The 'Point class' is set to 'classic'. The 'Point type' is 'Float32'. The 'Point source' is 'C'. The 'Eng Units' field is empty. The 'Display digits' are set to '-5'. The 'Exdesc' field contains the equation 'Event=sinusoid, 'sinusoid' * 2'.

Define a PE calculation

When you define a PE calculation, use the single- and double-quotes as indicated. For a complete guide to PE syntax, see the Data Archive Applications User Guide, available at the [Customer Portal](#).

1. To define the PE calculation that the PI Point will use,
2. Under **Collectives and Servers**, select the server or collective.
3. In the **System Management Tools** pane, select **Points > Performance Equations**.
4. Click the **Equation** tab.
5. Fill in the **Equation** fields.
6. Schedule the PE calculation. See [Schedule a PE calculation](#).

Equation fields

- **Event Tag**

For event-based calculations, the event tag appears in the read-only **Event tag** field.

- **Equation**

Enter an equation using PI Performance Equation syntax:

Enter point names in single quotes. For example, to simply add the current value of the sinusoid point to the current value of the ba:level.1 point:

Equation: `'sinusoid' + 'ba:level1'`

Enter timestamps in single quotes. To calculate the total time during the past hour where the sinusoid point had a value between 30 and 70:

Equation: `(timegt('sinusoid', "-1h", "30)) - timegt('sinusoid', "-1h", "70")`

To calculate the time-weighted average value of the **sinusoid** point for the last hour:

Equation: `tagavg('sinusoid', "-1h", "")`

Calculation expressions longer than 300 characters are difficult to work with. If longer expressions are required, consider breaking down an equation into parts handled by individual PE points, and then use those points in equations for other points.

Note: It is important that you use scan class offsets to schedule point calculations in the correct order.

- **Evaluate**

Click **Evaluate** to test an equation. If evaluation is disabled, be sure a Data Archive server is selected on the **General** tab.

- **Snapshot**

Click **Evaluate** to test an equation. If evaluation is disabled, be sure a Data Archive server is selected on the **General** tab.

See also:

- [Event-based scheduling](#)
- [Schedule a PE calculation](#)
- [Scan class offsets for performance equations](#)

Schedule a PE calculation

Two types of scheduling are available to calculate PE point values.

Event-based scheduling

Event-based scheduling triggers calculations when a specified point is updated with a new event value (based on snapshot events).

- **Event Tag**

Enter the name of a PI point to act as the trigger tag for calculation, or click  to search for tags on the Data Archive server. The event tag must be selected from the same server as the PE point being created.

Whenever the event tag receives a new value, the PE subsystem performs a calculation and updates the value of the PE point. For example, you might want a calculation performed whenever the point sinusoid gets a new value.

- **Set timestamp of resulting expression to**

Select one of two options to determine the source of the timestamp associated with each calculated PE value. In both cases the expression is computed based on current snapshot values:

- Set the PE point timestamp to the time at which the expression is evaluated.
- Set the PE point timestamp to match the time of the triggering event.

Clock scheduling

Clock scheduling uses scan classes to calculate values at regular intervals, such as every hour, five seconds, or 20 minutes.

- **Scan Class**

Choose a scan class to schedule calculations based on the scan frequency of an available scan class.

Note: All available scan classes are listed in the **Scan Class** menu. If the menu does not appear, make sure the **pipeschd.bat** file is registered in the PI Interface Configuration Utility (ICU).

- **Start Time field**

Specify an optional initial start time and offset for the calculation interval. For equations that use more than one point, use scan class offsets to perform calculations in a specific order.

Scan class and interface configuration

A scan class is a code that PI interfaces use to schedule data collection. You set the scan class in PI ICU when you configure an interface.

Scan classes have the following components:

Component	Description	Optional	Example
Period (Scan Frequency)	Specifies how often the interface collects data.	No	01:00:00 Get data every hour.
Offset	Specifies a start time for the calculation. Data Archive interprets the value starting from midnight of the current day.	Yes	01:00:00,13:00:00 Get data every hour, starting at 1:00PM.
UTC Time	Requires that the scheduling is synchronized with UTC. To use it, add ",U" after the scan class. UTC scan classes are not affected by daylight saving time because the scan class scheduling synchronizes with UTC* rather than local time. UTC Time has no effect if the scan-class period is 1 hour or less.	Yes, but recommended	02:00:00,13:00:00,U Get data every two hours, starting at 1:00PM UTC time.
Local Time	Specifies that: <ul style="list-style-type: none"> During a transition from daylight saving time to standard time, the scan-class period is 24 hours. 	Yes; using forces Wall Clock Scheduling	23:00:00,08:00:00,L During a transition from daylight saving time to standard time, get data after 24 hours. During a transition from standard time to daylight saving

Component	Description	Optional	Example
	<ul style="list-style-type: none"> During a transition from standard time to daylight saving time, the scan-class period is 22 hours. <p>To specify Local Time, add ",L" after the scan class. This setting has no effect when the scan-class period is 1 hour or less.</p>		time, get data after 22 hours.

Note: *Scheduling is synchronized with UTC means that the scan occurs based on the count of UTC seconds. To clarify, if you implement the scan class 02:00:00,13:00:00,U what will happen is at 1 PM clock time of your PI Server time the first scan will occur. It will then scan every 2 hours according to UTC seconds passed, not the difference between the last scan time and the current clock time.

Here is an example scan class:



This scan class uses the *hh:mm:ss* format and specifies a period, offset, and UTC time.

When you specify a scan class in the ICU, you can use any of the following formats:

```
ss
ss,ss
hh:mm:ss
hh:mm:ss, hh:mm:ss
hh:mm:ss, hh:mm:ss,t
```

where *hh* is hours, *ss* is seconds, *mm* is minutes, and *t* is either a U (for UTC Time) or an L (for Local Time).

Note: In PI ICU, scan class specifications cannot contain spaces.

Scan class offsets for performance equations

Performance equations (PE points) are calculated based on input data from PI tags. When multiple PE points share the same scan class, they are processed together at scheduled intervals. To ensure accurate results, dependent PE points must be calculated in a specific order. A scan class offset specifies a delay, allowing dependent equations to use the most up-to-date input data.

Why use scan class offsets?

Scan class offsets are critical when performance equations involve dependencies or complex mathematical

operations, such as multiplication and division. These operations often rely on precise and updated inputs. By staggering calculations with offsets, you can ensure accurate sequencing and prevent outdated data from being used.

Scenario: Dependent performance equations with complex operations

Consider three PE points (A, B, and C) that depend on each other and involve multiplication and division:

- **Point A** calculates the value of a PI tag called Sinusoid.
Formula: `Sin(TagVal('sinusoid'))`.
- **Point B** calculates the square of Sinusoid multiplied by a constant.
Formula: `Sqrt(TagVal('sinusoid') * 5)`.
- **Point C** combines the results of Points A and B in a more complex equation.
Formula: `(TagVal('A') * TagVal('B')) / (TagVal('B') + 1)`.

Since all three PE points share the same scan class, their calculations are processed together. However, Point C depends on the results of both Points A and B, so it must wait until these are updated. A scan class offset ensures the calculations occur in the correct sequence.

Example: Configuring scan class offsets for advanced operations

The following table demonstrates how to configure PE points with scan class offsets for these complex equations.

Tag	ExDesc	Scan class
A	<code>Sin(TagVal('sinusoid'))</code>	/f=01:00:00,01:00:00
B	<code>Sqrt(TagVal('sinusoid') * 5)</code>	/f=01:00:00,01:00:00
C	<code>(TagVal('A') * TagVal('B')) / (TagVal('B') + 1)</code>	/f=01:00:00,01:00:01

How it works

1. Point A and Point B are calculated together every hour on the hour.
2. Point C is calculated one second later, ensuring it uses the updated values of Points A and B.

By applying a one-second offset to Point C, you ensure its calculation reflects the results of the more advanced operations in Points A and B.

Key considerations for complex equations

- **Layered Dependencies:** For equations with multiple layers of dependency (for example, Point D depends on Point C), apply additional offsets to maintain sequencing. For instance, Point D could use a `/f=01:00:00,01:00:02` offset.
- **Testing:** Always validate your configuration in a controlled environment to confirm that calculations occur in the correct sequence.

the intended sequence.

- **System Performance:** More complex equations may require additional processing power. Monitor system resources to ensure calculations are completed within the configured intervals.

Set PE point archive options

Use the **Archive** tab to configure attributes that determine how PE point data is stored in the Data Archive server.

- **Typical Value**

Enter a reasonable sample value for the point. For a numeric tag, a typical value must be greater than or equal to the **Zero**, and less than or equal to the **Zero** plus the **Span**.

- **Zero**

Enter the lowest possible value for the point, as the bottom of the range used for scaling float16 values and trends. Recorded values lower than the **Zero** value are recorded in the Data Archive server with the digital state Under Range.

A **Zero** value is required for numeric data type points, but not for non-numeric points. The instrument zero is a logical zero value, and certain interfaces require that **Zero** and **Span** values match the instrument system range. See your interface documentation for details.

- **Span**

Enter the maximum difference between the top and bottom of the point range, as a positive value. The **Span** value is used for scaling float16 values and trends and is required only for numeric data type points.

Recorded values above the sum of the **Zero** and **Span** values are recorded in the Data Archive server with the digital state Over Range.

- **Scan**

Enable **Scan** if an interface that supplies data to the point requires a scan flag. Interfaces that require a scan flag do not update points when the flag is disabled. See your interface documentation for Scan attribute requirements.

- **Archiving**

Enable Archiving to store point values in the Data Archive server.

Step

Enable the Step attribute to treat archived point values as discrete units that would appear stepped on a chart. An archived value is assumed constant until the next archived value, and adjacent archived values are not interpolated. For example:

- at 12:00:00, the value 101.0 is archived
- at 12:01:00, the value 102.0 is archived
- a request for the value at 12:00:30 returns 101.0

Disable Step to treat archived point values as a continuous signal, and linearly interpolate adjacent values. For example:

- at 12:00:00 the value 101.0 is archived

- at 12:01:00 the value 102.0 is archived
- a request for the value at 12:00:30 returns 101.5

Note: By default, future PI points have their step attribute turned on (set to 1). This is because most future PI Points store discontinuous signals (for example, a series of discrete predictions) for which linear interpolation cannot be assumed. With the step attribute turned on, trends show a staircase pattern. Future PI points, by default, also have compression, excmin, excmax and excdev turned off (set to 0).

The Step attribute also affects compression calculations. When enabled, a linear change of value greater or equal to the compression deviation allows the point to pass compression.

The Step attribute is relevant only to numeric points. In general, data from continuous signals such as thermocouples and flow meters should be archived without stepping. Data from discrete measurements such as lab data and batch charge weight should be stepped.

To access the Step attribute setting:

1. **Open** PI SMT.
2. **Navigate** to Points > Point Builder.
3. **Search** for and **select** the point for which you would like to check the step attribute.
4. In the Archive tab, you can **view/change** the step attribute.

Shutdown

Use the Shutdown attribute to determine whether shutdown events are written. The timestamp of Shutdown events normally represents the actual shutdown time of the Data Archive server as recorded by the Snapshot Subsystem.

Beginning with PI Server PR1 SP1, shutdown events for most points are disabled. Unless you configure points to receive shutdown events, only test points such as sinusoid and sinusoidu will receive shutdown events.

- Enable Shutdown to record server shutdown events as point values with the timestamp of the server shutdown. If data is collected on the Data Archive server, shutdown events are helpful to clearly indicate gaps in data collection.
- Disable Shutdown when point data is collected on a distributed collection nodes, as buffering services collect, manage and retain the data until the server is up and running again. Disabled Shutdown PI points have a configurable attribute to determine whether shutdown events are written.

Note: The default behavior of Data Archive is to write the SHUTDOWN digital state to all PI points when Data Archive is started. The timestamp that is used for the SHUTDOWN events is usually updated every 15 minutes, which means that the timestamp for the SHUTDOWN events will be accurate to within 15 minutes in the event of a power failure.

Compression

Enable compression to apply compression algorithms and save only event values that indicate a change in point value. When disabled, all values collected and sent to the snapshot, including redundant values, are saved in the PI archive.

Note: By default, future PI points have *compression* turned off (set to 0).

Enable compression for all real-time points. Digital, BLOB and string data type values pass through compression only when the value changes. Compression is typically turned off for points that collect sample data, such as lab data, or other sparse data streams.

Deviation

Exception and compression deviation algorithms can be used to filter erroneous and redundant data before it is recorded in the archive.

Note: By default, future PI points have *excmin*, *excmax* and *excdev* turned off (set to 0).

Each algorithm is based on three specifications:

- Deviation in point value
- Minimum duration of time
- Maximum duration of time
- These specification are used to determine which points are recorded to the snapshot from the buffer (Exception) and to the archive from the snapshot (Compression).

Note: Exception deviation and compression deviation filter externally-generated events for archiving. Under no circumstances does this cause Data Archive to generate events.

Compression deviation

Once events are sent to the PI Snapshot subsystem, a compression algorithm can further filter data and reduce storage to only significant values as they are moved into the archive. An event is recorded:

- After a specified minimum duration of time since the previous event, if it exceeds a specified deviation in value from that event.
- After a specified maximum duration of time since the previous event.
- When activated, compression reporting filters events and stores only periodic values (including duplicates), unless an event represents a significant change in the short-term trend of values.

To turn off compression and archive every event that passes exception reporting, disable the *Compressing* attribute.

Compression Deviation — Enter the deviation in value required to record an event, either as a number of engineering units, or as a percentage of the point's *Span* value.

For most flows, pressures, and levels, use a deviation specification of 1 or 2 percent of *Span*. For temperatures, the deviation should usually be 1 or 2 degrees.

Min Time — Enter the minimum time that must elapse after an event before a compressed value can be recorded. The minimum time should be set to 0 if exception reporting is activated for the same point.

Max Time — Enter the maximum time that can elapse after an event before automatically recording the next event as a compressed value. The recommended maximum time is one work shift (for example, 8 hours). If this value is too low, the compression effects are too limited to save significant archive space. If this value is too high, useful data may be lost. Events that reach the Data Archive server in asynchronous order bypass the compression calculation and are automatically recorded to the archive.

The compression specifications consist of a deviation (*CompDev*), a minimum time (*CompMin*), and a maximum

time (*CompMax*).

Events are also archived if the elapsed time is greater than the maximum time. Duplicate values will be archived if the elapsed time exceeds *CompMax*. Under no circumstances does this cause Data Archive to generate events; it only filters events that are externally generated.

The most important compression specification is the deviation, *CompDev*. For non-numeric tags, *CompDev* and *CompDevPercent* are ignored. They will be displayed by applications as zero.

Note: *CompDev* specifies the compression deviation in engineering units; *CompDevPercent* specifies the compression deviation in percent of *Span*. If you change one, the other is automatically changed to be compatible. If you try to change both at once, *CompDevPercent* takes precedence.

Exception deviation

Exception reporting is used to define the precision of a data stream, and the amount of deviation that constitutes a significant change. Most interface programs can execute an exception-reporting algorithm to determine when to send a point value to the PI Snapshot subsystem. An exception is an event that occurs either:

- After a specified minimum duration of time since the previous event, while exceeding a specified deviation in value from that event.
- After a specified maximum duration of time since the previous event.

This means that when activated, exception reporting filters events and stores only periodic values, including duplicates, unless an event represents a significant change in the short-term trend of values. An exception event, both timestamp and value, is sent with the previous event to the Snapshot.

- **Exception Deviation**

The deviation in value required to store an event, either as a number of engineering units, or as a percentage of the point's *Span* value. The exception deviation should be less than the compression deviation by at least a factor of 2, and is ignored for digital, string and BLOB data type points.

- **Min Time**

The minimum time that must elapse after an event before an exception value can be stored.

- **Max Time**

The maximum time that can elapse after an event before automatically storing the next event as an exception value.

Set the minimum and maximum time values to 0 to turn off exception reporting.

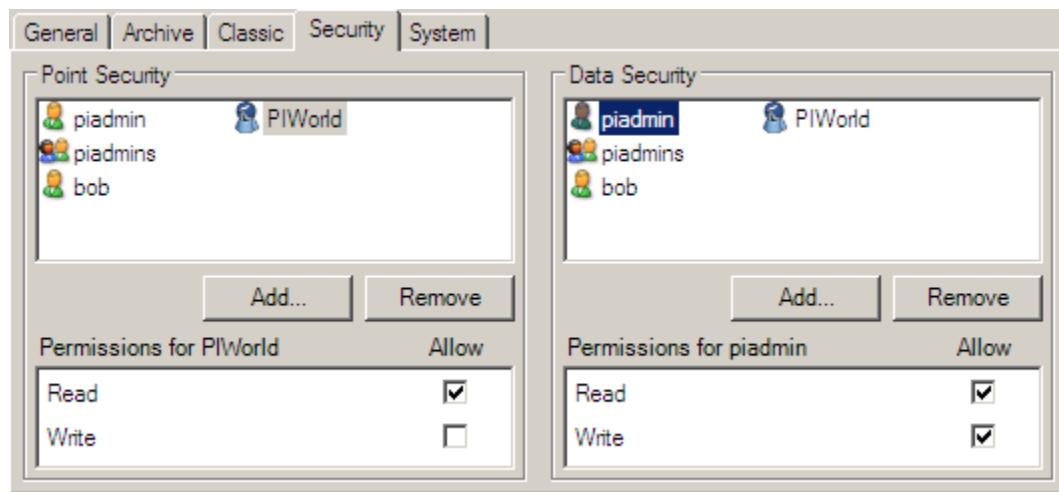
Set PE point security

To view or configure the security settings for a point, click the **Security** tab in the Point Builder tool.

Note: To have read and/or write access to PE points, you must configure read and/or write permission for both the point and the PIPOINT table.

- For a Data Archive versions 3.4.380 or later, the **Data Access** and **Point Access** lists specify the PI identities that are granted access to the PE point's data and configuration, respectively.

Click **Add** or **Remove** to update the PI identities lists for PI Totalizer and select or clear the **Read** and **Write** check boxes to update the permissions associated with each PI identity.



- For a PI Server versions 3.4.375 or earlier, the **Security** tab displays **Data Access** and **Point Access** settings for the PE point. The user and groups available for access assignments are determined by the Data Archive connection specified on the **General** tab.

Use the menus to specify permission levels for each user and group account such that only the correct individuals can use or edit PE point settings.



Define classic attributes for PE points

Use the **Classic** tab to view attributes that belong to the Classic point class, the default class for PE points. Most attributes are read-only, and only some of them are applicable to PE points.

- Location1 (ID)**

Enter an ID for the point. The point ID is referenced by PE Scheduler instances that use the point source, and the PI Recalculator module. The Recalculator adjusts PE point values automatically when the underlying PI point values are changed by another application, or out-of-order values are received by the snapshot.

Review PE Point system data

Use the **System** tab to view current information about the PE point including its snapshot value.

Set PE point validation options

To configure tag validation options, click the **Options**  button. A point name is validated as it is entered into the **Name** field or the **Event Tag** field, provided a server is selected.

On servers with high point counts, validation can take a second or two per character. Deselect **Validate tag after each letter is typed** to validate the point name only after the cursor leaves the **Name** text box.

Rename PE points

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Points > Performance Equations**.
3. Select the point in the Performance Equations tool, right-click, and choose **Rename**.
4. Enter a name in the **New name** field and click **OK**.

Note: If an error occurs with the rename, **OK** is disabled and a  appears. Hover your cursor over it to view the error message.

Delete calculated points

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Points > Performance Equations**.
3. Right-click on the point you want to delete, and select **Delete PI Point**.

Note: If the point is deleted, the Tag name, PointID and RecNo and the name of the user who is deleting the point are logged to the log files on the local machine on which SMT is run and the connected Data Archive server.

Performance equations (PE) syntax and functions reference

The performance equation (PE) subsystem provides an equation syntax and library of built-in functions that allow you to perform a wide variety of calculations on PI data.

PI performance equations can work with frequently updating snapshot and archive values, where tools like spreadsheets only have archive data and limited access to snapshot update values. Typical calculations include heat and material balances, unit performance, real-time cost accounting, and so on.

You typically use the PE subsystem in one of two ways:

- To create calculated points that have the PE subsystem as their source. The PE subsystem determines the value of these points by performing a PE calculation that the user specifies when creating the calculated

point. You can use PI SMT to create calculated points (Points > Performance Equations).

- To add calculations programmatically, through PI SDK or PI DataLink and PI ProcessBook client applications. For more information on these products, refer to the documentation included with each.

For detailed information regarding the PE functions, see [Alphabetical reference for PE functions](#).

Note: Client applications may not support all existing PI performance equations offered by the PE Subsystem.

For more information, see the Knowledge Base article: [PI Performance Equation \(PE\) Tips and Tricks](#).

Performance equation syntax

Writing a performance equation calculation expression is similar to writing an expression in arithmetic. You can use any of the standard arithmetic operators in a performance equation expression (such as +, -, and *) to add the values of two points together, add a number to the value of a point, and so on.

As with arithmetic expressions, the building blocks of a performance equation expression are operands and operators. Performance equations are simply expressions in which operators act on operands. A basic performance equation expression takes the form: *operand operator operand* as shown:

Operand	Operator	Operand	Resulting expression
'TagA'	+	'TagB'	TagA plus the value of TagB
3	-	'TagC'	3 minus the value of TagC
7	*	Sqr('TagD')	7 times the square root of TagD

You can construct more complex performance equation expressions, just as you can in arithmetic. The PE Scheduler performs most operations in the same order as they would be performed in a mathematical expression.

Use parentheses to group together expressions you want to evaluate first.

The following example evaluates as the sum of the values of 'TagA' and 'TagB', divided by the difference of 3 minus 'TagC':

```
( 'TagA' + 'TagB')/(3 - 'TagC')
```

This next example is TagA divided by the sum of TagA and TagB:

```
'TagA'/( 'TagA' + 'TagB' )
```

This next example is 3 plus the product of 7 and the square root of TagD:

```
3 + (7 * Sqr('TagD'))
```

Performance equation arguments

Performance equation functions have one or more arguments, or inputs, which are enclosed in parentheses following the function name. Some of the arguments may be optional. If there are several arguments, they are separated by commas:

```
functionName(argument1, argument2, argument3)
```

The following are examples of function expressions:

```
Max(3, 5, 12.6, 'sinusoid')
```

The Max function returns the maximum from a set of values. See [Max \(Tag-based PE function\)](#).

```
PrevEvent('sy:arc001', '*-2h')
```

The PrevEvent function returns the timestamp of the previous Archive event for tagname before time. See [PrevEvent \(Tag-based PE function\)](#).

```
Sqr(Abs(TagMax('tag', 'y', 't')))
```

The Sqr function returns the square root of x. See [Sqr](#).

The Abs function returns the square root of x. See [Abs \(Tag-based PE function\)](#).

The TagMax value returns the point's maximum value during a given time. See [TagMax \(Tag-based PE function\)](#).

```
Log(if 'tag'=2 then .5 else .2)
```

The Log topic returns the natural logarithm of x. See [Log](#).

Functions can also be nested and joined in expressions, as shown here with the Avg, TagVal, and Sin functions. See [Avg \(Tag-based PE function\)](#), [TagVal \(Tag-based PE function\)](#), and [Sin](#).

```
Avg(TagVal('TagA', 'y'), TagVal('TagB', 'y'), TagVal('TagC', 'y'))  
if TagVal('TagA', '*') < TagVal('TagB', '*') then sin('TagB') else  
sin('TagA')
```

You can use a tagname in any argument where a number or character string is called for. A tagname in single quotes is evaluated as if it had been written *TagVal(tagname)*, which is the same as *TagVal('tagname', '*')*. This gets the point's value at the "current" time for the calculation.

If the argument calls for a number, but the point's value is a digital state when the function is evaluated, a run-time error (*Calc Failed*) is generated.

Note: The pipetest utility can be used to check the syntax of a performance equations. See [Run the pipetest utility](#).

Performance equation operands

The operands that the PE Scheduler recognizes are listed in the following table. Note that certain operands must always be enclosed in single or double quotes.

Operands in performance equations

Operand Type	Syntax Requirements	Examples
Numbers	(none)	1342 98.6 .0015 1.2e2
Tagnames	In single quotes	'sinusoid' 'ba:level.1' 'ba.phase.1'
PI Time Expressions	In single quotes	'01-dec-03'

Operand Type	Syntax Requirements	Examples
		'16-jul-94' '*'
Strings	In double quotes	"string string string" "sinusoid"
Functions	Must be a Performance Equation function	TagVal('sinusoid') TagAvg('sinusoid') Cos('sinusoid')

Number operands

You can use numbers in performance equations. The PE Scheduler processes all numbers as floating point numbers. Examples of numbers are:

```
3.14159
299792458
299792458.
0.671
.671
6.71e-1
```

Note: The second and third examples are equal, as are the fourth, fifth, and sixth.

Tagname operands

You can use tagnames in performance equations to represent values from the Snapshot. You must put the tagname in single quotes, unless you are using the tagname as a string, in which case you must enclose it in double quotes. The PE Scheduler evaluates the tagname according to its use it in an expression: as a function argument, or as a time expression.

Tagnames in expressions

If you use the tagname in a performance equation expression, the tagname is evaluated as that point's value at the current time. For example:

```
3 + 'sinusoid'
```

is equal to the value of sinusoid at the current time (see note), plus three. The same value results from the expression:

```
3 + TagVal('sinusoid', '*')
```

Note: The exception is when this expression is used in a PE point, the PE point is event-based, and the Location3 attribute is set to one.

Tagnames as function arguments

If you use the tagname as an argument in one of the performance equation built-in functions, then the

Performance Equation Scheduler evaluates the tagname according to the type of value expected by that particular function.

For example, if the function expects a tagname, then the performance equation passes a tagname to the function. If the function expects any other data type such as a string or a number, PE Scheduler gets the current value of the point and passes that to the function—as whatever data type is expected.

For example, the Concat function expects two or more strings as arguments. It concatenates all the arguments into a single string:

```
Concat('sinusoid', 'ba:level.1')
```

To evaluate this expression, the PE Scheduler takes the current value of the sinusoid point and the ba:level.1 point and passes these to the Concat function as strings. Concat then returns a string that is composed of the value of the sinusoid point followed by the value of the ba:level.1 point. If the current values of these points are, respectively, 85.329 and 30.478, Concat returns the following string:

```
85.32930.478
```

Tagnames that are valid time expressions

Wherever possible, choose tagnames that cannot possibly be interpreted as time expressions. The tagname t-151d, for example, is also a valid time expression meaning "today minus 151 days."

If you must work with tagnames that are also valid time expressions, use the built-in function TagNum to ensure that the PE Scheduler does not treat the tagname as a time. For example, Abs(TagNum("t-151d")) would return the absolute Snapshot value of point t-151d. Note that TagNum interprets a double-quoted string as the argument.

To the PE Scheduler, an expression within single quotes can correspond either to a time or a tagname. The PE Scheduler treats expressions in single-quotes as tagnames for all the built-in functions that take a point as the first argument (examples include TagVal, TagAvg, and AlmCondition).

In all other cases, the PE Scheduler first attempts to resolve the expression as a time. If the expression is not a valid time, then the PE Scheduler tries to resolve it to a tagname. If the point does not exist, the subsystem returns the error Calc Failed.

For example, TagVal('t-151d') returns the Snapshot value for the point t-151d, if it exists. However, the expression t-151d returns the date corresponding to 151 days before today—because it is a valid time expression.

String operands

Strings are sequences of any printable characters. Strings are always enclosed in double-quotes. Some examples are:

```
"This is a string"  
"sinusoid"  
"14-Dec-97"
```

Note: Character strings might look like tagnames or time expressions, as in the second and third examples above. In some cases, the string in the third example might be interpreted as a time. The difference is that a character string is enclosed in double quotes (for example, "string") while a tagname must be enclosed in single quotes (for example, 'tagname') and a time expression may be enclosed in either single or double quotes.

Time expression operands

In a performance equation, you can use PI time syntax enclosed in single quotes. For information about PI time expressions, see [PI time](#).

Time as strings

You can also pass a time expression as a string to a function that expects a string. In this case, enclose the time expression in double quotes, rather than single quotes.

Numbers and strings as digital states

Digital state values consist of a state set specifier and a state number within that set. Each set has a list of text names for the states. You can set a digital point with an expression that results in either a number (specifying the offset) or a string (specifying the state name).

Comparing the value of digital and numeric points to strings

In performance equations, you can use expressions that compare the value of a digital or numeric point to a string. For example, if the string "Run" is in the state set for digital point PumpStatTag, then the following expression is valid:

```
If 'PumpStatTag' <> "Run" then 1 else 0
```

The state set for a numeric point is the System State Set. The System State Set is the default state set for all points and it contains a collection of all the states that any point can use, such as Shutdown, Over Range, and I/O Timeout.

For example, the following expression is true if the numeric point sinusoid contains the digital state Shutdown from the System Digital State Set.

```
'sinusoid' = "Shutdown"
```

Comparing a digital state to a string point

If you want to compare a digital state to a string point, use the DigState function to convert a string to a digital state explicitly. For example, the following are different:

```
If 'StringTag' = "Shutdown" then 0 else 1
If 'StringTag' = DigState("Shutdown") then 0 else 1
```

The former is true if the string point contains the string "Shutdown" while the latter is true if the point contains the digital state Shutdown.

Setting the digital state for a numeric or digital point

You can use a string to set the digital state for a numeric or digital point. When you do this, PE Scheduler looks first in that point's state set for a state that corresponds to the string. If the state does not exist in the point's state set, PE Scheduler searches the System Digital State Set for the state string. If PE Scheduler cannot find the state string in either the Digital State Set for that point or in the System Digital State Set, it returns *Calc Failed*. The state set for a numeric point is the System Digital State Set.

Performance equation operators

You use performance equation operators in performance equation expressions to act on operands such as tagnames, numbers, and time expressions. Operator priority works basically as it does in math—multiplication and division are performed before addition and subtraction, and so on. You can also use parentheses to group operations, just as you do in math.

The following topics detail all the performance equation operators, according to type.

Arithmetic operators

Performance Equation operators include the following simple arithmetic operators:

Operator	Meaning
+	Add
-	Subtract
*	Multiply
/	Divide
^	Raise to a power
mod	Find the modulus Note: The mod operator returns the remainder after its left operand is divided by its right operand. For example, 17 mod 3 returns 2.

Arithmetic operations on time values

You can perform certain arithmetic operations on times, such as adding two time expressions, or subtracting one absolute time expression from another. The result of the operation is one of the following:

- Timestamp
 - A date and time in the PI timestamp format. For example: 25-aug-02 12:00:00.
- Period
 - A number of seconds.
- Number
 - A numerical value.

The following table shows valid operations and results, where N represents a number, T represents an absolute or combined time expression, and P represents a period.

Operator	Expression	Result	Example
+	T + P	T	'*' + '+3h'
	T + N	T	'*' + 10
	P + N	P	('t' - 'y') + 10
	P + P	P	('t' - 'y') + ('t' - 'y')
- (infix)	T - P	T	'*' - '+3h'
	T - N	T	'*' - 10
	T - T	P	't' - 'y'
	P - N	P	('t' - 'y') - 10
	P - P	P	('t' - '*') - ('t' - 'y')
*	P * N	P	('t' - 'y') * 5
	N * P	P	5 * ('+1d' - '+1h')
/	P / P	N	('t' - '*') / ('t' - 'y')
	P / N	P	('t' - '*') / 2
	N / P	N	2 / ('t' - '*')
mod	T mod P	T (see note)	'*' mod ('*'- 't')
	T mod N	T (see note)	'*' mod 2
	P mod P	P	('t' - 'y') mod ('*'- 't')
	P mod N	P	('t' - 'y') mod 3
- (prefix)	-P	P	-('t' - 'y')

Note: The timestamp returned is the result of T mod P or T mod N added to January 1, 1970 Universal Coordinate Time (UTC). So depending on the time zone, different results are expected; in some case, even an error value is returned. In PI for OpenVMS systems, the use of T mod P or T mod N returns P.

Relational operators

A relational operator returns a value of 0 for false or 1 for true. You can use these operators to compare numbers, times, digital states, or character strings. With relational operators, you can compare bad values, or values of different types without generating an error. The relational operators in performance equations are:

Operator	Meaning
<	Less than
=	Equal to
>	Greater than
<=	Less than or equal to
<>	Not equal to
>=	Greater than or equal to

Comparing bad values

If you're comparing two operands of the same type, and one or both operands has a bad value, the expression returns 0, rather than an error value.

Comparing operands of different types

When you use the <> operator to compare any two operands of different types, the expression always returns a 1 (that is, true). When you use any other relational operator (anything except <>) to compare two operands of different types, the expression returns a 0 (that is, false) except in the following case:

If one of the two operands is the digital type, then the PI Performance Equation Scheduler compares the digital operand to the digital state of the other operand, if it exists. For example:

```
'sinusoid' = DigState("Shutdown")
```

Time comparisons

You can perform all comparisons, including the in operator, on times.

```
'*+20m' >= '*+300s'  
PrevEvent('tag1', '*') > PrevEvent('tag2', '*')
```

If a comparison is true, the result is 1; otherwise, it is 0.

Prefix operators

A prefix operator is simply an operator that appears to the left of its operand, for example, - x.

Operator	Meaning
-	Negation
Not	Complementation: Returns 1 if operand is 0 (or rounding to 0) and 0 otherwise

The expression following a prefix operator should be numeric. If you use a tagname as the operand, make sure that the point returns a number. Note, too, that even points that typically return a number sometimes return a

digital state, such as Shutdown. Valid examples include:

```
-3
Not 7
-TagVal ('sinusoid')
Not Cos('ba:level.1')
-StateNo('DigitalTag')
Not TagBad('StringTag'))
```

The last two examples use digital and string points (DigitalTag and StringTag) but these are used as arguments to functions that return numbers (StateNo and TagBad).

Conjunction, disjunction, and inclusion operators

Operator	Meaning	Returns
and	conjunction	True, if operands A and B both evaluate to true. If both A and B are integers, returns the result of a bitwise AND operation.
or	inclusive disjunction	True, if either operand A or operand B evaluates to true. If both A and B are integers, returns the result of a bitwise OR operation.
in ..	membership in a range	1 if true and 0 if false
in ()	membership in a discrete set	1 if true and 0 if false

Inclusion operator examples

The following are two examples that use inclusion operators.

```
If 1 in 0 .. 2 Then 1 Else 0
```

The result is **1**, since **1** is between **0** and **2**.

```
If 1 in (0, 2) Then 1 Else 0
```

The result is **0**, since **1** does not equal either **0** or **2**.

Using the inclusion operator with digital state functions

You can use the in .. operator with functions that return digital states, in which case the operator uses the offset within the Digital State Set for comparison. The digital states must all be in the same Digital State Set. Lexical comparisons are made with character strings.

Time comparisons using inclusion operators

You can use the Inclusion operators (in.., in()) on time expressions. If the comparison is true, the result is 1; otherwise, it is 0.

If-then-else expressions

The if-then-else operator is a compound operator whose operands are used as follows:

```
if expr0 then expr1 else expr2
```

where expr0, expr1, and expr2 are expressions. If expr0 is true (nonzero), this operator returns the value of expr1; if expr0 is false (zero), it returns the value of expr2.

Here are some examples:

```
if 'tag1' > 50 then "overlimit" else "good"
if 'tag1'= 1 then Sin('tag2') else if 'tag1'= 2 then Cos('tag2') else Tan('tag2')
if 'tag3'<> "shutdown" then (if 'tag1' > 'tag2' then 'tag1' else 'tag2') else "error"
'*' + (if 'tag2' = 0 then 3600 else 0)
```

Note: You must include the if, the then, and the else. Nested operations are supported.

Operator priority

The PE Scheduler executes operators in order of priority, from highest to lowest. When multiple operators have the same priority, the order of execution is either from left-to-right or right-to-left, depending on the operator, as listed in the following table.

Operator	Priority (highest number is done first)	Order
Post	8	L-R
^	8	R-L
Not	7	R-L
UMINUS		
)	6	L-R
(6	R-L
*	5	L-R
/		
mod		
in .. in()	5	R-L
+	4	L-R
-		
>	3	L-R
>=		
=		
<>		

Operator	Priority (highest number is done first)	Order
<=		
<		
Else		
And	2	L-R
Or	1	L-R

You can use parentheses to specify the order of calculation. Regardless of operator priority, operations within parentheses are evaluated before operations outside those parentheses. For example:

`(2 + 3) * 5 equals 25`

while

`2 + 3 * 5 equals 17`

Data types

The PE Scheduler recognizes these data types:

- Number
- Character string
- Tagname
- Time
- Period

Every variable has one of these data types; every function returns one of these data types. The PE Scheduler cannot typically use one data type where another is expected. For example, you cannot add two character strings, or multiply two times together. Additionally, the built-in functions might require particular data types for particular arguments.

Type checking

At compile time, the PE Scheduler checks type compatibility as far as is possible. This process catches some errors, such as trying to add a number to a character string.

However, not all type compatibility errors can be detected at compile time. The expression

`'sinusoid' / 2.0`

works well if sinusoid has a numeric value. However, if sinusoid is equal to the digital state Shutdown the expression returns an error (Calc Failed).

Note: Comparisons (expressions using relational operators) are an exception to this rule. Every comparison is valid, regardless of its operand types.

Error values

When the PE Scheduler cannot perform a calculation during runtime, it returns the error value Calc Failed. Error values propagate through most calculations. For example, an error value plus one is an error value. Exceptions to this rule are:

- The BadVal and Concat functions
- Any of the relational operators, when a value of 0 is returned

To check for compile time errors on Windows-based computers, check the PI SDK logs.

Testing the performance equation syntax

The pipetest utility is a command line function that checks the syntax of a performance equation. It can operate interactively, take its input from a file, or check the extended descriptor of a point. There is also an System Management Tool (SMT) plug-in to test a performance equation.

Run the pipetest utility

The pipetest utility is located in the pi\adm directory. To start pipetest, open a command window, change to the pi\adm directory, and type a pipetest command. For a complete list of pipetest commands, type:

```
pipetest -?
```

The pipetest utility is limited to equations that are 4095 characters or less and you can not use it to test dynamic response functions.

Using pipetest in interactive mode

To run the pipetest utility interactively from a command prompt window, open a command window, change to the pi\adm directory and enter:

```
pipetest
```

At the pipetest equation prompt, type in the equation you want to test. If the equation syntax is not valid, pipetest displays a syntax error. If the syntax is valid, pipetest displays the result of the equation.

Using pipetest in file input mode

You can also put one or more performance equations in a simple text file, and pass the entire file to pipetest using the -f switch. In the text file, you put each equation on a single line. You can include comment lines by beginning the line with an exclamation mark (!).

Here's the text from an example pipetest file, called peTestEquations.txt:

```
! test calculation for point A
if BadVal('sinusoid') then 0 else ('sinusoid')/25
! test calculation for point B
TimeLT('sinusoid', 'y' , 't', TagVal('sinusoid', '*'))
```

To test the equations in the file, type:

```
pipetest -f peTestEquations.txt
```

Each input line in turn is echoed and the evaluated result is displayed.

Built-in performance equation functions

The PE Scheduler provides a wide range of built-in functions that make it easier for you to perform calculations on PI data. (You can also use steam functions in performance equations.)

For a complete list of functions, see [Alphabetical reference for PE functions](#). To learn about performance equation syntax, see [Performance equation syntax](#).

PE functions by type

This topic provides comprehensive tables that categorize and describe the available Performance Equation (PE) functions by type, including mathematical, aggregate, logical, string, time-based, archive-related, and point attribute functions. These tables are intended to serve as a quick reference for understanding the purpose and syntax of each function.

Detailed information on how to use these functions, including syntax, parameters, and practical examples, is available in the [Expression Functions Reference](#).

A complete list of Performance Equation functions organized alphabetically can be found in the [Alphabetical reference for PE functions](#).

Math functions

Name	Description
Abs	Absolute value
Asin	Arc sine
Acos	Arc cosine
Atn	Arc tangent
Atn2	Arc tangent (two arguments)
Cos	Cosine
Cosh	Hyperbolic cosine
Exp	Exponential
Float	Conversion of string to number
Frac	Fractional part of number
Int	Integer part of number
Log	Natural logarithm
Log10	Common logarithm

Name	Description
Poly	Evaluate polynomial
Round	Round to nearest unit
Sgn	Numerical sign
Sin	Sine
Sinh	Hyperbolic sine
Sqr	Square root
Tanh	Hyperbolic tangent
Tan	Tangent
Trunc	Truncate to next smaller unit

Aggregate functions

Name	Description
Avg	Average
Max	Maximum
Median	Median selector
Min	Minimum
PStDev	Population standard deviation
SStDev	Sample standard deviation
Total	Sum

Miscellaneous functions

Name	Description
BadVal	See if a value is bad (not a number or time)
Curve	Get value of a curve
DigState	Get digital state from a string
IsDST	Test whether a time is in local daylight saving time period
IsSet	Test if a PI value is annotated, substituted, or questionable

Name	Description
StateNo	The code number of a digital state
TagBad	See if a point has an abnormal state

PI Archive retrieval functions

Name	Description
NextEvent	Time of a point's next archive event
NextVal	Point's next value after a time
PrevEvent	Time of a point's previous archive event
PrevVal	Point's previous value before a time
TagVal	Point's value at a time

PI Archive search functions

Name	Description
FindEq	Timestamp when point = value
FindGE	Timestamp when point >= value
FindGT	Timestamp when point > value
FindLE	Timestamp when point <= value
FindLT	Timestamp when point < value
FindNE	Timestamp when point != value
TimeEq	Total period when point = value
TimeGE	Total period when point >= value
TimeGT	Total period when point > value
TimeLE	Total period when point <= value
TimeLT	Total period when point < value

Name	Description
TimeNE	Total period when point != value

PI Archive statistics functions

Name	Description
EventCount	Number of archive events
PctGood	Percent of good time in a period
Range	Range of minimum to maximum value
StDev	Time-weighted standard deviation
TagAvg	Time-weighted average
TagMean	Event-weighted average
TagMax	Maximum value in a period
TagMin	Minimum value in a period
TagTot	Time integral over a period

Point attribute functions

Name	Description
TagDesc	Get a point's descriptor
TagEU	Get a point's engineering unit string
TagExDesc	Get a point's extended descriptor
TagName	Get a point's name
TagNum	Get a point's ID
TagSource	Get a point's point source string
TagSpan	Get a point's span
TagType	Get a point's type character
TagTypVal	Get a point's typical value

TagZero	Get a point's zero value
---------	--------------------------

Time functions

Name	Description
Bod	Timestamp for beginning of the day for given time
Bom	Timestamp for beginning of the month for given time
Bonm	Timestamp for first of the next month for given time
Day	Day of the month from a time
DaySec	Seconds since midnight from a time
Hour	Hour from a time
Minute	Minute from a time
Month	Month from a time
Noon	Timestamp for local noon of day of a time
ParseTime	Convert character string to time
Second	Second from a time
Weekday	Day of the week from a time
Year	Year from a time
Yeaday	Day of the year from a time

Dynamic response functions

Name	Description
Arma	Dynamic response from Auto Regressive Moving Average model
Delay	Introduce time delay
MedianFilt	Select the median value of time series

Name	Description
Impulse	Dynamic response characterized by impulse response shape

Alarm status functions

Name	Description
AlmAckStat	Alarm acknowledgment status code
AlmCondition	Condition code number for Alarm State
AlmCondText	Alarm condition as text
AlmPriority	Alarm priority number

String functions

Name	Description
Ascii	ASCII character code for a character
Char	String for ASCII character code(s)
Compare	Wild comparison of two strings
DigText	Text for a digital state
Format	Formatting of a numerical number
InStr	Instance of a sub-string
LCase	Conversion of all characters to lower case
Len	Length of a string
Left	First characters in a string
Trim	Removal of blanks on the left side of a string
Mid	Extraction of a sub-string from a string
Right	Last characters in a string

Name	Description
RTrim	Removal of blanks on the right side of a string
Trim	Removal of blanks on both sides of a string
UCase	Conversion of all characters to upper case

String conversion functions

Name	Description
Concat	Concatenate two or more strings
String	String representing any PI value
Text	Concatenation of strings for a series of PI value arguments

Alphabetical reference for PE functions

PE functions make it easy to perform calculations on PI data. You can also use steam table functions in performance equations, see [Steam functions reference](#).

This section contains topics that detail each PE function.

Abs (Tag-based PE function)

Abs function in PI Server's Performance Equations (PE) subsystem is used to return the absolute (non-negative) value of a number or the result of a numeric expression. This function is particularly helpful in scenarios where the direction of a value (positive or negative) is not relevant, such as when calculating differences or deviations between values.

Syntax

```
Abs(x)
```

Arguments

x

Must be an integer, real number, or tag that provides a numeric value.

Returns

The absolute (non-negative) value of x.

Exceptions

If x is not an integer or real number, an error returns value is returned.

Example

```
Abs(1) // Returns: 1
Abs(-2.2) // Returns: 2.2
Abs('tag1') // Returns the absolute value of the current value of 'tag1'
Abs('tag1' - 'tag2
') // Returns the absolute value of the difference between 'tag1' and 'tag2'
```

Usage

The Abs function can be incorporated into PI Performance Equations to ensure values remain non-negative, a useful approach for monitoring metrics or conditions where only the magnitude of a change matters. For example, you could define a new Performance Equation tag in PI System Management Tools (SMT) to track the absolute difference between two tags, using an expression like:

```
Abs('tag1' - 'tag2')
```

This setup helps capture deviations or comparative metrics without considering positive or negative signs, making it ideal for applications that require non-directional data. For more information, see [Performance equations \(PE\) syntax and functions reference](#).

Acos (Tag-based PE function)

Return the inverse cosine (arccos) of an integer or real number. The inverse cosine of x is the angle in radians whose cosine is equal to x .

Syntax

```
Acos(x)
```

Arguments

x

Must be a real number between -1.0 and 1.0, inclusive.

Returns

The inverse cosine of x , in radians.

Exceptions

If x is not a number, or is less than -1.0 or greater than 1.0, returns an error value.

Example

```
Acos(-0.5) [Returns 2.0944]  
Acos(0.75) [Returns 0.72273]
```

AlmAckStat

Return the acknowledgment status code for an alarm point.

Syntax

```
AlmAckStat(alm)
```

Arguments

alm

An alarm tagname.

Returns

The acknowledgement status code for the given Alarm State. Possible values are:

0 - acknowledged (or no alarm)

1 - unacknowledged

2 - missed alarm

Exceptions

If the argument is not a digital state tagname, an error condition is returned.

Example

```
AlmAckStat('alarmtag')  
AlmCondition (page ), AlmCondText (page ), AlmPriority (page )
```

AlmCondition (Tag-based PE function)

Returns the condition code for an Alarm State.

Syntax

```
AlmCondition(alm)
```

Arguments

alm

An alarm tagname.

Returns

The code number of the condition for the alarm tagname.

Exceptions

If the argument is not a digital state tagname, an error is returned.

Example

```
AlmCondition('alarmtag')  
AlmAckStat (page ), AlmCondText (page ), AlmPriority (page )
```

AlmCondText (Tag-based PE function)

Return the string for the condition of an Alarm State.

Syntax

```
AlmCondText(alm)
```

Arguments

alm

An alarm tagname.

Returns

The condition text for the condition represented by the alarm status.

Exceptions

If the argument is not a digital state tagname, an error condition is returned.

Example

```
AlmCondText('alarmtag')  
AlmAckStat (page ), AlmCondition (page ), AlmPriority (page )
```

AlmPriority (Tag-based PE function)

Return the priority of an Alarm State.

Syntax

```
almPriority(alm)
```

Arguments

alm

A digital state value from a PI Alarm State Set.

Returns

The priority of the given alarm. Priorities are small positive integers. Priority 0 is an alarm that is never unacknowledged.

Exceptions/Errors

If the argument is not from a valid Digital State Set for alarms, an error condition is returned.

Example

```
almPriority('alarmtag')  
almPriority(digstate("__ Lolo"))  
AlmAckStat (page ), AlmCondition (page ), AlmCondText (page )
```

Arma

Calculate dynamic response for Auto Regressive Moving average model as shown in Examples.

Syntax

```
Arma(in, runflag, (a1, a2, ... aN),(b0, b1, b2, ... bN))
```

Arguments

The denominator and numerator coefficient series are enclosed in parenthesis with a comma between the two. There must be only one more term (**b0**) in the numerator than denominator. Both **in** and **runflag** may be any expression that evaluates to a number.

in

Must be an integer or real number.

runflag

Non-zero enables filter to run.

a1, a2, ...

Coefficients of past output terms.

b0, b1, b2 ...

Coefficients of the present and past input terms of the model.

Returns

The next output value in time series response to past and present input.

```
ut = a1 * ut-1 + a2 * ut-2 + ... + an * ut-n + b0 * yt +  
     b1 * yt-1 + b2 * yt-2 + ... + bn * yt-n
```

Where **ut** is model output at time **t**, now. **ut-1** is output one sample interval in the past. **yt** is the input signal at time **t**. If **runFlag** expression result is **0**, the model is reset. Depending on the number of coefficients used, **Arma** stores the inputs and outputs until an evaluation of the model is available. For example, in

```
Arma( 'input_tag', 1, ( 0. ,1), ( 1, -1 ,1))
```

Arma stores two previous values of the input and output. Hence when the point is first created, no good results are returned until two prior values of the input are stored. From then on, the two most current previous values are stored.

Exceptions

Arma gives different results depending on which type of scheduling is used. In scan class scheduling, the interval between time series values depends on the scan class and gives values at evenly spaced time intervals.

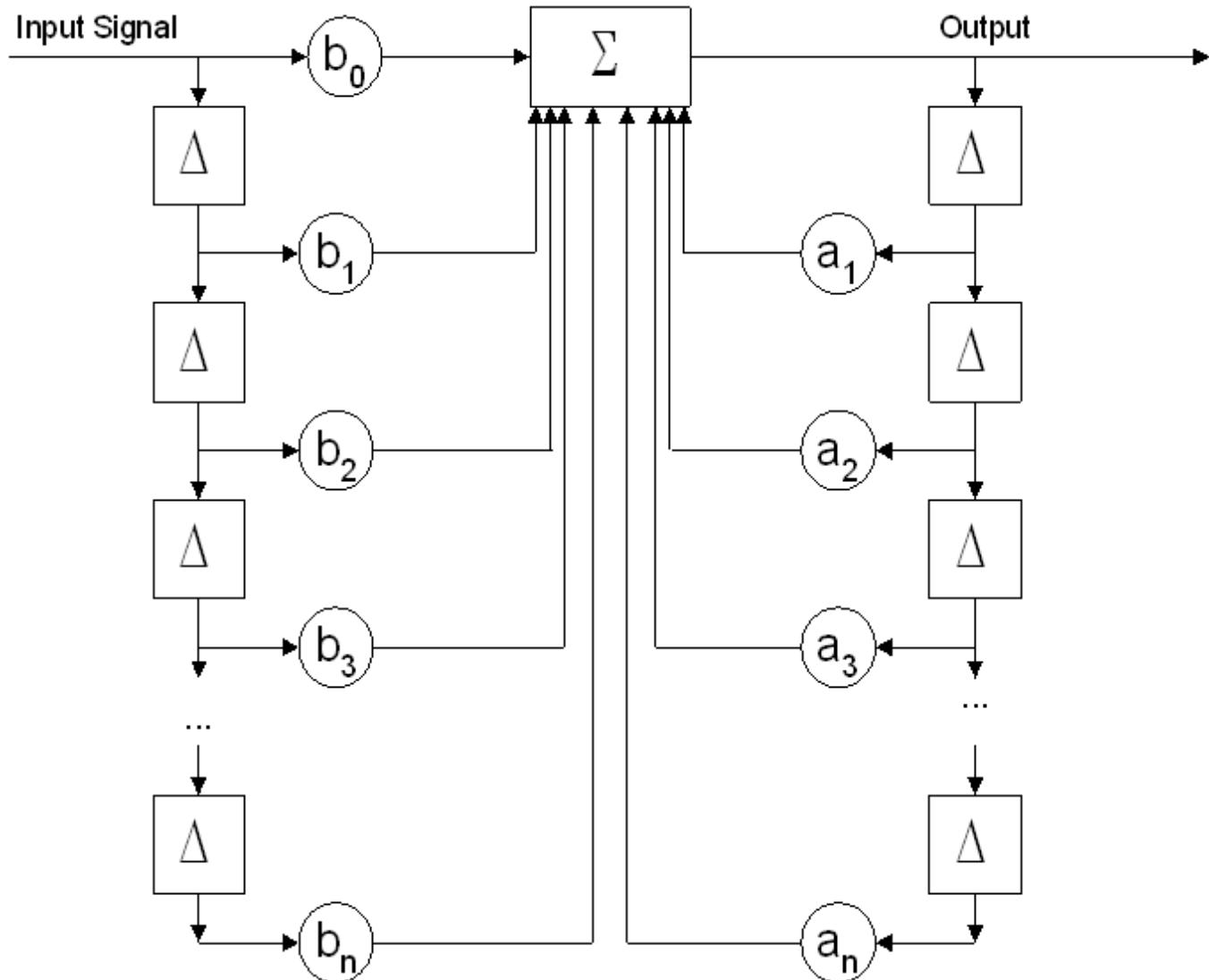
Event-based scheduling is dependent on a trigger from another point. With event-based scheduling, **Arma** sometimes gives results that are not trustworthy. Use **Arma** with event-based scheduling only:

- If the exception deviation is not zero.
- If the point used for event-based scheduling is exception-base rather than scan-based.

Arma is intended for the PE Scheduler only. Use of **Arma** in client applications might produce unexpected results. If the input point is not a real number or integer, Arma returns an error.

Example

```
Derivative: Arma('input_tag', 1, (0.), (1, -1))  
Integration: Arma('input_tag', 1, (1.), (.05, 0.))  
Second order filter: Arma('input_tag', 1, (.25,.25), (.1,.25,.15 ))
```



Ascii (Tag-based PE function)

Return the ASCII character code of the first character of a string.

Syntax

```
Ascii(s1)
```

Arguments

s1

Any expression evaluating to a string

Returns

The character code of the first character of the string

Exceptions

If the argument is not a string, an error value is returned

Example

```
Ascii("D") [Returns 68]  
Ascii("Program") [Returns 80, the ASCII character code for the first letter of the  
string]
```

Asin (Tag-based PE function)

Return the inverse sine (arcsin) of a number. The inverse sine of x is the angle in radians whose sine is equal to x .

Syntax

```
Asin(x)
```

Arguments

x

Must be a real number between -1.0 and 1.0, inclusive.

Returns

The inverse sine of x , in radians.

Exceptions

If x is not a number, or is less than -1.0 or greater than 1.0, returns an error value.

Example

```
Asin(-0.5) [Returns -0.5236]  
Asin(TagVal('att1','y'))  
Asin('att1')
```

Atn (Tag-based PE function)

Return the inverse (or arc) tangent of an integer or real number. The inverse tangent of x is the angle in radians whose tangent is equal to x .

Syntax

```
Atn(x)
```

Arguments

x

Must be an integer or real number

Returns

The inverse tangent of x, in radians.

Exceptions

Returns an error if x is not an integer or real number.

Example

```
Atn(1) [Returns 0.7854]
Atn(-2.2) [Returns -1.1442]
Atn('att1')
Atn('att1' - 'att2')
```

Atn2 (Tag-based PE function)

Return the inverse tangent (arctan) of a tangent value a/b. The inverse tangent is the angle measured in radians from the positive x-axis to a line whose endpoints are the origin and the Cartesian coordinates (b, a).

Syntax

```
Atn2(x, y)
```

Arguments

x

An integer or real number

y

A non-zero integer or real number

Returns

The inverse tangent in radians of the tangent value x/y.

Exceptions

Returns an error if x or y is not an integer or real number.

Example

```
Atn2(1,2) [Returns 0.46365]  
Atn2('att1', 'att2')  
Atn2(TagVal('att1','y'), TagVal('att2', 'y'))
```

Avg (Tag-based PE function)

Return the average of all the arguments.

Syntax

```
Avg (x1,[...])
```

Arguments

x1, [...]

May be numbers, times, or periods but all must be the same data type.

Returns

The average of the arguments. The result is the same data type as the operands.

Exceptions

Arguments whose run-time values are character strings or digital states are not included in the average. If all values are character strings or digital states, **Avg** returns an error value.

Example

```
Avg(TagVal('tag1', '*'),TagVal('tag2', '*'),1,2)
```

- The average of the value of tag1 at the current time ('*'), tag2 at the current time ('*'), and the values 1 and 2.

```
Avg('*', 't', '14-Dec-97', '14 8:00')
```
- The average of four different time values.

```
Avg('tag1', 'tag2')
```
- The average of the current values of tag1 and tag2.

Badval (Tag-based PE function)

Test a value to see if it is bad. For real and integer points, a bad value is a digital state value. For digital points, a bad value is a value outside the point's digital state set.

Syntax

```
Badval(x)
```

Arguments

x

A value to be tested.

Returns

1 if the value is bad

0 if the value is not bad

Exceptions

Returns 1 for blob points. Returns 0 for character strings.

Example

```
BadVal('tag1')
BadVal('digitaltag')
BadVal(TagVal('stringtag', '14-Dec-97 8:00:00'))
```

Bod (Tag-based PE function)

Return a timestamp for beginning of the day from a time expression.

Syntax

```
Bod(t1)
```

Arguments

t1

A time expression, enclosed in single quotes

Returns

Timestamp for the start of the day

Exceptions

None

Notes

This function is useful for establishing a time at a unique clock time independent of the length of particular days.

Example

```
Bod('*')
```

- [Return a timestamp for beginning of the day today.]

```
Bod('y')
```
- [Return a timestamp for beginning of the day yesterday.]

```
Bod(FindEq('att1', '-3d', '*', 50))
```
- [Return a timestamp for beginning of the day when 'att1' value was first equal to 50 in the past 72 hours.]

Bom (Tag-based PE function)

Return a timestamp for midnight on the first day of the month from a given time expression.

Syntax

```
Bom(t1)
```

Arguments

t1

A time expression

Returns

Timestamp for the start of the month.

Exceptions

None

Notes

This function is useful for establishing a time at a unique clock time independent of the length of particular days.

Example

```
Bom('*')
```

- [Return a timestamp for midnight on the first day of this month.]

```
Bom(PrevEvent('att1', '*'))
```
- [Return a timestamp for midnight on the first day of the month when 'att1' had a value before the current one.]

```
Bom(FindEq('att1', '-60d', '*', 50))
```
- [Return a timestamp for midnight on the first day of the month when the value of 'att1' was first equal to 50 in the past 60 days.]

Bonm (Tag-based PE function)

Return a timestamp for midnight on the first day of a following month from a given time expression.

Syntax

```
Bonm(t1)
```

Arguments

t1

Time expression, enclosed in single quotes

Returns

Timestamp for the start of the next month.

Exceptions

None

Notes

This function is useful for establishing a time at a unique clock time independent of the length of particular days.

Example

```
Bonm('*')
```

- [Return a timestamp for midnight on the first day of next month.]

```
Bonm('y')
```
- [Return a timestamp for midnight on the first day of the following month from yesterday's date.]

```
Bonm(FindEq('att1', '-60d', '*', 50))
```

- [Return a timestamp for midnight on the first day of the following month when the value of 'att1' was first equal to 50 in the past 60 days.]

Char

Build a string from ASCII character codes.

Syntax

Char(x1, ... xn)

Arguments

- **x1, ... xn**
Integers

Returns

A string built from the 80 character codes

Exceptions

Returns an error if an argument is not a number

Example

- Char(80, 73)
[Returns "PI"]
- Char(65)
[Returns "A"]
- Char(5 * 10)
[Returns "2"]

Compare (Tag-based PE function)

Compare two strings using wildcard characters ("*" and "?").

Syntax

Compare(s1, s2 [,caseSensitive])

Arguments

s1, s2

Strings (s2 can contain wildcard characters)

caseSensitive

Optional flag indicating if the comparison is case sensitive. If False (the default), the comparison is case-insensitive. If True, the comparison is case-sensitive.

Returns

1 if s1 = s2

0 otherwise

Exceptions

Wildcard characters in s1 are treated literally and not as wildcards.

Example

```
compare("What","what",True) = 0
compare("b","a") = 0
compare("What","wha?") = 1
compare("What","wh") = 0
```

Concat (Tag-based PE function)

Concatenate two or more strings.

Syntax

```
Concat(s1, s2[, ... sn])
```

Arguments

s1, s2, sn

Must be character strings, or expressions yielding character strings.

Returns

The character strings, concatenated together. This function does not insert blanks between its arguments. To include a space in the concatenated string, add an argument consisting of a string that is a single space enclosed in double quotes..

Note: Consider using Text, which is more general and more precise in many cases than Concat.

Example

```
Concat("shut", "down") = "shutdown"
```

Cos

Return the trigonometric cosine of an integer or real number.

Syntax

$\text{Cos}(x)$

Arguments

- x

Must be an integer or real number, which represents an angle in radians

Returns

The cosine of x

Exceptions

If x is not an integer or real number, returns an error value

Example

- $\text{Cos}(1.1)$
[Returns 0.4536]
- $\text{Cos}(1)$
[Returns 0.5403]
- $\text{Cos}('att1')$
[Return the trigonometric cosine of the value of 'att1' at trigger time]

Cosh

Return the hyperbolic cosine of a number.

Syntax

$\text{Cosh}(x)$

Arguments

- x

Must be an integer or real number

Returns

The hyperbolic cosine of x

Exceptions

If x is not an integer or real number, returns an error

Example

- Cosh(1)
[Returns 1.5431]
- Cosh(-1)
[Returns 1.5431]
- Cosh('att1')
[Return the hyperbolic cosine of the value of 'att1' at trigger time]

Curve (Tag-based PE function)

Return the y value of a curve given the x value.

Syntax

```
Curve( x, (x1,y1) (x2,y2) ... (xn,yn) )
```

Arguments

x

Expression evaluating to a number

x1, y1

The first point on the curve. The xi's and yi's are numeric constants evaluated at compile time. The values set for xi's must be in ascending order.

Returns

Returns the y value on the curve corresponding to the value of x. Linear interpolation is used between points defining the curve. If the value of x is less than x1 then y1 is returned and if it is greater than xn, yn is returned. The points are assumed to be ordered in the x direction from smallest to largest.

Exceptions

If the value of x is not an integer or real number, an error value is returned

Example

```
Curve(3, (0,100) (100,0) )
```

- [Returns 97]

```
Curve('tag1', (25,25) (75,75) )
```

Day

Extract the day of the month from a time expression.

Syntax

```
Day(t1)
```

Arguments

- **t1**
time expression enclosed in single quotes

Returns

The day of the month of a time expression in the range 1 through 31

Exceptions

None

Example

- Day('*')
[Return what day of the month today is]
- Day('y')
[Return what day of the month yesterday was]
- Day(FindEq('att1', '-28d', '*', 50))
[Return what day of the month it was when the value of 'att1' was first equal to 50 in the past 28 days]

DaySec

Return the total time in seconds between the start of day (midnight) and the time denoted in the argument.

Syntax

```
DaySec(t1)
```

Arguments

- **t1**

A time expression, enclosed in single quotes

Returns

Total seconds since the start of day (midnight) till *t1*, in the range 0-86399

Exceptions

None

Example

- `DaySec('*')`
[Return the number of seconds from the start of day (midnight) until now]

Delay

Delay line, the output tracks the input. For use in real time calculations, in **pipeschd.exe** for example, this function might be a better choice than PrevVal.

Syntax

```
Delay( x, runflag, n )
```

Arguments

x

Must be an integer or real number.

runflag

Non-zero enables filter to run.

n

Length of the delay, integer.

Returns

The input signal delayed by *n* calculation intervals. For scan class scheduling, the calculation interval is based on the scan class. For event based scheduling, the calculation interval is dependent on the trigger and the exception deviation.

Exceptions

Delay is not supported in the **pipetest** utility or in PI DataLink. If the input point is not a real number or integer, Delay returns an error. Delay returns **Calc Failed** until n scans have elapsed after startup.

Example

```
Delay('tag1',1,2)
```

DigState (Tag-based PE function)

Translate a character string representing a digital state into its corresponding digital state.

Syntax

```
DigState(s1 [, x])
```

Arguments

s1

A character string representing a digital state.

x

(Optional) A digital point in which the character string represents a digital state. If omitted, all digital state sets, starting with the system digital set, are searched for the given string.

Returns

An enumeration value. (For PI PE calculations, a digital state is returned.)

Exceptions

If the character string does not represent a digital state in the digital state set of the specified digital point, the function returns **Calc Failed**. If digital point is omitted and character string does not represent a digital state in any of the digital sets, **Calc Failed** is returned.

Example

```
DigState("digitalstring", 'digitaltag')
StateNo(DigState("digitalstring", 'digitaltag'))
```

DigText (Tag-based PE function)

Obtain the text corresponding to the current digital state of a point.

Syntax

```
DigText(tagname)
```

Arguments

tagname

A tagname that represents a digital state variable.

Returns

The text for the digital state.

Exceptions

If the argument is not a digital state tagname, an error condition is returned.

Example

```
DigText('alarmtag')
DigText('cdm158' )
DigText('nondigitaltag') would not compile and returns an error message
```

EventCount (Tag-based PE function)

Find the number of Archive events for a point over a given time.

Syntax

```
EventCount(tagname, starttime, endtime [, pctgood])
```

Arguments

tagname

A tagname, enclosed in single quotes. This point must represent a continuous variable.

starttime

A time expression representing the beginning of the time range to search.

endtime

A timestamp that is greater than starttime; the end of the time range to search.

Note: When endtime is a future time (such as '+1h'), TagCount might include the system digital state **No Data** and thus is larger than the number of events stored in the PI Archive. Avoid using a future time if possible.

pctgood

Optional. Minimum time percentage, over the given time range, that the point's archived values must be good.

Returns

Number of Archive events for the point within the specified interval.

Exceptions

If the point has no good values or the pctgood minimum is not reached for the given time range, returns an error value.

Example

```
EventCount('tag1', 'y', '*')
EventCount('tag1', '14-Dec-97', '+1d', 70)
EventCount('tag1', '14-Dec-97', '15-Dec-97')
```

Exp

Return the exponential of an integer or real number. This is the number e^x , where $e = 2.7182818\dots$

Syntax

Exp(x)

Arguments

- x

Must be an integer or real number

Returns

The exponential of x

Exceptions

If x is not an integer or real number, returns an error value

Example

- Exp(11)
[Returns 59874]
- Exp('att1')
[Return the exponential of the value of 'att1' at trigger time]
- Exp(TagVal('att1', 't'))
[Return the exponential of the value of 'att1' at the start of day (12am) today]

FindEq (Tag-based PE function)

Find the timestamp closest to starttime, within the given range, for which the point was equal to the given value.

Syntax

```
FindEq(tagname, starttime, endtime, value)
```

Arguments

tagname

A tagname enclosed in single quotes.

starttime

A time expression representing the beginning of the time range to search. Relative times are relative to endtime if endtime is not itself a relative time.

endtime

A time expression representing the end of the time range to search. Relative times are relative to starttime if starttime is not itself a relative expression. If endtime is earlier than starttime, the range is searched backwards.

value

Must be an integer or real number or digital state (character string), the value to search for.

Returns

The timestamp closest to starttime, within the given range, for which the point was equal to the given value.

Exceptions

If the point was never equal to the given value, **FindEq** returns an error value.

Notes

FindEq interpolates between Archive events, if necessary, to find the value it is looking for.

Example

```
FindEq('tag1', 't', '*', 40.0)
FindEq('digitaltag', '-1d', '*', TagVal('digitaltag', '14-Dec-97'))
FindEq('digitaltag', '14-Dec-97', '*', "On")
```

FindGE (Tag-based PE function)

Find the first or last time, within a range, when a point is greater than or equal to a given value.

Syntax

```
FindGE(tagname, starttime, endtime, value)
```

Arguments

tagname

A tagname.

starttime

A time expression representing the beginning of the time range to search or a time relative to endtime, if endtime is a time.

endtime

A time expression representing the end of the time range to or a time (in seconds) relative to starttime, if starttime is a time. If endtime is earlier than starttime, the range is searched backwards.

value

Must be an integer or real number or digital state (character string), the value to search for.

Returns

The timestamp closest to starttime, within the given range, for which the point was greater than or equal to the given value.

Exceptions

If the point was always less than the given value, **FindGE** returns an error value.

Notes

FindGE interpolates between archive events, if necessary, to find the value it is looking for.

Example

```
FindGE('tag1', 't', '**',40.0)
FindGE('digitaltag', '-1d', '*', TagVal('digitaltag', '14-Dec-97'))
FindGE('tag1', '-1d', '**','tag2')
```

FindGT (Tag-based PE function)

Find the first time, within a range, when a point is greater than a given value.

Syntax

```
FindGT(tagname, starttime, endtime, value)
```

Arguments

tagname

A tagname.

starttime

A time expression representing the beginning of the time range to search. Can be a time relative to endtime if endtime is a time.

endtime

End of the time range to search, time expression or time (in seconds) relative to starttime if starttime is a time. If this time is earlier than starttime, the range is searched backwards.

value

Must be an integer or real number or digital state (character string), the value to search for.

Returns

The timestamp closest to starttime, within the given range, for which the point was greater than the given value.

Exceptions

If the point was never greater than the given value, **FindGT** returns an error value.

Notes

FindGT interpolates between Archive events, if necessary, to find the value it is looking for.

Example

```
FindGT('tag1', 't', '*', 40.0)
FindGT('tag1', '-1d', '*', 40.0)
FindGT('digitaltag', '-1d', '*', TagVal('digitaltag', 'y'))
```

FindLE (Tag-based PE function)

Find the first time, within a range, when a point is less than or equal to a given value.

Syntax

```
FindLE(tagname, starttime, endtime, value)
```

Arguments

tagname

A tagname.

starttime

Beginning of the time range to search; time expression or time relative to endtime if endtime is a time.

endtime

End of the time range to search, timestamp or time (in seconds) relative to starttime if starttime is a time. If this time is earlier than starttime, the range is searched backwards.

value

Must be an integer or real number or digital state (character string), the value to search for.

Returns

The timestamp closest to starttime, within the given range, for which the point was less than or equal to the given value.

Exceptions

If the point was always greater than the given value, **FindLE** returns an error value.

Notes

FindLE interpolates between Archive events, if necessary, to find the value it is looking for.

Example

```
FindLE('tag1', 't', '*',40.0)
FindLE('tag1', -3600, '*',40.0)
FindLE('tag1', 'Saturday', '*',40.0)
```

FindLT (Tag-based PE function)

Find the first time, within a range, when a point is less than a given value.

Syntax

```
FindLT(tagname, starttime, endtime, value)
```

Arguments

tagname

A tagname.

starttime

Beginning of the time range to search; time expression or time relative to endtime if endtime is a time.

endtime

End of the time range to search, time expression or time (in seconds) relative to starttime if starttime is a time. If this time is earlier than starttime, the range is searched backwards.

value

Must be an integer or real number or digital state (character string), the value to search for.

Returns

The timestamp closest to starttime, within the given range, for which the point was less than the given value.

Exceptions

If the point was never less than the given value, FindLT returns an error value.

Notes

FindLT interpolates between Archive events, if necessary, to find the value it is looking for.

Example

```
FindLT('tag1', 't', 3600,40.0)
FindLT('tag1', -1h, '*',40.0)
FindLT('tag1', '14-Dec-97 01:00:00.0001', '*',40.0)
```

FindNE (Tag-based PE function)

Find the first time, within a range, when a point is unequal to a given value.

Syntax

```
FindNE(tagname, starttime, endtime, value)
```

Arguments

tagname

A tagname.

starttime

Beginning of the time range to search; time expression or time relative to endtime if endtime is a timestamp.

endtime

End of the time range to search, time expression or time (in seconds) relative to starttime if starttime is a time. If this time is earlier than starttime, the range is searched backwards.

value

Must be an integer or real number or digital state (character string), the value to search for.

Returns

The timestamp closest to starttime, within the given range, for which the point was unequal to the given value.

Exceptions

If the point was always equal to the given value, **FindNE** returns an error value.

Example

```
FindNE('tag1', 'y', '*', 40.0)
FindNE('tag1', '14-Dec-97', '*', 40.0)
FindNE('tag1', '14-Dec-97', 'Monday', 40.0)
```

Float

Convert a string to a number.

Syntax

```
Float(x)
```

Arguments

- **x**

A string or a number, or an attribute whose value evaluates to a number at time of evaluation

Returns

A number for a numeric string. If **x** is already a number, **x** is returned

Exceptions

If **x** is not a number or a numeric string, returns *Calc Failed*

Notes

Unit of measure of the argument, if it exists, is carried over to the result. **Float** also takes *timespan* and *boolean* as argument. Note, however, that **Float** only converts *timespan* format to the number of seconds from 12:00am Jan 1, 1970.

Example

- **Float("12.3")**
[Converts string to a number and returns 12.3]
- **Float(12.3)**
[Returns 12.3]
- **Float('*')**

[Return the total number of seconds passed since Jan 1, 1970]

Format (Tag-based PE function)

Convert a number to string according to a format expression.

Syntax

```
Format(x, format [,numType ])
```

Arguments

x

A numeric value (real or integer).

format

Format-control string. This is the same as that used by the C language function **Sprintf**.

numType

Optional. Character indicating number type, either R(eal) or I(ntege). The default is R.

Returns

A formatted string.

Example

```
Format('sinusoid', "%3.3f", "R") = "66.890"
Format(45, "%3.3d") = "045"
Format(45, "%3.3d", "I") = "045"
Format(45, "%3.3d", "R") = "000" (Don't do this!)
```

Frac

Return the fractional part of a real number. Returns 0 for integers.

Syntax

```
Frac(x)
```

Arguments

- x

Must be an integer or real number

Returns

The fractional part of x

Exceptions

If x is not an integer or real number, returns an error value

Notes

By definition: $\text{Int}(x) + \text{Frac}(x) = x$

Unit of measure of the argument, if it exists, is carried over to the result

Example

- `Frac(1)`
[Returns 0]
- `Frac(1.3)`
[Returns 0.3]
- `Frac(TagVal('att1', '*'))`
[Return the fractional part of the value of 'att1' at current time.]

Hour

Extract the hour from a time expression.

Syntax

`Hour(t1)`

Arguments

- **t1**
A time expression, enclosed in single quotes

Returns

The hour of time, in the range 0-23

Exceptions

None

Example

- Hour('*')
[Return the hour portion of current time]
- Hour('Saturday')
[Returns 0]
- Hour(FindEq('att1', '-1d', '*', 50))
[Return the hour of the time when the value of 'att1' was first equal to 50 in the past 24 hours]

Impulse

Dynamic response specified by the impulse response.

Syntax

```
Impulse(tagname, runflag, i1,i2 ... )
```

Arguments

tagname

Must be a tagname for a numerical point.

runflag

Non-zero enables filter to run.

i1, i2, ...

Unit impulse response specifying dynamic model, text sequence of numbers.

Returns

Dynamic model output as function of time.

```
u(t)=i1*u(t-1) + i2*u(t-2) + ...
```

Where u(t) is the current output and u(t-1) is the output one sample interval in the past.

Exceptions

Impulse gives different results depending on which type of scheduling is used. In clock scheduling, the interval between time series values depends on the scan class and gives values at evenly spaced time intervals.

Event-based scheduling is dependent on a trigger from another point. With event-based scheduling, **Impulse** sometimes gives results that are not trustworthy. Use **Impulse** with event-based scheduling only:

- If the exception deviation is not zero.
- If the point used for event-based scheduling is exception-based rather than scan-based.

If the input point is not a real number or integer, **Impulse** returns an error. **Impulse** is intended for the PE

Scheduler only. Use of **Impulse** in client applications might produce unexpected results.

Example

```
Impulse('tag1',1,1,1,1)
```

InStr

Return the location within a string where a sub-string match is first found.

Syntax

```
InStr([start,] s1, s2 [,caseSensitive])
```

Arguments

- **start**

Optional: An integer specifying which character in *s1* to start the comparison. Must be larger than or equal to 1.

- **s1, s2**

Two strings to be compared.

- **caseSensitive**

Optional: A flag indicating if the comparison is case-sensitive. If 0 (the default) the comparison is case-insensitive, if 1, the comparison is case-sensitive.

Returns

0 if *s2* is not a sub-string of *s1* starting from the start position; otherwise, the location of character where *s2* first matches the characters in *s1* from the start position.

Exceptions

Wildcard characters are not treated as wildcards.

Example

- InStr("What", "At")
[Returns 3]
- InStr("What What What", "What")
[Returns 1]
- InStr("what", "At", 1)
[Returns 0]
- InStr(4, "what", "At")

- [Returns 0]
- `InStr('att1', "Error")`
- [Returns 1 if the value of *att1* is "Error"]

Int (Tag-based PE function)

Return the integer part of an integer or real number.

Syntax

```
Int(x)
```

Arguments

x

A number or string.

Returns

The integer part of x. If x is a string, it is first converted into a number.

Exceptions

If x is not a number or a numeric string, returns *Calc Failed*.

Example

```
Int('tag1')
Int(1)
Int(2.1)
Int("2.1")
```

IsDST (Tag-based PE function)

Determine if a time expression is in a daylight saving time (DST) period on the local machine.

Syntax

```
IsDST(time)
```

Arguments

time

A time expression.

Returns

1 if the time is in a DST period and 0 otherwise.

Exceptions

If the argument is not a time value, an error condition is returned.

Example

```
IsDST('*')
IsDST('*-182.5d')
IsDST('t')
IsDST('timestringtag')
```

IsSet (Tag-based PE function)

Determine if a PI value is annotated, substituted, or questionable.

Syntax

```
IsSet(x, select)
```

Arguments

x

Any value. May be an integer, real number, digital state, or character string.

select

A string but only the first character is considered. "a" for annotated; "s" for substituted; and "q" for questionable. It is case-insensitive.

Returns

1 if true and 0 otherwise.

Exceptions

None.

Example

```
IsSet('sinusoid', "a")
IsSet('sinusoid', "annotated")
IsSet('sinusoid', "annotatted is misspelled")
IsSet('stringtag', "annotation is misspelled but it does not matter.")
IsSet('stringtag', "A")
```

```
IsSet('alarmtag1','q')
IsSet('stringtag','s')
```

LCASE

Convert a string to a lowercase string.

Syntax

```
LCASE(s1)
```

Arguments

- **s1**
string

Returns

A string that has been converted to lowercase

Exceptions

If the argument is not a string, returns an error value

Example

- `LCASE("String")`
[Returns "string"]

LEFT

Return a specified number of characters of a string from the left.

Syntax

```
LEFT(s1, len)
```

Arguments

- **s1**
String
- **len**
Integer

Returns

The first **len** characters of the string, starting from the left

Exceptions

If the arguments are not of the required types, returns an error

Example

- `Left("String_att", 3)`
[Returns "Str"]

Len (Tag-based PE function)

Determine the length of a string.

Syntax

```
Len(s)
```

Arguments

s

A string.

Returns

The length of a string.

Exceptions

If the argument is not a string, returns an error value.

Example

```
Len("Stringtag") = 9  
Len('Stringtag') = 5 if the Snapshot value for the stringtag equals "Error"
```

Log

Return the natural logarithm (base e = 2.7182818...) of an integer or real number.

Syntax

`Log(x)`

Arguments

- `x`

Must be an integer or real number greater than zero

Returns

The natural logarithm of x

Exceptions

If x is zero or negative, or not a number, returns an error value

Example

- `Log(14)`
[Returns 2.6391]
- `Log(TagVal('att1', '14-Dec-16'))`
[Return the natural log of the value of 'att1' at 12:00am on Dec 14, 2016]

Log10

Return the base 10 logarithm of an integer or real number.

Syntax

`Log10(x)`

Arguments

- `x`

Must be an integer or real number greater than zero

Returns

The base 10 logarithm of x

Exceptions/Errors

If x is zero or negative, or not a number, returns an error value

Example

- `Log10(100)`
[Returns 2]
- `Log10(TagVal('att1', '14-Dec-16'))`
[Return the base 10 logarithm of the value of 'att1' at 12:00am on Dec 14, 2016]

LTrim

Remove the leading blanks from a string.

Syntax

`LTrim(s1)`

Arguments

- `s1`
string

Returns

A string with leading blanks removed

Exceptions

If `s1` is not a string, an error value is returned.

Example

- `LTrim(" String")`
[Returns "String"]
- `LTrim("String ")`
[Returns "String "]

Max (Tag-based PE function)

Return the maximum of a set of values.

Syntax

`Max(x1, x2, ..., xn)`

Arguments

x1...xn

May be numbers, times, or time periods, but all must be the same.

Returns

The maximum value of the arguments. The result has the same data type as the arguments.

Exceptions

Arguments whose run-time values are digital states are ignored. If all values are digital states, **Max** returns an error value.

Example

```
Max('*', 'y', 'Saturday')
Max(14, 'tag1', 14.5, TagVal('tag2','14-Dec-97'))
Max('*-*-*h', 't'-'y', TimeEq('tag1', 'y', 't',50))
```

[Min \(page \)](#)

Median (Tag-based PE function)

Return the median (middle) value for a set of three or more values.

Syntax

```
Median(x1, x2, x3, [... xn])
```

Arguments

x1, x2, x3, [... xn]

All arguments must be the same type. Enter at least three arguments, all of the same data type (integers and real numbers, times, or time periods).

Returns

The median value of the input arguments. If the number of arguments is even, the average of the two middle values is returned.

Exceptions

Arguments whose run-time values are digital states are ignored. The function must have greater than two arguments that evaluate to non-digital states; otherwise, Median returns an error value.

Notes

Median allows for mixed integer and real data types. Median follows the data type of the first argument. Hence if the first argument is a point that evaluates to an integer then all the other entries are converted to integers by rounding.

Example

- To find the median of these timestamps: now, 12:00am yesterday, and 12:00am last Saturday:
`Median('*', 'y', 'Saturday')`
- To find the median of these values: 14, the current value of tag1, 14.5, and the value for tag2 at midnight on Dec 14, 2016:
`Median(14, 'tag1', 14.5, TagVal('tag2', '14-Dec-16'))`
- To find the median of these time durations: from 1 hour ago to now, from 12:00am yesterday to 12:00am today, and the total time tag1 was equal to 50 between 12:00am yesterday and 12:00am today:
`Median('*-*1h', 't'- 'y', TimeEq('tag1', 'y', 't', 50))`

MedianFilt

Return the median value of the last specified number of values of a time series.

Syntax

```
MedianFilt( tagname, runflag, number )
```

Arguments

tagname

Must be a numerical point.

runflag

Non-zero enables filter to run.

number

The number of series elements to be considered. A numeric constant greater than or equal to 3.

Returns

The median value of the last number values in the series of values.

Exceptions

Arguments whose run-time values are digital states are ignored. **MedianFilt** is not supported in the pipetest utility or in PI DataLink. If all values are digital states, **MedianFilt** returns an error value.

Example

```
MedianFilt('tag1', 1, 3)
```

Mid

Return a sub-string within a string.

Syntax

```
Mid(s1, start [,len])
```

Arguments

- **s1**
string
- **start**

An integer specifying the position of the first character within the string. The first character in the string is number 1

len

Optional: The maximum length of the returned string. The default is the length of the string

Returns

len characters of the string to the right of (and including) the first character whose position is specified by start

Exceptions

If the arguments are not of the required types, an error value is returned. The maximum number of characters that can be returned is 999

Example

- Mid("String", 3)
[Returns "ring"]
- Mid("String", 3, 2)
[Returns "ri"]

Min (Tag-based PE function)

Return the minimum of a set of values.

Syntax

```
Min(x1, ..., xn)
```

Arguments

x1..xn

May be numbers, times, or time periods, but all must be the same data type.

Returns

The minimum of the arguments. The result has the same data type as the arguments.

Exceptions

Arguments whose run-time values are digital states are ignored. If all values are digital states, Min returns an error value.

Example

```
Min('*', 'y', 'Saturday')
Min(14, 'tag1', 14.5, TagVal('tag2','14-Dec-97'))
Min('*'-'*-1h', 't'-'y', TimeEq('tag1', 'y', 't',50))
```

[Max \(page \)](#)

Minute

Extract the minute from a time expression.

Syntax

```
Minute(t1)
```

Arguments

- t1

A time expression, enclosed in single quotes

Returns

The minute of time, in the range 0-59

Exceptions

None

Example

- `Minute('*')`
[Extract the minute from current time]
- `Minute(FindGT('att1', '-1h', '*', 5))`
[Extract the minute from a timestamp when the value of 'att1' was first greater than 5 in the past hour.
Return error if it was never over 5]

Month

Extract the month from a given time expression.

Syntax

```
Month(t1)
```

Arguments

- **t1**
A time expression, enclosed in single quotes

Returns

The month of time, in the range 1-12

Exceptions

None

Example

- `Month('*')`
[Return the current month]
- `Month(FindEq('att1', '-10d', '*', 5))`
[Return the month from a timestamp when the value of 'att1' was first equal to 5 in the past 10 days]

NextEvent (Tag-based PE function)

Find the time of a point's next Archive event after a given time.

Syntax

```
NextEvent(tagname, time)
```

Arguments

tagname
A tagname.
time
A time expression.

Returns

The timestamp of the next Archive event for tagname after time.

Exceptions

If point has no Archive data after time, returns an error value.

Example

```
NextEvent('tag1','*')
NextEvent('digitaltag', '*')
NextEvent('tag1','*')
NextEvent('digitaltag', '*')
```

NextVal

Find the value of a point's next Archive event after a given time.

Syntax

```
NextVal(tagname, time)
```

Arguments

tagname
A tagname.
time
A time expression.

Returns

The value of the next Archive event for tagname after time.

Exceptions

If point has no Archive data after time, returns an error value.

Example

```
NextVal('tag1','*-1h')
NextVal('digitaltag', '14-Dec-97')
NextEvent (page), PrevEvent (page), PrevVal (page), TagVal (page)
```

Noon

Return a timestamp for noon on the day of a given time expression.

Syntax

```
Noon(t1)
```

Arguments

- **t1**
A time expression enclosed in single quotes

Returns

A timestamp corresponding to noon of the day of the input time

Exceptions

None

Notes

This function is useful for establishing a unique clock time independent of the length of particular days.

Example

- Noon('*')
[Return the timestamp for noon of current day]
- Noon(FindEq('att1', '-3d', '*', 50))
[Return the timestamp for noon of the day when 'att1' was first equal to 50 in the past 3 days]

NoOutput

Do not write current calculation result.

Syntax

```
NoOutput()
```

Arguments

None

Notes

It is important to include the parentheses after this function (use *NoOutput()* instead of *NoOutput* as *NoOutput* is an invalid syntax). This function applies only to the current calculation.

Example

- If 'att1' < 100 or 'att1' > 200 then 'att1' else NoOutput()

ParseTime (Tag-based PE function)

Translate a PI time expression to a timestamp. Use regular time expression inside single quotes for better performance.

Syntax

```
ParseTime(s1)
```

Arguments

s1

A character string in PI time format, enclosed in double quotes

Returns

The timestamp corresponding to s1

Exceptions

If s1 is not a character string, or if there is a syntax error, returns an error value

Notes

There is no difference between *ParseTime("14-Nov-92")* and the time expression '*14-Nov-92*', except that the **ParseTime** call takes more time. This is because the time expression (enclosed in single quotes) is evaluated at compile time, not run time. Therefore it is most efficient to use a time expression (enclosed in single quotes.)

If you write *ParseTime('14-Nov-92')* (using single quotes, not double quotes) the parser detects an error, because

the expression in single quotes is already translated to a timestamp at compile time

The expression `ParseTime(":12:00:00")` is not the same as the time expression '`:12:00:00`'. The **ParseTime** expression is evaluated at run time and translated using '*' as the relative time base, while the time expression is evaluated at compile time and uses the time the expression is parsed as the relative time base

Example

```
ParseTime(Concat("12", " -31", " -16"))
```

- [Returns 12/31/2016 12:00:00 AM, which is the same as '12/31/16']

```
ParseTime("14-Dec-16")
```
- [Renders the same result as '14-Dec-16'. Use only when string operations are necessary]

```
ParseTime("*")
```
- [Renders the same result as '*'. Use only when string operations are necessary]

PctGood (Tag-based PE function)

Find the percentage, over a given time range, when a point's archived values are good. The PctGood function finds the percentage of values returned by the expression that do not contain an error code from the System digital state set. For more information about, see [System digital state set](#).

Syntax

```
PctGood(tagname, starttime, endtime)
```

Arguments

tagname

A tagname.

starttime

Must be a time expression, the beginning of the time range to search.

endtime

Must be a time expression, greater than starttime; the end of the time range to search.

Returns

An integer or real number from 0.0 to 100.0: the percentage of the given time when the point had good values.

Example

```
PctGood('tag1', 'y', '*')
PctGood('tag1', '-1h', '*')
```

Poly

Return the polynomial $c_0 + c_1x + c_2x^2 + \dots + c_nx^n$.

Syntax

```
Poly(x, c0 [, ... cn])
```

Arguments

- **x**
Variable. An integer or real number
- **c0 [, ... cn]**
Coefficients. There must be at least one coefficient. All must be numbers.

Returns

The value of the polynomial

Exceptions

If x or any coefficient is not an integer or real number, **Poly** returns an error value

Example

- Poly(3, 4, 5)
[Returns 19]
- Poly('att1', 2, 3)

PrevEvent (Tag-based PE function)

Find the time of a point's previous Archive event before a given time.

Syntax

```
PrevEvent(tagname, time)
```

Arguments

- tagname
A tagname.
- time
A time expression.

Returns

The timestamp of the previous Archive event for tagname before time.

Exceptions

If point has no Archive data before time, returns an error value.

Example

```
PrevEvent('tag1', '*')
PrevEvent('tag1', '14-Dec-97')

NextEvent (page), NextVal (page), PrevVal (page), TagVal (page)
```

PrevVal (Tag-based PE function)

Find the value of a point's previous Archive event before a given time.

Syntax

```
PrevVal(tagname, time)
```

Arguments

tagname
A tagname
time
A time expression

Returns

The value of the previous Archive event for tagname before time

Exceptions

If point has no Archive data before time, returns an error value

Example

```
PrevVal('tag1', '*')
PrevVal('tag1', '14-Dec-97')

NextEvent (page), NextVal (page), PrevEvent (page), TagVal (page)
```

PStDev (Tag-based PE function)

Returns the *population standard deviation* for a population of two or more values.

Syntax

```
PStDev(x1, x2, ..., xn)
```

Arguments

x1...xn

May be numbers or time expressions, but all must be the same.

Returns

The *population standard deviation* for the arguments. For numerical arguments the result is a number. For arguments that are time expressions (absolute times or time periods), the result is a number indicating a time period expressed in seconds.

The population standard deviation of a population x1...xn is

$$\sqrt{\frac{\sum (x_i - \mu)^2}{n}}$$

where μ is the mean of the arguments; that is,

$$\frac{\sum x_i}{n}$$

Exceptions

Arguments whose run-time values are digital states are ignored. If all values are digital states, an error value is returned.

Notes

PStDev uses every value in a population to calculate the population standard deviation. However, it is common, especially for a large population, to estimate standard deviation from a sample of the population. **SStDev** uses a set of sample values to calculate sample standard deviation, which approximates the population standard deviation.

Example

```
PStDev('att1', 'att2')
```

```
PStDev('*', '14-Dec-97', 'y')
PStDev('*-'y', '14-Dec-97'-'*', '-1h')
```

Range (Tag-based PE function)

Find the difference between a point's maximum and minimum values during a given time, according to values stored in the PI Archive.

Syntax

```
Range(tagname, starttime, endtime [, pctgood])
```

Arguments

tagname

A tagname. This point should represent a continuous variable.

starttime

Must be a time expression, the beginning of the time range to search.

endtime

Must be a time expression, greater than starttime; the end of the time range to search.

pctgood

Optional. Minimum time percentage, over the given time range, that the point's archived values must be good.

Returns

The difference between the point's maximum and minimum values during the given time.

Exceptions

If the point has no good values or the pctgood minimum is not reached in the given time range, returns an error value.

Notes

Note: The **OverRangeStat** and **UnderRangeStat** digital states are not taken into account when calculating **Range**.

Example

```
Range('tag1', 'y', '*')
Range('tag1', '-1h', 'y')
Range('tag1','y', '+1h',70)
```

Right

Return a specified number of characters of a string from the right.

Syntax

```
Right(s1, len)
```

Arguments

- **s1**
string
- **len**
integer

Returns

len characters of the string from the right

Exceptions

If the arguments are not of the required types, an error value is returned.

Example

- Right("String", 3)
[Returns "ing"]
- Right("String", 20)
[Returns "String"]

Round (Tag-based PE function)

Round a number or time to the nearest unit.

Syntax

```
Round(x [, unit])
```

Arguments

x

Must be an integer or real number or time expression.

unit

Optional. The size of the unit to round to. If x is a number, unit must be a number. If x is a time expression or time period, unit must be a time period. If unit is omitted, **Round** rounds to the nearest integer (for a number) or second (for a time period).

Returns

The nearest value to x which is an integer multiple of unit. Returns the same data type as x . For more details, see the following examples.

Exceptions

If x is a string, or if unit is of the wrong data type, returns an error value.

Notes

If x is time and unit is omitted this routine has no effect: times are accurate only to 1 second.

Example

```
Round(12.499)
```

- Round to nearest integer (12.0)

```
Round(12.5)
```
- Half a unit rounds up (13.0)

```
Round(12.8, 10)
```
- Round to nearest ten (10.0)

```
Round('14-Dec-97 11:47', '+1h')
```
- Timestamp rounded to nearest hour (14-Dec-97 12:00)

```
Round('18:47' - '15:00', '+1h')
```
- Period measured in seconds rounded to nearest hour (10800)

Note: **Round** to the nearest day results in a timestamp of the closest day in UTC time and not local time.

RTrim

Trim trailing blanks from a string.

Syntax

```
RTrim(s1)
```

Arguments

- **s1**
string

Returns

The source string with trailing blanks removed

Exceptions

If *s1* is not a string, an error value is returned

Example

- RTrim("String ")
[Returns "String"]
- RTrim(" String")
[Returns " String"]

Second

Extract the second from a time expression.

Syntax

Second(*t1*)

Arguments

- **t1**
A time expression enclosed in single quotes.

Returns

The second of time, in the range 0-59

Exceptions

None

Example

- Second('*')
[Return the second from current time]
- Second(FindEq('att1', '-1m', '*', 2))
[Return the second from a timestamp when the value of 'att1' was first equal to 2 in the past minute]

Sgn

Return an indicator of the numerical sign of a number.

Syntax

$\text{Sgn}(x)$

Arguments

- x

Must be an integer or real number

Returns

-1 if $x < 0$; 0 if $x = 0$; 1 if $x > 0$

Exceptions

If x is not an integer or real number, returns an error value

Example

- $\text{Sgn}(1.1)$
[Returns 1]
- $\text{Sgn}(0)$
[Returns 0]
- $\text{Sgn}(-0.1)$
[Returns -1]
- $\text{Sgn}('att1')$
[Returns 1 if the value of 'att1' is positive, 0 if it equals 0, and -1 if it is negative]

Sin

Return the trigonometric sine of a number.

Syntax

$\text{Sin}(x)$

Arguments

- x

Must be an integer or real number, which represents an angle in radians

Returns

The sine of x

Exceptions

If x is not a number, returns an error value

Example

- `Sin(1)`
[Returns 0.84147]
- `Sin(1.1)`
[Returns 0.89121]
- `Sin('att1')`
[Return the trigonometric sine of the value of 'att1' at trigger time]

Sinh

Return the hyperbolic sine of a number.

Syntax

`Sinh(x)`

Arguments

- x
Must be an integer or real number

Returns

The hyperbolic sine of x

Exceptions

If x is not a number, returns an error value

Example

- `Sinh(1)`

[Returns 1.1752]

- `Sinh(1.1)`

[Returns 1.3356]

- `Sinh('att1')`

[Return the hyperbolic sine of the value of 'att1' at trigger time]

Sqr

Return the square root of a number.

Syntax

`Sqr(x)`

Arguments

- `x`

Must be an integer or real number equal to or greater than zero

Returns

The square root of x

Exceptions

If x is negative, or is not a number, returns an error value

Example

- `Sqr(2)`

[Returns 1.4142]

- `Sqr(2.1)`

[Returns 1.4491]

- `Sqr('att1')`

[Return square root of the value of 'att1' at trigger time]

SStDev (Tag-based PE function)

Return the sample standard deviation for two or more arguments that are a sample of a larger population. The standard deviation of a sample $x_1 \dots x_n$ is equal to

$$\sqrt{\frac{\sum(x_i - \mu)^2}{n-1}}$$

Where μ is the sample mean; that is,

$$\frac{\sum x_i}{n}$$

Syntax

```
SStDev(x1, x2, ..., xn)
```

Arguments

x1...xn

May be numbers or time expressions, but all must be the same.

Returns

The sample standard deviation of the arguments. If the arguments are numbers, the result is a number; if they are times or time periods, the result is a time period (number of seconds).

Exceptions

Arguments whose run-time values are digital states are ignored. If there are not at least two numeric values, **SStDev** returns a zero.

Notes

In the rare case where you have the entire population, rather than a sample, you might use the function **PstDev**, rather than **SStDev**.

Example

```
SStDev('tag1', 'tag2', TagVal('tag1', 'y'))
SStDev('y', 't', '14-Dec-97')
SStDev(1, 2, 1.1)
```

StateNo (Tag-based PE function)

Translate a digital state into its corresponding state number.

Syntax

```
StateNo(digstate)
```

Arguments

digstate

An enumeration value.

Returns

The offset into the Digital State Set corresponding to *digstate*.

Exceptions

If a point is passed as *digstate* that is not a digital point, returns an error value.

Notes

A digital state may appear more than once in the digital state table. In this case, the value that **StateNo** returns may vary. If *digstate* is the value of a digital point, **StateNo** returns a code number appropriate for that point.

Example

```
StateNo('digitaltag')
StateNo(TagVal('digitaltag', '*-1h'))
```

StDev (Tag-based PE function)

Find the time-weighted standard deviation of a point over a given time, according to values stored in the PI Archive.

Syntax

```
StDev(tagname, starttime, endtime [, pctgood])
```

Arguments

- **tagname:** The name of the tag (point) whose standard deviation you want to calculate. This point must represent a continuous variable.
- **starttime:** Must be a time expression representing the beginning of the time range to search.
- **endtime:** Must be a time expression, greater than starttime; representing the end of the time range to search.
- **pctgood(optional):** Minimum time percentage over the given time range, that the point's archived values must be good.

Returns

The point's time-weighted standard deviation over the given time.

Exceptions

If the point has no good values or the **PctGood** minimum is not reached for the given time range, returns an error value.

Notes:

- If the point has few good Archive values during the time period, this function's result may not be trustworthy. Use the **PctGood** function to find out what percentage of the values is good.
- The **pctgood** is ignored when it is passed as a string.

Examples

1. Using relative time expressions:

```
StDev('tag1', 'y', '*')
StDev('tag1', '14-Dec-97', '+1d', 85)
StDev('tag1', '14-Dec-97', '15-Dec-97')
```

This calculates the standard deviation of 'tag1' from yesterday ('y') to the current time ('*').

2. Using absolute and relative time expressions with a minimum percentage of good values:

```
StDev('tag1', '14-Dec-97', '+1d', 85)
```

This calculates the standard deviation of 'tag1' from December 14, 1997, to one day later ('+1d'), requiring at least 85% of the values to be good.

3. Using absolute time expressions:

```
StDev('tag1', '14-Dec-97', '15-Dec-97')
```

This calculates the standard deviation of 'tag1' from December 14, 1997, to December 15, 1997.

String

Convert any value to a string.

Syntax

```
String(x)
```

Arguments

- **x**

Any expression that is of normal PI value type

Returns

The string representing the value argument

Exceptions

None

Example

- `String("Hello, PI user!")`
[Returns "Hello, PI user!"]
- `String(12.23)`
[Returns "12.23"]
- `String('sinusoid')`
[Return current value of 'sinusoid' in string]
- `String('*')`
[Return the current time in string]

TagAvg (Tag-based PE function)

Find the time-weighted average value of a point over a given time, according to values stored in the PI Archive.

Syntax

```
TagAvg(tagname, starttime, endtime [, pctgood])
```

Arguments

tagname

A tagname. This point must represent a continuous variable.

starttime

Must be a time expression representing the beginning of the time range to search.

endtime

Must be a time expression, greater than starttime; representing the end of the time range to search.

pctgood

Optional. Minimum time percentage over the given time range, that the point's archived values must be good.

Returns

The point's time-weighted average value over the given time.

Exceptions

If the point has no good values or the pctgood minimum is not reached for the given time range, returns an error value.

Notes

Note: If the point has few good Archive values during the time period, this function's result may not be trustworthy. Use the **PctGood** function to find out what percentage of the values are good.

Example

```
TagAvg('tag1', 'y', '*')
TagAvg('tag1', '14-Dec-97', '+1d',70)
TagAvg('tag1', '14-Dec-97', '15-Dec-97')
```

TagBad (Tag-based PE function)

Test if a point has an abnormal state at a given time. If the point's type is R or I, any digital state is abnormal. If the point is type D, the states that are defined for that point are normal; all others are abnormal.

Syntax

```
Tagbad(tagname [, time])
```

Arguments

tagname

A tagname.

time

Optional. A time expression. If omitted, the current time (*) is used.

Returns

0 if the point's state at time is normal, 1 if it is abnormal.

Exceptions

If point does not exist, or has no archived value at time, returns an error value.

Notes

Badval can test any value or expression; **TagBad** can only test a point.

Example

```
TagBad('tag1', '*')
TagBad('digitaltag', '14-Dec-97')
TagBad('tag1', 'y')
```

TagDesc (Tag-based PE function)

Get a point's descriptor from the Point Database.

Syntax

```
TagDesc(tagname)
```

Arguments

tagname

A tagname.

Returns

The point's descriptor.

Exceptions

If point does not exist, returns an error value.

Example

```
TagDesc('tag1')
TagDesc('digitaltag')
```

TagEU (Tag-based PE function)

Get a point's engineering unit string from the Point Database.

Syntax

```
TagEU(tagname)
```

Arguments

tagname

A tagname.

Returns

The point's engineering units.

Exceptions

If point does not exist, returns an error value.

Example

```
TagEU('tag1')
```

TagExDesc (Tag-based PE function)

Get a point's extended descriptor from the Point Database.

Syntax

```
TagExDesc(tagname)
```

Arguments

tagname

A tagname.

Returns

The point's extended descriptor.

Exceptions

If point does not exist, returns an error value.

Example

```
TagExDesc('tag1')
```

TagMax (Tag-based PE function)

Find the maximum value of a point during a given time, according to values stored in the PI Archive.

Syntax

```
TagMax(tagname, starttime, endtime [, pctgood])
```

Arguments

tagname

A tagname.

starttime

A time expression indicating the beginning of the time range to search. Relative times are relative to endtime, if endtime is not itself a relative time.

endtime

End of the time range to search. Relative times are relative to starttime, if starttime is not itself a relative time. This time must be after starttime.

pctgood

Optional. Minimum time percentage over the given time range, that the point's archived values must be good.

Returns

The point's maximum value during the given time.

Exceptions

If the point has no good values or the pctgood minimum is not reached for the given time range, returns an error value.

Notes

Note: The **OverRange** digital state is not taken into account when evaluating TagMax.

Example

```
TagMax('tag1', '-1h', '*', 95)
```

- Here, the starttime is one hour before the endtime, which is now ('*'). During the time span, at least 95% of the values must be good.

```
TagMax('tag1', 'y', '*')
TagMax('tag1', '-1h', '*', 95)
TagMax('tag1', '14-Dec-97', '+1h')
```

TagMean (Tag-based PE function)

Find the average value of a point over a given time, according to values stored in the PI Archive.

Syntax

```
TagMean(tagname, starttime, endtime [, pctgood])
```

Returns

The point's average value over the given time. Notice that the average is not time-weighted.

Arguments

tagname

A tagname. This point must represent a continuous variable.

starttime

Must be a time expression representing the beginning of the time range to search.

endtime

Must be a time expression, greater than starttime; representing the end of the time range to search.

pctgood

Optional. Minimum time percentage over the given time range, that the point's archived values must good.

Exceptions

If the point has no good values or the pctgood minimum is not reached for the given time range, returns an error value. Unlike some other summary functions, TagMean does not interpolate any value on the boundary. Thus, if there is no Archive event between the specified interval, an error value is returned.

Notes

Note: If the point has few good Archive values during the time period, this function's result may not be trustworthy. Use the **PctGood** function to find out what percentage of the values is good.

Example

```
TagMean('tag1', 'y', '*')
TagMean('tag1', '14-Dec-97', '+1d', 70)
TagMean('tag1', '14-Dec-97', '15-Dec-97')
```

TagMin (Tag-based PE function)

Find the minimum value of a point during a given time, according to values stored in the PI Archive.

Syntax

```
TagMin(tagname, starttime, endtime [, pctgood])
```

Arguments

tagname

A tagname. This point should represent a continuous variable.

starttime

Beginning of the time range to search. Relative times are relative to endtime, if endtime is not itself a relative time.

endtime

Relative times are relative to starttime, if starttime is not itself a relative time. This time must be after starttime.

pctgood

Optional. Minimum time percentage over the given time range, that the point's archived values must good.

Returns

The point's minimum value during the given time.

Exceptions

If the point has no good values or the pctgood minimum is not reached for the given time range, returns an error value.

Notes

Note: The **UnderRange** digital state is not taken into account when calculating this value.

Example

```
TagMin('tag1', 'y', '*')
TagMin('tag1', '-1h', '*', 90)
TagMin('tag1', '14-Dec-97', '+1h')
```

TagName (Tag-based PE function)

Get a point's name from the Point Database.

Syntax

```
TagName(tag)
```

Arguments

tag

A tagname.

Returns

The point's name.

Exceptions

If point does not exist, returns an error value.

Example

```
TagName('tag1')
```

TagNum (Tag-based PE function)

Get a point's number from the Point Database.

Syntax

```
TagNum(string)
```

Arguments

string

A tagname in double quotes.

Returns

The point's number.

Exceptions

If point does not exist, returns an error value.

Example

```
TagNum("tag1")
```

TagSource (Tag-based PE function)

Get a point's point source string from the Point Database.

Syntax

```
TagSource(tagname)
```

Returns

The point's point source string.

Arguments

tagname

A tagname.

Exceptions

If point does not exist, returns an error value.

Example

```
TagSource('tag1')
```

TagSpan (Tag-based PE function)

Get a point's span from the Point Database.

Syntax

```
TagSpan(tagname)
```

Arguments

tagname

A tagname.

Returns

The point's span. If the point's type is Digital this is an integer whose value is the number of digital states defined for the point.

Example

```
TagSpan('tag1')  
TagSpan('digitaltag')
```

TagTot (Tag-based PE function)

Find the totalized value (time integral) of a point over a given time, according to values stored in the PI Archive.

Syntax

```
TagTot(tagname, starttime, endtime [, pctgood])
```

Arguments

tagname

A tagname. This point must represent a continuous process flow.

starttime

Beginning of the time range to search. Relative times are relative to endtime, if endtime is not itself a relative time.

endtime

End of the time range to search. Relative times are relative to starttime, if starttime is not itself a relative time.

pctgood

Optional. Minimum time percentage over the given time range, that the point's archived values must be good. For more information about pctgood, see [PctGood \(Tag-based PE function\)](#).

Returns

The point's totalized value over the given time.

Exceptions

If the point has no good values or the **PctGood** minimum is not reached for the given time range, returns an error value.

Note: If the point has few good Archive values during the time period, this function's result may not be trustworthy. Use the **PctGood** function to find out what percentage of the value is good.

Notes

- The system chooses a scale factor such that the integral is correct only if the flow is expressed in units per day. If the flow is expressed in units per hour, or per some other time unit, you must multiply this result by a conversion factor. The conversion factor equals the number of actual flow time units in a day.
- For instance, if you totalize a point measured in gallons per minute, multiply the result of **TagTot** by 1440 to get the answer in gallons. This conversion factor is not related to the time period you are totalizing over; it is strictly a function of the point's engineering units.
- Some PI sites have the default total period configured to be per hour rather than per day. If you are at one of these sites, your conversion factor will differ.
- When the percentage of good data is less than 100%, TagTot determines the total based on good data and divides the fraction of good data in the interval.

Example

```
TagTot('tag1', 'y', '*')
TagTot('tag1', '-1h', '*', 85)
TagTot('tag1', '14-Dec-97', '+1h')
```

TagType (Tag-based PE function)

Get a point's type character (I, R, or D) from the Point Database.

Syntax

```
TagType(tagname)
```

Arguments

tagname

A tagname.

Returns

The point's type character.

Exceptions

If point does not exist, returns an error value.

Example

```
TagType('tag1')
TagType('digitaltag')
```

TagTypVal (Tag-based PE function)

Get a point's typical value from the Point Database.

Syntax

```
TagTypVal(tagname)
```

Arguments

tagname

A tagname.

Returns

The point's typical value. If the point's type is R or I, this is a number; if the point's type is D, this is a digital state (character string).

Exceptions

If point does not exist, returns an error value.

Example

```
TagTypVal('tag1')
TagTypVal('digitaltag')
```

TagVal (Tag-based PE function)

Find a point's Archive value at a given time.

Syntax

```
TagVal(tagname [, time])
```

Arguments

tagname

A tagname.

time

Optional. A time expression. If you omit this argument, '*' is used.

Returns

The archived value of tagname at time. This value is interpolated unless the point has the Step attribute of 1 (or Resolution Code of 4 for PI2).

Exceptions

If point does not exist, or has no archived value at time, returns an error value.

Example

```
TagVal('tag1')
```

- Return of the value of 'tag1' at current time

```
TagVal('digitaltag', 't+11h')
```

- Return of the value of 'digitaltag' at 11am today

```
TagVal('enum_att1')
```

- Return value of enum_att1, an attribute from an enumeration set at current time

```
TagVal('tag1', '23-aug-2012 15:00:00'). - Return of the value of 'tag1' at 3pm on
August 23, 2012
```

- Return of the value of 'tag1' at 3pm on August 23rd, 2012

Note: To check the syntax of the performance equation, use the Pipetest Utility.

[NextEvent \(page\)](#), [NextVal \(page\)](#), [PrevEvent \(page\)](#), [PrevVal \(page\)](#)

TagZero (Tag-based PE function)

Get a point's zero value from the Point Database.

Syntax

```
TagZero(tagname)
```

Arguments

tagname

A tagname.

Returns

The point's zero value. If the point's type is R or I, this is a number; if the point's type is D, this is a digital state (character string).

Exceptions

If point does not exist, returns an error value.

Example

```
TagZero('tag1')
TagZero('digitaltag')
```

Tan

Return the trigonometric tangent of a number.

Syntax

Tan(x)

Arguments

- x

Must be an integer or real number, which represents an angle in radians

Returns

The tangent of x .

Exceptions

If x is not a number, returns an error

Example

- `Tan(1)`
[Returns 1.5574]
- `Tan(1.1)`
[Returns 1.9648]
- `Tan('att1')`
[Return the trigonometric tangent of the value of 'att1' at trigger time]

Tanh

Return the hyperbolic tangent of a number.

Syntax

`Tanh(x)`

Arguments

- x
Must be an integer or real number

Returns

The hyperbolic tangent of x

Exceptions

If x is not a number, returns an error value

Example

- `Tanh(1)`
[Returns 0.76159]
- `Tanh(1.1)`

[Returns 0.8005]

- `Tanh('att1')`

[Return the hyperbolic tangent of the value of 'att1' at trigger time]

Text

Concatenate strings representing argument values.

Syntax

```
Text(x1 [, x2, ... xn])
```

Arguments

- `x1 ... xn`
Any expression of normal PI value type

Returns

A string that is the concatenation of strings representing the argument values.

Example

- `Text(Round('sinusoid', 1), " is the current value of 'sinusoid' at ", '*')`
[Returns rounded current value of 'sinusoid' at current time]

TimeEQ (Tag-based PE function)

Find the total time in seconds, within a range, when a point is equal to a given value.

Syntax

```
TimeEq(tagname, starttime, endtime, value)
```

Arguments

tagname

A tagname.

starttime

Beginning of the time range to search. Relative times are relative to endtime, if endtime is not itself a relative time.

endtime

End of the time range to search. Relative times are relative to starttime, if starttime is not itself a relative time.

This time must be after starttime.

value

Must be an integer or real number or digital state (character string); the value to search for.

Returns

The time period in seconds within the given range, for which the point was exactly equal to the given value.

Example

```
TimeEq('tag1', 't', '*', 40.0)
TimeEq('digitaltag', '-1d', '*', TagVal('digitaltag', '14-Dec-97'))
TimeEq('digitaltag', '14-Dec-97', '*', "On")
```

[TimeGE \(page\)](#), [TimeGT \(page\)](#), [TimeLE \(page\)](#), [TimeLT \(page\)](#), [TimeNE \(page\)](#)

Note: For more information, see TimeGT.

TimeGE (Tag-based PE function)

Find the total time in seconds, within a range, when a point is greater than or equal to a given value.

Syntax

```
TimeGE(tagname, starttime, endtime, value)
```

Arguments

tagname

A tagname.

starttime

Beginning of the time range to search. Relative times are relative to endtime, if endtime is not itself a relative time.

endtime

End of the time range to search. Relative times are relative to starttime, if starttime is not itself a relative time.
This time must be after starttime.

value

Must be an integer or real number or digital state (character string); the value to search for.

Returns

The time period in seconds within the given range, for which the point was greater than or equal to the given value.

Exceptions

None.

Notes

TimeGE interpolates between Archive events, if necessary, to find the times when the point crossed the given value.

Example

```
TimeGE('tag1', 't', '*', 40.0)
TimeGE('digitaltag', '-1d', '*', TagVal('digitaltag', '14-Dec-97'))
TimeGE('digitaltag', '14-Dec-97', '*', "On")
TimeEq (page ), TimeGT (page ), TimeLE (page ), TimeLT (page ), TimeNE (page )
```

TimeGT (Tag-based PE function)

Find the total time in seconds, within a range, when a point is greater than a given value.

Syntax

```
TimeGT(tagname, starttime, endtime, value)
```

Arguments

tagname

A tagname.

starttime

Beginning of the time range to search. Relative times are relative to endtime, if endtime is not itself a relative time.

endtime

End of the time range to search. Relative times are relative to starttime, if starttime is not itself a relative time. This time must be after starttime.

value

Must be an integer or real number or digital state (character string); the value to search for.

Returns

The time period in seconds within the given range, for which the point was greater than the given value.

Exceptions

None.

Notes

TimeGT interpolates between Archive events, if necessary, to find the times when the point crossed the given value.

Example

```
TimeGT('tag1', 't', '*', 40.0)
TimeGT('digitaltag', '-1d', '*', TagVal('digitaltag', '14-Dec-97'))
TimeGT('digitaltag', '14-Dec-97', '*', "On")
```

Note: For more information about PI time, see [PI time abbreviations](#) and [PI time expressions](#).

TimeLE (Tag-based PE function)

Find the total time in seconds, within a range, when a point is less than or equal to a given value.

Syntax

```
TimeLE(tagname, starttime, endtime, value)
```

Arguments

tagname

A PI tag.

starttime

Beginning of the time range to search. Relative times are relative to endtime, if endtime is not itself a relative time.

endtime

End of the time range to search. Relative times are relative to starttime, if starttime is not itself a relative time.
This time must be after starttime

value

Must be an integer or real number or digital state (character string); the value to search for.

Returns

The time period in seconds within the given range, for which the point was less than or equal to the given value.

Exceptions/Errors

None.

Notes

TimeLE interpolates between Archive events, if necessary, to find the times when the point crossed the given

value.

Examples

```
TimeLE('att1', 't', '+1h',80)
```

- [Find the total time between 12:00am and 1:00am today when 'att1' was less than or equal to 80.]

```
TimeLE('att1', '-1h', '*',TagVal('att1', 't+8h'))
```
- [Find the total time in the past hour when the value of 'att1' was less than or equal its value at 8am today. Result is in seconds.]

```
TimeLE('digitaltag', '14-Dec-97', '*', "On")
```

[TimeEq](#) (page), [TimeGE](#) (page), [TimeGT](#) (page), [TimeLT](#) (page), [TimeNE](#) (page)

TimeLT (Tag-based PE function)

Find the total time in seconds, within a range, when a point is less than a given value.

Syntax

```
TimeLT(tagname, starttime, endtime, value)
```

Arguments

tagname

A tagname.

starttime

Beginning of the time range to search. Relative times are relative to endtime, if endtime is not itself a relative time.

endtime

End of the time range to search. Relative times are relative to starttime, if starttime is not itself a relative time. This time must be after starttime.

value

Must be an integer or real number or digital state (character string); the value to search for.

Returns

The time period in seconds within the given range, for which the point was less than the given value.

Exceptions

None.

Notes

TimeLT interpolates between Archive events, if necessary, to find the times when the point crossed the given value.

Example

```
TimeLT('tag1', 't', '*', 40.0)
TimeLT('digitaltag', '-1d', '*', TagVal('digitaltag', '14-Dec-97'))
TimeLT('digitaltag', '14-Dec-97', '*', "On")
```

[TimeEq](#) (page), [TimeGE](#) (page), [TimeGT](#) (page), [TimeLE](#) (page), [TimeNE](#) (page)

TimeNE (Tag-based PE function)

Find the total time in seconds, within a range, when a point is unequal to a given value.

Syntax

```
TimeNE(tagname, starttime, endtime, value)
```

Arguments

tagname

A tagname.

starttime

Beginning of the time range to search. Relative times are relative to endtime, if endtime is not itself a relative time.

endtime

End of the time range to search. Relative times are relative to starttime, if starttime is not itself a relative time. This time must be after starttime.

value

Must be an integer or real number or digital state (character string); the value to search for.

Returns

The time period in seconds within the given range, for which the point was unequal to the given value.

Exceptions

None.

Example

```
TimeNE('tag1', 't', '*', 40.0)
```

```
TimeNE('digitaltag', '-1d', '*', TagVal('digitaltag', '14-Dec-97'))  
TimeNE('digitaltag', '14-Dec-97', '*', "On")
```

[TimeEq](#) (page), [TimeGE](#) (page), [TimeGT](#) (page), [TimeLE](#) (page), [TimeLT](#) (page)

Total (Tag-based PE function)

The **Total** function calculates the sum of two or more values.

Syntax

```
Total(x1, x2, ..., xn)
```

Arguments

x1...xn

May be numbers or time periods, but all must be the same.

Returns

The total of the arguments. The result has the same data type as the arguments.

Exceptions

- Arguments whose run-time values are digital states are not included in the total.
- If all values are digital states, **Total** returns an error value.

Usage

1. Purpose: Use **Total** to sum numeric values, tag data, or time periods.
2. Context: Commonly used in PI Performance Equations, PI DataLink, or PI System Management Tools.
3. Examples:

- Sum current tag values:

```
Total('sinusoid', 'cdt158')
```

- Combine tag values and time periods:

```
Total('tag1', TagVal('tag2', '*-1h'), '+1h')
```

Tips: Ensure all inputs are the same data type to avoid errors.

Examples

The following examples demonstrate how to use the **Total** function in different scenarios, including numeric calculations, tag-based operations, and combining tags with time periods.

- Example 1: Summing numeric values

```
Total(10.5, 20.3, 5.2)
```

Result: 36.0

- Example 2: Combining tags and functions

```
Total('sinusoid', TagVal('cdt158', '*-1h'))
```

Results: Adds the current value of *sinusoid* and the value of *cdt158* one hour ago.

Trim

Trim blanks on both sides of a string.

Syntax

```
Trim(s1)
```

Arguments

- **s1**
string

Returns

The source string with leading and trailing blanks removed.

Exceptions

If *s1* is not a string, an error value is returned.

Example

- Trim(" String ")
[Returns "String"]
- Trim(" String is a string attribute. ")
[Returns "String is a string attribute."]

Trunc

Truncate a number or time to the next lower unit.

Syntax

```
Trunc(x [, unit])
```

Arguments

- **x**
An integer or real number, time expression, or time period.
- **unit**
Optional. The size of the unit to truncate to; *x* will be truncated to a multiple of *unit*. If *x* is a number, *unit* must be a number. If *x* is a time expression or time period, *unit* must be a time period. If *unit* is omitted, **Trunc** truncates to the next lower integer (for a number) or second (for a time period).

Returns

The largest multiple of *unit* that is less than *x*. For a negative *x*, it returns the lowest multiple of *unit* larger than *x*. The return is the same data type as *x*.

Exceptions

If *x* is a string, or if *unit* is of the wrong data type, an error is returned.

Notes

If *x* is a time, and *unit* is omitted, this routine has no effect, as times are only accurate to one second.

When $|x| < |unit|$, 0 is returned.

Example

- `Trunc(12.999)`
[Returns 12, truncated to the next lower integer]
- `Trunc(28.75, 10)`
[Returns 20, truncated to next lower multiple of 10]
- `Trunc('14-Dec-16 11:47', '+1h')`
[Returns 12/14/2016 11:00:00 AM, truncated to next lower hour]
- `Trunc('18:47'-'15:00', '+1h')`
[Returns 03:00:00, truncated period to next lower hour]

Note: Truncating to the next lower day results in a timestamp of the next lower day in UTC time, not local time.

UCase

Convert a string to an uppercase string.

Syntax

`UCase(s1)`

Arguments

- **s1**

String in double quotes

Returns

s1 in uppercase

Exceptions

If the argument is not a string, returns an error value.

Example

- `UCase("String")`
[Returns "STRING"]

Weekday (Tag-based PE function)

Extract the day of the week from a given time expression.

Note: The days of the week are represented in a range from 1-7, where 1 represents a Sunday.

Syntax

```
Weekday(t1)
```

Arguments

t1

A time expression, enclosed in single quotes

Returns

The day of the week, in the range 1-7, where 1 represents a Sunday.

Exceptions

None

Example

```
Weekday('*')
```

- [Return what day of the week today is.]
`Weekday(FindEq('att1', '-7d', '*', 50))`
- [Return what day of the week it was when the value of 'att1' was 50 for the first time in the past 7 days.]

Year (Tag-based PE function)

Extract the year from a time expression.

Syntax

```
Year(t1)
```

Arguments

t1

A time expression, enclosed in single quotes.

Returns

The year of time, in the range 1970-present.

Exceptions

None.

Example

```
Year('*')
```

- [Return what year it is now.]
`Year(FindGT('att1', '1/1/1970', '*', 50))`
- [Return what year it was when the value of 'att1' was first greater than 50 since 1/1/1970.]

Yearday (Tag-based PE function)

Extract the day of the year from a time expression. The day of the year (also known as a Julian day) is an integer ranging from 1 to 366, where 1 represents January 1.

Syntax

```
Yearday(t1)
```

Arguments

t1

A time expression, enclosed in single quotes.

Returns

The day of the year of time, in the range 1-366, where 1 represents January 1.

Exceptions

None.

Example

```
Yeaday('*')
Yeaday('t')
```

[Day](#) (page), [DaySec](#) (page), [Hour](#) (page), [Minute](#) (page), [Month](#) (page), [Second](#) (page), [Weekday](#) (page), [Year](#) (page)

PI Services

Use the PI Services tool to view, configure, and stop and start PI services for each connected Data Archive server. By default, the status of each service is updated every 30 seconds; you can change this refresh rate. You can also view the status, errors, and thread details for services used by the connected Data Archive server and export a list of PI Services.

To open the PI Services Tool, from the System Management Tools pane, select **Operation > PI Services**.

View PI services

Each PI service and its status is listed by service name in the PI Services tool. One of these statuses is listed next to each service name:

- **Running**: the service is running
- **Stopped**: the service is not running
- **Unknown**: the service status is unknown.

Use the check boxes next to each service to start or stop the service, configure the startup type, or export a list of services.

You can view thread details in the **Thread Details for Selection** pane. Use the Session Record to view any errors that occur when service statuses are retrieved.

PI process list

All PI processes listed in **pisrvstart.bat** and **pisrvsvrappsstart.bat** files are monitored for each connected Data Archive server. Processes listed in **pisrvsitestart.bat** in either of the following formats are also monitored:

- **net start ServiceName**
- **rem net start ServiceName**

These files are located in the PI home directory under the **\adm** directory, for example:

```
c:\PI\adm\pisrvstart.bat  
c:\PI\adm\pisrvsvrappsstart.bat  
c:\PI\adm\pisrvsitestart.bat
```

Note: The **pisrvstart.bat** and **pisrvsvrappsstart.bat** files should not be modified.

Add processes

To add, run and monitor existing PI processes that are not already part of the core PI system, you can edit the **pisrvsitestart.bat** file. Processes added to this file are started automatically when **pisrvstart.bat** is used to start Data Archive. When a process is included in this file, the process is monitored by PI Services manager, provided it is configured to run as a Windows service.

Processes may be added in one of the following formats:

- **net start ServiceName**, where *ServiceName* is the name of the service to be monitored. For example, the PI Performance Monitor Interface basic version would use the syntax:

```
net start piperfmon_basic
```
- **net start DisplayName**, where *DisplayName* is the display name of the service to be monitored. For example the same interface would use the syntax:

```
net start "PI-Performance Monitor Interface (basic version)"
```
- **Note:** Double quotes are required for **pisrvsitestart.bat** to function properly.

You may also add **rem** to either format to prevent automatic startup of the service. For example:

```
rem net start piperfmon_basic
rem net start "PI-Performance Monitor Interface (basic
version)"
```

The following is an example **pisrvsitestart.bat** file:

```
REM Non-Interactive Site Specific startup file. This file
REM should be modified to start site specific services
REM related to PI. This file will not be overwritten
REM on upgrade. Instead new versions will be written
REM as pisrvsitestart.bat.new for review and integration
REM by the PI Administrator.
REM $Workfile: pisrvsitestart.bat $ $Revision: 11$

echo Starting Site Specific PI System Services...
net start rmp_sk
net start random

rem
rem The following interfaces are not started by default.
rem Remove the remarks "rem" below to start the interfaces.
rem
rem net start piperfmon_basic
rem net start piping_basic
rem net start pisnmp_basic
rem net start pibagen
rem net start pirecalc
:theend
```

View thread details

Use thread detail information to tune thread usage, based on the information that indicates which subsystems are at maximum capacity and which are idle. Thread details are also useful for preventing or troubleshooting issues that involve PI Services.

Note: PI SMT does not provide thread information for Data Archive 3.4.395 and later versions.

1. Under **Collectives and Servers**, select the server or collective.
2. In the System Management Tools pane, select **Operation > PI Services**.
3. Select a service or a pisrvsitestart.bat icon in the PI Services tool.

Available thread information appears in the Thread Details pane. Here you can:

- Right-click on the heading row to include or remove columns that display in the Thread Details pane.
-
- Note:** Some subsystems do not currently expose thread information.
-
- Right-click an RPC thread and select **Suspend** to suspend an RPC thread or **Resume** to resume the tracking of RPC details or refresh the thread list.
 - Click  or right-click and select **Refresh** to the thread list information for the currently selected subsystem.

Determine service startup type

To view or change the service startup type:

1. Under **Collectives and Servers**, select the server or collective.
2. In the System Management Tools pane, select **Operation > PI Services**.
3. Right-click the service in the PI Services list.
4. Select a startup type from the available options.

Note: Toolbar and right-click options are enabled or disabled, depending on the service status.

Service startup types

The service startup type determines how a process is run:

- **Auto**

Specifies that a service starts automatically with the operating system at each system startup. If an automatically-started service depends on a manual service, that service is also started automatically.

- **Manual**

Specifies that a service is only started manually by a user through the Windows Service Control Manager, or by an application.

- **Disabled**

Indicates that the service has been disabled and cannot be started by a user or application, unless its startup type is changed to manual or auto.

- **Set All to Auto**

Specifies that the selected services start automatically with the operating system at each system startup. If an automatically-started service depends on a manual service, that service is also started automatically.

- **Set All to Manual**

Specifies that the selected services are only started manually by a user through the Windows Service Control Manager, or by an application.

- **Set All to Disabled**

Indicates that selected services have been disabled and cannot be started by a user or application, unless its startup type is changed to manual or auto.

Start PI services

1. Under **Collectives and Servers**, select the server or collective.
2. In the System Management Tools pane, select **Operation > PI Services**.
3. Select the service in the PI Services tool and click .

Note: Toolbar and right-click options are enabled or disabled, depending on the service status.

Stop PI services

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Operation > PI Services**.
3. Select the service in the PI Services tool and click the **Stop** button .

Note: Toolbar and right-click options are enabled or disabled, depending on the service status.

PI Service display options

To configure how the PI Services tool display, click the **Options** button . The Options dialog appears. Use the fields in the Options dialog for these settings:

- **Error Messages**

Select **SUPPRESS error messages about PI services that do not exist** to stop messages from being logged when a PI service is not found on a newly-connected Data Archive server.

- **Update Rates**

- Select **Update service statuses every:** and enter the frequency, in seconds, that the list of PI Services will automatically update.
- Select **Update thread information every:** and enter the number of seconds that will lapse between the time a user initiates a **Stop** and the process to stop the service will begin.

- **Services**

- Configure whether **Start AllServices** is selected in the **PI Services** tool will start services set to startup type **Automatic** only, or both **Automatic and Manual**.
- Enter the wait time, in seconds for the **PI Services** tool display to reflect status changes for services that are started and stopped.

- **Thread information options**

Select **Do not query for additional information** to allow the tool to retrieve and display connection and point information for the thread requests. To view this information move the cursor over an item Thread Details for Selection pane:

Thread Details for Selection									
Server	Subsystem	Time Retrieved	PoolName	ID	Handle	Priority	State	Calls	
XPISEVER	pimsgss	13-Jul-09 16:11:12.556	RPC	3504	8644	0	Wait	39166	
XPISEVER	pimsgss	13-Jul-09 16:11:12.556	Main	3496	4294967294			8584293	
XPISEVER	pimsgss	13-Jul-09 16:11:12.556	Message	3508	8648			8729163	
XPISEVER	pimsgss	13-Jul-09 16:11:12.556	Read	3512	8648			79956	
XPISEVER	pilicmgr	13-Jul-09 16:11:12.556	RPC	3580	8640	0	Wait	10965	
XPISEVER	pilicmgr	13-Jul-09 16:11:12.556	RPC	3584	8644	0	Wait	10346	
XPISEVER	pilicmgr	13-Jul-09 16:11:12.556	RPC	3588	8648	0	Wait	10485	
XPISEVER	pilicmgr	13-Jul-09 16:11:12.556	Connection: pilogsrv.exe	3592	8652	0	InUse	10500	

- **Logging**

Use these settings to determine which information is included in log files:

- Everything
- In use threads
- System threads
- Thread pools: Select Replication, Flush, RPC or EVQ

Note: The thread activity log is a local Comma-Separated Value (CSV) file stored locally for use with thread information. It is automatically updated, but is not part of the PI Message or Windows log files.

Export a list of PI services

To export a list of processes selected in the PI Services tool to a Comma-Separated Value (CSV) file:

1. Under **Collectives and Servers**, select the server or collective.
2. In the System Management Tools pane, select **Operation > PI Services**.
3. Click the **Export** button .
4. Specify the filename and folder location and click **Save**.

Note: To export all services on a specified server, select the Data Archive server rather than an individual service.

Remote login

The PI Services manager uses the remote login features of the piartool utility to determine when the base PI services have started up.

If a password is needed, you are prompted to enter a password for each connected server when accessing the PI Services tool.

Note: The PI Services tool stores passwords only during the current SMT session.

PI services quick reference

Note: Toolbar and right-click options are enabled and disabled based on the item and its current status.

Goal	Right-Click Option	Toolbar Option
Start the selected PI process on the selected server, or use to resume a paused thread.	Start Service	
Start all PI processes on the selected server. Start the base PI subsystems on the selected server (PI Network Manager, PI Message Subsystem, PI Update Subsystem, PI Base Subsystem, PI Snapshot Subsystem, PI Archive Subsystem, PI Shutdown Subsystem)	Start All or Start Base Services	
If the selected entry in the process tree is a PI process, then this means: Stop the selected PI process on the selected server. If the selected entry in the process tree is a Data Archive server, then this means: Stop all PI processes on the selected server.	Stop Service or Stop All Services	
Toggle ON or OFF to log the thread information.	N/A	
Expand list of PI processes under each listed server	N/A	
Collapse list of PI processes under all listed servers	N/A	
Configure Point Builder display options	N/A	
Export the service list to a CSV file		
Refresh the selected server's PI process statuses or refresh the selected PI process's status	Refresh Service or Refresh All	
Launch the online Help	Help	
Set the selected PI Process to be started by the operating system, at system start-up.	Auto	N/A
Set all PI Processes on the selected server to be started by the	Set All to Auto	N/A

Goal	Right-Click Option	Toolbar Option
operating system, at system start-up.		
Set the selected PI process to be started only manually, by a user (using the Service Control Manager) or by an application.	Manual	N/A
Set all PI process on the selected server to be started only manually, by a user (using the Service Control Manager) or by an application.	Set All to Manual	N/A
Set all PI process on the selected server to be disabled	Disabled	N/A

PI Version

The PI Version tool lists complete version information for Data Archive subsystems on all connected Data Archive servers. Subsystems are listed by server. Columns display the following information.

- Server name or IP address
- Collective name, if applicable
- Subsystem name
- Version in memory
- Most recent startup time
- Version on disk
- System timeout date, if applicable
- Extended information, if applicable

Ping

Use the Ping tool to view, create, and manage PI Ping points. PI Ping points store data about the response times of Internet Control Message Protocol (ICMP) echo requests, or *pings*, between a computer where the PI Ping Interface runs and any remote network computer. You can use PI Ping points to measure latency of a TCP/IP network. PI Ping points can be useful when you diagnose network connection problems.

We provide the PI Ping basic interface with every Data Archive; PI Ping Interface is sold separately and allows more points. For more information about PI Ping interfaces, see the [Customer Portal](#).

Managing PI Ping points

The **Point Browser** opens when you select the PI Ping tool. Use this browser to create new PI Ping points for connected Data Archive servers and interfaces and edit the properties of PI Ping points. See [View and edit PI Ping properties](#).

To select PI Ping points in the **Point Browser**, select a Data Archive server or PI Interface node and press Alt to select points under a selected interface or server. Or, enter Ctrl- or Shift-click to select multiple points.

You can use the PI Ping tool to add PI Ping interfaces and PI Ping points to the list as needed. See [PI Ping interfaces](#) and [PI Ping point configuration](#).

Note: Interfaces added to the Point Browser with the PI Ping tool have a purple icon and configuration information stored on the local machine (See [PI Ping interfaces](#)). Interfaces are loaded in the **Point Browser** only when the same Windows user account is used to run PI SMT. Interfaces managed through the PI Interface Configuration Utility (PI ICU) have a blue icon.

PI Ping interfaces

There are three ways that interfaces are added to the **Point Browser**:

- The **Point Browser** automatically loads PI Ping Interfaces that are managed by PI ICU and displays a hierarchy of interfaces and PI Ping points.
- You can also add interface nodes if you:
 - Use the Enter Interface Information dialog provided by the **Point Browser** to *manually* enter the interfaces, or
 - [Create an interface list batch file](#) to import a list interfaces to the **Point Browser**.

To add an interface:

- Right-click on a Data Archive node, select **Add Ping Interface**, and one option:
 - Select **Manually Enter** to enter the required information, including **Interface Name**, **Point Source** and **Interface ID** into the **Enter Interface Information** dialog.
 - Select **Open Configuration File** to open a batch (.bat) file containing the information needed to configure the Ping interface.

Note: When an interface is added, PI Ping points belonging to the interface are automatically retrieved by and visible in the **Point Browser**.

To display configuration information for an interface:

- Right-click on the interface node and select **Interface Information**.

Note: Scan classes are displayed only for interfaces managed by the **PI ICU**.

You can remove interfaces that users added to the **Point Browser** with the PI Ping tool:

- Right-click on the interface node and select **Remove Interface**.

PI Ping point configuration

You can create and update PI Ping points with the PI SMT Point Builder tool, however, the PI Ping tool offers a convenient way to:

- View the interfaces that use PI Ping points
- View configuration information about PI Ping Interfaces
- Create, update and manage PI Ping points for those interfaces
- Copy or move those points between interfaces.
- Add PI Ping interfaces that are not automatically added to the **Point Browser**

Note: You cannot manage points on secondary Data Archive server in a collective.

Best practices for PI Ping points

We recommend that you configure a PI Ping point for:

- Data Acquisition nodes
- Major network equipments, such as switches, routers and so on
- Internal application servers, such as mail servers, ERP servers, RDBMS, and so on
- External computers, such as those used on the Internet, or in remote offices

Configure default point properties

Each new PI Ping point is created with default point properties so that standard point information does not need to be entered for each new point. Before you can create new PI Ping points, you have to configure these default point properties.

Default point properties are editable and stored on the local Data Archive server for each Windows user. For a list of what values can be used, see [Point attribute values for PI Ping](#).

1. Under Collectives and Servers, select the server or collective.
2. In the System Management Tools pane, select **IT Points > Ping**.

3. Click **Default Point Properties**  on the toolbar.
4. Change the properties displayed in the Edit Default Point Properties dialog box.
 - The fields available match those provided in the Point Property pane . When entering values, the field background turns yellow when an invalid character is entered.
 - Press **Enter** to save or **Esc** to cancel all changes.
 - Click the arrow to access the menu of options for properties that have a set number of choices, such as Point Type.
 - Use the **Tag** field to change the default format of automatically generated points. By default, PI Ping points use this format:

• where:
 - *Prefix* is the prefix used in the name of new Ping points
 - *Ping Source* is the name of the Ping source, either the Data Archive name or interface node name. You can manually enter this value.
 - *Ping Target* is the name of the Ping target, either the host name or the IP address of the remote machine.
 - *Delimiter* is the delimiter used to separate each element in the name, which must be a single character valid for the point name.

To change the format of the Tag default property:

- a. Click the arrow next to the Tag field.
- b. Select the field for each default property formats and edit your changes. When entering values, the field background turns yellow when an invalid character is entered.

You can configure whether the host name or IP address of the remote machine should be shown in the InstrumentTag property of new PI Ping points. To change this setting, select the **Ping Target** field and use the drop-down menu to select **Host Name** or **IP Address**.



- c. Press Enter to save or Esc to cancel all changes.
- d. Click the arrow again to close the dialog box.
5. Click **Reset** to set all values to match the system default, if necessary.
6. Make other desired changes and click **OK** to save them to the local machine, or click **Cancel** to discard all changes.

Note: Point properties that differ from default values are shown in bold font.

For more information, see [PI point access permissions](#).

Point attribute values for PI Ping

You can use these values for PI Ping point attributes:

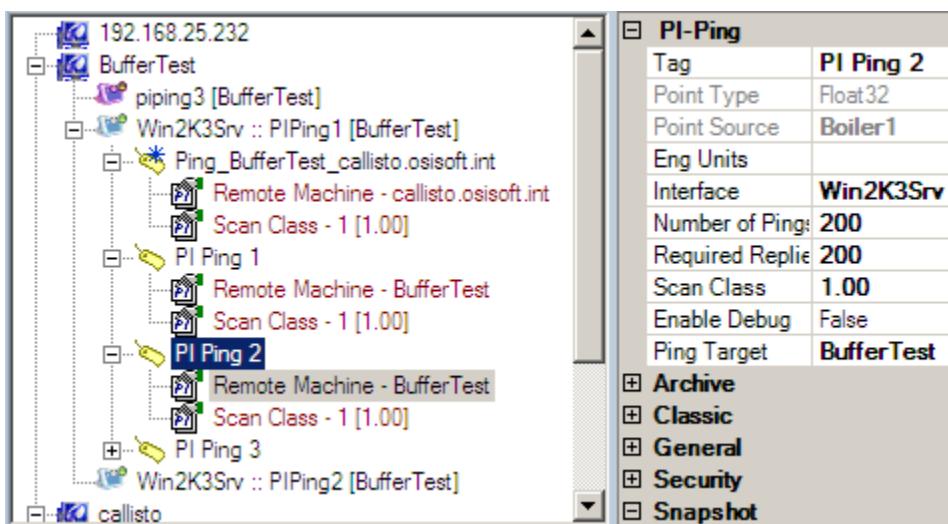
- **Point Type:** Float16, Float32, Float64, Int32, Int16
- **Point Source:** the PointSource parameter, as specified in the interface startup file, or in the Interface Configuration Utility (ICU); the default is J
- **InstrumentTag:** remote host name or IP address of the machine that the ping is sent to
- **Location1:** Interface ID number
- **Location4:** Scan class number

We recommend you use the default values for these settings:

- **Step:** 1
- **Span:** 10000
- **Engunit:** ms

Create PI Ping points

1. Under Collectives and Servers, select the server or collective.
2. In the System Management Tools pane, select **IT Points > Ping**.
3. Select the desired interface node in the tree.
4. Click **New Point** on the toolbar, or right-click the interface and select **Add New Point**.
A new node is created under the selected interface.
5. Enter a name for the new PI Ping point now selected in the tree, such as a host name or IP address of the ping target.
6. Select the new point in **Point Browser** to view or edit the point's properties, or click + to view the name of the ping target and the target's scan classes:



Note: To create PI Ping points for interfaces that you add to the Network Node List, see [Add points from the](#)

network node list.

Copy or move PI Ping points

You can also cut and paste PI Ping points between interfaces that belong to stand-alone and primary servers.

1. Under Collectives and Servers, select the server or collective.
2. In the System Management Tools pane, select **IT Points > Ping**.
3. Select the desired PI Ping points in the tree. Ctrl- or Shift-click to select multiple points. Choose **Select All** to select all points in the tree.
4. Right-click the selected points and select **Cut** or **Copy**, or use standard key commands to cut and copy the selected points.
5. Select the target interface where you want to place the points.
6. Right-click and select **Paste**.

Rename a PI Ping point

1. Under Collectives and Servers, select the server or collective.
2. In the System Management Tools pane, select **IT Points > Ping**.
3. Right-click the point and select **Rename**.

Delete a PI Ping point

To permanently delete a PI Ping point:

1. Under Collectives and Servers, select the server or collective.
2. In the System Management Tools pane, select **IT Points > Ping**.
3. Select the point and click **Delete** .

Search for PI Ping points

1. Under Collectives and Servers, select the server or collective.
2. In the System Management Tools pane, select **IT Points > Ping**.
3. Click the arrow next to **Find Next**  and select **Advanced Search** from the resulting menu.
4. Enter the desired search criteria in the Search Points dialog box and select **Find Next** to locate matching points, or **Select All** to select all points in the tree that match the search criteria.

View and edit PI Ping properties

When you select a point in the **Point Browser**, the Point Property pane is populated with the point's properties. Use the Point Property pane to view and edit detailed information for a PI Ping point selected in the **Point**

Browser.

Use the Point Property pane to:

- Edit properties for one or multiple points selected in the **Point Browser**. The number of points selected appears in the toolbar.
- Select a property to display its description at the bottom of the pane.
- Click in a cell containing a property value to edit the value. Grayed values are not editable.

Note: You cannot edit properties of points belonging to a secondary Data Archive server. Some properties are not editable when multiple points are selected.

- Click **Sort**  to sort point properties in alphabetical order, or click **Categorize**  to order them categorically.
- Right-click a property and select an option from the context menu to **Reset** the property to its default value, or to hide/display the property **Description**.

Note: Point properties that differ from default values are shown in bold font.

Import lists into the PI Ping tool

If you have PI Ping points to use with a large number of interfaces, you can use the PI Ping tool to create and import batch or Comma-Separated Value (CSV) files that list those interfaces. You can create both types of these files using the PI Ping tool.

Files are created, edited, imported and exported in the **Network Node List** of the PI Ping tool.

Use the **Show**  and **Hide**  buttons to show or hide the **Network Node List**. **Hide**  minimizes the **Network Node List** to a title bar at the bottom of the **Point Browser**. Click **Show**  , or double-click the title bar to expand the list.

Create an interface list batch file

To add a list of PI Ping Interfaces to the **PI Ping** tool, you can create or edit a batch file and use it to add the interfaces with the **Point Browser**.

Use the **Network Node List** to create the file, or [Import an interface list batch file for editing](#).

You can add nodes from the **Network Node List** to selected PI Ping interfaces in the **Point Browser**.

To add entries to the **Network Node List**:

1. Click the **Add New Node** button  or double-click an empty row to add a node to the **Network Node List**.
Note: The names of new points are created automatically based on the default point properties. See [Configure default point properties](#).
2. Click each entry in the list to edit:
 - Host name
 - IP address
 - Tag value of the node
3. Select **DNS** to resolve host names or IP addresses using the default DNS server. If an address fails to resolve,

an error message appears in the **Note** column.

4. Select **Ping Test** to measure the ping value of each network node in the list and display the value in the **Ping** column.
5. Enter a value in the **Timeout** field to set the time that ping measurement will allow before timing out. The default value is **200ms**.

To remove nodes: You can add nodes from the Network Node List to selected PI Ping interfaces in the Point Browser.

6. Optional: Click the **Delete** button  to remove a selected node.

Import an interface list batch file for editing

Use the **PI Ping** utility to import and then edit an existing batch file that lists PI Ping Interfaces.

1. Click the **Import** button  in the **Network Node List**.
2. Browse to the file location and select the file.
3. Click **Open**.
4. After the file is loaded into the **Network Node List**, edit the file according to the procedures in [Create an interface list batch file](#).

Create a CSV file interface list

You can create a list of PI Ping Interface nodes in a CSV file and import the list into the PI Ping utility.

1. Click the **Import** button  in the **Network Node List**.
2. Browse to the file location and select the file.
3. Click **Open**.
4. After the file is loaded into the Network Node List, you can edit the file according to the procedures in [Create an interface list batch file](#).

Export the network node list to a CSV file

You can export a list of PI Ping interface nodes to a CSV file for import elsewhere.

1. Click the **Export** button  in the **Network Node List**.
2. Browse to the directory where you want to save the file.
3. Click **Save**.

Add points from the network node list

To create PI Ping points, you can select the nodes you want to create PI Ping points for in the **Network Node List**, then either:

1. Select the nodes you want in the **Network Node List**, and then either:
 - Select the desired network nodes in the **Network Node List**, select an interface in the **Point Browser**, and then click **Create** .
 - Drag selected nodes and drop them onto the desired interface directly.
 - Drag selected nodes from the **Point Browser** to the **Network Node List**.
2. Click **Save**  to save the points on the Data Archive server.

Note: You cannot use duplicate names for PI Ping points that exist on the same Data Archive server.

Point Builder

Use the Point Builder tool to view, create, and edit PI points. This tool allows you to easily search for existing points, configure new points, and manage various point attributes. Using Point Builder, you can view a list of PI points along with key information, such as Server, Point name, Point Source, and Data Security.

The subsequent topics in this section cover specific aspects of point management like configuring attributes, performing searches, and adjusting security settings.

View PI points

Use **Tag Search** to find PI points, then use the Point Builder to view a list of PI points and the Server, Point name, Point Source, Point Class, Extended Descriptor, Point Security and Data Security for each point.

To clear the list of points in the Point Builder, right-click on the point and select **Clear List**.

To delete a PI point, right-click on the point you want to delete, and select **Delete PI Point**.

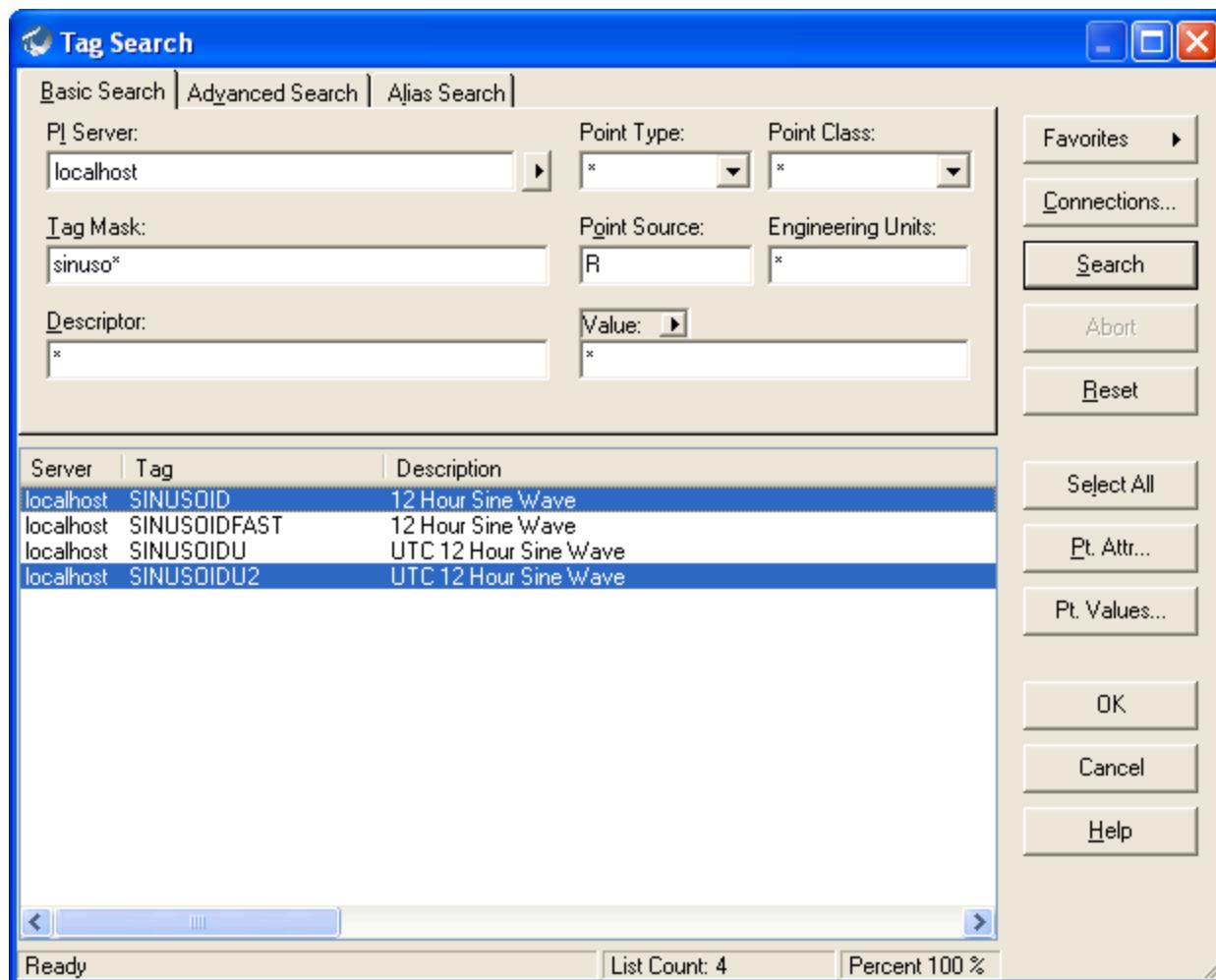
When a point is deleted, the Tag name, PointID and RecNo are logged to the local and server log files, as well as the name of the user who is deleting the point and the machine on which PI SMT is run.

Use Tag Search

Click the **Tag Search** button  to open the Tag Search dialog box.

The Tag Search dialog box provides three types of searches for historical or future PI points:

- **Basic Search** allows you to create a tag mask by specifying PI point attributes. The mask is used to find a list of PI points on the server with matching attributes.
- **Advanced Search** provides a query-building interface with access to more point attributes for complex searches.
- **Alias Search** provides a logical tree view of a Data Archive through the PI Module Database, which you can use to select tags by their descriptive aliases.



1. Click a tab to choose a **Basic**, **Advanced** or **Alias** search.
2. Enter the required search criteria and click **Search**.

Use * or ? as wildcard characters to search for tag names and attributes. For example, the tag mask *Tem** returns all point names that start with *Tem* while *Tem?** returns only points that start with *Tem* and end with another single character. All point mask fields are case insensitive.

You can also click **Favorites** to access previous searches.

3. Tags returned from a search appear listed in a search results panel. Select the desired tags in the results panel, and click **OK**.

Click column headers in the search results panel to sort the results. Press Ctrl+click or Shift+click to select multiple tags.

Configure point attributes

To configure the general attributes of a PI point, see General attributes.

To specify how PI point data is stored in the Data Archive server, see Archive attributes.

General attributes

To configure the Base Point Class attributes of a PI point, click the **General** tab. Type a point name in the **Name** text box on the **General** tab and edit the point.

Name — Enter a name that uniquely identifies the point. Names can contain letters, numbers, certain characters, and spaces. Names have the following constraints:

- The first character must be alphanumeric, an underscore (_), or a percent sign (%).
- Control characters, such as linefeeds or tabs, are not allowed.
- The following characters are not allowed:

* ' ? ; { } [] | \ ^ ' "

Though you can specify any length name, some API functions truncate names to 12 or 80 characters, and PI SQL Subsystem can only process tags with at most 1016 characters.

Note: The name text appears blue if the point does not exist on the selected Data Archive server. The text appears black if the point exists, or a server is not selected, preventing validation.

PI Server — Select a Data Archive server from the list of connected servers under **Collectives and Servers**.

Descriptor — The descriptor field is optional.

Point Class — Select a point class from the list of point classes that belong to the server listed in **PI Server**.

Stored Values — Select "Real-time data" while configuring a historical PI point for real-time data, and "Future data" when configuring a future PI point for forecast data.

Point Source — Specify a point source. A PointSource is an attribute that commonly specifies which PI Interface is collecting the data for the PI Point.

Point Type — Select the point type for the point. A Performance Equation can use Point Types Digital, Float64, Float32, Int32, Int16 and String.

Digital Set — Choose a digital set from the Data Archive server. This menu is activated only if the selected Point Type is Digital. The System digital set is otherwise used.

Engineering Units (Eng Units) — Enter optional engineering units to describe the default unit of measurement for the point.

Display Digits — Enter an optional number of display digits to determine the display precision of point values.

Extended Descriptor (Exdesc) — Enter an optional extended description for the point.

Source Tag — Enter a reference to the tag name of another PI point to use that point's data stream as the source of the new PI point. A source tag is not required to create a point, but can be supplied. Click  to display the **Tag Search** dialog box to search for points.

Note: The **Source Tag** attribute cannot be used to replicate data directly from one point to another.

Archive attributes

To specify how PI point data is stored in the Data Archive server, click on the **Archive** tab and enter values.

Typical value — Enter a reasonable sample value for the point. For a numeric tag, a typical value must be greater than or equal to the Zero, and less than or equal to the Zero plus the Span.

Zero — Enter the lowest possible value for the point, as the bottom of the range used for scaling float16 values

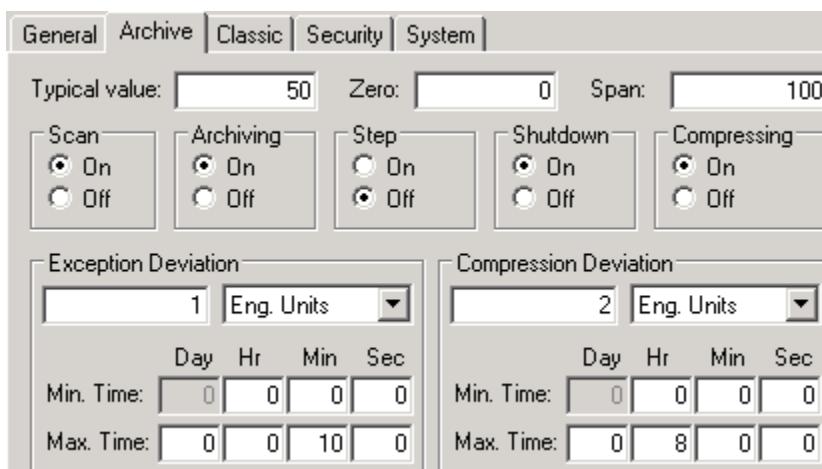
and trends. Recorded values lower than the Zero value are recorded in the Data Archive server with the digital state Under Range.

A Zero value is required for numeric data type points, but not for non-numeric points. The instrument zero is a logical zero value, and certain interfaces require that Zero and Span values match the instrument system range. See your interface documentation for details.

Span — Enter the maximum difference between the top and bottom of the point range, as a positive value. The Span value is used for scaling float16 values and trends and is required only for numeric data type points. Recorded values above the sum of the zero and Span values are recorded in Data Archive server with the digital state Over Range.

Scan — Enable Scan if an interface that supplies data to the point requires a scan flag. Interfaces that require a scan flag do not update points when the flag is disabled. See your interface documentation for Scan attribute requirements.

Archiving — Enable Archiving to store point values in the Data Archive server.



Step

Enable the Step attribute to treat archived point values as discrete units that would appear stepped on a chart. An archived value is assumed constant until the next archived value, and adjacent archived values are not interpolated. For example:

- at 12:00:00, the value 101.0 is archived
- at 12:01:00, the value 102.0 is archived
- a request for the value at 12:00:30 returns 101.0

Disable Step to treat archived point values as a continuous signal, and linearly interpolate adjacent values. For example:

- at 12:00:00 the value 101.0 is archived
- at 12:01:00 the value 102.0 is archived
- a request for the value at 12:00:30 returns 101.5

Note: By default, future PI points have their step attribute turned on (set to 1). This is because most future PI Points store discontinuous signals (for example, a series of discrete predictions) for which linear interpolation

cannot be assumed. With the step attribute turned on, trends show a staircase pattern. Future PI points, by default, also have compression, excmin, excmax and excdev turned off (set to 0).

The Step attribute also affects compression calculations. When enabled, a linear change of value greater or equal to the compression deviation allows the point to pass compression.

The Step attribute is relevant only to numeric points. In general, data from continuous signals such as thermocouples and flow meters should be archived without stepping. Data from discrete measurements such as lab data and batch charge weight should be stepped.

To access the Step attribute setting:

1. **Open** PI SMT.
2. **Navigate** to Points > Point Builder.
3. **Search** for and **select** the point for which you would like to check the step attribute.
4. In the Archive tab, you can **view/change** the step attribute.

Shutdown

Use the Shutdown attribute to determine whether shutdown events are written. The timestamp of Shutdown events normally represents the actual shutdown time of the Data Archive server as recorded by the Snapshot Subsystem.

Beginning with PI Server PR1 SP1, shutdown events for most points are disabled. Unless you configure points to receive shutdown events, only test points such as sinusoid and sinusoidu will receive shutdown events.

- Enable Shutdown to record server shutdown events as point values with the timestamp of the server shutdown. If data is collected on the Data Archive server, shutdown events are helpful to clearly indicate gaps in data collection.
- Disable Shutdown when point data is collected on a distributed collection nodes, as buffering services collect, manage and retain the data until the server is up and running again. Disabled Shutdown PI points have a configurable attribute to determine whether shutdown events are written.

Note: The default behavior of Data Archive is to write the SHUTDOWN digital state to all PI points when Data Archive is started. The timestamp that is used for the SHUTDOWN events is usually updated every 15 minutes, which means that the timestamp for the SHUTDOWN events will be accurate to within 15 minutes in the event of a power failure.

Compression

Enable compression to apply compression algorithms and save only event values that indicate a change in point value. When disabled, all values collected and sent to the snapshot, including redundant values, are saved in the PI archive.

Note: By default, future PI points have *compression* turned off (set to 0).

Enable compression for all real-time points. Digital, BLOB and string data type values pass through compression only when the value changes. Compression is typically turned off for points that collect sample data, such as lab data, or other sparse data streams.

Deviation

Exception and compression deviation algorithms can be used to filter erroneous and redundant data before it is recorded in the archive.

Note: By default, future PI points have *excmin*, *excmax* and *excdev* turned off (set to 0).

Each algorithm is based on three specifications:

- Deviation in point value
- Minimum duration of time
- Maximum duration of time
- These specification are used to determine which points are recorded to the snapshot from the buffer (Exception) and to the archive from the snapshot (Compression).

Note: Exception deviation and compression deviation filter externally-generated events for archiving. Under no circumstances does this cause Data Archive to generate events.

Compression deviation

Once events are sent to the PI Snapshot subsystem, a compression algorithm can further filter data and reduce storage to only significant values as they are moved into the archive. An event is recorded:

- After a specified minimum duration of time since the previous event, if it exceeds a specified deviation in value from that event.
- After a specified maximum duration of time since the previous event.
- When activated, compression reporting filters events and stores only periodic values (including duplicates), unless an event represents a significant change in the short-term trend of values.

To turn off compression and archive every event that passes exception reporting, disable the *Compressing* attribute.

Compression Deviation — Enter the deviation in value required to record an event, either as a number of engineering units, or as a percentage of the point's *Span* value.

For most flows, pressures, and levels, use a deviation specification of 1 or 2 percent of *Span*. For temperatures, the deviation should usually be 1 or 2 degrees.

Min Time — Enter the minimum time that must elapse after an event before a compressed value can be recorded. The minimum time should be set to 0 if exception reporting is activated for the same point.

Max Time — Enter the maximum time that can elapse after an event before automatically recording the next event as a compressed value. The recommended maximum time is one work shift (for example, 8 hours). If this value is too low, the compression effects are too limited to save significant archive space. If this value is too high, useful data may be lost. Events that reach the Data Archive server in asynchronous order bypass the compression calculation and are automatically recorded to the archive.

The compression specifications consist of a deviation (*CompDev*), a minimum time (*CompMin*), and a maximum time (*CompMax*).

Events are also archived if the elapsed time is greater than the maximum time. Duplicate values will be archived if the elapsed time exceeds *CompMax*. Under no circumstances does this cause Data Archive to generate events; it only filters events that are externally generated.

The most important compression specification is the deviation, *CompDev*. For non-numeric tags, *CompDev* and *CompDevPercent* are ignored. They will be displayed by applications as zero.

Note: *CompDev* specifies the compression deviation in engineering units; *CompDevPercent* specifies the compression deviation in percent of *Span*. If you change one, the other is automatically changed to be compatible. If you try to change both at once, *CompDevPercent* takes precedence.

Exception deviation

Exception reporting is used to define the precision of a data stream, and the amount of deviation that constitutes a significant change. Most interface programs can execute an exception-reporting algorithm to determine when to send a point value to the PI Snapshot subsystem. An exception is an event that occurs either:

- After a specified minimum duration of time since the previous event, while exceeding a specified deviation in value from that event.
- After a specified maximum duration of time since the previous event.

This means that when activated, exception reporting filters events and stores only periodic values, including duplicates, unless an event represents a significant change in the short-term trend of values. An exception event, both timestamp and value, is sent with the previous event to the Snapshot.

- **Exception Deviation**

The deviation in value required to store an event, either as a number of engineering units, or as a percentage of the point's *Span* value. The exception deviation should be less than the compression deviation by at least a factor of 2, and is ignored for digital, string and BLOB data type points.

- **Min Time**

The minimum time that must elapse after an event before an exception value can be stored.

- **Max Time**

The maximum time that can elapse after an event before automatically storing the next event as an exception value.

Set the minimum and maximum time values to *0* to turn off exception reporting.

Classic point attributes

The Classic point class is used by most PI interface programs. To configure attributes belonging to this class, click the **Classic** tab and enter values into the fields according to the specifications for the corresponding PI interface that serves as the source for the point. Refer to the interface documentation for these specifications.

Note: If the tab is disabled, the selected point class does not include Classic attributes.

Other point class attributes

If the selected point class contains an attribute set that is not part of the Base or Classic point classes, an additional tab will display with the point class name. For example, see the **Totalizer** tab in the following screen shot.

Click in the right-hand column to edit an attribute value.

General	Archive	Classic	Totalizer	Security	System
MovingCount				2	
Offset				+10m	
Offset2				+0m	
Options					
PctGood				85	
Period				+1h	
Period2				+2m	
RateSampleMode				natural	
ReportMode				PeriodEnd	
srcptid				0	
TotalCloseMode				clock	
zerobias				0	

Configure point security

To view or configure the security settings configured for a point, click the **Security** tab in the Point Builder. The security settings are different for points on Data Archive 3.4.380 than they are for earlier versions, because version 3.4.380 implements a new security model.

Note: In order to edit point data or configuration, you must have write access to the PIPOINT item in the Database Security tool. Similarly, in order to view point data or configuration you must have read access to PIPOINT.

PI point access permissions

Access to PI points is initially controlled by the *PIPOINT* entry in the Database Security tool in PI System Management Tools (SMT). While this sets a baseline for access permissions, it is important to note that individual PI points can be configured with more permissive access settings. This is achieved through the *DataSecurity* or *PtSecurity* attributes for each point, allowing granular control over who can read or write to specific points.

Types of security settings

PI point security consists of data security and point security.

Type of security	Description
Data security	Determines who can access the point's data values, including snapshot and archive data.
Point security	Governs who can access and configure the point's attributes like Zero, Span, Descriptor, and other settings.

You can have different access permissions for a point's attributes than for the point's data. For example, a user might be allowed to edit a point's data, but not to edit that point's attributes.

- **Data Security**

To view and edit point data, you also need read access to point security. If users do not have permission to view a point's attributes, they generally cannot see that point's data, because client applications require access to the point attributes to retrieve the data.

- **Point Security**

To view point attributes, you need read access to *PIPOINT*, and read access to the point security for the point itself. Similarly, to edit a point's attributes, you need read/write access to *PIPOINT*, and read/write access to the configuration for the point itself.

Configuring access permissions

You can set permissions at the point level to be more permissive than those defined by *PIPOINT*. This flexibility allows:

- **Enhanced operational control:** Grant broader access to critical points for specific users or groups.
- **Customized security settings:** Apply stricter access to points handling sensitive data, regardless of more lenient general access levels.

Common tasks and the required permissions

The following table lists required access permissions for basic tasks.

Task	Required access permissions
View point data	Read access to <i>PIPOINT</i> , <i>DataSecurity</i> , and <i>PtSecurity</i> for that point
Edit point data	Read-write access to <i>DataSecurity</i> for that point; read access to <i>PtSecurity</i> and <i>PIPOINT</i>
View point attributes	Read access to <i>PIPOINT</i> and to <i>PtSecurity</i> for that point
Edit point attributes	Read access to <i>PIPOINT</i> and read-write <i>Point Security</i> for that point
Create a point	Read-write access to <i>PIPOINT</i>
Delete a point	Read-write access to <i>PIPOINT</i> and to <i>PtSecurity</i> for that point
Add a user to data security	Read-write access to both <i>DataSecurity</i> and <i>PtSecurity</i>
Edit a user's data security settings	Read-write access to both <i>DataSecurity</i> and <i>PtSecurity</i>

Default access for new points and modules

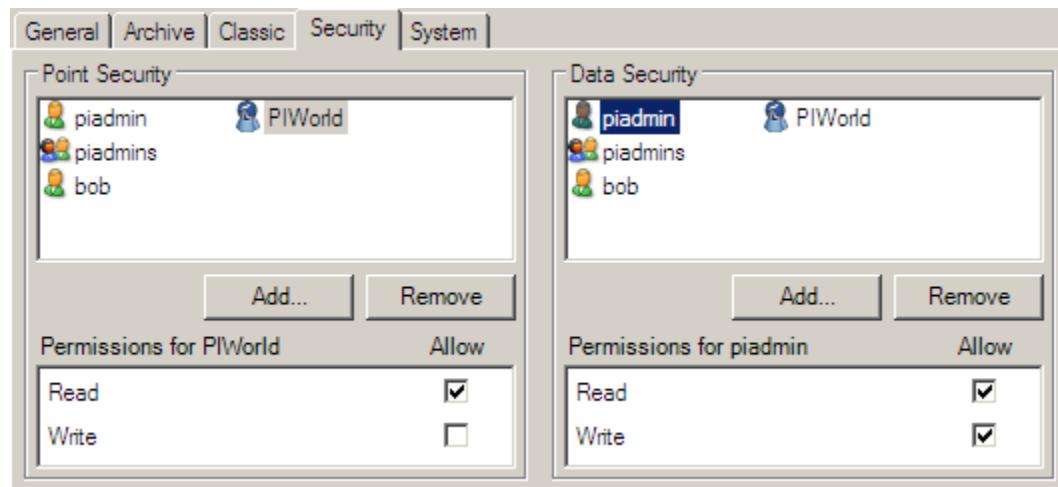
You can set default access permissions for points and modules. When you create a new point or module without

explicitly setting access permissions, the point or module gets the default access permissions.

- Default access permissions for all new points (both point data and point configuration) match the access permissions for the point database (PIPOINT). You can set permissions for PIPOINT using the Database Security plug-in for PI SMT.
- Similarly, default access permissions for root modules match the access permissions for the module database (PIModules). You can set permissions for PIModules in the Database Security tool. New modules below the root level inherit from their parent.

Set point and data security permissions for PI Server 2010 and later

PI Server 2010 (Data Archive 3.4.380) and later versions allow you to specify different access permissions for multiple PI identities, PI users, and PI groups. To set access permissions for a point, first select the point in Point Builder, then click the **Security** tab.



Identities, users, and groups that have defined access permissions for the selected point appear in the **Point Security** and **Data Security** lists. To see the access permissions for that identity, user, or group, select it in the list. You can add or remove identities, users, and groups using the **Add** and **Remove** buttons.

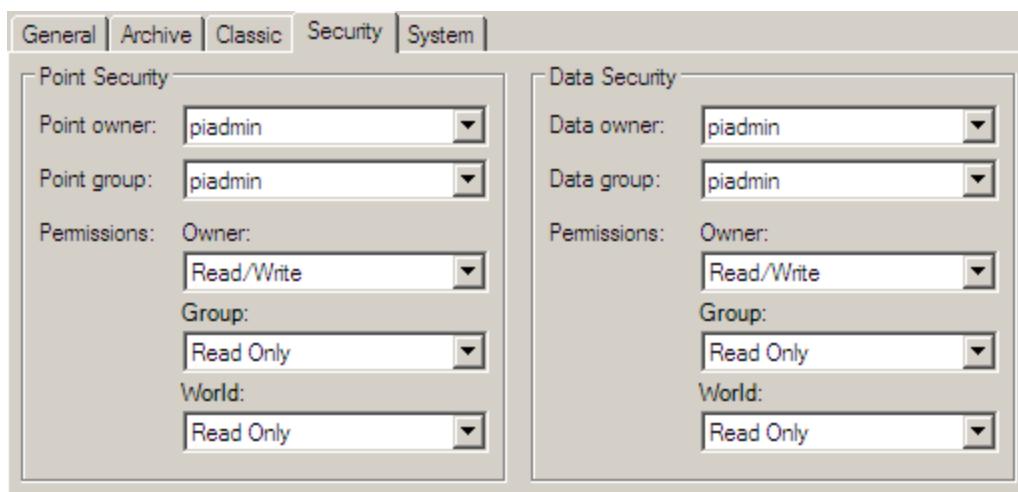
The possible levels of access are *read* and *write*. The possible access rights string can be "*r*", "*w*", "*r,w*" or "" (null). Note that there is no level for **deny**, as there is in Windows.

Set access permissions for versions earlier than PI Server 2010

Versions earlier than PI Server 2010 (Data Archive 3.4.380) use the owner/group/world model for access permissions. Each object can have an owner, which must be a PI user and a group, which must be a PI group. You can set access permissions for the owner, access permissions for the group, and the access for everyone else (called world access).

Perform the following steps to set access permissions for a point

1. Select the point in Point Builder, then click the **Security** tab.



2. Use the drop-down lists to choose an owner and a group and to specify access permissions for the owner and group. The possible levels of access are *read* and *write*. The possible access rights string can be "r", "w", "r,w" or "" (null). Note that there is no level for *deny*, as there is in Windows.

View PI point system data

Use Point Builder to view current information:

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Points > Point Builder**.
3. Click the **System** tab. You can view the following information:
 - **Creator:** The PI user or PI identity who created the point.
 - **Changer:** The PI user or PI identity who edited the point.
 - **Creation date:** The date the point was created.
 - **Last change date:** The date of the last point edits.
 - **PointID:** The PI point attribute that assigns and stores a unique number for a point after it is created.
 - **Record Number:** The read-only base PI point attribute that references its primary record in the PI archive.
 - **Source PointID:** If the point was based on another point, the PointID of that point is displayed.
 - **Snapshot:** If the point has been created and exists in PI, the current timestamp and snapshot values are displayed.

Note: The snapshot value displayed here is the end-of-stream value of the PI point.

Turn off character validation

A point name is validated as it is entered into the Name field or the Event Tag field, provided a server is selected.

On servers with high point counts, validation can take a second or two per character. Deselect **Validate tag after each letter is typed** to validate the point name only after the cursor leaves the **Name** text box.

1. Under **Collectives and Servers**, select the server or collective.
 2. In the **System Management Tools** pane, select **Points > Point Builder**.
 3. Click the **Options** button .
- The Options dialog box appears.
4. Clear the **Validate tag after each letter is typed** check box.

Rename PE points

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Points > Performance Equations**.
3. Select the point in the Performance Equations tool, right-click, and choose **Rename**.
4. Enter a name in the **New name** field and click **OK**.

Note: If an error occurs with the rename, **OK** is disabled and a  appears. Hover your cursor over it to view the error message.

Point Classes

Use the PI Point Classes tool to view PI attributes sets and PI point classes.

- **PI attribute sets**

Collections of PI point attributes that are named. One or more attribute sets are used to define point classes. Users apply lists of attributes and attribute sets to create or modify a point within a given PI Point Class.

- **PI point classes**

A collection of one or more attribute sets. Examples of point classes include Base, Classic, Alarm, and Totalizer. All point classes include the attributes from the Base attribute set. Other point classes add attributes needed to provide functionality for certain processes. The *PtClassName* attribute specifies the point class for every point.

For more information, including details on specific point classes, see the PI Server topics under [Point classes and attributes](#).

View point classes

A point class is defined by grouping together one or more attribute sets.

Procedure

1. In the Servers pane, select a Data Archive server.
2. In the System Management Tools pane, select **Points > Point Classes**.
3. In the tree view, do one of the following:
 - To view all the attribute sets on a Data Archive server, select the server ().
 - To view all the attributes for a specific point class, select the point class () under the server you want.
4. To sort attributes in sections headed by the name of the attribute set to which they belong, click **Categorized** (). To show all attributes for the selected server or set in alphabetical order, click **Alphabetized** ().

Results

In the attribute list, the default values for attributes are listed in the right column of the property grid. When you select an attribute, its name and type are listed at the bottom of the property grid.

Point Source Table

The PI Point Source Table tool displays a complete list of point sources stored in the Data Archive Point Source table for all connected Data Archive servers. Point sources are unique single or multi-character strings used to identify the source of data for a PI point. Each PI Point contains a PointSource attribute to enable PI Interfaces or other scanning software to provide data to that point. For example, when a PI Random Interface starts up, it searches the PI Point Database for every PI point that is configured with a PointSource *R*. The interface then typically examines other PI Point attributes before it loads the points, to ensure the points are valid for its use.

Point Sources

Point sources are listed for all connected Data Archive servers in the PI Point Source Table tool, with four columns to describe point source attributes:

- The source Data Archive server
- The Point Source code, which is the name used to describe the point source
- The Number of PI Points associated with the point source
- A Description of the point source and how it is used
- To refresh the Point Source list, click the Refresh button .

Add a point source

Procedure

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Points > Point Source Table**.
3. Click .

The Add a New Point Source dialog box appears.

4. Select the Data Archive server where you want to create the point source in the **Server** field.
5. Enter a unique point source code that is not in use on the current Data Archive server in the **Point Source** field.
6. Enter an optional **Description**.
7. Click **OK**.

This adds the new point source to the server's point source table.

Edit a point source description

Procedure

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Points > Point Source Table**.
3. Right-click a point source in the list and select **Edit Description**.
4. Enter your changes in the **Description** field.
5. Click **OK**.

Export a point source list

You can export the current Point Source list displayed in the Point Source Table tool to a CSV file.

Procedure

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Points > Point Source Table**.
3. Click .
4. In the Save Point Source List As dialog box, browse to the directory where you want to save the file.
5. Click **Save**.

Built-in point sources

Point sources available by default in Data Archive	Description
9	Ramp_soak
#	PI PerfMon
\$	PI SNMP
@	PI Alarm
C	PI Performance Equation
G	Alarm group
J	PI Ping
L	Lab tag (manual entry)
Q	PI RTSQC
R	Random

Point sources available by default in Data Archive	Description
T	Totalizer

Reason Tree

Use the Reason Tree tool to manage a structured collection of reason codes. A reason code, which consists of a name and description, provides a convenient way to standardize operator comments in Data Archive.

Use the Reason Tree tool to view reason codes beneath a hierarchical tree view of servers. Expand the server icons to view reason codes for that server. You can create, move, edit, and delete reason codes in the Reason Tree tool.

To open the Reason Tree tool, select **Operation > Reason Tree**.

View reason codes

Procedure

1. Under **Collectives and Servers**, select the server or collective.
2. In the System Management Tools pane, select **Operation > Reason Tree**.
3. In the Reason Tree tool, expand the server icon with the + sign next to the server name.

Note: If the server login fails, the server icon has a red X on it. Click the refresh button () , or right-click the server and select **Refresh**, to attempt a re-connection. If that fails, see [Connection checking](#).

Add reason codes

Procedure

1. Under **Collectives and Servers**, select the server or collective.
2. In the System Management Tools pane, select **Operation > Reason Tree**.
3. In the Reason Tree tool, click on a Data Archive server and click  .
The New Reason Code dialog appears.
4. Type in the title of the new reason code and a description.
The **Title** must be unique within the child reason codes under the selected parent reason code. The **Description** is optional.
5. Click **OK**.

Note: You cannot add a reason code to a secondary server of a collective. When you select a secondary server or a reason code under a secondary server, the **New** button is disabled.

Reason code title restrictions

The following rules apply for a reason code title:

- The first character must be alphanumeric; % and _ are also allowed.

- No control characters such as line feeds or tabs are allowed.
- The following characters are not allowed: '?;{}[]/\`\'\`

Edit reason codes

Procedure

1. Under **Collectives and Servers**, select the server or collective.
 2. In the **System Management Tools** pane, select **Operation > Reason Tree**.
 3. In the Reason Tree tool, select the reason code and click .
- The Edit Reason Code dialog box appears. Here you can edit the Description (but not the Name) of the Reason Code.
4. Use the **Description** field to edit the reason code description.

Note: On a Data Archive collective, you can only edit a reason code on a primary server.

Move reason codes

You can create a hierarchy of reason codes. This structure can be multi-leveled and may represent actual plant configurations.

Procedure

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Operation > Reason Tree**.
3. In the Reason Tree tool, click and drag the reason code and drop it onto the desired parent reason code.
 - If you want to move the reason code to the root, simply drop it onto the server name.
 - If you drag a reason code and hover over a collapsed reason code, the target code expands after one second. This makes it easy to drop the dragged reason code onto a hidden child reason code.
 - If you drag a reason code to the bottom of a scrollable reason tree, the tree scrolls down to display items below the bottom-most reason codes. The same is true for scrolling up.

Note: You cannot move a reason code to or from a server that is a secondary node on a Data Archive collective.

Delete reason codes

Procedure

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Operation > Reason Tree**.
3. In the Reason Tree tool, select the reason code in the server tree, and click .
4. Click **Yes** to delete the reason code.

You cannot delete a reason code from a secondary server of a collective. When you select a reason code under a secondary server, the **Delete** button is disabled.

About moving reason codes between Data Archive servers

When moving reason codes on the same server, the operation is a move. The reason code is added to the new location and removed from the previous parent reason code.

When dragging reason codes from one Data Archive server to another, the selected reason code and any child reason codes are copied. A reason code with the same title and description is created on the target server, but not removed from the source server.

To reverse this behavior, hold down the Shift key while dragging and dropping the reason code. This is the same behavior exhibited in Windows Explorer when dragging files between folders on different drives.

During the drag procedure, the mouse cursor updates to reflect whether a copy or a move operation would occur on drop. A move mouse cursor is an arrow with a small rectangle. The copy cursor is similar except uses a plus-sign.

Security settings

You can make the Data Archive server more secure by disabling less secure types of authentication. If possible, disable the least-secure authentication method, which is explicit login accounts on the Data Archive server (PI user accounts). If you are upgrading an existing Data Archive server , you might need to upgrade your security configuration before disabling explicit logins.

For more information about configuring security, see [Overview of Data Archive security](#).

Understanding the security levels

The available security levels are based on the relative security of different methods of authentication on the Data Archive server. We recommend that you use Windows authentication, wherever possible.

The security levels are:

- **Blank passwords not allowed**

Data Archive user passwords provide a minimal level of security for the Data Archive server. If you are going to use individual PI user accounts to manage the Data Archive server, at least require that the user accounts be protected by passwords. When this option is selected, PI user accounts that do not have passwords cannot authenticate on the Data Archive server.

Before you enable this security setting, create temporary passwords for all your user accounts. Ask users to change the passwords immediately.

- **Explicit login for piadmin disabled**

piadmin is the Data Archive super-user account. A person authenticated as piadmin can perform any task on the Data Archive server. Since explicit logins (PI user accounts and passwords) are the least secure authentication method, we recommend disabling this access for piadmin.

Note: The piadmin account can be still be accessed locally in PI SMT and other client applications through a default trust. You can create a mapping or a trust to access the piadmin account to allow remote access.

- **Explicit login disabled**

This is the recommended security level for Data Archive servers configured for Windows authentication. Before you disable explicit logins altogether, make sure that you have adequate access through mappings and trusts. Note also that you need SDK 1.3.6 or later for Windows authentication.

On new Data Archive installations, explicit logins are disabled by default. During upgrades, you have the option to disable them.

- **SDK trusts disabled**

PI SDK 1.3.6 and later supports Windows authentication. You can replace SDK trusts with Windows authentication.

- **API trusts disabled**

When API trusts are disabled, you can only access the Data Archive server through piconfig or through

Windows authentication. The API does not support Windows authentication, so all applications that connect through the API are locked out. This can include PI interfaces. This is not a recommended security configuration for many Data Archive servers.

Authentication options that are most secure

The most secure method for authenticating a user or application on the Data Archive server is Windows Active Directory (AD) authentication. We recommend you use this method wherever possible.

AVEVA does not recommend using PI trusts or explicit logins for authentication. For a more secure environment, AVEVA recommends Windows Integrated Security.

Note: PI API 2016 for Windows Integrated Security extends Windows authentication to API-based client applications. If you choose to install PI API 2016 for Windows Integrated Security, you can use only Windows Integrated Security for authentication. Both trusts and explicit logins will fail.

Authentication methods are listed in order, from most secure to least secure:

- **AD Authentication (Kerberos)**
- **Local Windows authentication (NTLM)**
- **PI Trusts**

When you define a PI trust, you can choose how secure the trust will be. You can create PI trusts for API connections or for SDK connections. However, since the SDK supports Windows authentication, you should not create PI trusts for SDK connections.

- **Data Archive User Accounts and Passwords (explicit logins)**

Local Data Archive user accounts are the least secure way to authenticate on the Data Archive server. We do not recommend using this method.

Configure security settings

Procedure

1. In SMT, select **Security > Security Settings**.

The Security Settings tool opens.

2. Select the Data Archive server in the Collectives and Servers pane. You can change settings for only one server at a time.
3. To apply the changes, stop and then re-start the Base subsystem.

Snapshot and Archive Statistics

Use the Snapshot and Archive Statistics tool to monitor snapshot and archive activity and usage statistics on connected Data Archive servers. If you periodically review these statistics, you can solve a system or data issue before it becomes a problem. Many of the statistics, such as Overflow Data Record Count, are informational; others are valuable for predictive maintenance.

The archive statistics displayed are aggregated statistics, not separated by historical and future archive set types. For example, if you are viewing the archiving flag in **Operation > Snapshot and Archive Statistics**, the archiving flag records the state of both archive sets, historic and future, combined. The values returned range from 0 (zero) to 3, as shown in this table. (The archiving flag indicates that events will be read off the event queue and archived.)

Value of archiving flag	Meaning of value
3	Both historic and future archive sets are writing to the archives
2	Only the future archive set is writing to the archive
1	Only the historic archive set is writing to the archive
0	Neither historic or future archive sets are writing to the archives

All data passes through both the snapshot and the archive subsystems. Indicators available in the Snapshot and Archive Statistics tool can alert you to potential data-flow problems. For example:

- Out-of-order events are events that arrive with an older timestamp than the current value. Out-of-order events may indicate that there is a clock problem on the Interface node.
- Rising event numbers on the Snapshot Overflow Queue indicates that the archive is not accepting new data or cannot keep up with the pace of the data transfer. This situation could have a number of causes and should be remedied immediately.

To open the Snapshot and Archive Statistics tool, select **Operation > Snapshot and Archive Statistics** under System Management Tools.

Set up automatic refresh

Procedure

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Operation > Snapshot and Archive Statistics**.
3. Click ➤ or right-click an item in the **Snapshot and Archive Statistics** viewer and select **Start Updating**, to begin automatic, periodic updates of the display. You can also change the refresh rate.

Freeze snapshot and archive statistics

Procedure

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Operation > Snapshot and Archive Statistics**.
3. Click  or right-click an item in the **Snapshot and Archive Statistics** viewer and select **Stop Updating** to stop updates and freeze the current display.

Note: Click **Refresh**  to manually update the display to reflect current values.

Change the refresh rate

Procedure

1. Under **Collectives and Servers**, select the server or collective.
 2. In the **System Management Tools** pane, select **Operation > Snapshot and Archive Statistics**.
 3. Click .
- The Options dialog appears.
4. Enter a new value in seconds and click **OK**.

All counters are reset when the subsystem is restarted.

View snapshot statistics only

Procedure

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Operation > Snapshot and Archive Statistics**.
3. Click the **Snapshot** radio button at the top of the **Snapshot and Archives Statistics** viewer.

View archive statistics only

Procedure

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Operation > Snapshot and Archive Statistics**.
3. Click the **Archive** radio button at the top of the **Snapshot and Archives Statistics** viewer.

View snapshot and archive statistics together

Procedure

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Operation > Snapshot and Archive Statistics**.
3. Click the **Both** radio button at the top of the **Snapshot and Archives Statistics** viewer.

Export snapshot and archive statistics

Procedure

1. Under **Collectives and Servers**, select the server or collective.
 2. In the **System Management Tools** pane, select **Operation > Snapshot and Archive Statistics**.
 3. Click  or right-click an item and choose **Export**.
 4. In the Save Current Value List As dialog box, specify the file name and location, and click **Save**.
- The Snapshot and Archive Statistics tool exports the displayed statistics to a Comma-Separated Value (CSV) file.

View snapshot and archive statistics

The Snapshot and Archives Statistics tool lists snapshot and archive statistics for each connected Data Archive server. Each row includes the following column values:

- **Type:**
The statistical source type of the value, either archive or snapshot.
- **Counter:**
The descriptive name of the counter to indicate the measurement.

SNMP

Note: This tool is no longer included with PI Server. For new PI Server installations, if you want to use this tool, you need to download and install this interface separately.

Use the SNMP tool to build and manage points for the PI SNMP Interface. PI SNMP points enable you to monitor IT resources.

SNMP (Simple Network Management Protocol) is a generic protocol used to make requests between an SNMP manager and the SNMP agent, or remote device. In the case of AVEVA™ PI System™, the PI SNMP Interface is the SNMP manager.

When you use the PI SNMP utility to associate PI Points with SNMP variables or Object Identifiers (OIDs), you can use PI SNMP points to gain important insights into network management challenges such as port traffic, system uptimes, and environmental statistics.

To open the PI SNMP tool, select **IT Points > SNMP** in the System Management Tools pane.

PI SNMP points support SNMP versions 1, 2c, and 3, excluding SHA decoding.

Information provided by SNMP

Various types of information are available through SNMP, although the most common are related to network performance and management. For example, most routers run SNMP Agent software that allows them to retrieve the following:

- Information defined by standards bodies (such as ISO)
- Proprietary information specific to a particular device or manufacturer

SNMP information exchange

The type of information exchanged between the SNMP Manager and the SNMP Agent is defined by a Management Information Base (MIB).The most common MIB is MIB-II (or MIB-2) which is related to network management, and defines the number of octets (groups of 8 bits) sent or received on a particular physical interface.

An Object Identifier (OID) is an element of an MIB. For example, the OIDs for the number of octets sent or received are:

```
.iso.org.dod.internet.mgmt.mib-
2.interfaces.ifTable.ifEntry.ifInOctets
.iso.org.dod.internet.mgmt.mib-
2.interfaces.ifTable.ifEntry.ifOutOctets
```

Because the use of MIB-II is widely prevalent, the above are often abbreviated as:

```
interfaces.ifTable.ifEntry.ifInOctets
interfaces.ifTable.ifEntry.ifOutOctets
```

and the numerical representations of these two OIDs are:

```
1.3.6.1.2.1.2.2.1.10
```

.1.3.6.1.2.1.2.2.1.16

A particular occurrence of an OID is called an instance. This instance number is added to the end of the OID. Continuing with the examples above, you can see that the number of octets received on the first physical interface is given by

```
interfaces.ifTable.ifEntry.ifInOctets.1  
.1.3.6.1.2.1.2.2.1.10.1
```

For OIDs where there is only a single occurrence, a zero is used. For example,

```
system.sysUptime
```

is the time since the network management portion of the system was last reinitialized. Thus, the only instance of *system.sysUptime* is

```
system.sysUptime.0
```

COUNTER values

Some OID values are given in terms of a COUNTER. A COUNTER is an unsigned 32-bit integer ranging from 0 to 4,294,967,295. When a COUNTER value reaches the maximum, it rolls over to 0. In particular, these OIDS are both COUNTERS:

```
interfaces.ifTable.ifEntry.ifInOctets  
interfaces.ifTable.ifEntry.ifOutOctets
```

Therefore, if the PI SNMP points use the raw values for these OIDs, such values will continuously increase numbers up to the maximum. A graphical trend of these numbers will probably not be meaningful.

Alternatively, you can configure the PI SNMP Interface to retrieve COUNTER values per unit time. If Location2 is set to 1, the PI SNMP Interface retrieves the difference between two successive readings divided by the scan time. For example,

```
scanned value = 2000  
previous value = 200  
scan time = 1 minute  
stored value = (2000 - 200)/60 = 30
```

A graphical trend of such values will be more meaningful because it provides the number of octets transferred per second.

Note: For more detailed information on SNMP, consult SNMP, SNMPv2, SNMPv3, and RMON 1 and 2, Third Edition, by William Stallings (Addison-Wesley, 1999, ISBN 0201485346), or your Windows help files.

Build SNMP points

Before you can build and configure PI SNMP points, you must have:

- a PI SNMP Interface installed
- an SNMP service running on the remote devices you plan to monitor

The PI SNMP Interface is the network node that supplies data for PI SNMP points. You can use the PI SNMP utility to update the PI SNMP Interface.

Note: SNMP interfaces must be configured with PI Interface Configuration Utility (PI ICU) to be available to SNMP point builder. Manually installed interfaces are not visible in this utility.

The PI SNMP point builder uses the following process to build SNMP points:

1. Select the PI SNMP Interface for which you will build points on the **Tag Settings** tab.
2. Configure the SNMP agent on the interface and select specific OIDs on the **SNMP Settings** tab.
3. Select OIDs and build PI points using SNMP MIBs and templates on the **Build Tags** tab.

Select a PI SNMP interface and configure tag settings

Before you create PI SNMP points, you must select the PI SNMP Interface node in the PI SNMP point builder for which you will build points and update some PI SNMP Interface settings.

For further details about location codes and point attributes, see [Default point attributes](#) and the *PI Interface for SNMP* available at the [Customer Portal](#).

1. Select the **Tag Settings** tab.
2. Select or the type in the name of an SNMP Interface in the Choose an existing SNMP interface field. If you enter a name, include the following information in the name:
 - Point Source
 - Interface ID
3. Select an available PI SNMP Interface from connected Data Archive servers in the **Choose an existing SNMP interface** field.
4. Configure or update **PI SNMP Interface** settings, as necessary:
 - **Point Source** — Enter a single character value that indicates the origin of point data in the Data Archive server and matches the value specified in the interface startup command file. Valid point source characters include any ASCII character, but should not conflict with any other PI data collection or system programs. This attribute is also known as Location4 Code or Scan Class.
 - **Program instance (Location 1)** — Enter a value to associates the point with an instance of PI SNMP, as multiple copies may run on the same machine. Location 1 is a positive integer that should match the -id= parameter in the interface startup command file.
 - **Scan class frequency** — Enter the point scan class that corresponds to a scan class set created at interface startup.
5. Configure or update the **Tag Details**, as necessary:
 - **Create Rate Tags, when applicable (Location 2)** — If you are using Counter-type SNMP variables, set Location 2 equal to 1 for time-normalized (or rate) points, as opposed to points with non-decreasing values.

Note: When set to 1, counter-valued OIDs selected on the **Build Tag** tab are converted from raw counters to rates, that is, counter difference divided by time difference. For non-Counter type variables, the interface writes the configure state to the point and the box should remain unchecked.

- **Group tags in sets (Location 3=1 if checked)** — Set equal to 1 to process SNMP points for the same agent, and therefore, the same Community Strings, and the scan class in one SNMP call, to increase performance and lower overall network overhead.
- **Omit community String from Extended Descriptor** — Select to exclude the community string in the extended descriptor field. Note that the administrator then becomes responsible for maintaining a list of community strings, as described in the *SNMP Interface manual*.

- **Use tag name substitution table** — Select to scan a substitution dictionary lookup table when building a point. If strings in the lookup table exist in the tag name, they are replaced by the matching table entry.

SNMP agent configuration

You configure the SNMP agent on the interface node where you plan to build tags using the tools on the **SNMP Settings** tab. You can enter the SNMP profile manually, or you can use a saved agent profile.

Create SNMP agent profile

You can enter SNMP authentication information directly, and save the details in a file:

1. On the **SNMP Settings** tab, enter an **Agent Prefix** to identify the interface node as the parent of SNMP PI points built from the interface.
2. Enter the **Agent IP/Hostname** of the device and an **SNMP Username** for version 3 connections.
3. Enter credentials to connect to and poll an SNMP agent.
 - For SNMP versions 1 and 2, the credential is a **Read Community String**, usually **public**.
 - For SNMP version 3, you must provide a username and possibly an **Authentication** and **Privacy Password**.
4. Click **Save to File** to save the information to an agent file in XML format.

Select a saved agent profile file

To use a saved agent file to enter the SNMP version information:

1. Click  to browse to and select the desired agent file.
2. Select the desired agent from the **Agents in file** pane.
3. After agent information is supplied through an agent file, you can select a file in the **Agents in File** list to validate the SNMP agent and get interface information.

View network SNMP interfaces

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **IT Points > SNMP**.
3. Click the **SNMP Settings** tab to see a list of the network interfaces on the remote device.

Validate SNMP agent

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **IT Points > SNMP**.
3. Select the **SNMP Settings** tab.

4. Click **Validate Only** to validate your SNMP devices and interface settings.

You should see a message stating that the agent was validated. Otherwise, an error message will indicate the interface validation failed. Contact the [Customer Portal](#) for further assistance.

Get interface information

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **IT Points > SNMP**.
3. Select the **SNMP Settings** tab.
4. Select the interface or interfaces that you want information about.
5. Click **Get Interface Info** to get information about the PI SNMP Interfaces and SNMP agents. For example:

```
6/10/2009 1:27:06 PM SNMPPoints> System information for bruin
SysDescr: Hardware: x86 Family 15 Model 4 Stepping 1 AT/AT COMPATIBLE -
Software: Windows 2000 Version 5.1 (Build 2600 Multiprocessor Free)
Number of Interfaces = 4
```

Map OIDs to PI Points

Before you build PI SNMP points, you must select the PI SNMP Interface node. See [Select a PI SNMP interface and configure tag settings](#).

Points are built for each interface-OID combination, based on the SNMP settings and the OIDs you select in the **Build Tags** tab.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **IT Points > SNMP**.
3. Select the **Build Tags** tab.
4. Select OIDs in the **Build Tags** tab, or use an OID template to map typical OID configurations on parallel devices to common sets of tags.

Select OIDs

Points are built for each interface-OID combination, based on the SNMP settings and the OIDs you select in the **Build Tags** tab. See [Map OIDs to PI Points](#).

1. Select the Build Tags tab to view the OID tree. Each OID is an entry in the MIB-II tree which stores definitions for SNMP messages.

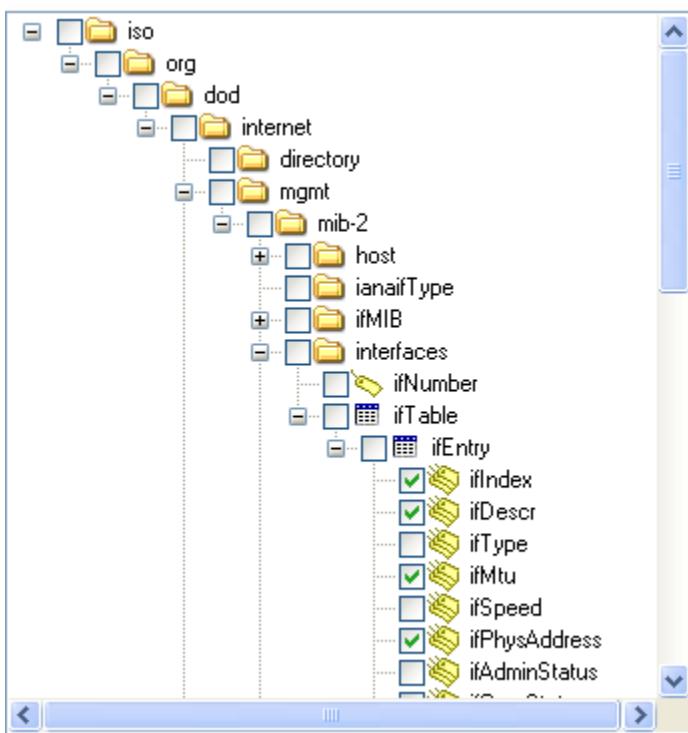
All OIDs in the interfaces branch of the MIB-II table refer to the interfaces selected in the panel. In other words, if you select the *IfIndex* OID, these points are built for each selected interface:

```
iso.org.dod.internet.mgmt.mib- 2.interfaces.ifTable.ifIndex.ifInOctets
```

Typically, you want to choose a number of OIDs as the basis for a collection of PI points that communicate detailed information about the status of the interface.

2. Select each OID you want to build points for.
3. Select a parent node to automatically select all child nodes.

4. Once all OIDs are selected, click **Assign Tags** to preview the PI SNMP points that will be created based on the selections in the **Tag Preview Pane**.
5. Create PI SNMP points.



6. To load a different MIB file:
 - a. Click **Load an MIB file** on the toolbar and navigate to the MIB file location.
 - b. Select the file.
 - c. Click **Open**.

Create PI SNMP points

After you configure your tag and SNMP settings, you can build PI SNMP points directly on the Data Archive server or [Export PI SNMP Point Configurations](#) to a Comma-Separated Value (CSV) file.

1. Select **Create tags on PI Server** and select an available server from the menu.
2. Click **Create Tags**.

Note: If you try to name a PI SNMP point with a name that is already used by an existing PI point, you are prompted whether to skip or overwrite the existing point.

3. To write to a CSV file:
 - a. Select Write tags to CSV File and navigate to the location where you want to save the file.
 - b. Click **Create Tags**.

Modify point attribute properties

You might want to change point attributes other than Interface ID and Point Source (Locations 1 and 4).

Change point values on the Properties dialog box. The dialog box has two tabs:

- **General:** allows you to change the sampling interval and displays fixed PI Interface information.
 - **Advanced:** includes typical point attribute controls and highlights any attributes that differ across selected tags in yellow.
1. To access properties, right-click selected tag entries and select **Properties**.
The Properties dialog appears.
 2. Select one or more tag names.
 3. To view and change attribute values, right-click over specific attributes and choose **Details** to see the value of that point attribute for each selected tag.
 4. Select **Change Value**.
 5. Choose the value:
 - Select **To Selected Value** and choose one of the attribute values displayed to apply that value to all the tags in the list.
 - Select **To Other Value** and enter a new attribute value to apply that value to all the tags in the list.
 6. Click **OK**.

Manage OID templates

You can create templates of OID selections that can be saved and reused. With these templates, you can quickly build parallel point configurations for arrays of identical devices.

After selecting OIDs from the OID tree, use the **Template Management** tools to create and organize your OID templates.

Load an OID template

To load an existing template into the **Template Manager**:

1. Click the **Load** button .
2. Navigate to the location of the OID template file.
3. Verify that the template name appears in the **Current Template(s)** field.

Remove an OID template

To remove the OID template from the **Template Manager**:

1. Select the template.
2. Click the **Unload** button .

Apply an OID template

To apply the OID template that is loaded in the **Current Template** field to a set of PI SNMP points:

1. Select the OID template.
2. Click **Apply**.
The template specifications are applied to the **Tag Preview Pane**.
3. Use the selected OID tags to assign the tags to PI points.
4. [Create PI SNMP points](#).

Create OID templates

When you create a template, the Template Builder dialog box appears. The template builder displays OID information and provides access to default point attribute properties that you can also set for points created using the template. To create an OID template for PI SNMP points:

1. Click create in the **Template Management** area of the **Build Tags** tab.
2. Use the Template Builder to update default point attributes select templates to load.
3. Click **Apply**.
4. Click **Save**.

Preview SNMP points

Point names based on the OIDs selected are displayed in the **Build Tags** tab to the right of the **OID Tree**.

1. Select or deselect points or shift-click to select multiple tags in the list.
2. Right-click to select SNMP point preview options for selected points.

SNMP point preview options

Option	Purpose
Clear Highlighted Entries	Remove selected entries from the list
Clear all Entries	Remove all entries from the list
Check Highlighted Entries	Select all entries highlighted by user
Uncheck Highlighted Entries	Deselect all entries highlighted by user
Rename Highlighted Entries	Display the Rename Tag dialog to rename highlighted entries
Check All	Select all entries
Uncheck All	Deselect all entries
Resize Columns	Resize columns to auto-fit the current list
Properties	Display the Properties dialog to edit point attributes of

Option	Purpose
	highlighted entries

Rename PI SNMP points

The SNMP Points builder uses a rigid naming scheme, which can produce long point tag attributes.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **IT Points > SNMP**.
3. Right-click a point name, or Shift-click multiple point names, in the SNMP Points list and select **Rename Highlighted Entries** to change default point names prior to building points.

Note: You can use the **Search and Replace** features in the Rename Tags dialog box to globally change certain naming conventions.

Point name restrictions

Longer tag names may pose a problem for some versions of the SNMP interface and client tools. The following utilities have tag length limitations, with the number of characters allowed denoted in parentheses:

- PI DataLink 2.x (80)
- PI SNMP versions prior to 1.3.0.1 (80)
- PI Performance Equations (73); this limit is for points on which event-based processing is based.

Export PI SNMP Point Configurations

To edit tag configuration before building PI SNMP points, you can export point configurations to a Comma-Separated Value (CSV) file.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **IT Points > SNMP**.
3. Select the **Build Tags** tab.
4. In the lower-right section of the window, select **Write tags to CSV file** and click **Create Tags**.
5. Specify a filename and location for the file.
6. Import the file into MS Excel and make any desired changes.
7. Export the tag information to a selected Data Archive server with the PI Builder application.

Default point attributes

Certain OIDs are used often. The SNMP point builder automatically creates the following PI point attributes:

ifInOctets point attributes

Point Attribute	Value
compdev	250
compmax	28800
compmin	0
compressing	1
convers	1
descriptor	Traffic In Rate
digitalset	N/A
engunits	Bytes/Sec
excdev	0
excdevpercent	0
excmax	600
excmin	0

ifOutOctets point attributes

Point Attribute	Value
compdev	250
compmax	28800
compmin	0
compressing	1
convers	1
descriptor	Traffic Out Rate
digitalset	N/A
engunits	Bytes/Sec
excdev	1

Point Attribute	Value
excdevpercent	1
excmax	600
excmin	0

ifInErrors point attributes

Point Attribute	Value
compdev	0.0002
compmax	28800
compmin	0
compressing	1
convers	1
descriptor	Errors In Rate
digitalset	N/A
engunits	Errors/Sec
excdev	0
excdevpercent	0
excmax	600
excmin	0

ifOutErrors point attributes

Point Attribute	Value
compdev	0.0002
compmax	28800
compmin	0
compressing	1

Point Attribute	Value
convers	1
descriptor	Errors Out Rate
digitalset	N/A
engunits	Errors/Sec
excdev	0
excdevpercent	0
excmax	600
excmin	0

ifAdminStatus point attributes

Point Attribute	Value
compdev	0
compdevpercent	0
compmax	28800
compmin	0
compressing	1
convers	1
descriptor	Interface Admin Status
digitalset	PortStatus
engunits	N/A
excdev	0
excdevpercent	0
excmax	600
excmin	0

ifOperStatus point attributes

Point Attribute	Value
compdev	0
compdevpercent	0
compmax	28800
compmin	0
compressing	1
convers	1
descriptor	Interface Oper Status
digitalset	PortStatus
engunits	N/A
excdev	0
excdevpercent	0
excmax	600
excmin	0

SysUptime point attributes

Point Attribute	Value
compdev	2
compdevpercent	2
compmax	28800
compmin	0
compressing	1
convers	0.00000011574
descriptor	System Up Time
digitalset	N/A

Point Attribute	Value
engunits	Days
excdev	1
excdevpercent	1
excmax	600
excmin	0

SQC Alarms

Use the SQC Alarms tool to manage real-time SQC alarms, a system of related PI points that record real-time SQC values. The alarms use real-time extensions stored in PI Alarm Subsystem.

If you run PI SQC Alarm Subsystem, a separately licensed enhancement to the standard Data Archive, your Data Archive can evaluate SQC pattern tests and manage the alarms generated from those tests.

You can also build SQC alarms using tools like **piconfig** and PI Builder. Since proper operation of an SQC alarm on the Data Archive server relies not only on the configuration of the alarm itself, but also on the order in which the points associated with the alarm are created; you might encounter difficulties when using those tools. You eliminate such potential problems when you create and configure SQC alarms with PI SMT.

When using the SQC Alarms tool, you should be familiar with statistical quality control (SQC) terms and principles and understand basic PI point configuration.

SQC Alarms tool

The SQC Alarms tool displays SQC alarm points on all connected Data Archive servers.

- To retrieve and display SQC alarms, click the **Search** button . You can filter your search results and retrieve only specific tags by entering a search mask in the **Tag Mask** field. For example *sqc** would load all alarms beginning with the letters *sqc*. The default mask is ***, which loads all alarms.
- To refresh the current values of all listed SQC alarms and their associated points, click the **Refresh** button or press F5.
- **Note:** If you have many alarm points and you want to display values more quickly, use Shift+click or Ctrl+click to select only the alarms you want to refresh.

The list displays properties of each alarm point in several columns:

- **Alarm Name:** The tag name of the SQC alarm point.
- **PI Server:** The server where the alarm point resides.
- **Alarm Status:** The current point value of the SQC alarm. SQC alarm point values represent alarm status based on whether a source point has passed or failed one or more SQC pattern tests.
- Possible alarm values are listed in the pisqcalarm digital state set and discussed in the *Applications User Guide*.
- **Value of Source:** The current value of the source point monitored by the SQC alarm.

Note: The value of source changes based on chart type.

- **UCL:** The value of the upper control-limit point, used in alarm calculations.
- **CL:** The value of the center-line point, used in alarm calculations.
- **LCL:** The value of the lower control-limit point, used in alarm calculations.
- **Execution State:** The current value of the reset point monitored by the SQC alarm, which tells SQC Alarm Subsystem whether the alarm should be running normally, placed on hold, or whether alarm is in a transitional state. An example of a transitional state is Alarm-On; this state indicates that the alarm had been

placed in hold, but has been turned on again and is waiting for a new source tag event.

- **Description:** The descriptor attribute of the SQC alarm point.

Create SQC alarms

Log into the Data Archive sever with permissions to create and edit points.

You can create or edit alarms using the same tools in the SQC Alarms wizard.

1. Under **Collectives and Servers**, select the server or collective.
2. In the System Management Tools pane, select **Alarms > SQC Alarms**.
3. Click **Create a New SQC Alarm**  on the toolbar, or right click and choose **New SQC Alarm**.
The SQC Alarms wizard appears.
4. Follow the instructions on the wizard to make the desired changes to the alarm configuration.
For details on the parameters you specify in the wizard, see [SQC Alarms wizard](#).
5. Click **Finish** to save the SQC alarm and close the wizard.
If you have used automatic point generation, all the associated points along with the alarm point are created at this time, and a message is written to the session record to indicate success or failure.

Edit SQC alarms

Log into the Data Archive server with permissions to create and edit points.

You can create or edit alarms using the same tools in the SQC Alarms wizard. When editing an alarm, most of the alarm information already exists, and some parts of the wizard are inaccessible.

1. Click **Edit SQC Alarm**  on the toolbar, or right click and choose **Edit SQC Alarm**.
The SQC Alarm wizard appears.
2. Follow the instructions on the wizard dialog boxes to make the desired changes to the alarm configuration.
For details on the parameters you specify in the wizard, see [SQC Alarms wizard](#).
3. Click **Finish** to save the SQC alarm and close the wizard.

SQC Alarms wizard

Basic SQC alarm attributes

Enter the basic attributes used to identify the SQC alarm point:

- **Server:** The Data Archive server where the alarm point is defined and stored. This attribute may not be changed once the alarm point is created.
- **SQC Alarm Tag Name:** The tag name for the SQC alarm point. This attribute may not be changed once the alarm point is created.

- **Description:** A functional description of the alarm which is used for the descriptor attribute of the alarm point.
- **Alarm Group:** Select a group if you want to associate the new SQC alarm with an existing PI alarm group on the server.
- **Clear on Subsystem Startup:** Select to clear existing alarm calculations on startup. When not selected, the subsystem seeds alarm calculations with archive values from the source point.
- **Clear on Control Limit Change:** Select to start calculations when limit point values are changed. When not selected, the alarm retains the existing pattern test buffers.
- **Auto-Acknowledge:** Select to automatically acknowledge alarms.
- **SQC Alarm Priority:** Specify an SQC alarm priority when not using the **Auto-Acknowledge** feature.

Note: If the **Auto-Acknowledge** check box is selected, the **SQC Alarm Priority** property is set to 0 and **SQC Alarm Priority** is disabled.

Automatic point generation

Automatic point generation allows the wizard to automatically generate all the additional alarm points associated with an SQC alarm, and provide consistent point naming.

Note: Automatic point generation is available for new alarm points, but cannot be added to existing points.

Perform the following steps to enable automatic point generation:

1. Select the **Use Automatic Point Generation** check box.
2. Specify the parameters for generating points:
 - **Position:** Choose to add identifying extensions to alarm point tag names as prefixes or suffixes.
 - **Delimiter:** Specify the delimiter to add between the SQC alarm tag name and the extension. For example, if your alarm name is called *SQCAlarmTag1* and you have a suffix extension with a . delimiter, the UCL tag name is *SQCAlarmTag1.UCL*. If you have a prefix extension and a - delimiter, the tag name is *UCL-SQCAlarmTag1*.
 - **Generate Optional Points:** Check to create points that you may need for your specific implementation of SQC. For example, if you need to see all of the pattern tests that fail in addition to the highest precedence test, then you should create the optional status point.
 - **Extensions:** Specify the extension to use for each related alarm tag type. The extensions are saved as default values for future alarm points.

Source chart and sampling

An SQC alarm point monitors the status of PI point values with regard to standard statistical tests. The next step in creating an SQC alarm is to specify the statistical calculations underlying the alarm point and the sampling methodology:

- **Source Tag:** The source tag provides data to which the pattern tests are applied. For a chart of individuals this can be any point on the same Data Archive server. For all other chart types, the source tag is created by the wizard and its data comprise appropriately aggregated samples of the raw data tag.

Different statistical calculations and corresponding chart types require different types of source points. An

Individuals chart allows any type of source point. EWMA chart types require a Performance Equation point. All other chart types require a Totalizer source point.

- **Chart Type:** Choose a chart type. One of eight charts may be selected to define the calculations performed and display the SQC results in PI ProcessBook SQC charts.

The configuration options provided change depending on the selected chart type:

- **Raw Data Tag:** If you are creating a non-individuals type SQC alarm, then enter or search for the name of the point containing the raw data to be sampled.

- **Event or Time Based Sampling:** Select a sampling mode to choose how events are included in a sample group. The sampling configuration greatly affects the behavior of the source point.

For event-based sampling, simply specify the number of **Events per Sample**. For time-based sampling, specify an offset from midnight and an interval of time to define windows used for sampling.

- **Filter Expression:** Enter an optional filter expression using PI Performance Equation syntax to filter out any values that you do not wish to include in sampling.

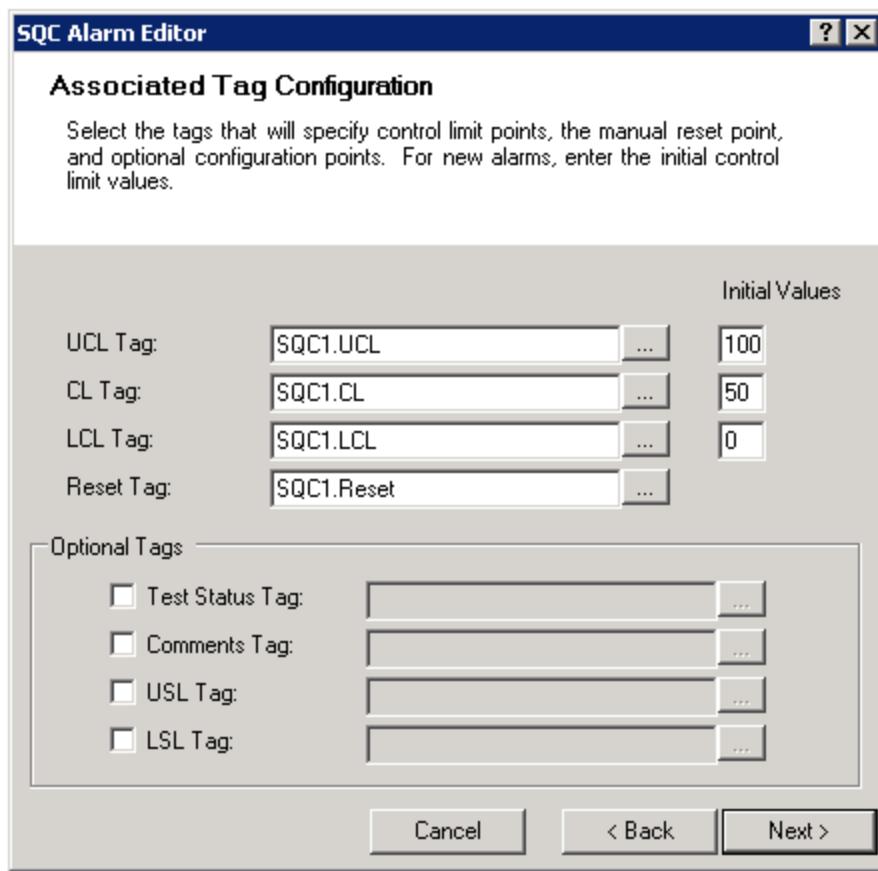
- **Lambda and Scan Class:** Enter values for these items when specifying an EWMA (exponentially-weighted moving average) chart, where the source point is a Performance Equation point of the following form:

```
'RawDataTag' + if badval('SourceTag') then <Lambda> *  
PrevVal('RawDataTag') else <Lambda> * PrevVal('SourceTag')
```

- The **Scan Class** refers to a valid scan class of the Performance Equation Scheduler.

Associated points

Associated points generally provide limit and reference values used in SQC calculations.



Perform the following steps to specify the points required for SQC calculations:

- Use the provided fields to enter or change the tag names of associated points, if necessary. You can also click the ellipsis button to search for tags on the Data Archive server. If you are creating a new alarm using Automatic Point Generation, the tag names are already created for you.
- Use the provided fields next to each tag name to enter initial limit values for each point.

If desired, you can add and specify optional points:

- Check the box next to each optional point type you want to add, then enter a tag name or click [...] to search for a tag on the server.

The following default points must be defined for a real-time SQC alarm:

Tag Alias	Description
Source Tag	The point on which alarm calculations are performed
Reset Tag	The point whose value sets the execution status of the pattern test evaluation
UCL Tag	The point whose value is the upper control limit for the chart

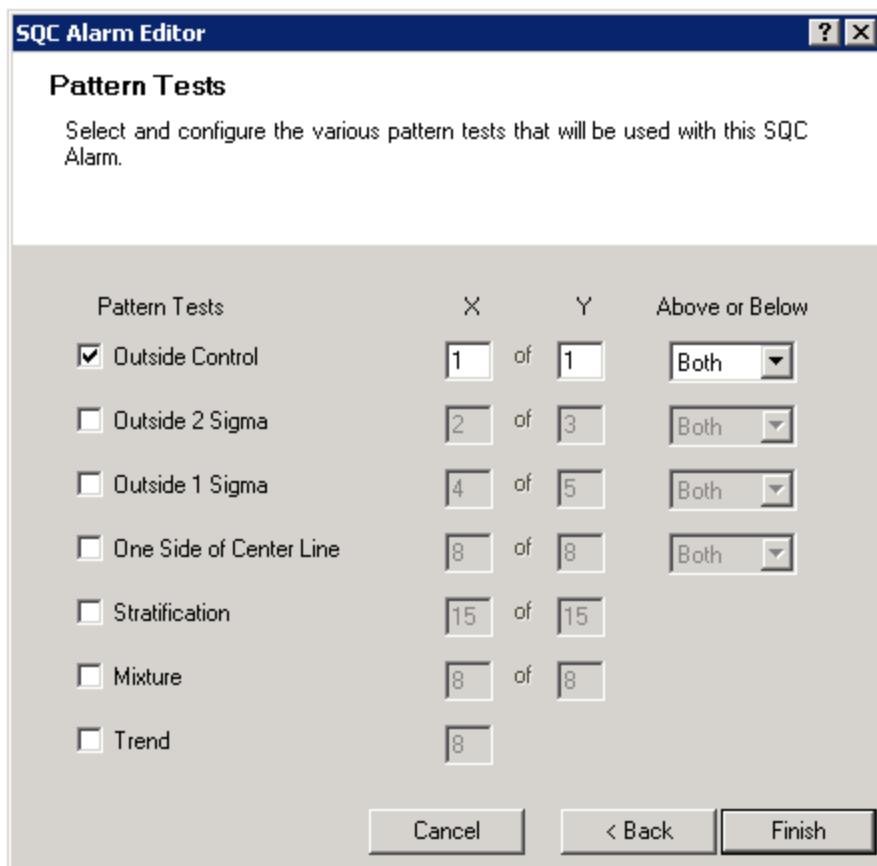
Tag Alias	Description
CL Tag	The point whose value is the center line for the chart
LCL Tag	The point whose value is the lower control limit for the chart

The following optional points can be associated with a real-time SQC alarm:

Tag Alias	Description
Test Status Tag	The point whose value corresponds to all pattern tests currently in alarm condition
USL Tag	The point whose value is the upper specification limit for the chart
LSL Tag	The point whose value is the lower specification limit for the chart
Comment Tag	The point used to store comments associated with the SQC alarm

Specify conditions for the SQC pattern tests

Pattern tests are run on point data to determine if alarm conditions exist, based on your limit specifications.



Perform the following steps to specify conditions for the SQC pattern tests:

- Select the checkbox next to each pattern test you want to apply in the SQC alarm. Values that reach the specified limits of any active pattern test will place the SQC point in alarm state. By default only the 3 sigma limit test is applied.
- For selected pattern tests, specify the number of values (X) out of a number of sampled values (Y) that are **Above or Below** the corresponding pattern test limits.
- The Western Electric suggested patterns are shown by default, and by default tests are applied on both sides of the center line.

The following tests may be selected:

- **OutsideControl:** Counts the number of samples outside the control limit on one side of the center line.
- **OutsideTwoSigma:** Evaluates the sample against a limit drawn 2/3 of the way between the center line and the control limit.
- **OutsideOneSigma:** Evaluates the sample against a limit drawn 1/3 of the way between the center line and the control limit.
- **OneSideofCL:** Counts the number of samples on one side of the center line.
- **Stratification:** Counts the number of samples that fall within the upper and lower One Sigma limits on both sides of the center line.
- **Mixture:** This test counts the number of samples that fall outside the upper and the lower One Sigma limits on both sides of the center line.

- **Trend:** Counts the number of samples which are monotonically increasing or decreasing.

Note: The term *sigma* generally refers to one standard deviation in a PI SQC alarm. SQC control limits are not limited to three times the standard deviation of process measurements. Control limits may be set to other values depending on process needs. Thus *sigms*, as used in the PI SQC alarm pattern tests is one-third the difference between the value of the center line and the control limits.

Change the execution state

The SQC Alarms tool interprets the execution state of an alarm point. An alarm can be in one of three different execution states at any time:

- Normal: The SQC Alarm point is running and pattern tests should be evaluated.
 - Hold: Evaluation of pattern tests should be suspended.
 - Clear: Clear out the point's pattern test buffers (used in conjunction with the Normal state).
1. Under **Collectives and Servers**, select the server or collective.
 2. In the **System Management Tools** pane, select **Alarms > SQC Alarms**.
 3. Select the alarms you want to change and click the arrow next to **Change the Execution State**  on the tool bar.
 4. Choose a reset option for the execution state.

You can place the alarm in Hold or Normal states, or clear and reset the alarm's pattern test buffer. When you clear and reset, the reset point is momentarily set to Clear, immediately followed by the Normal state.

Change control limits

You can edit the control limits for an alarm point in list view. To edit control limits:

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Alarms > SQC Alarms**.
3. Click **Change Control Limits**  on the tool bar.
4. Edit the desired control limit values and click **OK**.

Delete alarms

You can delete alarm points from the system as needed. Remember that deleted points are removed permanently. In many cases the Source Tag holds all the history for an instrument in the control system, or from the lab data entry system. Since deleting such points deletes ALL history associated with them, be sure to verify the value of tags before deletion.

1. Under **Collectives and Servers**, select the server or collective.
2. In the **System Management Tools** pane, select **Alarms > SQC Alarms**.
3. Select the alarm or alarms that you want to delete in the list view.

4. Click the **Delete SQC Alarm** button  on the tool bar.

You are prompted to delete the associated SQC alarm points.

5. Select corresponding check boxes to confirm the points you want to delete.
6. Click **Delete** to delete the SQC alarm point and specified associated points from the Data Archive server.

Note: Some alarms may share associated points, and deleting these points cause problems for other alarms that reference them. Verify these connections before deleting associated points.

Stale and Bad Points

Use the Stale and Bad Points tool to find and view stale and bad points, so you can diagnose the condition that produced the errors.

- The Stale state indicates that the PI point has not updated within a specified time. By default, a tag is stale if the current value is over four hours in the past.
- The Bad state indicates that the current value of a PI point is a digital state from the [System digital state set](#) in the archive. These values are inserted into the current value when an error condition occurs. For example, a state of I/O Timeout is a common symptom when a PI interface determines it has lost contact with the device it is configured to read.
- Not all digital states indicate errors. For example, the Scan Off digital state marks points that are no longer used, such as points from obsolete equipment, while the PtCreated digital state indicates a new point that has not yet begun collecting data. You can exclude specific digital states when you search for stale or bad points.
- You cannot use the Stale and Bad Points tool to diagnose future PI points.

Note: For information on monitoring and troubleshooting Data Archive, see the *Data Archive Administration* guide.

Find malfunctioning points

The Stale and Bad Points tool identifies points that have not received data for a long time or that have current values representing error or bad conditions, such as I/O timeout, PtCreated, bad input, and Shutdown.

When you find bad or stale points, research the reason for the condition. Some possible scenarios are:

- No network connection between the Data Archive server and the interface.
- The interface computer has been shut down or has lost connection with the device.
- Someone has changed the point attributes.

If point values are stale or bad for no known reason, you should immediately determine the cause.

When you find points that are no longer useful, such as points that represent data from obsolete equipment, decommission them.

Decommission a point with Point Builder

1. Start PI SMT and select the Data Archive server for that point.
2. Under **System Management Tools**, select **Points > Point Builder**.
3. Search for the point.
4. Click the **Archive** tab.
5. Under **Scan**, select **Off**.

Decommission multiple points with PI Builder

1. Open a spreadsheet in Microsoft Excel.
2. Click **PI Builder**.
3. Select the appropriate data server, asset server, and database.
4. Click **PI Points > Find PI Points**.
5. Search for and select the points you want to decommission. In the Select Object Types and Column Headers window, be sure that **scan** and **pointsource** are selected.
The points display on the spreadsheet.
6. In the **scan** column on the spreadsheet, change **1** to **0** for all the points.

Note: Some interfaces use the pointsource bit and not the scan bit to turn off points. To decommission points for such an interface, change the **pointsource** attribute to a value that you use only for decommissioned points.

7. To publish the changes:
 - a. Click **Publish**.
 - b. In the **Edit Mode** list, select **Edit only**.
 - c. Click **OK**.

Search parameters for stale or bad points

Use the following parameters to search for stale and bad PI points.

Search Option	Description
Show	Specifies whether to retrieve stale tags, bad tags, or both.
Stale	Specifies the <i>time range</i> that defines when a PI point is stale. Default is 4 hours. You can use PI time expressions to define the range.
Exclude tags with scan off	Excludes decommissioned PI points, that is, points that are not collecting data because their scan point attribute is set to Off.
Bad States	Specifies whether to include all bad states, or only <i>specific</i> bad states that indicate errors. For example, you might want to include points with the I/O Timeout state and exclude points with the PtCreated state. To select specific states, select the Specific States option button, click the Specific States button, and then use the dialog to select the states to include or exclude. You need to specify a Data Archive before you can select the Specific States option button.

Search Option	Description
Tag Search	Selects whether to perform a Quick Search with limited criteria for selecting points, or to use the Tag Search dialog (Advanced search): For Quick Search, click the Quick Search tab, specify a Tag mask , a Pointsource , and a Point Class , and then click  . For Advanced search, click the Advanced Search tab and then click Select Tags . Use the Tag Search dialog box to specify the PI points you want. See Use Tag Search .

To repeat the most recent search, click the **Quick** or **Advanced** tab and then click  on the toolbar.

View Stale and Bad Points

The following table shows the columns of data in the list of state and bad points that you retrieve:

Column	Description
Server	The Data Archive where the point resides.
Collective	The PI Collective to which the server belongs, if applicable.
Stale	An X indicates the point is stale.
Bad	An X indicates the point is bad.
Tag	The tag name for the point.
Pointsource	The point source of the point.
Timestamp	The timestamp associated with the point.
Value	The value of the point.

Note: Right-click anywhere in the point list to restart the search (**Go**), to **Export** a list of points to a CSV file, or to **Copy** the point list to the Clipboard.

Export a list of stale and bad points to file

- Under **Collectives and Servers**, select the server or collective.
- In the **System Management Tools** pane, select **Data > Stale and Bad Points**.
- Click the **Export** button .

The list of stale and bad points is exported in CSV format.

TCP Response

Note: This interface is no longer included with PI Server. If you want to use this tool, you will need to download and install this interface separately.

Use the TCPResponse tool to measure the response time of various network services.

The TCPResponse tool can obtain the actual result (not the response time) of a DNS operation.

The TCPResponse tool facilitates the creation and maintenance of TCPResponse points. It also supports HA (High Availability) Data Archive. Use the tool to browse TCPResponse points in primary and secondary Data Archive servers, and create and edit TCPResponse points in primary Data Archive servers.

Create a TCPResponse point

1. Under **Collectives and Servers**, select the server or collective.
The Data Archive server is added to the Point browser.
2. Expand the tree. If the TCPResponse interface is not managed using ICU, add it to the server manually by right clicking on the server.
3. Configure the default point properties in Point Property pane.
4. Right click on the interface and select **Create New Point** in the resulting menu.
After you enter the host name or IP address of the desired target, a new TCPResponse point is added in Point Browser. If you have multiple points to create, use import function () of Network Node List.
5. Select the new point in the Point browser, then check and edit the values of its properties in Point Properties pane.
6. Click the Save button () to save the new TCPResponse point to the Data Archive server.

Using the PI TCPResponse plug-in

The plug-in has three main function components:

- Point browser
- Network Node list
- Point Property pane

Details of the layout and functions of each component are explained in the following sections.

Point Browser

The Point Browser shows the hierarchy of Data Archive servers, interfaces and TCPResponse points. Click a point node to select a TCPResponse point, use Ctrl or Shift key to select multiple points, or use Alt key to select points under the selected interface or server. When points are selected, you can check or edit detail properties of the

selected points in Point Property Pane. In Point Browser, you can also change the name of a single point by clicking the point or using F2 key.

When the plug-in starts up, TCPResponse interfaces that are managed by ICU will be loaded to Point Browser automatically. For TCPResponse interfaces that are not managed by ICU, you can add the interface to Point Browser manually. right-click on a server. A context menu appears to let you manually add an interface. You have the following two options:

- **Open Configuration File**

Open the configuration .bat file, which is used to configure the TCPResponse interface.

- **Manually Enter Interface Information**

Enter interface information, such as the interface name, point source and interface ID.

Once a TCPResponse interface is added, the plug-in will search the Data Archive server for TCPResponse points belonging to the interface and show these points in Point Browser.

Note that the manually added interfaces have a different icon () compared to the icon of interfaces that are managed by ICU ().

The information of manually added interfaces are stored in local machine where SMT host is running. These interfaces will be automatically loaded when the same Windows user starts up PI TCPResponse Plug-in in SMT host. To remove manually added interfaces, right-click the interface node and click **Remove Interface**.

When you right-click an interface, you see a context menu that includes the following options:

- **Add New Point (Ins)**

Add a new point in the selected interface. You need to enter the host name or IP address of the target of the new TCPResponse point. This option will be disabled if the interface belongs to a secondary Data Archive server.

- **Paste Points (Ctrl+V)**

Paste previously cut or copied points to the selected interface. See the context menu of a point to find out how to cut and copy points. This option will be disabled if the interface belongs to a secondary Data Archive server.

- **Interface Information**

Show the configuration of selected interface.

Note that scan classes of interfaces not managed by ICU will not be listed in the dialog box.

- **Remove Interface**

Remove the selected interface from the browser. Note that interface managed by ICU cannot be removed.

You can right-click on a point to edit and select points. The available menu options include:

- **Cut (Ctrl+X)**

Cut selected points. This option will be disabled if the interface belongs to a secondary Data Archive server.

- **Copy (Ctrl+C)**

Copy selected points.

- **Delete (Del)**

Delete selected points. Note that this action removes the selected points from Data Archive permanently.

This option will be disabled if the interface belongs to a secondary Data Archive server.

- **Rename (F2)**

Rename the selected point. This option is only available when you select one point, and the point does not belong to secondary Data Archive server.

- **Select All (Ctrl+A)**

Use this option to select all points in the tree.

- **Search Points (Ctrl+F)**

This option brings up Search Points dialog box. Its function is the same as the **Advanced Search** button ().

The key combinations shown in the bracket are the keyboard shortcuts for these functions. After you use Cut and Copy functions, you can paste these points to an interface. Note that, a list of points' name is also copied to the system clipboard, so you can paste these points' name to other tools, such as Microsoft Excel.

Save button

The Save button () saves the modified or new points in the Point Browser. The modified points are characterized by the icon (). The new points are characterized by the icon ().

Delete button in Point Browser

The Delete button () deletes the selected points in the Point Browser. Note that these points will be removed from Data Archive permanently, and all archive data of these points will be discarded.

Create point button

The Create Point button () creates a new point in the selected interface. After you click this button, a new point will be added to the selected interface, and you will be asked to enter the host name or IP address of the remote machine. Default point properties will be applied to the new point.

Edit default point properties button

The **Edit Default Point Properties** button () typically opens a dialog to let you configure the default point properties, which will be applied to new points added to Data Archive. The button is usually located in the System Management Tools (SMT) toolbar, often under **IT Points > Ping** or within the Points section, depending on your version or SMT display settings. However, its exact location may vary based on your SMT version, screen resolution, or display configuration. These default properties are stored on the local machine where the SMT host is running and are automatically loaded when the same Windows user starts up the **PI TCPResponse** Plug-in on the SMT host.

You can use **Reset** button to set the default point properties to their program set values, which are loaded when the plug-in is started at the first time.

Among the default point properties, two properties, Tag and Target Setting, need to be configured specially. Target Setting property in the default point properties is used to store the information of the remote machine, including Device, Port, Input and Reply. You can set these default values that will be applied to new points.

However, The properties Device or Input will be updated automatically when you create a new point for a remote machine.

The automatically-generated Tag has the following format:

[Prefix][Delimiter][TCP Source][Delimiter][TCP Target]

- **Prefix**

The prefix used in the name of new TCPResponse points.

- **TCP Source**

The name of the source that sends out the service request. You can use the Data Archive name or interface node name as the name of the source. You can also manually enter a name of the source.

- **TCP Target**

The name of target that the service request is sent to. You can choose to use either the host name or the IP address of the remote machine as the name of the target.

- **Delimiter**

The delimiter is used to separate Prefix, TCP Source and TCP Target. It must be a single character and valid for point name.

The background of above four fields will turn to yellow when you type in any invalid character for a tag. After you finish editing the tag configuration, use Enter key or click outside the tag configuration dialog to accept the changes, or you can press Esc key to cancel all changes.

After you finish making changes on the default point properties, click the OK button or close the dialog to save these changes, or you can click Cancel button to discard all changes. Note that these changes will be saved in local machine once you click **Accept**.

Search points

The TCP response tool can search all points shown in Point browser and select the first point containing the keys that you type in the **Find** text box.

You can find the next point containing the keys by clicking **Find Next** . If you click the drop-down button beside the **Find Next** button, you can select **Advanced Search**  which opens the Search Points dialog.

Use the Search Points dialog to browse through or select all points that match the specified search criteria.

Refresh button

The Refresh button () causes the plug-in to re-load the Data Archive servers, interfaces and points listed in Point Browser. A warning message will pop up to remind you to save the modified points. If you decide not to save the changes, these changes will be discarded permanently.

Network node list

In the Network Node list, you can manage a list of network nodes by entering the host names or IP addresses of desired target, or importing a list of targets. You can create new TCPResponse points based on the list.

New node button

The New Node button () adds a new node in Network Node list. You need to enter the host name or IP address of the remote machine. You can also add a new node by double clicking empty rows in the list. The names of new points are created automatically based on the default point properties. You can edit the host name, IP address, tag and service type by clicking each entry in the Network Node list. A context menu is available to change the service type of a collection of nodes. Select nodes in the list and right click, you will see a context menu to change service type. If the **Auto Measure** box is checked, the content in the **Value** column is refreshed, which may take a certain amount of time.

Delete button

The Delete button () removes the selected nodes in the Network Node list.

Create button

The Create button () adds the selected nodes in the Network Node list to the selected TCPResponse interface. These points are not created in the Data Archive server until you save them.

There are two ways you can create new TCPResponse points for the selected network nodes to a desired Data Archive server:

1. Select the network nodes that you want to create TCPResponse points
2. Select an interface or any single point under an interface in Point Browser.
3. Click the Create button () to create TCPResponse points for these nodes in the server.

You can also use mouse to drag selected nodes in Network Node List and drop to the desired interface directly. Similarly, you can drag selected points in Point Browser to Network Node List. Note that you cannot drop points to a secondary Data Archive server in a collective.

Note: If a point already exists with the same name on the selected Data Archive server, or another new point with the same name was already added to the Point Browser under the same Data Archive server, the new point will not be able to add to the tree, and it will stay in the Network Node list and be highlighted. A message will pop up and summarizes the number of network nodes that failed to create points.

Import button

The Import button () brings up an open file dialog. You can select the CSV file that contains a list of targets, which can be host names or IP addresses. During the process of importing, you can click the stop button () to stop the importing process. After the importing process is finished, a message will be shown in the Session Record of SMT host to summarize the number of network nodes imported.

Note: Each line in the CSV file should contain only one target. You can also import a previous exported list (see below).

Export button

The Export button () is used to export the Network Node List to a CSV (comma separated values) file. You can import this file to the plug-in again later.

Stop button

The Stop button () is used to stop the process of importing a list of network nodes.

Resize button

The resize buttons ( and ) are used to hide or show the Network Node List. If you click the hide button ()

, the Network Node List is minimized as a title bar at the bottom of Point Browser, shown as



. Click the Show button () or double click the title bar to show the Network Node List again.

DNS check box

If this option is enabled, the plug-in will try to resolve the host names or IP addresses you entered using default DNS server. If the resolve fails, an error message of "DNS lookup failure" will be shown in the Note column.

Test Measure check box

If this option is enabled, the plug-in will try to measure the response time of servers of each network node you entered in Network Node List, and display the value in the Value column. In the Timeout textbox, you can manually change the value of TCPResponse timeout used in measurement. The default Timeout value is 2000ms.

Point Property pane

The following is the layout of the Point Property Pane:

PI-TCPResponse

Tag	TCP_Bruin.osisoft.int_bruin_4
Point Type	Float32
Point Source	P
Eng Units	
Interface	Bruin :: pitcpresp1
Service Type	HTTP
Timeout Duration	0
Scan Class	00:10:00
Enable Debug	False
Target Setting	DEVICE=bruin; bruin
Device	bruin
Port	
Input	
Reply	
Archive	
Typical Value	50
Zero	0
Span	100
Scan	ON
Archiving	ON
Step	OFF
Shutdown	OFF
Compressing	ON
Exception Deviation	
Compression Deviation	
Classic	
General	
Security	
Snapshot	
System	

Target Setting
[InstrumentTag] The attribute holds various keywords/value pairs that, together with Service Type [Location2], uniquely define the operation that the Interface performs.

The Point Property Pane shows the detailed configurations of selected points. Click each property to see its description at the bottom of the pane. You can edit its value on the right side of the property name. If multiple points are selected, the change of a property value will be applied to all selected points. Note that all properties will not be editable if any selected points belong to a secondary Data Archive.

The toolbar in Point Property Pane is used to sort the properties. Click the button to sort point properties in alphabetical order, or click the button to sort them in categorized order. The label beside these two buttons shows the number of points selected.

Right-click a property to open a context menu with two functions: reset property value, and hide/display property description. If the value of a point property is different to its default value, the value is shown with bold font.

Note: If you select multiple points, some properties are not available to edit. For example, the Tag property can not be changed when multiple points are selected. If the selected points belong to different interfaces, the Scan

Class property is also not editable.

Supported Data Archive versions

The PI TCPResponse Plug-In for SMT 3.2.0.0 and above is supported on the following Data Archive versions and platforms:

Server Version	Minimum Supported Version
PI 3	PI 3.2.357.17
PI 2	Not Supported
Interface Nodes	Not Supported
PINet	Not Supported

Totalizers

The Totalizers tool allows you to create and edit calculations performed by the PI Totalizer Subsystem. PI Totalizers calculate data on a PI point in the snapshot and store the results in another PI point, ensuring accuracy by using data before compression. This makes PI Totalizers ideal for accurately representing production summary data. You can start and reset totalizers based on time or events, ensuring precise calculations for flow volumes and other critical variables essential for monitoring product transfers or production performance.

Note: PI Totalizer calculations can only use data collected up to the current moment. Do not include future-dated PI points.

To open the **Totalizers** tool, navigate to **Points > Totalizers** in the System Management Plug-ins panel. The **Totalizers** window opens.

Search for a PI totalizer

Procedure

1. In the Collectives and Servers pane, select the server or collective.
2. In the System Management Tools pane, select **Points > Totalizers**.
3. Enter a valid string in the **Tag Mask** field. To see all totalizers on the selected servers and collectives, leave the default value, asterisk (*).
4. Click the **Search** button .

Key parameters for PI Totalizer tags

PI Totalizer tags are essential for accurately summarizing and analyzing data over specified periods. Understanding these parameters will help you set up and optimize your totalizer tags for precise data accumulation and reporting.

Totalizer tag configuration

Configure the following parameters to define how the totalizer gathers and processes data:

- **SourceTag:** The tag from which the totalizer will gather data.
- **Calculation Basis:** Specifies whether the totalizer should use event-weighted or time-weighted calculations.
- **Time Interval:** Defines the period over which the data will be accumulated.
- **Output Tag:** The tag where the totalized value will be stored.

Sampling and reporting

Set these parameters to control the frequency of data sampling and reporting:

- **Sample Interval:** The frequency at which the source tag is sampled.
- **Report Interval:** How often the totalizer writes the totalized value to the output tag.

Optional configurations

Use these optional settings to refine the totalizer's behavior:

- **Reset Condition:** Conditions under which the totalizer should reset its calculations.
- **Deadband:** The minimum change in value that triggers a new totalizer calculation.
- **Filter Expression:** A condition that must be met for the totalizer to include a value in its calculation.

For more information, see [PI Totalizer Configuration on PISquare](#).

Types of summary calculations

Use summary calculations when you need to aggregate and analyze data points over a specified period to gain insights into trends, performance, and anomalies. These parameters are set in the Totalizer tool when you create or edit a totalizer.

When to set these parameters

- **Initial Configuration:** When you are setting up a new totalizer, you will configure these parameters as part of the initial setup process.
- **Updating or Tuning:** You may need to adjust these settings if you find that the existing configuration does not meet your reporting or monitoring needs. This could be due to changes in operational requirements or to improve accuracy and performance.
- **Periodic Review:** Regularly reviewing and updating these configurations can help ensure that the totalizer calculations remain relevant and accurate as your data and operational contexts evolve.

Summary calculations

You can use PI Totalizers to perform these types of summary calculations:

- Total
- Average
- Minimum
- Maximum
- Range
- Standard Deviation
- Median

In addition, you can use PI Totalizers to perform these types of update event counts for a point:

- All Events

- Event Equal to a Value
- Event Not Equal to a Value
- Event Greater Than a Value
- Event Greater Than or Equal To a Value
- Event Less Than a Value
- Event Less Than or Equal To a Value
- Event change from Greater Than or Equal To Less Than
- Event change from Less Than to Greater Than or Equal To

Edit an existing PI totalizer

Procedure

1. In the Collectives and Servers pane, select the server or collective.
2. In the System Management Tools pane, select **Points > Totalizers**.
3. Select the **Name & Type** tab.
4. Type the name of the PI totalizer into the **Name** field.

Note: If you don't know the exact name, search for the PI totalizer and select it from the search results.

5. The configuration loads into the tabbed **Editor** pane, where you can make changes.
6. When finished, click **Save**  to save your results to the Data Archive server.

After you finish

For information on configuring PI Totalizer, see [PI totalizer configuration](#).

Create a new PI totalizer

Procedure

1. In the Collectives and Servers pane, select the server or collective.
 2. In the System Management Tools pane, select **Points > Totalizers**.
 3. Select the **Name & Type** tab.
 4. Click the **New Totalizer** button .
- This clears the currently-displayed totalizer, if any, from the editor.
5. On the **Name and Type** tab, select a Data Archive server from the **PI Servers** menu.
 6. Type a unique name in the **Name** field.
 - If you type in the name of an existing PI totalizer on the Data Archive server, then the **Totalizer** tool loads the configuration of the existing point. You are then editing that totalizer, rather than creating a new one.
 - If a point with the name you entered already exists on the Data Archive server but the point is not a PI Totalizer, then the **Save** button is not enabled.

7. To complete the **SourceTag** field, click the **Tag Search** button  and choose the source tag from the search results. For additional configuration information, see [PI totalizer configuration](#).
8. Use each tab in the Editor pane to define the appropriate sampling, reporting, and optional configurations for the new PI Totalizer.
9. Click **Save** .

The new PI totalizer is created on the Data Archive server.

Note: To use an existing PI totalizer as a template for a new one, search for PI totalizers, select one from the **Search Results** list, and type the name for the new PI totalizer in the **Name** field of the **Name & Type** tab. You can also select a different server after selecting an existing PI totalizer and create the same PI totalizer on that server.

PI totalizer configuration

After you either create a new PI totalizer (see [Create a new PI totalizer](#)) or find an existing PI totalizer to edit (see [Edit an existing PI totalizer](#)) the configuration for the totalizer is displayed in the Editor pane.

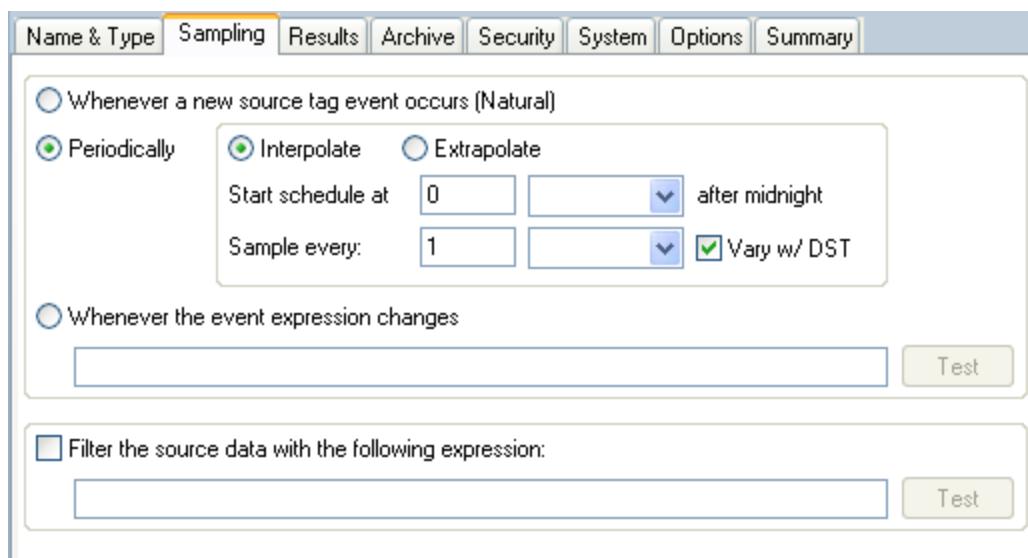
Specify the sampling

You specify how the **SourceTag** will be sampled on the Sampling tab. The choices you make here determine when the PI Totalizer will look at the **SourceTag**: whenever there is a new event, on a timed basis, or when an expression changes.

- To create a PI Totalizer that processes the value every time there is a new event for the **SourceTag**, select **Whenever a new source tag event occurs (Natural)**.
- To create a PI Totalizer to evaluate the **SourceTag** whenever an expression changes, select **Whenever the event expression changes**. You can test the expression by clicking the **Test** button.
- To create a calculation that samples the **SourceTag** on a timed basis (for example, every 20 minutes beginning at midnight, select **Periodically**.
 - Then select how the periodically sampled value is derived: **Interpolate** or **Extrapolate**. The time period when the sample is to be taken is not guaranteed to correspond to a new event on the **SourceTag** (unlike **Natural** sampling) so you must determine whether you will **extrapolate** the previously recorded value or wait until the next value and **interpolate**.
 - For new PI Totalizers, the **Totalizer** tool uses the **Step** attribute of the **SourceTag**. If **Step** is turned on, the tool will default to **Extrapolate**. If **Step** is turned off, the tool will default to **Interpolate**. For details, see [Set archive attributes](#).

Caution: You may override the default, but you should use caution due to possible data representation conflicts.

- If you edit an existing PI Totalizer with timed sampling, the tool displays the configured choice.



To filter the source events that the PI Totalizer uses, check the **Filter source data with the following expression** option. Then enter an expression to filter incoming **SourceTag** events. For example, you can look at a flow only when the fresh water valve is closed. You can test the expression by clicking the **Test** button.

Specify how to write results

1. In the Collectives and Servers pane, select the server or collective.
2. In the System Management Tools pane, select **Points > Totalizers**.
3. Select the **Results** tab.
4. Select one of the following options in the **Write final results** section:
 - **After a time period elapses:** Specify the start time and the duration of the time period. You can also specify whether or not you want the PI Totalizer to honor Daylight Saving Time (DST), that is, produce **n + 1** hour totals at the spring time change and **n - 1** hour totals at the autumn time change.
 - **Based on a trigger event:** Enter an expression in PI Performance Equation syntax and the PI Totalizer evaluates the expression each time any of the PI tags in the expression update. You can choose to have the PI Totalizer write results whenever the result of the expression changes, or whenever the expression evaluates to zero.
 - **After a number of source events:** Results are written when a specified number of **SourceTag** events occur. In this case, no interim results are written and only the final result is recorded.
 - **Continue forever [interim results ONLY]:** If you are building or editing a time-weighted moving total, maximum, or minimum summary calculation, you can choose to have the PI Totalizer continue forever. This behavior is like that of an integrator; you see interim results ramp up. If you choose the **Setable** option on the options tab you can write a new value to the PI Totalizer and force final results to be written. The PI Totalizer will start again using the newly entered value as its base.
5. Enter the appropriate values in the **Details** section.
6. Select an option in the **Write interim results** section.
7. The options are enabled or disabled depending on other configuration choices:
 - Stamped one second after start

- At source time [ramp]
- Projection based on source value — A new result is sent to the PI Snapshot as each **SourceTag** event is processed. The value is an estimate of the result if the rate point were to hold steady at its current value.
- Projection based on average — A new result is sent to the PI Snapshot as each **SourceTag** event is processed. The value is an estimate of the result if the rate point were to hold steady at the average observed so far in this accumulation interval.
- Do not write interim results

Set archive attributes

Configure the PI archive attributes on the **Archive** tab.

- **Typical value**

Enter a reasonable sample value for the point. For a numeric tag, a typical value must be greater than or equal to the Zero, and less than or equal to the Zero plus the Span.

- **Zero**

Enter the lowest possible value for the point, as the bottom of the range used for scaling float16 values and trends. Recorded values lower than the Zero value are recorded in the Data Archive server with the digital state Under Range.

A Zero value is required for numeric data type points, but not for non-numeric points. The instrument zero is a logical zero value, and certain interfaces require that Zero and Span values match the instrument system range. See your interface documentation for details.

- **Span**

Enter the maximum difference between the top and bottom of the point range, as a positive value. The Span value is used for scaling float16 values and trends and is required only for numeric data type points.

Recorded values above the sum of the zero and Span values are recorded in the Data Archive server with the digital state Over Range.

- **Scan**

Enable Scan if an interface that supplies data to the point requires a scan flag. Interfaces that require a scan flag do not update points when the flag is disabled. See your interface documentation for Scan attribute requirements.

- **Archiving**

Enable Archiving to store point values in the Data Archive server.

Step

Enable the Step attribute to treat archived point values as discrete units that would appear stepped on a chart. An archived value is assumed constant until the next archived value, and adjacent archived values are not interpolated. For example:

- at 12:00:00, the value 101.0 is archived
- at 12:01:00, the value 102.0 is archived
- a request for the value at 12:00:30 returns 101.0

Disable Step to treat archived point values as a continuous signal, and linearly interpolate adjacent values. For example:

- at 12:00:00 the value 101.0 is archived
- at 12:01:00 the value 102.0 is archived
- a request for the value at 12:00:30 returns 101.5

Note: By default, future PI points have their step attribute turned on (set to 1). This is because most future PI Points store discontinuous signals (for example, a series of discrete predictions) for which linear interpolation cannot be assumed. With the step attribute turned on, trends show a staircase pattern. Future PI points, by default, also have compression, excmin, excmax and excdev turned off (set to 0).

The Step attribute also affects compression calculations. When enabled, a linear change of value greater or equal to the compression deviation allows the point to pass compression.

The Step attribute is relevant only to numeric points. In general, data from continuous signals such as thermocouples and flow meters should be archived without stepping. Data from discrete measurements such as lab data and batch charge weight should be stepped.

To access the Step attribute setting:

1. **Open** PI SMT.
2. **Navigate** to Points > Point Builder.
3. **Search** for and **select** the point for which you would like to check the step attribute.
4. In the Archive tab, you can **view/change** the step attribute.

Shutdown

Use the Shutdown attribute to determine whether shutdown events are written. The timestamp of Shutdown events normally represents the actual shutdown time of the Data Archive server as recorded by the Snapshot Subsystem.

Beginning with PI Server PR1 SP1, shutdown events for most points are disabled. Unless you configure points to receive shutdown events, only test points such as sinusoid and sinusoidu will receive shutdown events.

- Enable Shutdown to record server shutdown events as point values with the timestamp of the server shutdown. If data is collected on the Data Archive server, shutdown events are helpful to clearly indicate gaps in data collection.
- Disable Shutdown when point data is collected on a distributed collection nodes, as buffering services collect, manage and retain the data until the server is up and running again. Disabled Shutdown PI points have a configurable attribute to determine whether shutdown events are written.

Note: The default behavior of Data Archive is to write the SHUTDOWN digital state to all PI points when Data Archive is started. The timestamp that is used for the SHUTDOWN events is usually updated every 15 minutes, which means that the timestamp for the SHUTDOWN events will be accurate to within 15 minutes in the event of a power failure.

Compression

Enable compression to apply compression algorithms and save only event values that indicate a change in point

value. When disabled, all values collected and sent to the snapshot, including redundant values, are saved in the PI archive.

Note: By default, future PI points have *compression* turned off (set to 0).

Enable compression for all real-time points. Digital, BLOB and string data type values pass through compression only when the value changes. Compression is typically turned off for points that collect sample data, such as lab data, or other sparse data streams.

Deviation

Exception and compression deviation algorithms can be used to filter erroneous and redundant data before it is recorded in the archive.

Note: By default, future PI points have *excmin*, *excmax* and *excdev* turned off (set to 0).

Each algorithm is based on three specifications:

- Deviation in point value
- Minimum duration of time
- Maximum duration of time
- These specification are used to determine which points are recorded to the snapshot from the buffer (Exception) and to the archive from the snapshot (Compression).

Note: Exception deviation and compression deviation filter externally-generated events for archiving. Under no circumstances does this cause Data Archive to generate events.

Exception deviation

Exception reporting is used to define the precision of a data stream, and the amount of deviation that constitutes a significant change. Most interface programs can execute an exception-reporting algorithm to determine when to send a point value to the PI Snapshot subsystem. An exception is an event that occurs either:

- After a specified minimum duration of time since the previous event, while exceeding a specified deviation in value from that event.
- After a specified maximum duration of time since the previous event.

This means that when activated, exception reporting filters events and stores only periodic values, including duplicates, unless an event represents a significant change in the short-term trend of values. An exception event, both timestamp and value, is sent with the previous event to the Snapshot.

- **Exception Deviation**

The deviation in value required to store an event, either as a number of engineering units, or as a percentage of the point's Span value. The exception deviation should be less than the compression deviation by at least a factor of 2, and is ignored for digital, string and BLOB data type points.

- **Min Time**

The minimum time that must elapse after an event before an exception value can be stored.

- **Max Time**

The maximum time that can elapse after an event before automatically storing the next event as an

exception value.

Set the minimum and maximum time values to *0* to turn off exception reporting.

Compression deviation

Once events are sent to the PI Snapshot subsystem, a compression algorithm can further filter data and reduce storage to only significant values as they are moved into the archive. An event is recorded:

- After a specified minimum duration of time since the previous event, if it exceeds a specified deviation in value from that event.
- After a specified maximum duration of time since the previous event.
- When activated, compression reporting filters events and stores only periodic values (including duplicates), unless an event represents a significant change in the short-term trend of values.

To turn off compression and archive every event that passes exception reporting, disable the *Compressing* attribute.

Compression Deviation — Enter the deviation in value required to record an event, either as a number of engineering units, or as a percentage of the point's *Span* value.

For most flows, pressures, and levels, use a deviation specification of 1 or 2 percent of *Span*. For temperatures, the deviation should usually be 1 or 2 degrees.

Min Time — Enter the minimum time that must elapse after an event before a compressed value can be recorded. The minimum time should be set to *0* if exception reporting is activated for the same point.

Max Time — Enter the maximum time that can elapse after an event before automatically recording the next event as a compressed value. The recommended maximum time is one work shift (for example, 8 hours). If this value is too low, the compression effects are too limited to save significant archive space. If this value is too high, useful data may be lost. Events that reach the Data Archive server in asynchronous order bypass the compression calculation and are automatically recorded to the archive.

The compression specifications consist of a deviation (*CompDev*), a minimum time (*CompMin*), and a maximum time (*CompMax*).

Events are also archived if the elapsed time is greater than the maximum time. Duplicate values will be archived if the elapsed time exceeds *CompMax*. Under no circumstances does this cause Data Archive to generate events; it only filters events that are externally generated.

The most important compression specification is the deviation, *CompDev*. For non-numeric tags, *CompDev* and *CompDevPercent* are ignored. They will be displayed by applications as zero.

Note: *CompDev* specifies the compression deviation in engineering units; *CompDevPercent* specifies the compression deviation in percent of *Span*. If you change one, the other is automatically changed to be compatible. If you try to change both at once, *CompDevPercent* takes precedence.

Define security settings

To view or configure the security settings configured for a Totalizer point, click the **Security** tab in the Totalizers tool and set the security just as you would for any other PI Point. See [Configure point security](#).

Configure optional functions

The **Options** tab contains a list of optional functionalities that can be used in your PI Totalizer. Options can be enabled or disabled, depending on the type of PI Totalizer that you are creating or editing.

- **Allow External Reset** — If you enable this option, you can write a new value to the PI Totalizer and force final results to be written. The PI Totalizer starts again using the newly entered value as its base.
- **Use negative source values** — By default, the PI Totalizer counts negative values of its **SourceTag** as **0**. If you select this option, you override the default behavior and count negative source values as negative numbers.
- **Source tag is a DCS integrator** — If the **SourceTag** is a DCS PI Totalizer (integrator), you can use this option. When enabled, the PI Totalizer senses when the integrator is reset on the DCS, automatically writes its final results, and restarts.
- **Close at end of the Sampling Period** — This option is valid for event driven PI Totalizers. Select this option if you must close event driven PI Totalizers with periodic sampling at the end of their sampling period, rather than waiting for the next **SourceTag** event.
- **Source OverRange is ZERO + SPAN** — If you enable this option, **SourceTagOverRange** updates will be counted as the **SourceTagZERO** plus its **SPAN**.
- **Use SourceTag BAD in place of "Bad Total"** — If you enable this option, the **SourceTag BAD** status is used in place of the System State "Bad Total" when the percentage of good **SourceTag** values is insufficient for a valid PI Totalizer result to be generated
- **Source UnderRange is: zero / bad** — If you enable this option, you count **SourceTagUnderRange** updates as either **BAD** or **ZERO**, depending on which option you select.
- **Final Result at: start/end/both** — Specify whether to have the total written at the beginning, end, or both beginning and end of the sampling period you select.
- **Conversion Factor** — The number entered here is a conversion factor to scale the results of the PI Totalizer. This is especially important for a time weighted PI Totalizer. The PI Totalizer acts as though the **SourceTag** is in engineering units per day. The final result of the PI Totalizer is in engineering units. If the **SourceTag** has a different time scale (for example, gallons/minute), a conversion must be made to get the engineering units result. The conversion factor should be the time scale of the **SourceTag**, divided by one day, divided by the time scale (for example, [**one min / (one day/1440 min)**] or **1440**). The default is **1.0**.
- **Source = Zero below** — **SourceTag** events with values less than the number entered here will count as zero for totalization.
- **Pct good values needed** — The number entered here is the percentage of **GOODSourceTag** updates required to produce a valid PI Totalizer result.

Turn off character validation in the Totalizer tool

As each letter of the tag name is typed, the **Totalizer** tool validates the tag name in the **Name** text box or the **SourceTag** text box. On systems with large point counts, this can consume a lot of time. If character-by-character validation is turned off (unchecked), then the tag validation occurs after the cursor leaves the text box.

1. In the Collectives and Servers pane, select the server or collective.
2. In the System Management Tools pane, select **Points > Totalizers**.
3. Click the **Options** button  and clear the **Validate tag after each letter is typed** check box.

Copy a PI totalizer to another Data Archive

Procedure

1. In the Collectives and Servers pane, select the server or collective.
2. In the System Management Tools pane, select **Points > Totalizers**.
3. Click the **Search** button .
4. Select a totalizer from the search results.
5. Choose the target Data Archive server from the **PI Server** menu in the **Name and Type** tab.

Note: If a PI Totalizer with the same name exists on the target server, its configuration loads into the **Editor** pane.

6. Click the **Save**  button.

Note: The **Save** button is disabled if PI Totalizer exists on the target server or if the **SourceTag** does not exist on the target server. The toolbar **Save** button is enabled and the PI Totalizer can be saved to the target server.

The PI totalizer is created on the Data Archive server.

View information about a PI totalizer

The **System** tab is a read-only tab that displays information about who created the point, when it was created, who last changed the point, and so on. The **System** tab also displays the current snapshot value of the point.

The **Summary** tab provides the details of the PI totalizer, as specified in the other tabs on the **Editor** pane.

Tuning Parameters

Use the Tuning Parameters tool to view and edit Data Archive settings for all connected Data Archive servers. Tuning parameter settings are also known as Timeout table parameters. PI SMT displays the tuning parameter sorted by name.

Use tuning parameters to:

- Configure the Data Archive server authentication policy
- Set the maximum size of files, such as auditing and message log files
- Set the amount of time that lapses before a secondary member of a Data Archive Collective is marked as unavailable

To edit tuning parameters, you need read/write permissions to the PITUNING entry in the Database Security tool. To open the **Tuning Parameters** tool, select **Operation > Tuning Parameters** in the System Management Tools pane.

Note: If the Data Archive server is a member of a collective, any tuning parameter changes you make are not replicated on other members of the collective.

Configurable tuning parameters

Use the Tuning Parameters tool to view and edit settings for all connected Data Archive servers. To open the Tuning Parameters tool, select **Operation > Tuning Parameters**.

By default, each tab in the Tuning Parameters tool provides a list of the most commonly used server settings for each category. When you select a tuning parameter, a description of its function is shown in the lower pane.

Note: When you select a tuning parameter, a description of what it controls is shown in the field below.

Settings are displayed on these tabs:

- **General:** Command line tool and server application settings.
- **Archive:** PI Archive Subsystem settings.
- **Backup:** PI Backup Subsystem settings.
- **Base:** PI Base Subsystem settings, including module database parameters.
- **Net Manager:** PI Network Manager settings.
- **Snapshot:** PI Snapshot Subsystem settings, including event queue settings for buffered values that are not yet archived.
- **Update Manager:** Update Manager settings that affect programs that sign up for point or data updates, including ProcessBook and most interfaces.
- **Security:** Security settings that affect authentication, PI identities, and mappings.

If the tuning parameter that you want is not displayed in these lists, then you can add it.

Add a tuning parameter to the list

By default, only the most commonly used tuning parameters are displayed in the PI SMT Tuning Parameters list. If you need to modify a tuning parameter that is not on the list, contact the [Customer Portal](#).

Procedure

1. In the Collectives and Servers pane, select the Data Archive server on which you want to add the parameter.
2. Clear the check boxes for all other Data Archive servers.
3. On the toolbar, click the **New Parameter** icon ().
4. In **Parameter name**, select the parameter that you want to add to the list. If you know the name, you can enter it exactly.
5. Optional: Enter a value for the tuning parameter.
6. Click **OK**.
The parameter is added to the list.
7. Stop and restart the Data Archive server or the subsystem associated with the updated parameter.

Edit tuning parameters

Procedure

1. Under Collectives and Servers, select the server or collective.
2. In the System Management Tools pane, select **Operation > Tuning Parameters**.
3. Double-click a row to edit a setting, or, select a row, right-click and select **Edit** from the context menu.
4. Select a setting name from the menu to change the setting that is saved, if desired.
You can add parameters that are not included by default in the Parameter name menu, provided the setting exists on the source Data Archive server.
5. Edit the setting value and click **OK** or, click **Apply** to save changed values and continue using the setting dialog box.

After you edit tuning parameters, the changes do not take effect until you stop and restart the Data Archive or the subsystem associated with the updated parameter. Clients that are connected to the Data Archive server while you edit the tuning parameters do not reflect your edits until you disconnect and then reconnect to the Server.

Note: Tuning parameters are not replicated. If this Data Archive server is a member of a Collective, the changes you make here will not be replicated on other members of the Collective.

Export a list of tuning parameters

You can export the list of parameters and their settings on a tab into a list of comma-separated values (CSV file).

Procedure

1. Under Collectives and Servers, select the server or collective.

2. In the System Management Tools pane, select **Operation > Tuning Parameters**.
3. Click **Export** .
4. Save the CSV file to the location of your choice.

You can view the CSV file in a text editor or in a spreadsheet.

Update Manager

Use the Update Manager tool for troubleshooting as directed by AVEVA Technical Support.

Update Manager keeps track of *producers* of updates (such as the snapshot and archive) and *consumers* of updates (such as interfaces). It lets clients and other consumers know when a specified value is updated.

View consumer details

Procedure

1. From the System Management Tools pane, select **Operation > Update Manager**.
2. In the right pane, from the **Server** drop-down list, select the Data Archive server that you want.
3. Select the **Consumers** tab. The consumers for the selected server are listed in this tab. See [Consumer columns](#) for an explanation of the columns in the output.
4. To view detailed information on a specific consumer, click the name of the consumer and view the information in the lower pane. See [Consumer details](#) for an explanation of the columns in the output.
5. To troubleshoot, look for the following information.
 - Is it the consumer registered?
 - Is the consumer timed out?
 - Is the consumer signed up?
 - When was the last time the consumer retrieved an update?

Consumer columns

Column	Description
Consumer	Name of the consumer
Connection ID	Consumer connection ID number.
Connection	Description of the consumer. The information in this column displays solely when the Query for additional information check box in the Options dialog is selected.
Persist	References whether signups persist when you restart the Update Manager subsystem.
Status	Shows whether a consumer is registered to receive updates. A value of Connected means that the consumer is registered.

Column	Description
Last Time	The last time the consumer collected events.
Pending	Events waiting for the consumer to retrieve.
Signups	Signups for the consumer.
Changes	Modifications to signups for the consumer.

Consumer details

Column	Description
Producer	The source of update notifications. Currently there are five producers. PI Snapshot Subsystem is a producer of snapshot events. PI Base Subsystem is a producer of Point Database and Module Database changes. PI Archive Subsystem is a producer of archive changes. PI Batch Subsystem is a producer of Batch Database changes.
Consumer	An application that currently signed up as a consumer of specified producer. For PI API applications, the consumer name is usually the first four letters of the login name of the user running the application. These names are not unique so the PI Update Manager assigned ID is appended to the name. PI API applications also have the PI Network Manager ID appended. These integers are appended to help you find specific consumers. For the PI SDK, the consumer name is the complete application name with a colon and a PI SDK supplied identifier followed by a pipe character and a PI Update Manager assigned ID.
Qualifier	A producer specific integer. For example, Snapshots update stores the requested point ID in the qualifier. When the Query for additional information check box in the Options dialog is selected, this column displays the tag name (Tag point attribute).
Flag	A producer-specific value.
Pending	Number of events available for the consumer to retrieve. The value goes up and down as events come in and the consumer pulls them out. Values that increase continuously might indicate that the consumer is not working properly or disconnected.

Column	Description
	The Pending number shows a maximum of 50000 events, even if more events are in the queue. You can configure this limit with the MAXUPDATEQUEUE Tuning Parameter. The Update Manager might limit individual consumers from accumulating too many pending events. This is a display limitation and does not imply data loss.

View producer details

Procedure

1. Select **Operation > Update Manager** from the System Management Tools pane.
2. In the right pane, from the **Server** drop-down list, choose the Data Archive server or collective that you want.
3. Select the **Producers** tab.

See [Producer columns](#) for an explanation of the columns in the output. To troubleshoot, look for the following information.

- Is the producer connected?
 - Is the producer sending updates?
 - When is the last time it sent an update?
 - How many sign-ups does it have?
4. To view detailed information on a specific producer, click the name of the producer and view the information in the lower pane.

See [Producer details](#) for an explanation of the columns in the output. To troubleshoot, look for the following information.

- Send failures
- Retrying
- Is the producer responsive?

Producer columns

Column	Description
Producer	Name of the producer
Last Time	The last time the producer was updated.
Status	Shows whether a producer is registered to send updates. A value of Connected means that the consumer is registered.

Column	Description
Signups	Number of signups for the producer.
Remove	Number of signups for which the producer no longer needs to send updates.
Pending	Events waiting for the producer to retrieve.

Producer details

Column	Description
Parameter	Producer parameter name.
Value	Producer parameter value.
Change	Status of parameter value since the last time the information was refreshed, either through automatic update  or the Refresh button  on the toolbar.

Producers and associated subsystems

Producer	Description	Subsystem
Snapshots	Snapshot	Snapshot
Archive	Archive	Archive
PtUpdates	Point updates	Base
MDBUpdates	Module database	Base
PIChangeRecordUpdates	Changes for replication	Base
DigitalSets	Digital sets	Base
BDBUpdates	Batch database updates	Archive
PIBatchUpdates	Batch updates	Archive
PIUnitBatchUpdates	Unit batches	Archive
PIUnitBatchOnUnitUpdates	Unit batch updates for a specific unit	Archive
PICampaignUpdates	Campaigns	Archive

Producer	Description	Subsystem
PITransferRecordUpdates	Transfer records	Archive
TimeSeries	Combines both archive and snapshot updates	Archive and Snapshot

View statistics summary

To view a summary of Update Manager statistics for a specific Data Archive server:

Procedure

1. Select **Operation > Update Manager** from the System Management Tools pane.
2. From the **PI Server** drop-down menu, choose the Data Archive server that you want to view. This menu is populated with all the Data Archive servers and collectives selected in the Collectives and Servers pane.
3. Select the **Statistics** tab. The statistics for the selected Data Archive server are listed in this tab. See [Statistics columns](#) for an explanation of the columns in the output.

Statistics columns

Column	Description
Parameter	Update Manager parameter name.
Value	Update Manager parameter value.
Change	Status of parameter value since the last time the information was refreshed, either through automatic update  or the Refresh button  on the toolbar.

Change the update refresh rate

Procedure

1. Select **Operation > Update Manager** from the System Management Tools pane
2. Click  on the toolbar.
The Options dialog box appears.
3. In the auto-update rate field, type in the refresh rate, in seconds.

Reduce displayed content for high-latency connections

By default, the Update Manager tool displays details on consumers. To retrieve these details, the Update Manager tool queries the Data Archive server, which requires extra time. If consumers are connected to the Data Archive server through a high-latency connection, then reduce the level of detail displayed in the Update Manager.

Procedure

1. Select **Operation > Update Manager** from the System Management Tools pane.
2. Click  on the toolbar.
3. The Options dialog box appears.
4. Clear the **Query for additional information** check box to reduce the level of detail displayed.

SMT security requirements

Alarm groups security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
View Alarm Groups and Points	PIPOINTS (r), PtSecurity (r)
View Alarm Groups and Points Status	PIPOINTS (r), PtSecurity (r), Datasecurity (r)
Specify Point Sources	PIPOINTS (r), ptsecurity (r)
Add Alarm Groups	PIPOINTS (r,w)

Archive editor security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
Retrieve Point List	PIPOINT(r); PtSecurity (r) is also required for each point
Write events	PIPOINT(r); DataSecurity (r,w) is also required for each point

Archives security permissions

Permission Type	Requirements
File System	Administrative share privileges and access to archives for some Data Archive-SDK combinations.
Registry Access	Read access to HKLM\Software\PISystem on local and remote computers.
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
List archives	PIARCDATA (r)
Create/Register/Unregister/Force Shift	PIARCADMIN (w)

Note: Archive backup is obsolete for this tool in the current version. Use the **Backups** tool instead.

AutoPointSync security permissions

This tool only creates the root PIModule the first time it runs. This requires PIMODULES (r,w) and module (r,w) to **\%OSI**.

Subsequent uses of this tool requires PIMODULES (r), module (r,w) to the specific root module only.

Root Module used: **\%OSI\AutoPointSync** .

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
List Interfaces	PIMODULES (r), module (r) Note: module (r) also assumes (r) for all modules along the hierarchy path above it.
Enable/Disable Synchronization	PIMODULES (r), module (r,w) Note: module (r) also assumes (r) for all modules along the hierarchy path above it.

Backups security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
View Backup History	PIBACKUP (r)
Perform Backups	PIBACKUP (r,w)

Batch custom names security permissions

This tool only creates the root PI Module the first time it runs. This requires PIMODULES (r,w) and module (r,w) to **\%OSI**.

Subsequent uses of this tool require PIMODULES (r), module (r,w) to the specific root module only.

Root Module used: **%OSI\BatchView\CustomNames**

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
View custom name sets	PIMODULES (r), module (r) Note: module (r) also assumes (r) for all modules along the hierarchy path above it.
Create and delete custom name sets	MDB access to write. PIMODULES (r,w), module (r,w) Note: module (r) also assumes (r) for all modules along the hierarchy path above it.
Edit custom name sets:	PIMODULES (r), module (r,w) Note: module (r) also assumes (r) for all modules along the hierarchy path above it.

Batch database security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
View PIUnitBatch, PISubBatch	PIModules (r) module (r) Note: module (r) also assumes (r) for all modules along the hierarchy path above it.
Create/Edit/Delete PIUnitBatch, PISubBatch	PIModules (r); (r) is also required for all parent objects along the hierarchy path
View PIBatch	PIBatch (r)
Create/Edit/Delete PIBatch	PIBatch (r,w)

Batch generator security permissions

This tool only creates the root PIModule the first time it runs. This requires PIMODULES (r/w) and module (r,w) to %OSI.

Subsequent uses of this tool require PIMODULES (r), module (r,w) to the specific root module only.

Root Module used: %OSI\

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
PiBaGenMake a module a Batch Unit	PIModules (r) module (r,w) Note: module (r) also assumes (r) for all modules along the hierarchy path above it.
Create new module	PIModules (r,w) parent module (r,w) Note: module (r) also assumes (r) for all modules along the hierarchy path above it.

Current values security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
Current values	PIPOINT (r)
Each point also requires	DataSecurity (r) and PtSecurity (r)

Database security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
View Database Security settings	PIDBSEC (r)
Edit Database Security settings	PIDBSEC (r,w)

Digital states security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
View Digital States, State Sets	PIDS (r)
Create/Edit/Delete Digital States, State Sets	PIDS (r,w)
Create Digital Points	PIPoint (r,w)
	Note: PIPOINT (r) includes PIDS (r)

Firewall security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
Configure PI Firewall	PITUNING (r,w)

Identities, users, & groups security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
Create/edit/delete users, groups, identities	PIUSER (r,w)
View users, groups, identities	PIUSER (r)

Interface list security permissions

This tool creates only the root PIModule the first time it runs. This operation requires PIMODULES (r,w) and module (r,w) permissions on **\%OSI**.

Subsequent uses of this tool require PIMODULES(r), module (r, w) permissions on the specific root module, only.
Root Module used: **\%OSI\Interfaces**.

Permission Type	Requirements
File System	Requires local and remote read permissions on the interface executable file to retrieve interface file version.
Registry Access	None required.
Service Control Manager	Requires read and service control privileges on local and remote computers. For Windows 7 and Windows Server 2003, and later, you are prompted to restart SMT as an administrator when this is attempted by the operating system.

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
View Interface List	PIModules (r), module (r) <u>Note:</u> module (r) also assumes (r) for all modules along the hierarchy path above it.
Configure Interface Services	None required; write access not needed.

IT Organizer security permissions

This tool creates a hierarchy module tree.

Requires PIMODULES (r,w) and module (r,w)* to **\%OSI**.

Root module used: **\%OSI**.

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
Each PI point	<p>PIPOINT (r), DataSecurity (r) and PtSecurity (r)</p> <p>Note: module (r) also assumes (r) for all modules along the hierarchy path above it.</p>

Licensing security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
View licensing information	None required

Message Logs security permissions

Permission Type	Requirements
File System	If PIPC.Log files are to be viewed, read access to these files is required. Read access to the \PI\log
Registry Access	If Data Archive message logs (pimsg_xxxxxxx.dat) are to be opened through the Open Dat or Log File option, the local pimsgss subsystem is used to parse these files. This plug-in reads the installation path for the pimsgss subsystem from the registry. The following registry keys on the SMT3 machine must have read access for the user account under which SMT3 is running: HKEY_LOCAL_MACHINE

Permission Type	Requirements
	HKEY_LOCAL_MACHINE\SOFTWARE\PISystem\PI
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
Requires access to the Data Archive log file via the PI SDK	PIMSGSS (r)

Mappings security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
Create/edit/delete mappings	PIMAPPING (r,w)
View mappings	PIMAPPING (r)

MDB to AF synchronization security permissions

Permission Type	Requirements
PI	AF Link read access to view statistics AF Link write access to enable tracing or to reset MDB PIModules write access when you edit settings Write access to %OSI\MDB-AFMigrationData when you edit settings. To run the preparation wizard, you need additional access permissions.
AF	You need to be able to read the AF server, AF database, and Data Archive Element. To run the preparation wizard, you need additional access

Permission Type	Requirements
	permissions.
File System	None required
Registry Access	None required
Service Control Manager	None required

Access permissions required to run the preparation wizard

The PI MDB to AF Preparation wizard requires certain access permissions in order to prepare for migration.

When running the preparation wizard directly on the Data Archive computer itself, you need the following access permissions:

- On the Data Archive server, you need write access on the PIBACKUP and PIModules table in Database Security (in PI SMT, choose **Security > Database Security**).

If you don't have the correct access permissions, a pop-up dialog box appears showing a -10401 (no write access) error.

Note: On PI Server versions 3.4.370 or earlier, PIBACKUP is not required, but piadmin privileges are required. If necessary, the preparation wizard prompts you for the elevated credentials.

- When running the wizard remotely, you need the access permissions listed above, and you also need to be an administrator on both the remote Data Archive server and the local client. If necessary, the preparation wizard prompts you for the elevated credentials.

Module Database security permissions

Permission type	Requirements
Data Archive	Required access permissions vary by task (see the following table)
File System	None required.
Registry Access	None required.
Service Control Manager	None required.

The following table lists Data Archive access permissions required for module-related tasks.

Task	Database security permissions	Other permissions
Modules: Create	PIModules (r,w)	parent module (r*,w)
Modules: Delete	PIModules (r,w)	parent module (r,w)

Task	Database security permissions	Other permissions
		module (rw)
Modules: Edit	PIModules (r)	module (r,w)
Modules: Edit – Link / Unlink	PIModules (r)	new parent module (r*,w) module (r*,w)
Modules: Edit – Add / Remove alias	PIModules (r) PIPOINT(r)	module (r*,w) PtSecurity (r)
Modules: Edit – Add / Remove heading	PIModules (r) PIHeadingSets (r) If you have PIModules (r), then PIHeadingSets (r) is automatically set.	module (r*,w) heading (r) heading set (r)
Modules: View	PIModules (r)	module (r*)

* module (r) also assumes (r) for all modules along the hierarchy path above it.

Network Manager Statistics security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
View connection details	None required
View Statistics	None required

Performance counters security permissions

Permission Type	Requirements
File System	None
Registry Access	None

Permission Type	Requirements
Service Control Manager	None

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
View Performance Monitor Interfaces	PIMODULE (r), module (r) for \\%OSI\Interfaces and its sub-modes Note: module (r) also assumes (r) for all modules along the hierarchy path above it.
View Existing Points	PIPOINT (r), PtSecurity (r) for the individual points
Create Points	PIPOINT (r,w)
Edit Points	PIPOINT (r), PtSecurity (r,w) for the individual points

Performance Equations security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
View Existing Points	PIPOINT (r), PtSecurity (r) for the individual points
Create Points	PIPOINT (r,w)
Edit Points	PIPOINT (r), PtSecurity (r,w) for the individual points

PI Point Source Table security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
List Point Sources	PIPOINT (r)
Edit Point Source Description	PIPOINT (r,w)

PI Services security permissions

Permission Type	Requirements
Data Archive	None required.
File System	Requires local and remote read permissions on the contents of the \PI\adm directory through the admin share (\System\c\$).
Registry Access	Requires read access to the HKLM\Software\PISystem\PI registry key on any Data Archive to be monitored.
Service Control Manager	Requires read and service control privileges on local and remote computers.

PI Version security permissions

Permission Type	Requirements
PI	List subsystem versions: None required
File System	Requires local and remote read permissions on the contents of the \PI\adm directory through the admin share (\System\c\$).
Registry Access	Requires read access to the HKLM\Software\PISystem\PI registry key on any Data Archive to be monitored.
Service Control Manager	None required

Ping security permissions

Permission Type	Requirements
File System	Not used
Registry Access	Not used
Service Control Manager	Not used

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
View Existing Points	PIPOINT (r), PtSecurity (r) for the individual points
Create Points	PIPOINT (r,w)
Edit Points	PIPOINT (r), PtSecurity (r,w) for the individual points

Point Builder security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
Create points	PIPOINT (r,w)
Delete points	PIPOINT (r,w), PtSecurity (r,w)
Edit points	PIPOINT (r), PtSecurity (r,w)

Point Classes security permissions

Permission Type	Requirements
PI	View point classes: PIPOINT (r) View attribute sets: PIPOINT (r)
File System	None required.
Registry Access	None required.
Service Control Manager	None required.

Reason Tree security permissions

Permission Type	Requirements
PI	Creating new reason codes requires access to create modules under %OSI: PIModules (r,w) %OSI module (r,w)
File System	None required
Registry Access	Not supported
Service Control Manager	None required

Security security permissions

Permission Type	Requirements
File System	None required
PIUSER (r,w) Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks. There are two pieces of information displayed in this tool: Tuning parameter, which controls API/SDK enabled, explicit login disabled, and blank password disabled. Tuning requires PITuning database security access. The explicit login disabled for piadmin is read directly from the PIUSER database.

Permission Type	Requirements
View Settings	PIUSER (r), PITUNING(r)
Edit Settings	PIUSER (r,w), PITUNING(r,w)

Snapshot and Archive Statistics security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
Archive statistics	PIArcData(r)
Snapshot statistics	None required

SNMP Points security permissions

Permission Type	Requirements
File System	This tool requires read/write access to the template files in PIPCHome Folder\SMT3\PISNMPConfig and its subfolders.
Registry Access	None required
Service Control Manager	None required

Note: module (r) also assumes (r) for all modules along the hierarchy path above it.

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
View SNMP Interfaces	PIMODULE (r), module (r) for \%OSI\Interfaces and its sub-modes
View Existing Points	PIPOINT (r), PtSecurity (r) for the individual points
Create Points	PIPOINT (r,w)

Permission Type	Requirements
Edit Points	PIPOINT (r), PtSecurity (r,w) for the individual points

SQC Alarms security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks. This plug-in writes to the Module Database to store preferences. You need PIModules (r,w) to write to the Module Database.

Permission Type	Requirements
View Existing Points	PIPOINT (r)
Create or Edit Points	PIPOINT (r,w)

Stale and Bad Points security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
For the points being displayed	PIPOINT (r), DataSecurity (r) and PtSecurity (r) for each point

TCP Response security permissions

The PI TCPResponse Plug-In for PI SMT 3.x has the following security requirements

Permission Type	Requirements
File System	Not used
Registry Access	Not used
Service Control Manager	Not used

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
View Existing Points	Requires access through a PI user that has permissions to read existing points (through login or trust).
Create or Edit Points	Requires access through a PI user that has permissions to edit or create points (through login or trust).

Totalizers security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI point-related tasks.

Permission Type	Requirements
Create points	PIPOINT (r,w)
Delete points	PIPOINT (r,w), PtSecurity (r,w)
Edit points	PIPOINT (r), PtSecurity (r,w)

Trusts security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for PI trust-related tasks.

Permission Type	Requirements
Create/edit/delete trusts	PITRUST (r,w)
View trusts	PITRUST (r)

Tuning Parameters security permissions

Permission Type	Requirements
File System	None required
Registry Access	None required
Service Control Manager	None required

The following table lists access permissions required for tasks related to tuning parameters.

Permission Type	Requirements
Create / Edit / Delete Tuning Parameters	PITUNING (r,w)
View Tuning Parameters	PITUNING (r)

Update Manager security permissions

Permission Type	Requirements
PI	Producer information is available with PIWorld (r) access. If PIWorld is disabled, then piadmin or piadmins is required. Optionally resolve point names: PIPOINT (r). Also requires PtSecurity (r) for each point.
File System	None required
Registry Access	None required
Service Control Manager	None required

AVEVA™ PI SMT 2023 Release Notes

Overview

PI System Management Tools (PI SMT) released in 2023, provides PI System administrators with a suite of applications that provide management functionality for the PI Data Archive. These tools include:

1. **PI System Management Tools:** A host application and a suite of plug-ins that provide management functionality.
2. **PI Collective Manager:** Create and monitor the health of PI Collectives.
3. **PI Module Database Builder:** Excel add-in which allows you to create, modify, and delete items in your PI Module Database.
4. **PI Builder:** Excel add-in which allows you to create, modify, and delete PI Points on your PI Data Archive as well as objects on your PI AF Server
5. **PowerShell Tools for the PI System:** Microsoft PowerShell module that allows you to use scripting to manage your PI Data Archive and now AF Server.
6. **PI Interface Configuration Utility:** A tool for configuring and managing interfaces.

PI SMT 2023 introduces support for OpenID Connect (OIDC) mapping when it is used with PI Server 2023 or later. Refer to the [Customer Portal Products](#) page for additional information.

Fixes and enhancements

The table lists all resolved issues for the PI SMT 2023 release.

WI / PLI	Description
77377	File browser does not populate the backup location if run remotely
75306	When loading the Archives plugin in SMT, it says that there are corrupt archives, even after reprocessing them. Automatic archive corruption detection is incorrect (showing a non-corrupt archive file).

Enhancements

- **OpenID Connect Mapping Support**

PI System Management Tools introduces support for OpenID Connect Mappings in the 2023 release when it is used with PI Server 2023 and later.

- **OpenID Connect Server Status Verification**

The PI System Management Tools 2023 release includes support for verifying the OpenID Connect (AVEVA Identity Manager) Server status when it is used with PI Server 2023 and later.

Deprecations

PI System Tray has been deprecated and will no longer be distributed.

Known issues

Refer to the [Customer Portal Products](#) page for a list of all known issues for PI SMT.

Security information and guidance

We are [committed to releasing secure products](#). This section is intended to provide relevant security-related information to guide your installation or upgrade decision.

We [proactively disclose](#) aggregate information about the number and severity of security vulnerabilities addressed in each release. The tables below provide an overview of security issues addressed and their relative severity based on standard scoring.

Vulnerabilities Fixed in this Release

The following table lists the number of fixed vulnerabilities for the PI SMT 2023 release by severity category and CVSS base score range.

Severity Category	CVSS Base Score Range	Number of Fixed Vulnerabilities in PI SMT 2023
Critical	9.0 - 10	0
High	7.0 - 8.9	0
Medium	4.0 - 6.9	0
Low	0 - 3.9	0

Vulnerabilities Mitigated in this Release

There were no new vulnerabilities for PI SMT 2023 mitigated in this release.

Distribute Kits

Product	Software Version
Microsoft Runtimes VB6	1.0.1
Microsoft Visual C++ 2012 Update Redistributable (x64 & x86)	11.0.61030
Microsoft Visual C++ 2015-2022 Redistributable	14.34.31931.0

(x64 & x86)	
Microsoft Visual Studio 2010 Tools for Office Runtime	10.0.60724
PI Network Subsystem Support (PINS)	3.5.500.88
PI SDK 2018 SP1 Patch 3 (x86 and x64)	1.4.7.620
PI AF Client 2023	3.0.0.790
PI System Management Tools 2023	3.7.0.272
PI Powershell Tools for the PI System	3.0.0.126
PI Collective Manager	1.4.3.25
PI Module Database Builder	1.3.0.0
PI Interface Configuration Utility	1.5.1.10
PI SPT	1.2.4.0
PIGenericNamesDLL	1.3.5.25

©2024 AVEVA Group plc and its subsidiaries. All rights reserved.



AVEVA Group plc
High Cross
Madingley Road
Cambridge
CB3 0HB
UK

Tel +44 (0)1223 556655

www.aveva.com

To find your local AVEVA office, visit **www.aveva.com/offices**

AVEVA believes the information in this publication is correct as of its publication date. As part of continued product development, such information is subject to change without prior notice and is related to the current software release. AVEVA is not responsible for any inadvertent errors. All product names mentioned are the trademarks of their respective holders.