SIEMENS

XHQ

SDK Reference Guide

About This Guide

Client Data API 1

Web Services 2

XHQ Web Interface 3

XHQ BI Data Provider Web Service 4

Data Retrieval Modes for XHQ 5

For Performance Management A

XHQ Web Services WSDL Interface B

Database Views C

06/2019 V6.0.0.1

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

⚠ DANGER
Indicates that death or severe personal injury **will** result if proper precautions are not taken.

⚠ **WARNING**
Indicates that death or severe personal injury **may** result if proper precautions are not taken.

⚠ **CAUTION**
Indicates that minor personal injury can result if proper precautions are not taken.

NOTICE
Indicates that property damage can result if proper precautions are not taken.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage. See the topic, Visual Cues for Online Viewing, for additional XHQ-specific notices.

Qualified Personnel

The product/system described in this documentation may be operated only by personnel qualified for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

⚠ **WARNING**
Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner. For a complete list, see the Copyright topic.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Copyright © 1998-2019 Siemens AG. All rights reserved. Protected by U.S. Patents Nos. 6,700,590, 7,069,514, 7,478,128, 7,689,579, 7,698,292, 7,814,123, 7,840,607, 8,001,332, 8,078,598, 8,260,783, 8,442,938, 8,566,781, 8,700,671 and 8,700,559; Patents Pending.

Siemens Product Lifecycle Management Software, Inc.
6 Journey, Suite 200
Aliso Viejo, CA 92656-5318, USA
siemens.com/xhq

XHQ® is a registered trademark of Siemens AG in the United States. This License does not grant LICENSEE any rights to trademarks or service marks of Siemens AG.

All other company, product and service names and logos may be trademarks or service marks of their respective companies. Any rights not expressly granted herein are reserved. LICENSEE may not remove or alter any trademark, logo, copyright or other proprietary notices, legends, symbols or labels from the Licensed Software or the Documentation.

This software is proprietary and confidential. Siemens AG or its suppliers own the title, copyright, and other intellectual property rights in the Software. The Software is licensed, not sold.

Adobe, the Adobe logo, Acrobat, the Adobe PDF logo, PostScript, and the PostScript logo, Distiller, and Reader are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Microsoft, Active Directory, ActiveX, Authenticode, Developer Studio, DirectX, Microsoft, MS-DOS, Outlook, Excel, PowerPoint, Visual Basic, Visual C++, Visual C#, Visual J#, Visual SourceSafe, Visual Studio, Win32, Windows, Windows Server, WinFX, Windows 7, Windows 10, Windows Server 2008, Windows Server 2012, Windows Server 2016, and the Windows logo are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries, or both.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Oracle, Java, and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates. Oracle, or its licensor, shall at all times retain all rights, title, interest, including intellectual property rights, in Oracle Programs and media.

SAP, SAP R/3, R/3, R/3 software, mySAP, mySAP.com, xApps, xApp, ABAP, BAPI, and SAP NetWeaver are trademarks or registered trademarks of SAP AG in Germany and in several other countries.

Documentum, OpenText Documentum, OpenText and the Corporate Logo are trademarks or registered trademarks of OpenText in the United States and throughout the world.

IBM, the IBM logo, DB2, and ibm.com are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide.

InstallShield® is a registered trademark and service mark of Macrovision Corporation and/or Macrovision Europe Ltd. in the United States and/or other countries. DemoShield, InstallFromTheWeb and PackageForTheWeb are service marks and registered trademarks of Macrovision Corporation and/or Macrovision Europe Ltd. in the United States and/or other countries. InstallShield Express, InstallShield for Windows Installer, InstallShield for Windows CE, Express Wizard, InstallShield Objects, WebUpdate, FastReg and NetInstall are trademarks and/or service marks of Macrovision Corporation and/or Macrovision Europe Ltd. InstallShield Software Corporation. InstallShield is a member of Macrovision Corporation.

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, systems, machines and networks.

In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial security concept. Siemens' products and solutions only form one element of such a concept.

Customer is responsible to prevent unauthorized access to its plants, systems, machines and networks. Systems, machines and components should only be connected to the enterprise network or the internet if and to the extent necessary and with appropriate security measures (e.g. use of firewalls and network segmentation) in place.

Additionally, Siemens' guidance on appropriate security measures should be taken into account. For more information about industrial security, please visit https://www.siemens.com/industrialsecurity.

Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends to apply product updates as soon as available and to always use the latest product versions. Use of product versions that are no longer supported, and failure to apply latest updates may increase customer's exposure to cyber threats.

For the Siemens Security Advisory, visit https://www.siemens.com/industrialsecurity.

To stay informed about product updates, subscribe to the Siemens Industrial Security RSS Feed under https://www.siemens.com/industrialsecurity.

While every effort is made to ensure the accuracy of content, the XHQ product documentation set (which includes online help) could contain inaccuracies or out-dated material (which includes product screenshots and images) due to the large number of product enhancements being added. As such, the documentation set is subject to change at any time without notice. Refer to the README for documentation corrections and addendum. Please note, updates to the documentation set are reflected in the next general availability major release of XHQ.

Table of Contents

About This Guide **19**
Conventions Used in This Guide 19
Visual Cues for Online Viewing 20
Related XHQ Product Documentation 21
Contacting Customer Support 23
General Feedback and Comments 24
1 | Client Data API **25**
Coverage Overview 26
Solution API Hierarchies 26
Model API Hierarchies 28
About the API Support Files 30
The DLL Files for .NET API 31
The JAR Files for Java API 32
The Packages 32
Using the Client Data .NET API 33
Using the Client Data Java API 34
Using the ANS API 35
Connecting to and Disconnecting from the ANS Server 36
Creating, Updating, Reading, and Deleting Condition Definitions 36
Creating, Updating, Reading, and Deleting Condition Definition Lists 38
Creating, Updating, Reading, and Deleting Definition List Sections 39
Creating, Updating, Reading, and Deleting Subscriptions 40
Creating, Updating, Reading, and Deleting Subscription Lists 42
Creating, Updating, Reading, and Deleting ANS Codes like Categories and Interval Types 44
Creating, Updating, Reading, and Deleting Notification Routing 45
Solution Server Client API Examples 47
Java Examples 47
.NET Example 57
Getting Help 61
To access help files 61
2 | Web Services **62**
Benefits of Web Services 63
Web Services Set-up 64

Setting the Solution Property ... 65
To set the solution property ... 65
Setting the default XHQ host server for connecting the remote/distributed XhqWse web service ... 65
Modifying the WAKA XML configuration file for the remote/distributed XhqWse web service ... 65
Setting the Default Time Zone ... 67
Additional Configurable Properties ... 67
Web Services Architecture ... 69
Using the External Web Service, XhqWse ... 70
Web Service Operations ... 74
XhqWse Web Service Class ... 75
XhqWse Web Service Methods ... 76
Importing the XHQ Model ... 116
Web Service Methods for Importing ... 117
WSDL Definition, Input and Output Parameters ... 118
Using the Administrative Web Service, XhqAdminWs ... 122
XhqAdminWs Web Service Methods ... 125
Using the ANS Web Service, AnsWs ... 153
AnsWs Web Service Methods ... 157
REST-compliant XHQ Web Services ... 168
REST Operations ... 168
Ping ... 169
IsSolutionAccessible ... 169
GetPointCurrentData ... 170
GetPointAtValue ... 170
GetPointActualValue ... 171
GetPointAverageValue ... 171
GetPointCountValue ... 171
GetPointDeltaValue ... 172
GetPointFirstValue ... 172
GetPointLastValue ... 172
GetPointIntegralValue ... 172
GetPointMaximumValue ... 173
GetPointMinimumValue ... 173
GetPointSumValue ... 173
GetTimeSeriesData ... 174
GetCollectionData ... 174
GetCollectionDataWhere ... 174
GetCollectionDataOrderBy ... 175

GetCollectionDataFilter 175
GetCollectionDataXml 175
GetCollectionDataJson 175
GetCollectionDataWhereXml 176
GetCollectionDataWhereJson 176
GetCollectionDataOrderByXml 176
GetCollectionDataOrderByJson 177
GetCollectionDataFilterXml 177
GetCollectionDataFilterJson 177
GetCollectionColumnAverage 177
GetCollectionColumnCount 178
GetCollectionColumnMaximum 178
GetCollectionColumnMinimum 178
GetCollectionColumnSum 178
GetCollectionCount 179
GetGlobalCollections 179
GetSolutionRoot 179
GetChildPrimitiveMembers 179
GetChildComponentMembers 180
GetChildCollectionMembers 180
GetAllChildMembers 180
WritePointValue 181
CanReadMember 181
CanWriteMember 181
GetReadableMembers 181
GetWriteableMembers 182
GetUserRoleList 182
IsUserInRole 182
GetRolesWithReadAccess 182
GetRolesWithWriteAccess 182
GetRolesProvidingReadAccess 184
GetRolesProvidingWriteAccess 184
GenerateSolutionTreeNode 184
GetPathListAccess 184
InsertCollectionData 184
UpdateCollectionData 185
ReplaceCollectionData 185
DeleteCollectionData 185

GetBulkCurrentValues 185
GetBulkAtValues 185
GetBulkActualValues 185
GetBulkAverageValues 185
GetBulkCountValues 185
GetBulkDeltaValues 185
GetBulkFirstValues 186
GetBulkLastValues 186
GetBulkIntegralValues 186
GetBulkMaximumValues 186
GetBulkMinimumValues 186
GetBulkSumValues 186
Testing the Web Service 188
Retreiving Large Collection Data 189
To increase the maximum message size quota 189
Configuring SSL (HTTPS) Access 190
SSL Access for the XhqWse Web Service 190
To enable the XhqWse web service to work with SSL (HTTPS) 190
SSL Access for the XHQ Applet and Web Applications 193
To configure the SSL certificate for the indx virtual directory 193
HTTPS (SSL) and HTTP Links for WAKA 194
3 | XHQ Web Interface 195
Accessing Online Help 195
XHQ Admin Module Methods 198
Audit 198
LimitClasses 198
Paths 198
Perm 199
ReasonCodes 199
Severities 199
SubTypes 200
Types 200
XHQ ANS Module Methods 202
AnchorViews 202
AnsId 202
ASD 202
Backfill 202
CodeKey 203

CondDef 203
DefList 204
ExcursionLog 204
Export 205
Import 205
Interval 206
ListSection 206
LO 207
Locale 207
Notif 207
NotifRouting 208
Perm 208
Role 208
Schedule 209
SubscriptionItem 209
SubscriptionList 210
Template 210
User 211
XHQ Core Module Methods 212
Access Calls 212
CanReadMember 212
CanWriteMember 212
PathList 212
ReadableMembers 213
ReadPathRoles 213
UserRoles 213
WriteableMembers 213
WritePathRoles 213
Collections Calls 213
ColumnAverage 214
ColumnCount 214
ColumnMaximum 214
ColumnMinimum 214
ColumnSum 214
Records 214
RecordsCount 215
Metadata Calls 215
AllMembers 215

CollectionMembers ... 215
ComponentMembers ... 215
GlobalCollections ... 215
PrimitiveMembers ... 215
SolutionRoot ... 216
SolutionTreeNode ... 216
Points Calls ... 216
ActualValue ... 216
AtValue ... 216
AverageValue ... 216
CountValue ... 217
CurrentValue ... 217
DeltaValue ... 217
FirstValue ... 217
IntegralValue ... 218
LastValue ... 218
MaximumValue ... 218
MinimumValue ... 218
SumValue ... 218
TimeSeries ... 219
Utility Calls ... 219
CheckSolutionAccess ... 219
Collections ... 219
GlobalProps ... 220
License ... 220
Logger ... 220
Login ... 220
PathPicker ... 220
Ping ... 221
XHQ eLogs Module Methods ... 222
AssocTypes ... 222
Conditions ... 222
CondLists ... 222
CondTypes ... 223
Defaults ... 223
ExportConfigs ... 224
GenericUsers ... 224
ImportConfigs ... 224

KeyWords 225
LogClasses 225
LogTypes 225
MultiDefaults 225
OperLogs 225
ParamGroups 226
ParamTypes 227
Priorities 227
Reports 227
RoutineParameters 227
Routines 228
ShiftReports 228
Units 228
XHQ RDS Module Methods 230
Logs 230
SubsCol 230
XHQ Target Management Module Methods 231
ExportKpis 231
ImportKpis 231
Kpi 231
Limits 232
Subscribers 232
Targets 232
How to retrieve information from XHQ using the REST web services 233
Get tag data 235
Get a list of assets 243
Get attributes for a particular pump 246
Using the XHQ Performance Management Mobile App 254
Installing the App 254
Launching the XHQ Mobile App 255
The XHQ Mobile Home Screen 258
The Settings Menu 259
Adding a Log 260
The Log Explorer 264
The ANS Alert Status 269
The ANS Excursion Report 273
The RestClient Class 278
Whitelisting XHQ Servers 282

To specify a list of whitelisted XHQ Servers ... 282

4 | XHQ BI Data Provider Web Service ... 283

Physical Path ... 283
Virtual Directory Path ... 283
Application Pool ... 283
Security ... 283
Application Configuration ... 283
XHQ BI Data Provider Home Page URL ... 284
URL Components ... 284
The XHQ BI Data Provider Web Service Root (Service Document) URL ... 285
Metadata Document Request (Service Metadata Document) URL ... 285
Requesting Data Using Resource Path ... 285
Addressing Entities ... 286
Basic OData Support ... 286
Querying Collections ... 286
Querying Solution Values and Solution Time-Series (Trend) Data ... 289
Business Intelligence (BI) Tools Integration ... 290

5 | Data Retrieval Modes for XHQ ... 292

About Time Slices ... 294
Raw Values and Computation of Aggregates ... 294
Views and Server Side Expressions ... 294
Trends ... 295
Web Services ... 295

Appendices ... 296

A - For Performance Management ... 297
Performance Management Client API ... 297
getConnection ... 297
disconnect ... 297
createPerformanceIndicator ... 297
updatePerformanceIndicator ... 298
deletePerformanceIndicator ... 298
getPerformanceIndicator ... 298
getPerformanceIndicatorList ... 298
getPiCount ... 299
getPiNames ... 299
getPiMetrics ... 299
getAppCodesList ... 300

getCode ... 300
setAppCodeAccess ... 300
setCode ... 301
getTarget ... 301
updateTarget ... 301
getTargetAuditTrail ... 301
getLimitsAuditTrail ... 302
getComponentMembers ... 302
getTagAliases ... 302
importPerformanceIndicators ... 303
exportPerformanceIndicators ... 303
authenticateUser ... 303
getRoleList ... 303
getPermissionList ... 304
getRolePermissions ... 304
getRoleCategories ... 304
getRolePaths ... 305
updateRolePermissions ... 305
updateRoleCategories ... 305
updateRolePaths ... 306
getUserList ... 306
getOwnerPiList ... 307
changePiOwner ... 307
Web Services Methods for Performance Management ... 308
getPerformanceIndicatorList ... 308
getPerformanceIndicator ... 308
getPerformanceIndicatorCount ... 308
bulkLoadPerformanceIndicators ... 308
bulkUpdatePerformanceIndicators ... 308
To access the Performance Management Web Services ... 308
B - XHQ Web Services WSDL Interface ... 310
WSDL for XhqWse Web Service ... 311
service XhqWse ... 317
binding BasicHttpBinding_XHQ_x0020_Web_x0020_Service ... 318
porttype XHQ_x0020_Web_x0020_Service ... 340
message XHQ_x0020_Web_x0020_Service_DeleteCollectionData_InputMessage ... 356
message XHQ_x0020_Web_x0020_Service_DeleteCollectionData_OutputMessage ... 356
message XHQ_x0020_Web_x0020_Service_GetAllChildMembers_InputMessage ... 356

message XHQ_x0020_Web_x0020_Service_GetAllChildMembers_OutputMessage 357
message XHQ_x0020_Web_x0020_Service_GetBulkActualValues_InputMessage 357
message XHQ_x0020_Web_x0020_Service_GetBulkActualValues_OutputMessage 357
message XHQ_x0020_Web_x0020_Service_GetBulkAtValues_InputMessage 358
message XHQ_x0020_Web_x0020_Service_GetBulkAtValues_OutputMessage 358
message XHQ_x0020_Web_x0020_Service_GetBulkAverageValues_InputMessage 358
message XHQ_x0020_Web_x0020_Service_GetBulkAverageValues_OutputMessage 359
message XHQ_x0020_Web_x0020_Service_GetBulkCountValues_InputMessage 359
message XHQ_x0020_Web_x0020_Service_GetBulkCountValues_OutputMessage 359
message XHQ_x0020_Web_x0020_Service_GetBulkCurrentValues_InputMessage 360
message XHQ_x0020_Web_x0020_Service_GetBulkCurrentValues_OutputMessage 360
message XHQ_x0020_Web_x0020_Service_GetBulkDeltaValues_InputMessage 361
message XHQ_x0020_Web_x0020_Service_GetBulkDeltaValues_OutputMessage 361
message XHQ_x0020_Web_x0020_Service_GetBulkFirstValues_InputMessage 361
message XHQ_x0020_Web_x0020_Service_GetBulkFirstValues_OutputMessage 362
message XHQ_x0020_Web_x0020_Service_GetBulkIntegralValues_InputMessage 362
message XHQ_x0020_Web_x0020_Service_GetBulkIntegralValues_OutputMessage 362
message XHQ_x0020_Web_x0020_Service_GetBulkLastValues_InputMessage 363
message XHQ_x0020_Web_x0020_Service_GetBulkLastValues_OutputMessage 363
message XHQ_x0020_Web_x0020_Service_GetBulkMaximumValues_InputMessage 363
message XHQ_x0020_Web_x0020_Service_GetBulkMaximumValues_OutputMessage 364
message XHQ_x0020_Web_x0020_Service_GetBulkMinimumValues_InputMessage 364
message XHQ_x0020_Web_x0020_Service_GetBulkMinimumValues_OutputMessage 364
message XHQ_x0020_Web_x0020_Service_GetBulkSumValues_InputMessage 365
message XHQ_x0020_Web_x0020_Service_GetBulkSumValues_OutputMessage 365
message XHQ_x0020_Web_x0020_Service_GetChildCollectionMembers_InputMessage 366
message XHQ_x0020_Web_x0020_Service_GetChildCollectionMembers_OutputMessage 366
message XHQ_x0020_Web_x0020_Service_GetChildComponentMembers_InputMessage 366
message XHQ_x0020_Web_x0020_Service_GetChildComponentMembers_OutputMessage 367
message XHQ_x0020_Web_x0020_Service_GetChildPrimitiveMembers_InputMessage 367
message XHQ_x0020_Web_x0020_Service_GetChildPrimitiveMembers_OutputMessage 367
message XHQ_x0020_Web_x0020_Service_GetCollectionColumnAverage_InputMessage 368
message XHQ_x0020_Web_x0020_Service_GetCollectionColumnAverage_OutputMessage 368
message XHQ_x0020_Web_x0020_Service_GetCollectionColumnCount_InputMessage 368
message XHQ_x0020_Web_x0020_Service_GetCollectionColumnCount_OutputMessage 369
message XHQ_x0020_Web_x0020_Service_GetCollectionColumnMaximum_InputMessage 369
message XHQ_x0020_Web_x0020_Service_GetCollectionColumnMaximum_OutputMessage 369
message XHQ_x0020_Web_x0020_Service_GetCollectionColumnMinimum_InputMessage 370

message XHQ_x0020_Web_x0020_Service_GetCollectionColumnMinimum_OutputMessage 370
message XHQ_x0020_Web_x0020_Service_GetCollectionColumnSum_InputMessage 370
message XHQ_x0020_Web_x0020_Service_GetCollectionColumnSum_OutputMessage 371
message XHQ_x0020_Web_x0020_Service_GetCollectionCount_InputMessage 371
message XHQ_x0020_Web_x0020_Service_GetCollectionCount_OutputMessage 371
message XHQ_x0020_Web_x0020_Service_GetCollectionData_InputMessage 372
message XHQ_x0020_Web_x0020_Service_GetCollectionData_OutputMessage 372
message XHQ_x0020_Web_x0020_Service_GetCollectionDataFilter_InputMessage 373
message XHQ_x0020_Web_x0020_Service_GetCollectionDataFilter_OutputMessage 373
message XHQ_x0020_Web_x0020_Service_GetCollectionDataOrderBy_InputMessage 373
message XHQ_x0020_Web_x0020_Service_GetCollectionDataOrderBy_OutputMessage 374
message XHQ_x0020_Web_x0020_Service_GetCollectionDataWhere_InputMessage 374
message XHQ_x0020_Web_x0020_Service_GetCollectionDataWhere_OutputMessage 374
message XHQ_x0020_Web_x0020_Service_GetGlobalCollections_InputMessage 375
message XHQ_x0020_Web_x0020_Service_GetGlobalCollections_OutputMessage 375
message XHQ_x0020_Web_x0020_Service_GetPointActualValue_InputMessage 375
message XHQ_x0020_Web_x0020_Service_GetPointActualValue_OutputMessage 376
message XHQ_x0020_Web_x0020_Service_GetPointAtValue_InputMessage 376
message XHQ_x0020_Web_x0020_Service_GetPointAtValue_OutputMessage 376
message XHQ_x0020_Web_x0020_Service_GetPointAverageValue_InputMessage 377
message XHQ_x0020_Web_x0020_Service_GetPointAverageValue_OutputMessage 377
message XHQ_x0020_Web_x0020_Service_GetPointCountValue_InputMessage 378
message XHQ_x0020_Web_x0020_Service_GetPointCountValue_OutputMessage 378
message XHQ_x0020_Web_x0020_Service_GetPointCurrentData_InputMessage 378
message XHQ_x0020_Web_x0020_Service_GetPointCurrentData_OutputMessage 379
message XHQ_x0020_Web_x0020_Service_GetPointDeltaValue_InputMessage 379
message XHQ_x0020_Web_x0020_Service_GetPointDeltaValue_OutputMessage 379
message XHQ_x0020_Web_x0020_Service_GetPointFirstValue_InputMessage 380
message XHQ_x0020_Web_x0020_Service_GetPointFirstValue_OutputMessage 380
message XHQ_x0020_Web_x0020_Service_GetPointIntegralValue_InputMessage 380
message XHQ_x0020_Web_x0020_Service_GetPointIntegralValue_OutputMessage 381
message XHQ_x0020_Web_x0020_Service_GetPointLastValue_InputMessage 381
message XHQ_x0020_Web_x0020_Service_GetPointLastValue_OutputMessage 381
message XHQ_x0020_Web_x0020_Service_GetPointMaximumValue_InputMessage 382
message XHQ_x0020_Web_x0020_Service_GetPointMaximumValue_OutputMessage 382
message XHQ_x0020_Web_x0020_Service_GetPointMinimumValue_InputMessage 383
message XHQ_x0020_Web_x0020_Service_GetPointMinimumValue_OutputMessage 383
message XHQ_x0020_Web_x0020_Service_GetPointSumValue_InputMessage 383

message XHQ_x0020_Web_x0020_Service_GetPointSumValue_OutputMessage 384
message XHQ_x0020_Web_x0020_Service_GetSolutionRoot_InputMessage 384
message XHQ_x0020_Web_x0020_Service_GetSolutionRoot_OutputMessage 384
message XHQ_x0020_Web_x0020_Service_GetTimeSeriesData_InputMessage 385
message XHQ_x0020_Web_x0020_Service_GetTimeSeriesData_OutputMessage 385
message XHQ_x0020_Web_x0020_Service_InsertCollectionData_InputMessage 385
message XHQ_x0020_Web_x0020_Service_InsertCollectionData_OutputMessage 386
message XHQ_x0020_Web_x0020_Service_IsSolutionAccessible_InputMessage 386
message XHQ_x0020_Web_x0020_Service_IsSolutionAccessible_OutputMessage 386
message XHQ_x0020_Web_x0020_Service_ReplaceCollectionData_InputMessage 387
message XHQ_x0020_Web_x0020_Service_ReplaceCollectionData_OutputMessage 387
message XHQ_x0020_Web_x0020_Service_UpdateCollectionData_InputMessage 388
message XHQ_x0020_Web_x0020_Service_UpdateCollectionData_OutputMessage 388
message XHQ_x0020_Web_x0020_Service_WritePointValue_InputMessage 388
message XHQ_x0020_Web_x0020_Service_WritePointValue_OutputMessage 389
element DeleteCollectionData 389
element DeleteCollectionDataResponse 390
element GetAllChildMembers 390
element GetAllChildMembersResponse 390
element GetBulkActualValues 391
element GetBulkActualValuesResponse 391
element GetBulkAtValues 392
element GetBulkAtValuesResponse 392
element GetBulkAverageValues 393
element GetBulkAverageValuesResponse 393
element GetBulkCountValues 394
element GetBulkCountValuesResponse 394
element GetBulkCurrentValues 395
element GetBulkCurrentValuesResponse 395
element GetBulkDeltaValues 396
element GetBulkDeltaValuesResponse 396
element GetBulkFirstValues 397
element GetBulkFirstValuesResponse 397
element GetBulkIntegralValues 398
element GetBulkIntegralValuesResponse 398
element GetBulkLastValues 399
element GetBulkLastValuesResponse 399
element GetBulkMaximumValues 400

element GetBulkMaximumValuesResponse 400
element GetBulkMinimumValues 401
element GetBulkMinimumValuesResponse 401
element GetBulkSumValues 402
element GetBulkSumValuesResponse 402
element GetChildCollectionMembers 403
element GetChildCollectionMembersResponse 403
element GetChildComponentMembers 403
element GetChildComponentMembersResponse 404
element GetChildPrimitiveMembers 404
element GetChildPrimitiveMembersResponse 405
element GetCollectionColumnAverage 405
element GetCollectionColumnAverageResponse 406
element GetCollectionColumnCount 406
element GetCollectionColumnCountResponse 407
element GetCollectionColumnMaximum 407
element GetCollectionColumnMaximumResponse 408
element GetCollectionColumnMinimum 408
element GetCollectionColumnMinimumResponse 409
element GetCollectionColumnSum 409
element GetCollectionColumnSumResponse 410
element GetCollectionCount 410
element GetCollectionCountResponse 410
element GetCollectionData 411
element GetCollectionDataFilter 412
element GetCollectionDataFilterResponse 412
element GetCollectionDataOrderBy 413
element GetCollectionDataOrderByResponse 413
element GetCollectionDataResponse 414
element GetCollectionDataWhere 414
element GetCollectionDataWhereResponse 415
element GetGlobalCollections 415
element GetGlobalCollectionsResponse 415
element GetPointActualValue 416
element GetPointActualValueResponse 416
element GetPointAtValue 417
element GetPointAtValueResponse 417
element GetPointAverageValue 418

element GetPointAverageValueResponse 418
element GetPointCountValue 419
element GetPointCountValueResponse 419
element GetPointCurrentData 420
element GetPointCurrentDataResponse 420
element GetPointDeltaValue 421
element GetPointDeltaValueResponse 421
element GetPointFirstValue 422
element GetPointFirstValueResponse 422
element GetPointIntegralValue 423
element GetPointIntegralValueResponse 423
element GetPointLastValue 424
element GetPointLastValueResponse 424
element GetPointMaximumValue 425
element GetPointMaximumValueResponse 425
element GetPointMinimumValue 426
element GetPointMinimumValueResponse 426
element GetPointSumValue 427
element GetPointSumValueResponse 427
element GetSolutionRoot 428
element GetSolutionRootResponse 428
element GetTimeSeriesData 429
element GetTimeSeriesDataResponse 429
element InsertCollectionData 430
element InsertCollectionDataResponse 430
element IsSolutionAccessible 431
element IsSolutionAccessibleResponse 431
element ReplaceCollectionData 432
element ReplaceCollectionDataResponse 432
element UpdateCollectionData 433
element UpdateCollectionDataResponse 433
element WritePointValue 434
element WritePointValueResponse 434
C - Database Views 435

About This Guide

Conventions Used in This Guide

The following formatting cues are designed to allow you to quickly locate and understand the information provided in this guide.

Formatting Conventions

Convention	Example
Acronyms are spelled out the first time they appear.	Alert Notification System (ANS)
Bold is used for menu names, command options, and dialog box names in primary task procedures.	From the **XHQ Workbench**, go to the **Add** menu and click **New Component**.
Italic is used for glossary terms.	The first step in building this model is to develop reusable software building blocks, called *components*.
A **`monospaced font`** is used for program and code examples.	The subdirectory `\log` is automatically created below the location you choose. All log files are written to this subdirectory. **`C:\XHQ`**
Key combinations appear in uppercase, bold. If joined with a plus sign (+), press and hold the first key while you press the remaining keys.	**CTRL+B**
The *.x* (in italics) is used to indicate release numbers of a product.	Enable (by checking) the **Use Java *x.x.x_xx* for <applet>** option.
In See Also notices, sub-chapter headings are in italics, chapter headings are in quotes, and guide titles are in bold.	For more information, go to the *About install.properties* topic located in the "Working with PROPERTIES Files" chapter of the **XHQ Administrator's Guide**.

Visual Cues for Online Viewing

This document uses the following styled paragraphs.

Notes are used to offer information that supplement important points of the main text. Tips suggest certain techniques and procedures that may help you achieve your task quickly.

Depending on your network configuration, include domain information only if the domains are different.

See Also notices provide you with additional references to similar topics and/or concepts within the documentation set. Sub-chapter headings are in italics, chapter headings are in quotes, and guide titles are in bold.

For more information, go to the About the Options Menu topic located in the "Working with PROPERTIES Files" chapter of the **XHQ Administrator's Guide**.

Web References point you to external web sites that give additional information on the given topic.

Refer to Microsoft support information with regards to the various server settings for application performance and network utilization.

`http://support.microsoft.com`

Tips provide additional hints to help you use the product more efficiently.

Use the `NavbarWestVerticalOffset` property to make fine adjustments in pixels. The upper, left-hand corner is the origin. The positive horizontal direction moves to the right and the positive vertical direction moves down.

Important notices provide information that are required to completing a given task.

XHQ must run as a domain user.

Warnings tell you that failure to take or avoid a certain action could result in loss of data or application malfunction.

WARNING

Do not modify the `shutdown.dat` template file.

Related XHQ Product Documentation

The XHQ documentation set includes the following titles.

XHQ Documentation Set

Title	Target Audience
XHQ Administrator's Guide Provides the steps required to begin administering XHQ. It also covers security and access, property settings, redundancy, and localization.	Administrators
XHQ ANS User's Guide Learn how to use and administer the XHQ Alert Notification System (XHQ ANS).	ANS Users, Administrators
XHQ Backup and Recovery Guide Learn how to properly backup XHQ.	Administrators
XHQ Connection Guide Provides information on injecting an XHQ-supported connector type and configuring the connection.	Connector Developers
XHQ Developer's Guide Introduces the XHQ Development Client (Workbench and Solution Builder) user interface and provides information on how to set-up XHQ, develop reusable components, create views, and build a solution hierarchy.	Content and Solution Developers
XHQ Getting Started Gives you step-by-step instruction on how to set up your model and solution.	Content, Connector, and Solution Developers
XHQ Installation Guide Provides the system requirements, installation instructions, and upgrade information for the current release of the XHQ System.	Administrators
XHQ Integrated Data Gateway Guide Includes information on the ADO.NET and the XHQ OPC UA Server.	Application Engineers, Integrators
XHQ Performance Analytics Guide Learn how to use the Engineering Environment to enable the generation of the processes necessary to extract and transform data for source systems, and populate the XHQ Data Store and Data Mart.	Solution Developers/Users, Analysts
XHQ Performance Management Guide Learn how to use Target Management to monitor performance indicators and eLogs to create shift reports.	Administrators, End Users
XHQ Reference Guide Lists the functions and methods used in XHQ, and provides examples,	Content and Solution Developers

Title	Target Audience
usage notes, and parameter descriptions.	
XHQ Reporting Services Guide Introduces the XHQ Reporting Services and provides instruction on how to connect to an XHQ data source.	Application Engineers, End Users
XHQ SDK Reference Guide Provides a set of development tools that allows you to create applications that extend XHQ. Includes information on the Client API and Web Services.	Application Engineers, Integrators
XHQ Solution Design and Architecture Provides best-practice examples for XHQ solution design. Includes information on tag synchronization.	Solution Architects
XHQ Solution Viewer User's Guide Gives you step-by-step instruction on how to access your solution through a browser client and set browser preferences.	All End Users
XHQ System Guide Contains information regarding secure handling of an XHQ implementation.	Administrators, Application Engineers, Integrators
XHQ Trend Viewer User's Guide Learn how to use the XHQ Trend Viewer to view both real-time and historical data.	All End Users
XHQ Visual Composer Guide Provides end-user information for the XHQ Visual Composer and associated programs, which are used in the development of presentation content.	Content Developers

Contacting Customer Support

For general XHQ product support or related questions, pre-registered customer or partner support staff with a valid XHQ customer support agreement may contact the XHQ Customer Support Team using any of the following means:

- **Web Portal**

  The support portal leverages a system called GTAC (Global Technical Access Center). GTAC provides one common support entry point for many Siemens products. It is available via this URL:

  https://www.siemens.com/gtac

  Customers must be pre-registered to be able to use the web portal. A log-in can be requested at any time by self-registering in the GTAC portal. Note, the end user "sold to" identifier is needed in order to register.

  Use of the support portal is the preferred means to report incidents to the XHQ Customer Support Team unless immediate interactive telephone assistance is required. The support portal is available twenty four hours per day/seven days per week ("24/7").

- **E-mail**

  support.xhq@siemens.com

- **Phone Support and Hours of Coverage**

  International: +1 (949) 448-7463

  U.S. only: +1 (877) 700-4639

  The following paid support levels are available:

Bronze Support: 9/5

9 x 5 hours support. 9 hours per day, 5 days per week. Monday to Friday. Daylight Saving Time is honored.

Choice of one coverage zone out of the following options (default: Americas):

- Americas (15-1 GMT)
- Europe (8-17 GMT)
- Asia (1-10 GMT)

Excludes national holidays as defined by the following countries for the related coverage zone:

- USA (Americas)
- Germany (Europe)
- Singapore (Asia)

Example Americas: *Implies coverage from 7:00 AM to 5:00 PM, Pacific Time, Monday to Friday, excluding US national holidays.*

Silver Support: 24/5

24 x 5 hours support. 24 hours per day, 5 days per week. Monday to Friday. Daylight Saving Time is honored.

Choice of one coverage zone out of the following options (default: Americas):

- Americas
- Europe
- Asia

The weekly start/end times of coverage follow the local times of the following countries in each coverage zone:

- California/USA (Americas)
- Germany (Europe)
- Singapore (Asia)

Example Americas: *Implies coverage from midnight on Sunday until midnight on Friday, Pacific Time, Monday to Friday*.

Gold Support: 24/7

24 x 7 hours support. 24 hours per day, 7 days per week.

- **Postal Mail**

  Siemens Product Lifecycle Management Software, Inc.

  XHQ Operations Intelligence

  Attn: XHQ Customer Support Department

  6 Journey, Suite 200

  Aliso Viejo, CA 92656, USA

General Feedback and Comments

Please send an e-mail to:

info.xhq@siemens.com

Siemens Product Lifecycle Management Software, Inc. and affiliated Siemens Industry Software companies (collectively referred to as "SISW") are committed to working with our customers. Your comments, suggestions, and ideas for improvements are very important to us. Thank you for taking the time to send us your feedback.

1 | Client Data API

The XHQ Client Data API consists of the following general features:

- Supports Java.
- Supports Microsoft .NET Framework.

  Refer to the topic, *Requirements and Pre-requisites*, located in the XHQ Installation Guide, for the supported version of the .NET Framework.

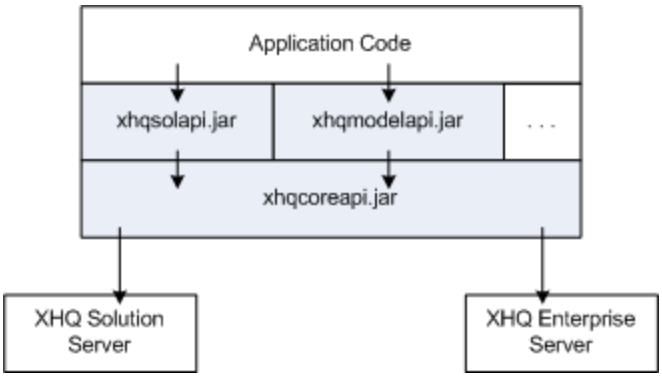
- Provides access to runtime primitive and collection data.
- Provides access to the full namespace.
- Provides access to model information.
- Includes logging capability.
- Provides limited metadata.
- Supports XHQ Security Model.
- Is separately installable.
- Is usable in stand-alone applications or application servers.

  Also refer to the topic, *Performance Management Client API*.

The XHQ Client Data API Is . . .

- **Modular**
  Areas of functionality are grouped together:
  - Solution Runtime Data, Solution Configuration
  - View Configuration, View Runtime Interaction
  - Model, Model Configuration, ANS, etc.
- **Cohesive**
  All elements in a module fit naturally.
- **Intuitive**
  - A "natural" Java interface.
  - Uses familiar java paradigms.
- **Realistic**
  - Accurately represents the target system.
  - Smoothes "rough edges" to allow overall cohesion.
- **Third-Party Oriented**
  - Meaningful error messages.
  - OID, MID are de-emphasized in favor of namespace.

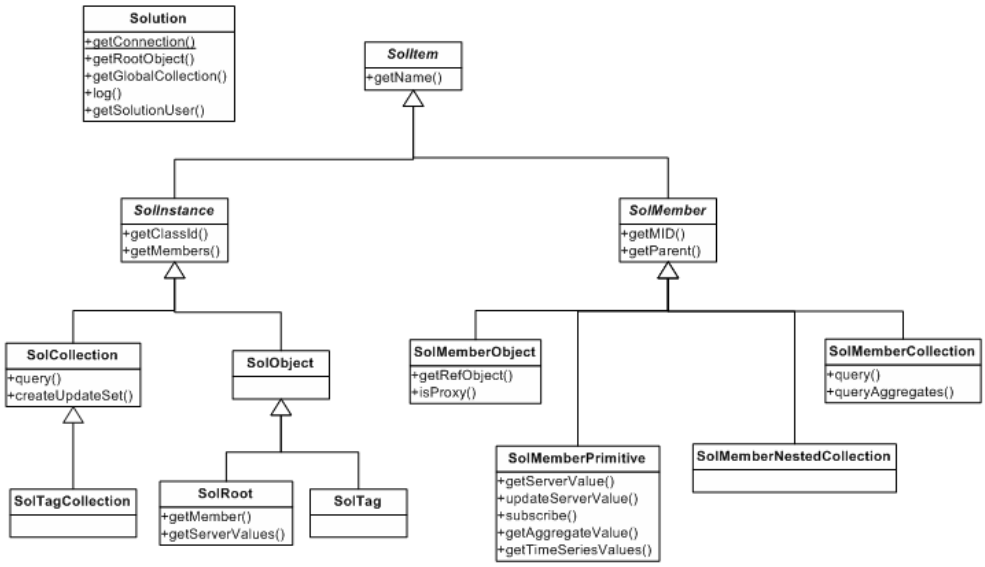
Coverage Overview



API Coverage Overview

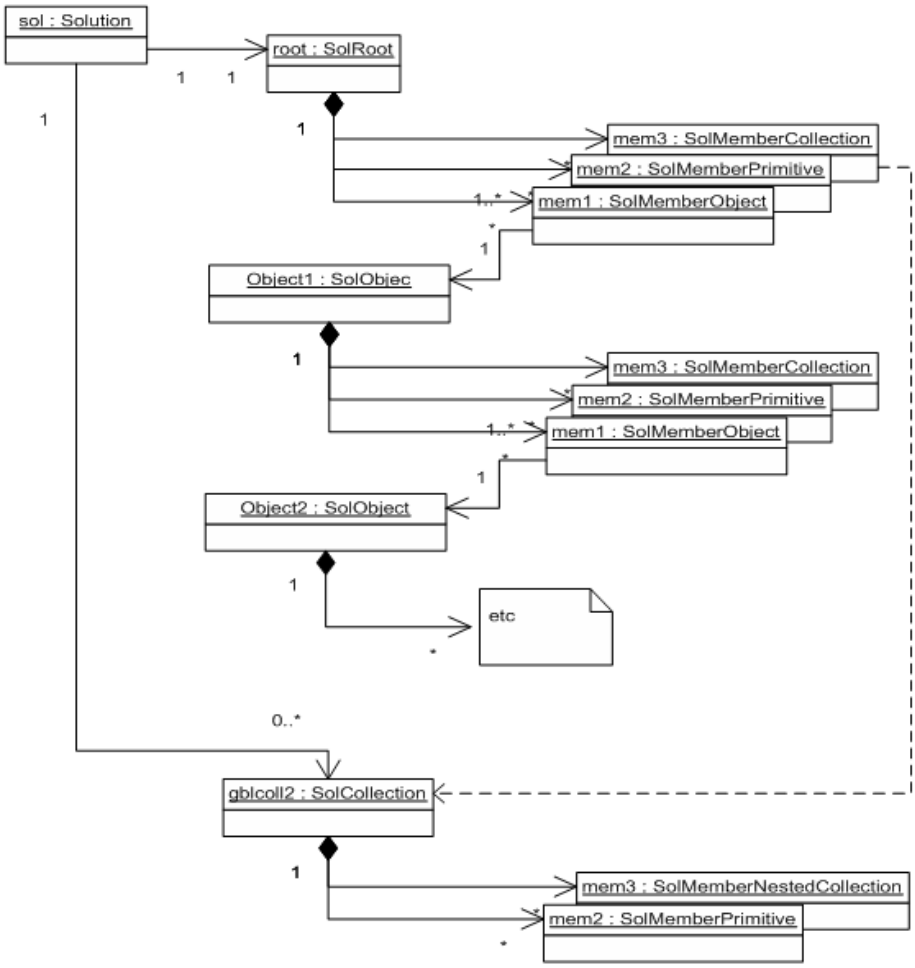
Solution API Hierarchies

Solution API Class Hierarchy



Solution API Class Hierarchy

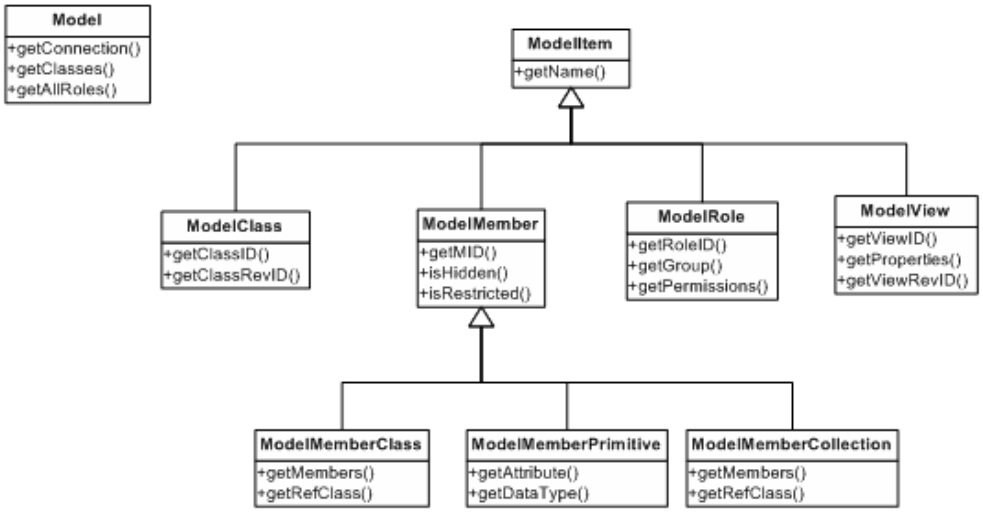
Solution API Instance Hierarchy



Solution API Instance Hierarchy

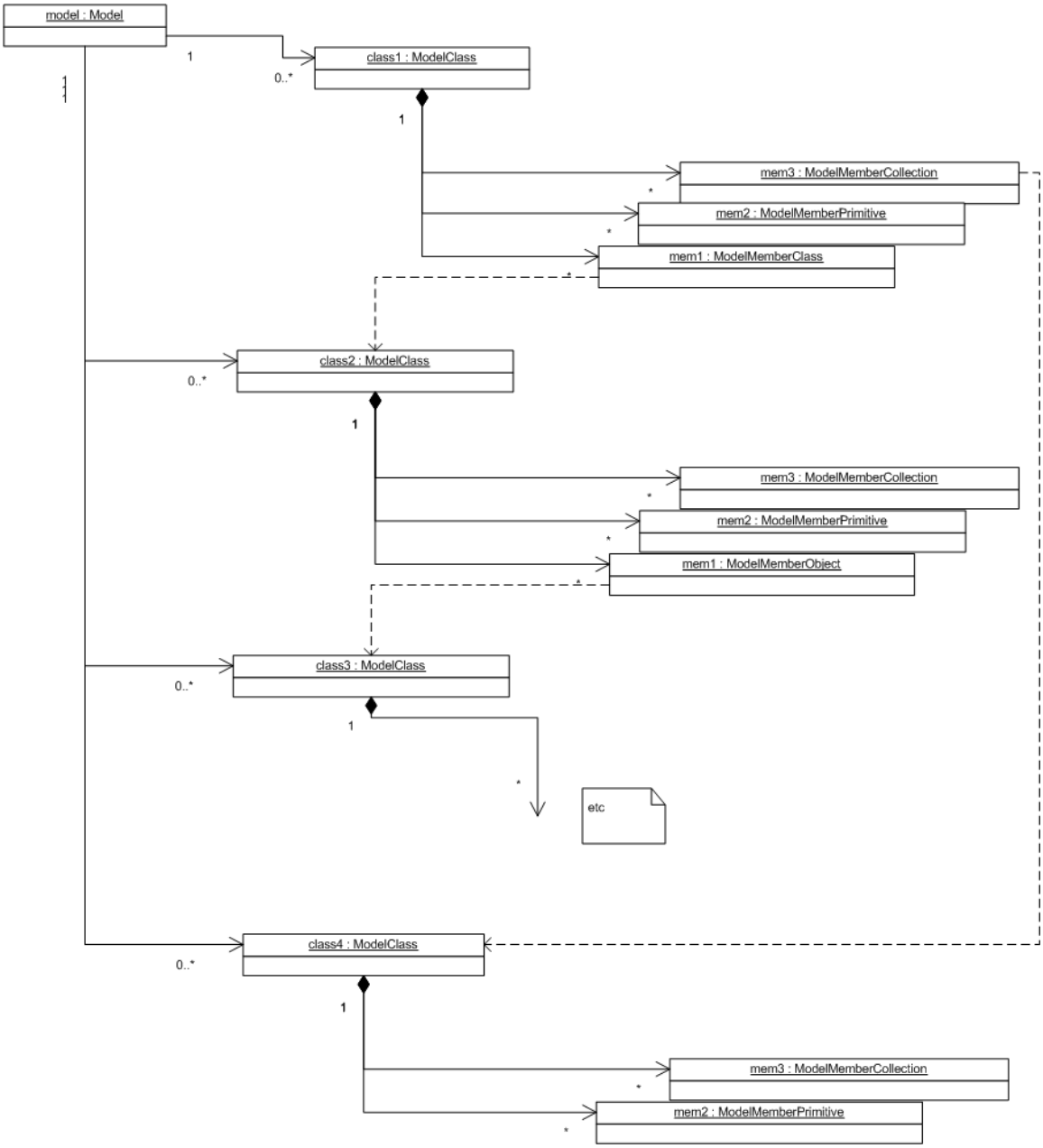
Model API Hierarchies

Model API Class Hierarchy



Model API Class Hierarchy

Model API Instance Hierarchy



Model API Instance Hierarchy

About the API Support Files

The following support files are installed during a **Basic Developer SDK installation**.

XHQ Core SDK Install vs. Basic Developer SDK Install
For a detailed list of what the XHQ SDK Installer does, see the topic, *For the XHQ Client Data API*, located in the XHQ Installation Guide.

Navigate to the location where you installed the API support files (which is typically `\Program Files\XHQ\XHQ Platform API\<x.x.xxxxx>`). The following directories and files are included.

API Support Files

Directories/Files	Description
`api_net_readme.txt`	Read Me for the XHQ .NET API
`api_readme.txt`	Read Me for the XHQ Java API
`lib`	This directory contains the XHQ API assemblies. Contains the following: • DLL files Support for the XHQ .NET API assemblies. (See the complete list of DLLs in the following section.) • JAR files Support for the XHQ Java API assemblies. (See the complete list of JARs in the following section.) • xhqapi.chm MSDN-style help file for the assemblies.
`doc`	This directory contains the Java Doc documentation set for the Java-based API.

The DLL Files for .NET API

The following DLLs are included with the XHQ Client API.

DLL Files

DLL	Description
`ICSharpCode.SharpZipLib.dll`	An open source implementation of the ZIP compression algorithm.
`IKVM.AWT.WinForms.dll`	Part of IKVM. The port of the Java AWT library to .NET WinForms.
`IKVM.OpenJDK.ClassLibrary.dll`	Part of IKVM. The port of the Java OpenJDK to .NET.
`IKVM.Runtime.dll`	Part of IKVM. A native-mode DLL used to support various internal (threading, security, etc.) IKVM mechanisms.
`IKVM.Runtime.JNI.dll`	Part of IKVM. A native-mode DLL used to support Java JNI.
`ikvm-native-win32-x64.dll`	Part of IKVM. A native-mode DLL support.
`Indx.Xhq.Client.Common.dll`	Common XHQ Client classes.
`Indx.Xhq.Client.Model.dll`	Data-access API to the Model server.
`Indx.Xhq.Client.Pm.dll`	Data-access API to the Performance Management server.
`Indx.Xhq.Client.Solution.dll`	Data-access API to the Solution Server.
`Indx.Xhq.Data.XhqClient.dll`	The XHQ ADO.NET Data Provider.
`collections.dll`	

The JAR Files for Java API

The following JAR files are included with the XHQ Client Data API.

JAR files included with the XHQ Client Data API

JAR File	Description
`xhqcoreapi.jar`	Contains the **core API** classes and resource files (for example, the connector.jar file).
`xhqmodelapi.jar`	Contains the **Model Server** client API-specific classes and resource files (everything that is documented).
`xhqpmapi.jar`	Contains the Application Server client API-specific classes and resource files (everything that is documented) that is used by **XHQ Performance Management** (Target Management and eLogs).
`xhqsolapi.jar`	Contains the **XHQ Solution Server** client API-specific classes and resource files (everything that is documented).
`collections.jar`	Contains collections resource files.

The Packages

The following packages are included with the XHQ Client API.

Packages included with the XHQ Client Data API

Package	Description
`com.indx.xhq`	Generic XHQ exceptions and error classes.
`com.indx.xhq.model`	Generic model server-related exceptions and error classes.
`com.indx.xhq.model.clientapi`	The main package for interfacing to the model server for role and class (component) meta data.
`com.indx.xhq.pm`	Generic application server-related exceptions and error classes.
`com.indx.xhq.pm.clientapi`	The main package for interfacing to the application server for runtime and XHQ Performance Management business logic.
`com.indx.xhq.solution`	Generic solution server-related exceptions and error classes.
`com.indx.xhq.solution.clientapi`	The main package for interfacing to the solution server for runtime and collection data.
`com.indx.xhq.util`	Utility classes of general use with XHQ (generally used by the XHQ Alert Notification System and the XHQ Client Data API).

Using the Client Data .NET API

To use the .NET API files you must reference the following assemblies in your Visual Studio project. These DLLs are found in the **`lib` directory** (located with the API support files, which by default is under `\Program Files\XHQ\XHQ Platform API\<x.x.xxxxx>`).

- IKVM.OpenJDK.Core
- IKVM.OpenJDK.Security
- IKVM.OpenJDK.SwingAWT
- IKVM.OpenJDK.Text
- IKVM.OpenJDK.Util
- IKVM.Runtime
- IKVM.Runtime.JNI
- Indx.Xhq.Client.Common.
- Indx.Xhq.Client.Model
- Indx.Xhq.Client.Solution

Previous versions of IKVM must be removed from your references.

Important Things to Note

- In addition to the above assemblies, the **`ikvm-native-win32-x64.dll`** file must be added to your path or to your project's bin directory. You can access this `ikvm-native` DLL from the XHQ Installation media, in the `Support\IKVM` folder, or from the `\Program Files\XHQ\XHQ Platform API\<x.x.xxxxx>` directory.

  The `ikvm-native` DLL cannot be added as a reference because it is a native DLL.
  It is copied to the `%windir%\system32` directory.

- If the application is using the XHQ ADO.NET Data Provider, the following assembly references must be added:
  - `CalithaLib`
  - `GoldParserEngine`
  - `Indx.Xhq.Data.XhqClient`

EXAMPLE: How to use Java-style Collections in .NET

The following sample code demonstrates how to convert a java.util.Collection into a .NET `CLR` compatible array and processed with a `foreach`.

```
Solution sol = Solution.getConnection( null );
java.util.Collection children = sol.getRoot().getMembers();
foreach( SolMember mem in children.toArray() ) {
  /* process each mem */
}
```

Using the Client Data Java API

To use the Java API files you must reference the following assemblies in your Visual Studio project. These JAR files are found in the **`lib` directory** (located with the API support files, which by default is under `\Program Files\XHQ\XHQ Platform API\<x.x.xxxxx>`).

- `xhqcoreapi`
  Contains the **core API** classes and resource files.
- `xhqmodelapi`
  Contains the **Model Server** client API-specific classes and resource files.
- `xhqpmapi`
  Contains the Application Server client API-specific classes and resource files used by **Performance Management**.
- `xhqsolapi`
  Contains the **Solution Server** client API-specific classes and resource files.

A set of Java examples using the XHQ Client Data API can be found in the topic, *Java Examples*.

Important Things to Note

- In addition to the above assemblies, the **`ikvm-native-win32-x64.dll`** file must be added to your path or to your project's bin directory. This `ikvm-native` DLL is available on the XHQ Installation media, in the `Support\IKVM` folder.

  The `ikvm-native` DLL cannot be added as a reference because it is a native DLL.
  It is copied to the `%windir%\system32` directory.

Using the ANS API

The ANS API allows third party developers to create and update, read, and delete ANS configurations through a public API that exposes the majority of the ANS Server transactions and operations, which is currently accessible only to the ANS web application client.

The ANS API allows full CRUD (Create/Read/Update/Delete)-style operations performed by the XHQ Alert Server on the ANS configurations.

The Java JAR for the ANS API is **`ansapi.jar`**.

The DLL for the .NET API is **`Indx.Xhq.Client.ANS.dll`**.

The main ANS API package where all the public API methods and ANS entity classes are defined is **`com.indx.xhq.ans.clientapi`**.

The main class where all the ANS data and configuration methods for the ANS server are defined in "com.**`indx.xhq.ans.clientapi.ANSConnection`**.

The `AnsWs` WCF web service provides a SOAP and a REST interface.

The programming pattern to access various ANS API configuration, follows these 3 basic steps:

Step 1 Connect to the ANS Server.

Step 2 Call the specific API method, or group of methods, using the ANS connection returned in step 1.

Step 3 Disconnect from the ANS Server.

Example:

```
ANSConnection ansConn = null;
BulkResponse resp = null;

try
{
     ansConn = ANSConnection.getConnection("localhost");
     ArrayList<ConditionDefEntity> defList = new
ArrayList<ConditionDefEntity>();

     //populate the "defList" list with ConditionDefEntity objects

     resp = ansConn.createConditionDefinitions(defList);
     while(resp.next())
     {
          System.out.println("Def ID: " + resp.getID());
          System.out.println("Error Message: " + resp.getErrorMsg());
          System.out.println(" ");
     }
}
catch (Exception e)
{
     e.printStackTrace();
}
finally
{
     if (ansConn != null)
          ansConn.disconnect();
}
```

More detailed information about this ANS API can be found in the Java Docs help files, located at the same location as the rest of the XHQ Public API Java doc files.

Connecting to and Disconnecting from the ANS Server

To **connect to the ANS server** using the public API use one the following methods:

Method:		`ANSConnection getConnection(String ansHost) throws AnsException`
Description:		Gets a non-shared instance of this class for interfacing with ANS on the specified ANS server host. The caller is responsible for managing the life cycle of this connection.
Parameters:	*ansHost*	The name of the machine that hosts the ANS Server.
Returns:		A connection to ANS.
Throws:		`AnsException` if there is a problem communicating with ANS.

Method:		`ANSConnection getConnection(String ansHost, String username, String password) throws AnsException`
Description:		Gets a non-shared instance of this class for interfacing with ANS on the specified ANS server host. The caller is responsible for managing the life cycle of this connection
Parameters:	*ansHost*	The name of the machine that hosts the ANS Server.
	username	The username for authentication.
	password	The password for authentication.
Returns:		A connection to ANS.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **disconnect from the ANS server** using the public API use the following method:

Method:	`void disconnect()`
Description:	Disconnect from ANS. Any subsequent ANS API calls will fail.

Creating, Updating, Reading, and Deleting Condition Definitions

To **create new condition definitions** in ANS using the public API use the following method:

Method:		`BulkResponse createConditionDefinitions (List<ConditionDefEntity> defList) throws AnsException`
Description:		Creates new ANS condition definitions using bulk processing.
Parameters:	*defList*	The list of ConditionDefEntity to be created.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **update/edit existing condition definitions** in ANS using the public API use the following method:

Method:		`BulkResponse updateConditionDefinitions (List<ConditionDefEntity> defList) throws AnsException`
Description:		Edits (updates) existing ANS condition definitions using bulk processing.
Parameters:	*defList*	The list of ConditionDefEntity to be edited/updated.
Returns:		A bulk value response in the form of BulkRespon.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **read existing condition definitions** in ANS using the public API use one the following methods:

Method:		`BulkResponseEx<ConditionDefEntity> getConditionDefinitionsById(List<Integer> defIdList) throws AnsException`
Description:		Retrieves a list of ANS condition definitions given their correspondent IDs using bulk processing.
Parameters:	*defList*	The list of condition definition IDs to be retrieved.
Returns:		A bulk response of ConditionDefEntity objects.
Throws:		`AnsException` if there is a problem communicating with ANS.

Method:		`BulkResponseEx<ConditionDefEntity> getConditionDefinitionsByName(List<String> defNameList) throws AnsException`
Description:		Retrieves a list of ANS condition definitions given their names using bulk processing.
Parameters:	*defNameList*	The list of condition definition names to be retrieved.
Returns:		A bulk response of ConditionDefEntity objects.
Throws:		`AnsException` if there is a problem communicating with ANS.

Method:		`BulkResponseEx<ConditionDefEntity> getConditionDefinitionsByList(String defListName) throws AnsException`
Description:		Retrieves a list of all ANS condition definitions for a given definition list, using bulk processing.
Parameters:	*defListName*	The definition list name.
Returns:		A bulk response of ConditionDefEntity objects.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **delete condition definitions** in ANS using the public API use the following method:

Method:		`BulkResponse deleteConditionDefinitions(List<Integer> defIdList) throws AnsException`
Description:		Deletes existing ANS condition definitions using bulk processing
Parameters:	*defIdList*	The list of condition definition IDs to be deleted.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

Creating, Updating, Reading, and Deleting Condition Definition Lists

To **create new condition definition lists** in ANS using the public API use the following method:

Method:		`BulkResponse createConditionDefinitionLists (List<ConditionDefListEntity> defList) throws AnsException`
Description:		Creates new ANS condition definition lists using bulk processing
Parameters:	*defList*	The list of ConditionDefListEntity to be created.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **update existing condition definition lists** in ANS using the public API use the following method:

Method:		`BulkResponse updateConditionDefinitionLists (List<ConditionDefListEntity> defList) throws AnsException`
Description:		Edits (updates) existing ANS condition definition lists using bulk processing
Parameters:	*defList*	The list of ConditionDefListEntity to be edited/updated.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **read existing condition definition lists** in ANS using the public API use one the following methods:

Method:		`BulkResponseEx<ConditionDefListEntity> getDefinitionListsById(List<Integer> defListIds) throws AnsException`
Description:		Retrieves a collection of ANS condition definition lists given their correspondent IDs using bulk processing.
Parameters:	*defListIds*	The list of condition definition list IDs to be retrieved.
Returns:		A bulk response of ConditionDefListEntity objects.
Throws:		`AnsException` if there is a problem communicating with ANS.

Method:		`BulkResponseEx<ConditionDefListEntity> getDefinitionListsByName(List<String> defListNames) throws AnsException`
Description:		Retrieves a collection of ANS condition definition lists given their names using bulk processing.
Parameters:	*defListNames*	The list of condition definition list names to be retrieved.
Returns:		A bulk response of ConditionDefListEntity objects.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **delete condition definition lists** in ANS using the public API use the following method:

Method:		`BulkResponsen deleteConditionDefinitionLists(List<Integer> defListIds) throws AnsException`
Description:		Deletes existing ANS condition definition lists using bulk processing.
Parameters:	*defListIds*	The list of condition definition list IDs to be deleted.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

Creating, Updating, Reading, and Deleting Definition List Sections

To **create new definition list sections** in ANS using the public API use the following method:

Method:		`BulkResponse createListSections(List<ListSectionEntity> sectionList) throws AnsException`
Description:		Creates new ANS definition list sections using bulk processing.
Parameters:	*sectionList*	The list of ListSectionEntity to be created.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **update existing definition list sections** in ANS using the public API use the following method:

Method:		`BulkResponse updateListSections(List<ListSectionEntity> sectionList) throws AnsException`
Description:		Edits (updates) existing ANS definition list sections using bulk processing
Parameters:	*sectionList*	The list of ListSectionEntity objects to be edited/updated.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **read existing definition list sections** in ANS using the public API use one the following methods:

Method:		`BulkResponseEx<ListSectionEntity>` `getListSectionsById(Map<Integer, Integer> idsMap) throws AnsException`
Description:		Retrieves a collection of ANS definition list sections given their correspondent IDs and the definition list IDs, using bulk processing
Parameters:	*idsMap*	The IDs map with the section ID as the "key" and the definitions list ID as the "value".
Returns:		A bulk response of ListSectionEntity objects.
Throws:		`AnsException` if there is a problem communicating with ANS.

Method:		`BulkResponseEx<ListSectionEntity>` `getListSectionsByName(Map<String, String> namesMap) throws AnsException`
Description:		Retrieve.s a collection of ANS definition list sections given their correspondent names and the definition list names, using bulk processing
Parameters:	*namesMap*	The name map with the section name as the "key" and the definitions list name as the "value".
Returns:		A bulk response of ListSectionEntity objects.
Throws:		`AnsException` if there is a problem communicating with ANS.

Method:		`BulkResponseEx<ListSectionEntity>` `getListSectionsByList(String defListName) throws AnsException`
Description:		Retrieves all ANS sections for a given definition list, using bulk processing.
Parameters:	*defListName*	The definition list name.
Returns:		A bulk response of ConditionDefEntity objects.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **delete definition list sections** in ANS using the public API use the following method:

Method:		`BulkResponse deleteListSections(List<Integer> listSectionIds) throws AnsException`
Description:		Deletes existing ANS definition list sections using bulk processing.
Parameters:	*listSectionIds*	The definition list section IDs to be deleted.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

Creating, Updating, Reading, and Deleting Subscriptions

To **create new subscriptions** in ANS using the public API use the following method:

Method:		`BulkResponse createSubscriptions(List<SubscriptionEntity> subsList) throws AnsException`
Description:		Creates new ANS subscriptions using bulk processing.
Parameters:	*subsList*	The list of SubscriptionEntity to be created.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **update existing subscriptions** in ANS using the public API use the following method:

Method:		`BulkResponse updateSubscriptions(List<SubscriptionEntity> subsList) throws AnsException`
Description:		Edits (updates) existing ANS subscriptions using bulk processing.
Parameters:	*subsList*	The list of SubscriptionEntity to be edited/updated.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **read existing subscriptions** in ANS using the public API use one the following methods:

Method:		`BulkResponseEx<SubscriptionEntity>` `getSubscriptionsById(List<Integer> subscIdList) throws AnsException`
Description:		Retrieves a collection of ANS condition subscriptions given their correspondent IDs using bulk processing.
Parameters:	*subscIdList*	The list of subscription IDs to be retrieved.
Returns:		A bulk response of SubscriptionEntity objects.
Throws:		`AnsException` if there is a problem communicating with ANS.

Method:		`BulkResponseEx<SubscriptionEntity>` `getSubscriptionsByList(String subscListName) throws AnsException`
Description:		Retrieves all ANS subscriptions for a given subscription list, using bulk processing.
Parameters:	*subscListName*	The subscription list name.
Returns:		A bulk response of SubscriptionEntity objects.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **delete existing subscriptions** in ANS using the public API use the following method:

Method:		`BulkResponse deleteSubscriptions(List<Integer> subscrIds) throws AnsException`
Description:		Deletes existing ANS subscriptions using bulk processing.
Parameters:	*subscrIds*	The list of subscription IDs to be deleted.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

Creating, Updating, Reading, and Deleting Subscription Lists

To **create new subscription lists** in ANS using the public API use the following method:

Method:		`BulkResponse createSubscriptionLists (List<SubscriptionListEntity> subsList) throws AnsException`
Description:		Creates new ANS subscription lists using bulk processing.
Parameters:	*subsList*	The list of SubscriptionListEntity to be created.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **update existing new subscription lists** in ANS using the public API use the following method:

Method:		`BulkResponse updateSubscriptionLists (List<SubscriptionListEntity> subsList) throws AnsException`
Description:		Edits (updates) existing ANS subscription lists using bulk processing.
Parameters:	*subsList*	The list of SubscriptionListEntity to be edited/updated.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **read existing subscription lists** in ANS using the public API use one the following methods:

Method:		`BulkResponseEx<SubscriptionListEntity> getSubscriptionListsById(List<Integer> subsListIds) throws AnsException`
Description:		Retrieves a collection of ANS subscription lists given their correspondent IDs using bulk processing.
Parameters:	*subsListIds*	The list of subscription list IDs to be retrieved.
Returns:		A bulk response of SubscriptionListEntity objects.
Throws:		`AnsException` if there is a problem communicating with ANS.

Method:		`BulkResponseEx<SubscriptionListEntity> getSubscriptionListsByName(List<String> subsListNames) throws AnsException`
Description:		Retrieves a collection of ANS subscription lists given their names using bulk processing
Parameters:	*subsListNames*	The list of subscription list names to be retrieved.
Returns:		A bulk response of SubscriptionListEntity objects.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **delete existing subscription lists** in ANS using the public API use the following method:

Method:		`BulkResponse deleteSubscriptionLists(List<Integer> subscListIds) throws AnsException`
Description:		Deletes existing ANS subscription lists using bulk processing
Parameters:	*defListIds*	The list of condition definition list IDs to be deleted.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

Creating, Updating, Reading, and Deleting ANS Codes like Categories and Interval Types

To **create new codes** in ANS using the public API use the following method:

Method:		`BulkResponse createCodes(List<CodeKeyEntity> codeList) throws AnsException`
Description:		Creates new ANS codes (e.g. categories, interval types) using bulk processing.
Parameters:	*codeList*	The list of CodeKeyEntity to be created.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **update existing codes** in ANS using the public API use the following method:

Method:		`BulkResponse updateCodes(List<CodeKeyEntity> codeList) throws AnsException`
Description:		Edits (updates) existing ANS codes (e.g. categories, interval types) using bulk processing
Parameters:	*codeList*	The list of CodeKeyEntity to be created.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **read existing ANS codes** using the public API use one the following methods:

Method:		`BulkResponseEx<CodeKeyEntity> getCodesById(List<Integer> codeIdList) throws AnsException`
Description:		Retrieves a list of ANS codes (e.g. categories) given their correspondent IDs using bulk processing.
Parameters:	*codeIdList*	The list of code IDs to be retrieved.
Returns:		A bulk response of CodeKeyEntity objects.
Throws:		`AnsException` if there is a problem communicating with ANS.

Method:		`BulkResponseEx<CodeKeyEntity>` `getCodesByName(Map<String, CodeType> namesMap) throws AnsException`
Description:		Retrieves list of ANS codes (e.g. categories) given their correspondent names and code types, using bulk processing
Parameters:	*namesMap*	The name map with the code name as the "key" and the code type as the "value".
Returns:		A bulk response of CodeKeyEntity objects.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **delete existing codes** in ANS using the public API use the following method:

Method:		`BulkResponse deleteCodes(List<Integer> codeIdList) throws AnsException`
Description:		Deletes existing ANS codes using bulk processing.
Parameters:	*codeIdList*	The code key IDs to be deleted.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

Creating, Updating, Reading, and Deleting Notification Routing

To **create new notification routings** in ANS using the public API use the following method:

Method:		`BulkResponse createNotificationRoutings(List<RoutingEntity> routingList) throws AnsException`
Description:		Creates new ANS notification routings using bulk processing.
Parameters:	*routingList*	The list of RoutingEntity to be created.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **update existing notification routings** in ANS using the public API use the following method:

Method:		`BulkResponse updateNotificationRoutings(List<RoutingEntity> routingList) throws AnsException`
Description:		Edits (updates) existing ANS notification routings using bulk processing.
Parameters:	*routingList*	The list of RoutingEntity to be created.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

To **read existing notification routings** in ANS using the public API use one the following methods:

Method:		`BulkResponseEx<RoutingEntity>` `getNotificationRoutingsById(List<Integer> routeIdList) throws AnsException`
Description:		Retrieves a collection of ANS notification routings given their correspondent IDs using bulk processing.
Parameters:	*routeIdList*	The routeIdList.
Returns:		A bulk response of RoutingEntity objects.
Throws:		`AnsException` if there is a problem communicating with ANS.

Method:	`BulkResponseEx<RoutingEntity> getMyNotificationRoutings() throws AnsException`
Description:	Retrieves all ANS notification routings for the caller user, using bulk processing
Returns:	A bulk response of RoutingEntity objects.
Throws:	`AnsException` if there is a problem communicating with ANS.

To **delete existing notification routings** in ANS using the public API use the following method:

Method:		`BulkResponse deleteNotificationRoutings(List<Integer> routeIdList) throws AnsException`
Description:		Deletes existing ANS notification routings using bulk processing.
Parameters:	*routeIdList*	The routing IDs to be deleted.
Returns:		A bulk value response in the form of BulkResponse.
Throws:		`AnsException` if there is a problem communicating with ANS.

Solution Server Client API Examples

There are two parts to this section: the Java examples followed by the .NET examples.

Java Examples

These are simple representative **JAVA examples** showing the use of the XHQ Client Data API. They include the following tasks:

- EXAMPLE 1 Connecting Disconnecting The Basic Procedure
- EXAMPLE 2 Examining Global Collections
- EXAMPLE 3 Retrieving the Current Value of a Member
- EXAMPLE 4 Retrieving a Tag Member Value
- EXAMPLE 5 Computing a Sum Historical Aggregate Value
- EXAMPLE 6 Retrieving Time Series Trend Values
- EXAMPLE 7 Finding the Count of Rows in Link Collection
- EXAMPLE 8 Querying the Link Collection

EXAMPLE 1: CONNECTING/DISCONNECTING – THE BASIC PROCEDURE

In this example, you will:

- Connect to the XHQ Solution Server.
- Display XHQ Solution Server information.
- Show root object information.
- Disconnect from the XHQ Solution Server.

Given the following source code:

```
import com.indx.xhq.solution.*;
import com.indx.xhq.solution.clientapi.*;
public class sample1 {
  private static void log( String m ) { System.out.println( m ); }
  public static void main( String[] args ) {
    Solution s = null;
    try {
      s = Solution.getConnection( "localhost" );
      log( "TZ: " + s.getSolutionTimeZone().getDisplayName() );
      log( "Security enabled: " + s.getSecurityEnabled() );
      SolRoot r = s.getRootObject();
      log("Root ID: " + r.getOID() );
      log("Root Path: " + r.getPath() );
      log("Root Class: " + r.getClassName() );
    } catch(SolException e) {log("Error: "+e.getMessageChain());
    } finally { if( s != null ) s.disconnect(); }  }
}
```

The **output** looks like this:

```
TZ: Pacific Standard Time
Security enabled: false
Root ID: 1
Root Path: ::InterpTest
Root Class: root
```

EXAMPLE 2: EXAMINING GLOBAL COLLECTIONS

In this example, you will:

- Connect to the XHQ Solution Server.
- Iterate through the global collections.
- Detect the "Tag Sync" collection.
- Display the name of each global collection.
- Disconnect from the XHQ Solution Server.

Given the following source code:

```
import java.util.*;
import com.indx.xhq.solution.*;
import com.indx.xhq.solution.clientapi.*;
public class sample2 {
  private static void log( String m ) { System.out.println( m ); }
  public static void main( String[] args ) {
    Solution s = null;
    try {
      s = Solution.getConnection( "localhost" );
      Iterator it = s.getGlobalCollections().iterator();
      while( it.hasNext() ) {
         SolCollection c = (SolCollection)it.next();
         String t = c instanceof SolTagCollection ? "Tag " : "";
         log( t + "Collection: " + c.getName() );
      }
    } catch(SolException e) {log("Error: "+e.getMessageChain());
    } finally { if( s != null ) s.disconnect(); }  }
  }
}
```

The **output** looks like this:

```
Tag Collection: all_tags
Collection: EQITEMS
Collection: EQWO
Collection: mssql_tagdefs
Collection: odbc_tagdefs
Collection: odbctest
Collection: phd_tags
Collection: pi_tags
Collection: RSS_dans
Collection: TagDefs
Collection: TimeSeriesSource
Collection: VTQStreams
Collection: xml_wo
```

EXAMPLE 3: RETRIEVING THE CURRENT VALUE OF A MEMBER

In this example, you will:

- Connect to the XHQ Solution Server.
- Get the root object.
- Get the primitive member specified on the command-line.
- Show the member's path and current value (including timestamp, quality, and source).
- Disconnect from the XHQ Solution Server.

Given the following source code:

```
import com.indx.xhq.solution.*;
import com.indx.xhq.solution.clientapi.*;
public class sample3 {
  private static void log( String m ) { System.out.println( m ); }
  public static void main( String[] args ) {
    Solution s = null;
    try {
      s = Solution.getConnection( "localhost" );
      SolRoot r = s.getRootObject();
      SolMember mem = r.getMember( args[0] );
      SolMemberPrimitive pm = (SolMemberPrimitive)mem;
      log( pm.getPath() + " = " + pm.getServerValue() );
    } catch(SolException e) {log("Error: "+e.getMessageChain());
    } finally { if( s != null ) s.disconnect(); }  }
  }
}
```

The **output** looks like this:

```
::InterpTest.avgtest.val = V: 77.03096 T: 02/24/2005 08:35:27.543 Q: 0x00C0
S: CONNECTOR (Simulator)
```

The Member must be subscribed or the value will be uninitialized.

EXAMPLE 4: RETRIEVING A TAG MEMBER VALUE

In this example, you will:

- Connect to the XHQ Solution Server.
- Get the tag specified on the command-line.
- Show the member's path and current value (including timestamp, quality, and source).
- Disconnect from the XHQ Solution Server.

Given the following source code:

```
import java.util.*;
import com.indx.xhq.solution.*;
import com.indx.xhq.solution.clientapi.*;
public class sample4 {
  private static void log( String m ) { System.out.println( m ); }
  public static void main( String[] args ) {
    Solution s = null;
    try {
      s = Solution.getConnection( "localhost" );
      SolTag t s.getTagObject( args[0] );
      log( "Members of tag: " + t.getPath() );
      Iterator it = t.getMembers().iterator();
      while( it.hasNext() ) {
         Object o = it.next();
         if( o instanceof SolMemberPrimitive ) {
            SolMemberPrimitive pm = (SolMemberPrimitive)o;
            log( pm.getPath() + " = " + pm.getServerValue() );
         }
      }
    } catch(SolException e) {log("Error: "+e.getMessageChain());
    } finally { if( s != null ) s.disconnect(); }  }
  }
}
```

The **output** looks like this:

```
Members of tag: 50VI106
50VI106.Config = V:  T: 12/31/1969 16:00:00.000 Q: 0x0120 S: NONE
50VI106.Description = V: K10 IB LEFT RADIAL T: 02/22/2005 15:04:57.606 Q: 0x00C0 S: NONE
50VI106.IValue = V: 0 T: 12/31/1969 16:00:00.000 Q: 0x0120 S: NONE
50VI106.RValue = V: 0.0 T: 02/22/2005 15:04:57.606 Q: 0x0120 S: NONE
50VI106.SValue = V:  T: 12/31/1969 16:00:00.000 Q: 0x0120 S: NONE
50VI106.TagName = V:  T: 12/31/1969 16:00:00.000 Q: 0x0120 S: NONE
50VI106.Units = V: MILS T: 02/22/2005 15:04:57.606 Q: 0x00C0 S: NONE
```

EXAMPLE 5: COMPUTING A SUM (HISTORICAL AGGREGATE) VALUE

In this example, you will:

- Connect to the XHQ Solution Server.
- Get the root object.
- Get the primitive member specified on the command-line.
- Define the time span (for t1 = now, t2 = now – 30 minutes).
- Get the SUM of P.M. values over the time range.
- Disconnect from the XHQ Solution Server.

Given the following source code:

```
import com.indx.xhq.util.*;
import com.indx.xhq.solution.*;
import com.indx.xhq.solution.clientapi.*;
public class sample5 {
  private static void log( String m ) { System.out.println( m ); }
  public static void main( String[] args ) {
    Solution s = null;
    try {
      s = Solution.getConnection( "localhost" );
      SolRoot r = s.getRootObject();
      SolMemberPrimitive pm = (SolMemberPrimitive)r.getMember( args[0] );
      Timestamp t1 = Timestamp.now();
      Timestamp t2 = new Timestamp( t1 );
      t1.adjust( -1000*60*30 );  // Thirty minutes ago
      VTQ vtq = pm.getAggregateValue( t1, t2, TSAGG.SUM );
      log( pm.getPath() + " SUM = " + vtq.floatValue() );
    } catch(SolException e) {log("Error: "+e.getMessageChain());
    } finally { if( s != null ) s.disconnect(); }  }
  }
}
```

The **output** looks like this:

```
::InterpTest.avgtest.val SUM = 7703.406
```

EXAMPLE 6: RETRIEVING TIME-SERIES (TREND) VALUES

In this example, you will:

- Connect to the XHQ Solution Server.
- Get the root object.
- Get the primitive member specified on the command-line.
- Define the time span (for t1 = now, t2 = now – 30 minutes).
- Get values of Span in FITS mode.
- Show the "All Values" or "No History" message.
- Disconnect from the XHQ Solution Server.

Given the following source code:

```
import com.indx.xhq.util.*;
import com.indx.xhq.solution.*;
import com.indx.xhq.solution.clientapi.*;
public class sample6 {
  private static void log( String m ) { System.out.println( m ); }
  public static void main( String[] args ) {
    Solution = null;
    try {
      s = Solution.getConnection( "localhost" );
      SolRoot r = s.getRootObject();
      SolMemberPrimitive pm = (SolMemberPrimitive)r.getMember( args[0] );
      Timestamp t1 = Timestamp.now();
      Timestamp t2 = new Timestamp( t1 );
      t1.adjust( -1000*60*30 );  // Thirty minutes ago
      TimeSeriesResponse tsr = pm.getTimeSeriesData( t1, t2, TSMODE.FITS, 10);
     if( tsr.getSize() == 0 )
        log( pm.getPath() + " has no history data for that period" );
     for( int i = 0; tsr.next(); i++ )
        log( "#"+i+": " +tsr.getVTQ() );
    } catch(SolException e) {log("Error: "+e.getMessageChain());
    } finally { if( s != null ) s.disconnect(); }  }
  }
}
```

The **output** looks like this:

```
#0: V: 17.746721 T: 02/24/2005 11:23:03.947 Q: 0x00C0 S: NONE
#1: V: 14.814265 T: 02/24/2005 11:23:07.543 Q: 0x00C0 S: NONE
#2: V: 0.056571905 T: 02/24/2005 11:23:47.543 Q: 0x00C0 S: NONE
#3: V: 99.94343 T: 02/24/2005 11:26:17.543 Q: 0x00C0 S: NONE
#4: V: 52.855724 T: 02/24/2005 11:52:27.543 Q: 0x00C0 S: NONE
#5: V: 32.3052 T: 02/24/2005 11:52:47.543 Q: 0x00C0 S: NONE
```

EXAMPLE 7: FINDING THE COUNT OF ROWS IN LINK COLLECTION

In this example, you will:

- Connect to the XHQ Solution Server.
- Get the root object.
- Get the member collection specified on the command-line.
- Show the Count of Rows in the collection.
  Note: The query is restricted by the criteria.
- Disconnect from the XHQ Solution Server

Given the following source code:

```
import com.indx.xhq.util.*;
import com.indx.xhq.solution.*;
import com.indx.xhq.solution.clientapi.*;
public class sample6 {
  private static void log( String m ) { System.out.println( m ); }
  public static void main( String[] args ) {
    Solution s = null;
    try {
      s = Solution.getConnection( "localhost" );
      SolRoot r = s.getRootObject();
      SolMember o = r.getMember( args[0] );
      SolMemberCollection c = (SolMemberCollection)o;
      log( c.getPath() + " has " +
           c.queryCount(new SolQueryCriteria(args[1]+" IS NOT NULL", null))
           + " rows" );
    } catch(SolException e) {log("Error: "+e.getMessageChain());
    } finally { if( s != null ) s.disconnect(); }  }
  }
}
```

The **output** looks like this:

```
::InterpTest.AllTags.Tags has 10 rows
```

EXAMPLE 8: QUERYING THE LINK COLLECTION

In this example, you will:

- Connect to the XHQ Solution Server.
- Get the root object.
- Get the member collection specified on the command-line.
- Query with column mapping and ordering.
- Show the values from the query.
- Disconnect from the XHQ Solution Server.

Given the following source code:

```
import com.indx.xhq.util.*;
import com.indx.xhq.solution.*;
import com.indx.xhq.solution.clientapi.*;
public class sample6 {
  private static void log( String m ) { System.out.println( m ); }
  public static void main( String[] args ) {
    Solution s = null;
    try {
      s = Solution.getConnection( "localhost" );
      SolRoot r = s.getRootObject();
      SolMember o = r.getMember( args[0] );
      SolMemberCollection c = (SolMemberCollection)o;
      SolQueryCriteria qc = new SolQueryCriteria( null, args[1] );
      String[] cols = new String[]{ args[1], args[2] };
      dumpSet( c.query( cols, qc ), cols );
    } catch(SolException e) {log("Error: "+e.getMessageChain());
    } finally { if( s != null ) s.disconnect(); }  }
  }
private static void dumpSet( SolCollectionSet set, String[] cols )
    throws SolException
  {
    while( set.next() ) {
      StringBuffer buf = new StringBuffer();
      buf.append( "ROW: " ).append( set.getRowNumber() );
      int size = set.getNumberColumns();
      for( int c = 1; c <= size; c++ ) {
        buf.append( "\n\tC#" ).append( c ).append( ": " );
        if( !set.isValid( c ) )
          buf.append( "NOT VALID " );
        if( set.isNULL( c ) )
          buf.append( "NULL " );
        buf.append( cols[c-1]+": " ).append( set.getStringValue( c ) );
      }
      log( buf.toString() );
    }
    log( "\n" );
  }
}
```

The **output** looks like this:

```
ROW: 1
        C#1: TagName: 50VI105A
        C#2: Description: K10 OB RIGHT RADIAL
ROW: 2
        C#1: TagName: 50VI106
        C#2: Description: K10 IB LEFT RADIAL
ROW: 3
```

```
        C#1: TagName: 50VI106A
        C#2: Description: K10 IB LEFT RADIAL
ROW: 4
        C#1: TagName: 50VI107
        C#2: Description: K10 IB RIGHT RADIAL
ROW: 5
        C#1: TagName: 50VI107A
        C#2: Description: K10 IB RIGHT RADIAL
ROW: 6
        C#1: TagName: 50VI140
        C#2: Description: TK140 IB LEFT RADIAL
ROW: 7
        C#1: TagName: 50VI140A
        C#2: Description: TK140 IB LEFT RADIAL
```

.NET Example

The following **.NET example** is comprised of several tasks:

Example 1	Connecting to and disconnecting from the XHQ Solution Server.
Example 2	Examining global collections.
Example 3	Retrieving the current value of a member.
Example 4	Retrieving a Tag member value.
Example 5	Computing the SUM value.
Example 6	Retrieving the time-series values.
Example 7	Querying the Link collection.

These tasks are similar to the Java examples given in the previous topic. To see the **output** for each case, refer to the corresponding Java example (for Example 7, see Java Example 8).

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using com.indx.xhq.solution;
using com.indx.xhq.util;
using com.indx.xhq.solution.clientapi;

namespace Examples
{
    public class Program
    {
        private static void log( String m ) { Console.WriteLine( m ); }
        public static void Main(string[] args)
        {
            Example1();
            Example2();
            Example3();
            Example4();
            Example5();
            Example6();
            Example7();
        }

        public static void Example1()
        {
            Solution s = null;
            try
            {
                s = Solution.getConnection( "vtl2k8api" );
                log( "TZ: " + s.getSolutionTimeZone().getDisplayName() );
                log( "Security enabled: " + s.getSecurityEnabled() );
                SolRoot r = s.getRootObject();
                log("Root ID: " + r.getOID() );
                log("Root Path: " + r.getPath() );
                log("Root Class: " + r.getClassName() );
            }
            catch(SolException e)
```

```
        {
            log("Error: "+e.getMessageChain());
        }
        finally
        {
            if( s != null ) s.disconnect();
        }
    }

    public static void Example2()
    {
        Solution s = null;
        try
        {
            s = Solution.getConnection( "vtl2k8api" );
            java.util.Iterator it = s.getGlobalCollections().iterator();
            while (it.hasNext())
            {
                SolCollection c = (SolCollection)it.next();
                String t = c is SolTagCollection ? "Tag " : "";
                log(t + "Collection: " + c.getName());
            }
        }
        catch(SolException e)
        {
            log("Error: "+e.getMessageChain());
        }
        finally
        {
            if( s != null ) s.disconnect();
        }
    }

    public static void Example3()
    {
        Solution s = null;
        try
        {
            s = Solution.getConnection("vtl2k8api");
            SolRoot r = s.getRootObject();
            SolMember mem = r.getMember("::SDL.HistData.a_raw_real");
            SolMemberPrimitive pm = (SolMemberPrimitive)mem;
            log(pm.getPath() + " = " + pm.getServerValue());
        }
        catch (SolException e)
        {
            log("Error: " + e.getMessageChain());
        }
        finally
        {
            if (s != null) s.disconnect();
        }
    }

    public static void Example4()
    {
        Solution s = null;
        try
        {
            s = Solution.getConnection("vtl2k8api");
            SolTag t = s.getTagObject( "11FC0123" );
```

```
                log( "Members of tag: " + t.getPath() );
                java.util.Iterator it = t.getMembers().iterator();
                while( it.hasNext() ) {
                Object o = it.next();
                if( o is SolMemberPrimitive ) {
                SolMemberPrimitive pm = (SolMemberPrimitive)o;
                log( pm.getPath() + " = " + pm.getServerValue() );
                }
                }
            }
            catch (SolException e)
            {
                log("Error: " + e.getMessageChain());
            }
            finally
            {
                if (s != null) s.disconnect();
            }
        }

        public static void Example5()
        {
            Solution s = null;
            try
            {
                s = Solution.getConnection("vtl2k8api");
                SolRoot r = s.getRootObject();
                SolMemberPrimitive pm = (SolMemberPrimitive)r.getMember
("::SDL.HistData.a_raw_real");
                Timestamp t1 = Timestamp.now();
                Timestamp t2 = new Timestamp(t1);
                t1.adjust(-1000 * 60 * 30); // Thirty minutes ago
                VTQ vtq = pm.getAggregateValue(t1, t2, TSAGG.SUM);
                log(pm.getPath() + " SUM = " + vtq.floatValue());
            }
            catch (SolException e)
            {
                log("Error: " + e.getMessageChain());
            }
            finally
            {
                if (s != null) s.disconnect();
            }
        }

        public static void Example6()
        {
            Solution s = null;
            try
            {
                s = Solution.getConnection("vtl2k8api");
                SolRoot r = s.getRootObject();
                SolMemberPrimitive pm = (SolMemberPrimitive)r.getMember
("::SDL.HistData.a_raw_real");
                Timestamp t1 = Timestamp.now();
                Timestamp t2 = new Timestamp(t1);
                t1.adjust(-1000 * 60 * 30); // Thirty minutes ago
                TimeSeriesResponse tsr = pm.getTimeSeriesData(t1, t2, TSMODE.FITS,
                10);
                if (tsr.getSize() == 0)
                    log(pm.getPath() + " has no history data for that period");
```

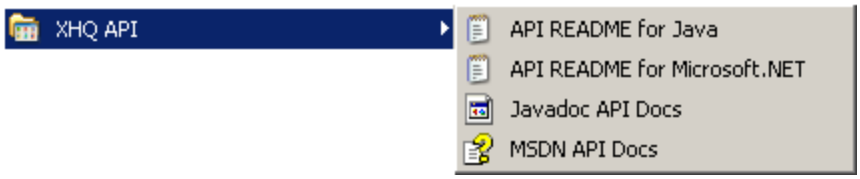
```
                for (int i = 0; tsr.next(); i++)
                    log("#" + i + ": " + tsr.getVTQ());
            }
            catch (SolException e)
            {
                log("Error: " + e.getMessageChain());
            }
            finally
            {
                if (s != null) s.disconnect();
            }
        }

        public static void Example7()
        {
            Solution s = null;
            try
            {
                Timestamp t1 = new Timestamp(new java.util.Date());
                t1.adjust(-10 * 60 * 1000);
                log(t1.toString());

                s = Solution.getConnection("vtl2k8api");
                ICollectionQuery icq = s.getGlobalCollection("::TSR2");
                SolQueryCriterion[] criterions = new SolQueryCriterion[] {
                   new SolQueryCriterion("RealValue", SolQueryCriterion.LE, new
VariantValue(28.0)),
                   new SolQueryCriterion("DateTimeValue", SolQueryCriterion.GE, new
VariantValue(t1)),
                };
                int count = icq.queryCount(new SolQueryCriteria(criterions));
                log("Count: " + count);
                log("**");
            }
            catch (SolException e)
            {
                log("Error: " + e.getMessageChain());
            }
            finally
            {
                if (s != null) s.disconnect();
            }
        }
    }
}
```

Getting Help

To access help files

Do one of the following:

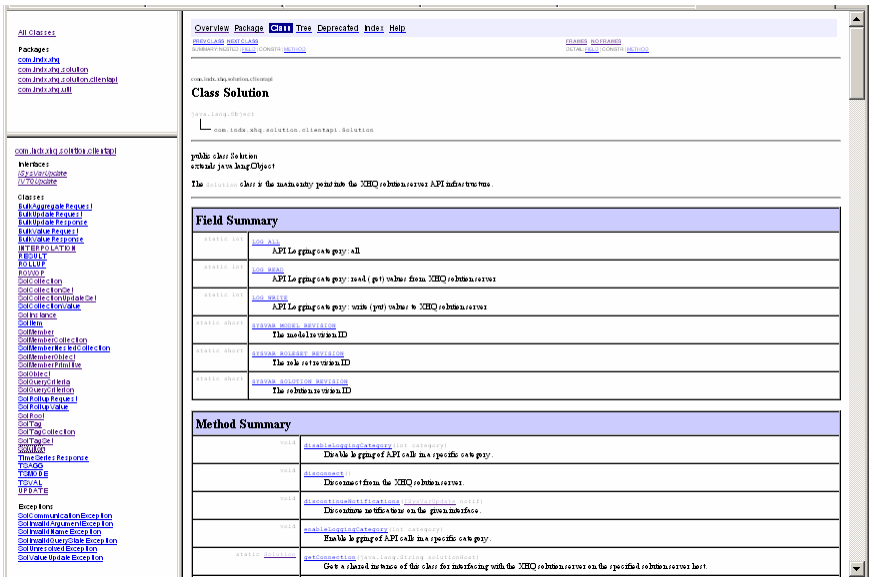
- From the **Start menu**, click **All Programs**, and then click **XHQ API** and then **MSDN API Docs**. From this submenu, you can access these documents.



or

- Navigate to the location where you installed the API support files (which is typically `\Program Files\XHQ\XHQ Platform API\4.0.xxxxx`). You can access the following files.

Documentation Files	Description
`doc\api\index.html`	This directory contains the JavaDoc documentation set for the Java-based API.
`lib\xhqapi.chm`	MSDN-style help file for the assemblies.
`Readme's`	Up-to-the-minute product information, including requirements, how to install, and known issues.



JavaDoc for XHQ Client Data API

2 | Web Services

The term Web Services describes a standardized way of integrating applications over Internet protocols, such as HTTP or SMTP, using the following open source standards:

- XML (Extensible Markup Language)
- SOAP (Simple Object Access Protocol)
  This is a protocol that provides a standard packaging structure for transporting XML documents over a variety of standard Internet technologies like HTTP, SMTP and FTP. It also defines encoding and binding standards for encoding non-XML RPC invocations in XML for transport.
- WSDL (Web Service Description Language)
  This is an XML-based language that describes the interface of a Web Service in a standardized way, by specifying how a Web Service represents the input and output parameters of an invocation externally, the method's structure, the nature of the invocation (in only, in/out, and so forth) and the service's protocol binding.
- UDDI (Universal Description, Discovery and Integration)
  This is a registry service for Web Services with regards to advertisement, discovery and integration purposes. UDDI provides a structure for representing businesses, business relationships, available Web Services, specification metadata and Web Services access points.
- OData (Open Data Protocol)
  OData is an open protocol that allows the creation and consumption of queryable and interoperable RESTful APIs in a simple and standard way. OData RESTful APIs are easy to consume. The OData metadata, a machine-readable description of the data model of the APIs, enables the creation of powerful generic client proxies and tools.

In Web Services, XML is used to tag the data, SOAP is used to transfer the data, WSDL is used for describing the services available, and UDDI is used for listing what services are available. Web Services allow businesses to easily communicate with each other. They provide the means for organizations to exchange data without the potential hindrances caused by technical dependencies and firewall restrictions.

Benefits of Web Services

Web Services have the potential to offer significant benefits to XHQ customers. Because it is based on open-source standards, Web Services technology is relatively inexpensive and easy to implement. It works with any application language on any platform. It works for both internal and external integration projects and is a loosely coupled technology. It is a safe choice because nearly all vendors support it, vendors who are committed to ensuring interoperability.

While the Client API (in Java or .NET form) and XHQ ADO.NET Data Provider can supply information resident in XHQ and receive information from external sources, it requires that Java or .NET calls be made to exchange information. Web Services provide great ease of use and implementation in both scenarios and it removes the dependence on platform and language calls.

Here are some practical scenarios when Web Services technology is implemented:

- XHQ can make available real-time business-operational data via Web Services for use by other applications running in the customer environment. The integration developers can invoke the XHQ Web Service to bring XHQ data into other applications.
- XHQ end users may want to view (and consume) data supplied by other internal and external applications via Web Services. XHQ solution developers can invoke this Web Service in XHQ views.

The XHQ Web Services implementation is an API addition and does not affect the current XHQ System, the XHQ Solution Viewer (browser client), or the XHQ Solution Server.

Web Services Set-up

Web Services Set-up Checklist Summary
☐ Set the solution property.
☐ Set the default XHQ host server for connecting the remote/distributed XhqWse web service.
☐ Modify the WAKA XML configuration file for remote/distributed XhqWse web service.
☐ Set the default time zone through the `web.config` file.
☐ Set optional `web.config` properties.

Setting the Solution Property

In addition to the software requirements, you must set the solution property in order to access the solution.

The **`solutionsettings.properties`** file allows you to set tunable parameters that deal with the solution server's performance. This file is stored at the location specified by the environment variable **`%XHQ_SERVER_REPOS%`** on the XHQ Enterprise Server (which is typically **`%SystemDrive%\XHQ\data\repos`**).

This file is not included with the XHQ installation and must be created by the user. Changes to these parameters require server **restart**.

The property and it's setting is as follows:

```
net.indx.repository.solution.server.solutionserver.allowapiclients=true
```

For more information on the properties file, go to the topic, *Using solutionsettings.properties*, located in the XHQ Administrator's Guide.

To set the solution property

1. Using a text editor, open a new file and saved it as **`solutionsettings.properties`** in the directory specified by the **`%XHQ_SERVER_REPOS%`** environment variable (which is typically `%SystemDrive%\XHQ\data\repos`).

   Make sure you save it as a **`.properties`** file and NOT a .txt file.

2. In this file, **enter** the following property and given value:

```
net.indx.repository.solution.server.solutionserver.allowapiclients
=true
```

3. **Save** and **close** the file.

Setting the default XHQ host server for connecting the remote/distributed XhqWse web service

To do this, set the **`DefaultXhqServerHost`** property in the Web Services **`web.config` file**. By default (after the XHQ installation), the value of this property is `localhost`.

```
<!-- The name of the XHQ Server host used to connect to when the host name provided in
the web service client/invoker is "NULL" or an empty string
-->  <add key="DefaultXhqServerHost" value="localhost"/>
```

You must manually change this value in the `web.config` file to match the name of the XHQ Server you want the remote/distributed XhqWse Web Service to connect to when NULL is specified in the Web Service connection string call.

The `web.config` file is located in the **`%XHQ_WEB_HOME%\webapps\xhqwse`** directory (which by default is `<systemdrive>:\Program Files (x86)\XHQ\XHQ Web Root\webapps\xhqwse`).

Modifying the WAKA XML configuration file for the remote/distributed XhqWse web service

From the XML configuration file, change the XhqWse section so that it is similar to the following (highlighted text):

```
<Site name="XhqWse">
    <Settings>
```

```
      <Setting name="FastInterval" value="2" />
      <Setting name="Interval" value="10" />
      <Setting name="Debug" value="true" />
    </Settings>
    <Paths>
      <Path name="XhqWse Service [%WS_HOST_MACHINE_NAME%]" value="http://[%WS_HOST_
MACHINE_NAME%]/indx/xhqwse/Default.aspx" />
    </Paths>
  </Site>
```

Note in the example above that localhost is replaced with the name(s) of the machine hosting the stand-alone XhqWse.

If there is an XhqWse Web Service that is local to the XHQ Server and there are multiple Web Services server machines that need to be "warmed up" because they are connecting to the same XHQ Server, then change the XhqWse section to the following.

```
<Site name="XhqWse">
    <Settings>
      <Setting name="FastInterval" value="2" />
      <Setting name="Interval" value="10" />
      <Setting name="Debug" value="true" />
    </Settings>
    <Paths>
      <Path name="XhqWse Service" value="http://localhost/indx/xhqwse/Default.aspx" />
      <Path name="XhqWse Service [%WS_HOST_MACHINE_NAME_1%]" value="http://[%WS_HOST_
MACHINE_NAME_1%]/indx/xhqwse/Default.aspx" />
...
     <Path name="XhqWse Service [%WS_HOST_MACHINE_NAME_n%]" value="http://[%WS_HOST_
MACHINE_NAME_n%]/indx/xhqwse/Default.aspx" />
    </Paths>
  </Site>
```

The above example is for "n" number of XhqWse Web Services.

If SSL (HTTPS) is used to access the remote Web Service, then the XhqWse section is similar to the following (again, note the highlighted text):

```
<Site name="XhqWse">
    <Settings>
      <Setting name="FastInterval" value="2" />
      <Setting name="Interval" value="10" />
      <Setting name="Debug" value="true" />
    </Settings>
    <Paths>
      <Path name="XhqWse Service"
value="https://localhost/indx/xhqwse/Default.aspx?useSSL=true"/>
      <Path name="XhqWse Service [%WS_HOST_MACHINE_NAME_1%]" value="https://[%WS_HOST_
MACHINE_NAME_1%]/indx/xhqwse/Default.aspx?useSSL=true"/>
...
     <Path name="XhqWse Service [%WS_HOST_MACHINE_NAME_n%]" value="https://[%WS_HOST_
MACHINE_NAME_n%]/indx/xhqwse/Default.aspx?useSSL=true"/>
    </Paths>
  </Site>
```

Setting the Default Time Zone

The `web.config` file includes a property that allows you to set the default time zone.

```
<appSettings>
<!-- The Time Zone used when a "NULL" or empty string timezone is specified as the input
parameter for the web services operations that take the timezone as an input parameter.
-->
<add key="DefaultTimeZone" value="UTC"/>
</appSettings>
```

The `web.config` file is located in the **`%XHQ_WEB_HOME%\webapps\xhqwse`** directory (which by default is `<systemdrive>:\Program Files (x86)\XHQ\XHQ Web Root\webapps\xhqwse`).

By default, this property is disabled (commented out) in the `web.config` file. To enable, you must open `web.config`, locate the property, uncomment the property, change the value, and save the file.

If a NULL or empty time zone is specified as the input parameter for any XHQ Web Services operation that actually has a time zone input parameter, then one of two things will occur:

- If the `DefaultTimeZone` property exists and has a valid value, then the Web Service uses the DefaultTimeZone time zone.
- If the `DefaultTimeZone` property does not exist (or it exists but does not have a valid value), then the Web Service uses the XHQ Server time zone (that is, the time zone of the XHQ Server to which the Web Service is connected by default). This will either be the XHQ Application Server (Web Services-only installation) or the XHQ Server (where the Solution Server is running).

Additional Configurable Properties

The XhqWse web service also uses the following `web.config` properties. Because these properties are *optional*, they can be omitted from the `web.config` file; in which case, the default values are used.

The `web.config` file is located in the **`%XHQ_WEB_HOME%\webapps\xhqwse`** directory (which by default is `<systemdrive>:\Program Files (x86)\XHQ\XHQ Web Root\webapps\xhqwse`).

For SubscriptionWaitTime:

```
<appSettings>
<!-- The number of milliseconds to wait for a subscribed value with quality other
than "BAD, Not Initialized" for Current Server Values (single or bulk) calls.
-->
<add key="SubscriptionWaitTime " value="200"/>
</appSettings>
```

The default value for the `SubscriptionWaitTime` property is 200 milliseconds.

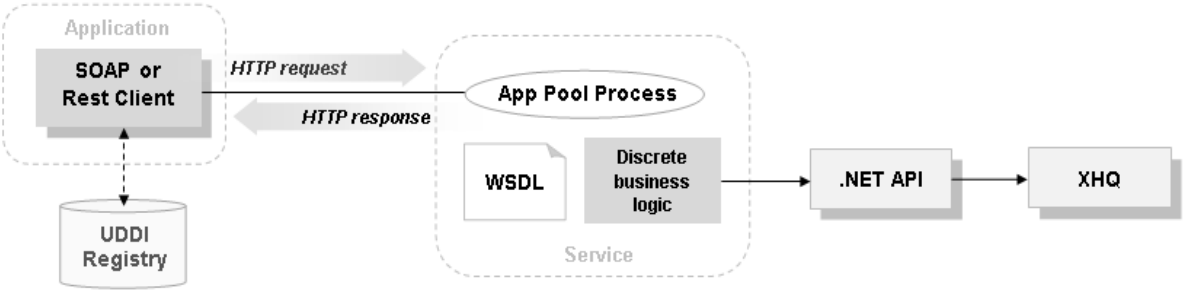
For KeepSubscribedTime:

```
<appSettings>
<!-- The number of milliseconds to keep the member subscribed for Current Server Values
(single or bulk) calls.
-->
<add key="KeepSubscribedTime" value="0"/>
</appSettings>
```

The default value for the `KeepSubscribedTime` property is 0 milliseconds.

Web Services Architecture

The following diagram shows how a client application invokes the XHQ Web Service.



The client invokes the Web Service through a proxy class. The SOAP request message is built by serializing the client code (C# or Java) into XML and is transmitted to the Web Service through HTTP. On the server side, the SOAP server/processor will de-serialize the XML from the SOAP message request into C# and execute the invoked C# method on the server. The corresponding C# method uses the XHQ ADO.NET Data Provider API to actually query the XHQ Solution Server for real-time data or metadata requested by the service client/consumer. Then, the Web Service serializes the response back into XML and using HTTP sends the SOAP responds message to the client. The client de-serializes the XML from the SOAP response body into its code and uses accordingly the XHQ data provided by the Web Service.

The Web Service that was implemented to allow the query of XHQ realtime data and metadata is called **XhqWse** and is defined in the **Indx.Xhq.Wse** namespace**.** The operations or web methods for the Web Service rely on the XHQ ADO.NET Data Provider and the .NET Client API, which provides the actual functionality for querying XHQ. For this reason these Web Services is implemented in C#.

The SOAP server/processor used is provided by IIS with .NET Framework and, as such, IIS must be used.

Refer to the topic, *Requirements and Pre-requisites*, located in the XHQ Installation Guide, for the supported versions of the .NET Framework and IIS.

The Web Service is a Windows Communication Foundation (WCF) service, and uses the document style. And all operations are of the "in/out" type.

In addition, the XHQ Web Service has a test client/consumer web application called **TestXhqWse** that serves two purposes:

- It allows the complete testing of all the Web Service's operations.
- It provides examples and sample code that shows how to properly implement client consumers for the Web Service and invoke its web methods.

Using the External Web Service, XhqWse

The URL to access the XHQ external web service, XhqWse, is:

`http://<XHQ Web Server>/indx/xhqwse/XhqWse.svc`

Example: `http://localhost/indx/xhqwse/XhqWse.svc`

The WSDL URL for the WCF service is:

`http://<XHQ Weber Server>/indx/xhqwse/XhqWse.svc?wsdl`

Example: `http://corp.acme.com/indx/xhqwse/XhqWse.svc?wsdl`

The following is a list of the SOAP operations (web methods) for the **`XhqWse`** external Web Service.

XHQ also offers an administrative Web Service, XhqAdminWs, which is used to invoke a majority of xhqci commands over HTTP.

For detailed information on each XhqWse web method, refer to the topic, *XhqWse Web Service Methods*.

XhqWse Web Methods QuickList

Web Method	Description
`DeleteCollectionData`	Performs a delete operation on a collection dataset.
`GenerateSolutionTreeNode`	Generates one level of the XHQ Solution tree (for a given node defined by an absolute path) as XML.
`GetAllChildMembers`	Gets all the child members' names for a component specified by a path.
`GetBulkActualValues`	Gets the VTQ data for a group/bulk of points/primitives at a specified moment in time (ACTUAL) as a VTQ array.
`GetBulkAtValues`	Gets the VTQ data for a group/bulk of points/primitives at a specified moment in time (AT) as a VTQ array.
`GetBulkAverageValues`	Gets the VTQ average data for a group/bulk of points/primitives for a given time range, as a VTQ array.
`GetBulkCountValues`	Gets the VTQ count data for a group/bulk of points/primitives for a given time range, as a VTQ array.
`GetBulkCurrentValues`	Gets the current VTQ data for a group/bulk of points/primitives as a VTQ array.
`GetBulkDeltaValues`	Gets the VTQ delta data for a group/bulk of points/primitives for a given time range, as a VTQ array.
`GetBulkFirstValues`	Gets the VTQ first value data for a group/bulk of points/primitives for a given time range, as a VTQ array.
`GetBulkIntegralValues`	Gets the VTQ integral data for a group/bulk of points/primitives for a given time range, as a VTQ array.

Web Method	Description
GetBulkLastValues	Gets the VTQ last value data for a group/bulk of points/primitives for a given time range, as a VTQ array.
GetBulkMaximumValues	Gets the VTQ maximum data for a group/bulk of points/primitives for a given time range, as a VTQ array.
GetBulkMinimumValues	Gets the VTQ minimum data for a group/bulk of points/primitives for a given time range, as a VTQ array.
GetBulkSumValues	Gets the VTQ sum data for a group/bulk of points/primitives for a given time range, as a VTQ array.
GetChildCollectionMembers	Gets the child collection members' names for a component specified by a path.
GetChildComponentMembers	Gets the child component/object members' names for a component specified by a path.
GetChildPrimitiveMembers	Gets the child primitive members' names for a component specified by a path.
GetCollectionColumnAverage	Gets the average value for a certain column.
GetCollectionColumnCount	Gets the count value for a certain column.
GetCollectionColumnMaximum	Gets the maximum value for a certain column.
GetCollectionColumnMinimum	Gets the minimum value for a certain column.
GetCollectionColumnSum	Gets the sum value for a certain column.
GetCollectionCount	Gets the collection records count.
GetCollectionData	Gets the collection dataset result for a collection query.
GetCollectionDataFilter	Gets the collection dataset result for a collection query having an ORDER BY and a WHERE clause.
GetCollectionDataOrderBy	Gets the collection dataset result for a collection query having only a ORDER BY clause.
GetCollectionDataWhere	Gets the collection dataset result for a collection query having only a WHERE clause.
GetCollectionMembers	Gets the member names for a collection specified by a path.
GetGlobalCollections	Gets all the XHQ global collections names.
GetPointActualValue	Gets the VTQ data for a single point/primitive at a specified moment in time (ACTUAL) as a String array with three elements.
GetPointAtValue	Gets the VTQ data for a single point/primitive at a specified moment in time (AT) as a String array with three elements.
GetPointAverageValue	Gets the VTQ average data for a single point/primitive calculated for a time range as a String array with three elements.
GetPointCountValue	Gets the VTQ count data for a single point/primitive calculated for a time range as a String array with three elements.

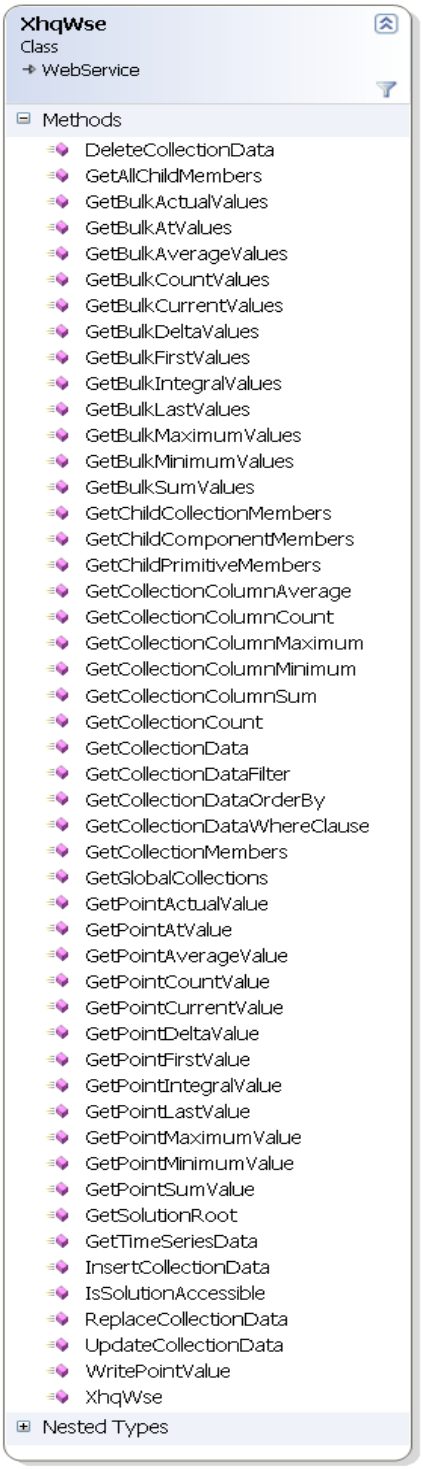
Web Method	Description
`GetPointCurrentData`	Gets the current VTQ data for a single point/primitive as a string array with three elements.
`GetPointDeltaValue`	Gets the VTQ point data for a single point/primitive calculated for a time range as a String array with three elements.
`GetPointFirstValue`	Gets the VTQ first data for a single point/primitive calculated for a time range as a String array with three elements.
`GetPointIntegralValue`	Gets the VTQ integral data for a single point/primitive calculated for a time range as a String array with three elements.
`GetPointLastValue`	Gets the VTQ last data for a single point/primitive calculated for a time range as a String array with three elements.
`GetPointMaximumValue`	Gets the VTQ maximum data for a single point/primitive calculated for a time range as a String array with three elements.
`GetPointMinimumValue`	Gets the VTQ minimum data for a single point/primitive calculated for a time range as a String array with three elements.
`GetPointSumValue`	Gets the VTQ sum data for a single point/primitive calculated for a time range as a String array with three elements.
`GetSolutionRoot`	Gets the XHQ solution root name.
`GetTimeSeriesData`	Gets the time series data for a primitive specified through a path or alias as a String array of triplets Value Time Quality (VTQ) structure from the time range specified with the given characteristics.
`InsertCollectionData`	Performs an insert operation on a collection dataset.
`IsSolutionAccessible`	Checks if a connection to the XHQ Solution Server can be established.
`ReplaceCollectionData`	Performs a replace operation on a collection dataset.
`UpdateCollectionData`	Performs an update operation on a collection dataset.
`WritePointValue`	Updates the value in the XHQ Solution Server for the member specified in the request.
`WriteValuesToBackend`	Writes values directly to backend (for example, a PI historian) for the members specified in the request.
For Access Validation:	
`CanReadMember`	Determines if the given global path or alias can be read for the current user.
`CanWriteMember`	Determines if the given global path or alias can be written for the current user.
`GetPathListAccess`	Retrieves the read and write access permissions for an array of XHQ points (specified by paths or aliases/tags), for the specified user.
`GetReadableMembers`	Retrieves the list of readable member names for the given path for the current user.

Web Method	Description
`GetRolesProvidingReadAccess`	Retrieves the list of XHQ Roles names that allow the specified user read access to the given path.
`GetRolesProvidingWriteAccess`	Retrieves the list of XHQ Roles names that allow the specified user write access to the given path.
`GetRolesWithReadAccess`	Retrieves the XHQ Roles names that have read access to the given path.
`GetRolesWithWriteAccess`	Retrieves the XHQ Roles names that have write access to the given path.
`GetUserRoleList`	Returns the list of XHQ Roles names for the specified user.
`GetWriteableMembers`	Retrieves the list of writeable member names for the given path, for the current user.
`IsUserInRole`	Flag that indicates if the current user belongs or not to the specified role.

For Xhq.Wse methods related to importing, refer to the topic, *Importing the XHQ Model*.

Web Service Operations

The following image shows the class diagram for the **XhqWse** Web Service:



XhqWse Web Service Class

Class Details	Descriptions
Namespace	**`Indx.Xhq.Wse`**

Class	**`XhqWse`**
	This class is a Web Service that provides operations for the XHQ real-time data and metadata.
	`System.Object`
	`  System.ComponentModel.MarshalByValueComponent`
	`  System.Web.Services.WebService`
	`  Indx.Xhq.Ws.XhqWse`

public class XhqWse	WebService
Thread Safety	Public static (**Shared** in Visual Basic) members of this type are safe for multithreaded operations. Instance members are **not** guaranteed to be thread-safe.
Remarks	Details about the class XhqWse.

XhqWse Web Service Methods

About VTQ

All XHQ Web Services methods that have a VTQ, or VTQ[] array, return (output) type consists of value, time, and quality, as well as the tag/point units and description. These methods have an input parameter named "units."

If the "units" input parameters is not NULL or empty, then the output value is converted to the given units of measure. Otherwise, no unit conversion is performed for the returned value. If the unit requested does not exist or is not in the same category as the output units, then the VTQ returns with a value of "NaN" (undefined).

Method		Description

`XhqWse.GetPointCurrentData`		Gets the current VTQ data for a single point/primitive as a String array with these elements.
		`public VTQ GetPointCurrentData(`
		`  string solutionHost,`
		`  string path,`
		`  string units,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Relative or absolute path (to a component or to a link collection) or tag name.
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (**::**) indicates an absolute path. • A path beginning with a period (**.**) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method	Description
`XhqWse.GetPointAtValue`	Gets the VTQ data for a single point/primitive at a specified moment in time (AT) as a String array with these elements.

Method		Description

		`public VTQ GetPointAtValue(`
		`  string solutionHost,`
		`  string path,`
		`  string atTime,`
		`  string units,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Relative or absolute path (to a component or to a link collection) or tag name.
	atTime	The time for which the value is requested. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetPointActualValue`		Gets the VTQ data for a single point/primitive at a specified moment in time (ACTUAL) as a String array with these elements.
		`public VTQ GetPointActualValue(`
		`  string solutionHost,`
		`  string path,`
		`  string actualTime,`
		`  string units,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Relative or absolute path (to a component or to a link collection) or tag name.
	actualTime	The time for which the value is requested. String format: yyyy-MM-dd

Method		Description
		HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetPointAverageValue`		Gets the VTQ average data for a single point/primitive calculated for a time range as a String array with these elements.
		`public VTQ GetPointAverageValue(`
		`  string solutionHost,`
		`  string path,`
		`  string startTime,`
		`  string endTime,`
		`  string units,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Relative or absolute path (to a component or to a link collection) or tag name.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified.

Method	Description
	This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value	The current VTQ structure for the point.
Remarks	The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetPointCountValue`		Gets the VTQ count data for a single point/primitive calculated for a time range as a String array with these elements.
		`public VTQ GetPointCountValue(`
		`  string solutionHost,`
		`  string path,`
		`  string startTime,`
		`  string endTime,`
		`  string units,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Relative or absolute path (to a component or to a link collection) or tag name.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location:

Method	Description
	HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value	The current VTQ structure for the point
Remarks	The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetPointDeltaValue`		Gets the VTQ delta data for a single point/primitive calculated for a time range as a String array with these elements.
		`public VTQ GetPointDeltaValue(`
		`  string solutionHost,`
		`  string path,`
		`  string startTime,`
		`  string endTime,`
		`  string units,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Relative or absolute path (to a component or to a link collection) or tag name.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone

Method	Description
	is used. The string value for server time is: "XHQ Solution Server Time".
Return Value	The current VTQ structure for the point.
Remarks	The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetPointFirstValue`		Gets the VTQ first data for a single point/primitive calculated for a time range as a String array with these elements.
		`public VTQ GetPointFirstValue(`
		`  string solutionHost,`
		`  string path,`
		`  string startTime,`
		`  string endTime,`
		`  string units,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Relative or absolute path (to a component or to a link collection) or tag name.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.

Method	Description
Remarks	The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetPointLastValue`		Gets the VTQ last data for a single point/primitive calculated for a time range as a String array with these elements.
		`public VTQ GetPointLastValue(`
		`  string solutionHost,`
		`  string path,`
		`  string startTime,`
		`  string endTime,`
		`  string units,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Relative or absolute path (to a component or to a link collection) or tag name.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path.

Method	Description
	• A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetPointIntegralValue`		Gets the VTQ integral data for a single point/primitive calculated for a time range as a String array with these elements.
		`public VTQ GetPointIntegralValue(`
		`  string solutionHost,`
		`  string path,`
		`  string startTime,`
		`  string endTime,`
		`  string units,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Relative or absolute path (to a component or to a link collection) or tag name.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss. [fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss. [fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

XhqWse.GetPointMaximumValue		Gets the VTQ maximum data for a single point/primitive calculated for a time range as a String array with these elements.
		`public VTQ GetPointMaximumValue(`
		`  string solutionHost,`
		`  string path,`
		`  string startTime,`
		`  string endTime,`
		`  string units,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Relative or absolute path (to a component or to a link collection) or tag name.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss. [fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss. [fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See XhqWse Web Service Methods for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.
Remarks		The following restrictions apply to the **path** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method	Description

XhqWse.GetPointMinimumValue	Gets the VTQ minimum data for a single point/primitive calculated for a time range as a String array with these elements.
	`public VTQ GetPointMinimumValue(`
	`  string solutionHost,`

Method		Description

		`string path,`
		`string startTime,`
		`string endTime,`
		`string units,`
		`string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Relative or absolute path (to a component or to a link collection) or tag name.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method	Description

`XhqWse.GetPointSumValue`	Gets the VTQ sum data for a single point/primitive calculated for a time range as a String array with these elements.
	`public VTQ GetPointSumValue(`
	`string solutionHost,`
	`string path,`
	`string startTime,`
	`string endTime,`
	`string units,`
	`string timeZone`
	`);`

Method		Description
Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Relative or absolute path (to a component or to a link collection) or tag name.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.WritePointValue`		Updates the value in the XHQ Solution Server for the member specified in the request.
		`public string WritePointValue(`
		`  string solutionHost,`
		`  string path,`
		`  string value,`
		`  string time,`
		`  string quality,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Relative or absolute path (to a component or to a link collection) or tag name.
	value	A String representing the actual update value in the VTQ set.

Method		Description
	time	A String representing the update time in the VTQ set. String format: `mm/dd/yy h:mmss AM/PM`
	quality	A String representing the update quality in the VTQ set.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		A String representing the status of the update request.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.WriteValueToBackend`		Writes values directly to backend (for example, a PI historian) for the members specified in the request. Paths must either be absolute (that is, fully qualified) or aliases (for example, tags). Absolute paths are prefixed with the double-colon (`::`) marker; otherwise, it is considered an alias.
		`public string[] WriteValueToBackend(`
		` string solutionHost,`
		` string[] pathList,`
		` VTQ[] vtqList,`
		` string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	pathList	A String array of valid XHQ paths.
	vtqList	A VTQ array.
	timeZone	The time zone in which the input time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used.
Return Value		A String array representing the status of the backend WRITE request.

Method	Description
	If the status is invalid, the code and the error are returned as strings in the array (with the format: "Code - Error").
Remarks	The following restrictions apply to the **`pathList`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetBulkCurrentValues`		Gets the current VTQ data for a group/bulk of points/primitives as a VTQ array.
		`public VTQ GetBulkCurrentValues(`
		`  string solutionHost,`
		`  VTQ paths,`
		`  string units,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	paths	A String array of relative or absolute paths, aliases or tag names.
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method	Description

`XhqWse.GetBulkAtValues`	Gets the VTQ data for a group/bulk of points/primitives at a specified moment in time (AT) as a VTQ array. Time format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM.
	`public VTQ GetBulkAtValues(`
	`  string solutionHost,`
	`  VTQ paths,`
	`  string atTime,`

Method		Description

		`string units,`
		`string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	paths	A String array of relative or absolute paths, aliases or tag names.
	atTime	The time for which the bulk values are requested. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See "*About VTQ*" for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetBulkActualValues`		Gets the VTQ data for a group/bulk of points/primitives at a specified moment in time (ACTUAL) as a VTQ array. Time format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM.
		`public VTQ GetBulkActualValues(`
		`string solutionHost,`
		`VTQ paths,`
		`string actualTime,`
		`string units,`
		`string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	paths	A String array of relative or absolute paths, aliases or tag names.
	actualTime	The time for which the bulk values are requested. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See "*About VTQ*" for details.

Method		Description
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetBulkAverageValues`		Gets the VTQ average data for a group/bulk of points/primitives for a given time range, as a VTQ array. Time format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM.
		`public VTQ GetBulkAverageValues(`
		`  string solutionHost,`
		`  VTQ paths,`
		`  string startTime,`
		`  string endTime,`
		`  string units,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	paths	A String array of relative or absolute paths, aliases or tag names.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used.

Method	Description
	The string value for server time is: "XHQ Solution Server Time".
Return Value	The current VTQ structure for the point.
Remarks	The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetBulkCountValues`		Gets the VTQ count data for a group/bulk of points/primitives for a given time range, as a VTQ array. Time format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM.
		`public VTQ GetBulkCountValues(`
		`  string solutionHost,`
		`  VTQ paths,`
		`  string startTime,`
		`  string endTime,`
		`  string units,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	paths	A String array of relative or absolute paths, aliases or tag names.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetBulkDeltaValues`		Gets the VTQ delta data for a group/bulk of points/primitives for a given time range, as a VTQ array. Time format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM.
		`public VTQ GetBulkDeltaValues(`
		`  string solutionHost,`
		`  VTQ paths,`
		`  string startTime,`
		`  string endTime,`
		`  string units,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	paths	A String array of relative or absolute paths, aliases or tag names.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method	Description

`XhqWse.GetBulkFirstValues`	Gets the VTQ first data for a group/bulk of points/primitives for a given time range, as a VTQ array. Time format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM.
	`public VTQ GetBulkFirstValues(`
	`  string solutionHost,`
	`  VTQ paths,`
	`  string startTime,`
	`  string endTime,`
	`  string units,`

Method		Description
		`string timeZone` `);`
Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	paths	A String array of relative or absolute paths, aliases or tag names.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetBulkLastValues`		Gets the VTQ last data for a group/bulk of points/primitives for a given time range, as a VTQ array. Time format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM
		`public VTQ GetBulkLastValues(`
		`string solutionHost,`
		`VTQ paths,`
		`string startTime,`
		`string endTime,`
		`string units,`
		`string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	paths	A String array of relative or absolute paths, aliases or tag names.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff]

Method		Description
		AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetBulkIntegralValues`		Gets the VTQ integral data for a group/bulk of points/primitives for a given time range, as a VTQ array. Time format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM.
		`public VTQ GetBulkIntegralValues(`
		`  string solutionHost,`
		`  VTQ paths,`
		`  string startTime,`
		`  string endTime,`
		`  string units,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	paths	A String array of relative or absolute paths, aliases or tag names.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location:

Method	Description
	HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value	The current VTQ structure for the point.
Remarks	The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetBulkMaximumValues`		Gets the VTQ maximum data for a group/bulk of points/primitives for a given time range, as a VTQ array. Time format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM.
		`public VTQ GetBulkMaximumValues(`
		`  string solutionHost,`
		`  VTQ paths,`
		`  string startTime,`
		`  string endTime,`
		`  string units,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	paths	A String array of relative or absolute paths, aliases or tag names.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.
Remarks		The following restrictions apply to the **`path`** parameter:

Method	Description
	• A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetBulkMinimumValues`		Gets the VTQ minimum data for a group/bulk of points/primitives for a given time range, as a VTQ array. Time format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM.
		`public VTQ GetBulkMinimumValues(`
		`  string solutionHost,`
		`  VTQ paths,`
		`  string startTime,`
		`  string endTime,`
		`  string units,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	paths	A String array of relative or absolute paths, aliases or tag names.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetBulkSumValues`		Gets the VTQ sum data for a group/bulk of points/primitives for a given time range, as a VTQ array. Time format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM.
		`public VTQ GetBulkSumValues(`
		`  string solutionHost,`
		`  VTQ paths,`
		`  string startTime,`
		`  string endTime,`
		`  string units,`
		`  string timeZone`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	paths	A String array of relative or absolute paths, aliases or tag names.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See *"About VTQ"* for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
Return Value		The current VTQ structure for the point.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method	Description

`XhqWse.GetTimeSeriesData`	Gets the Time Series data for a primitive specified through a path or alias as a String array of triplets Value Time Quality (VTQ) structure from the time range specified with the given characteristics.
	`public VTQ[] GetTimeSeriesData(`
	`  string solutionHost,`
	`  string path,`
	`  string startTime,`
	`  string endTime,`

Method		Description

		`string units,`
		`string timeZone,`
		`string samplingMode,`
		`int noPoints`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Relative or absolute path or tag name.
	startTime	The earliest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	endTime	The latest time in the series. String format: yyyy-MM-dd HH:mm:ss.[fffffffff] AM/PM (fractional seconds, which are supported to 9 places, are optional).
	units	The units of measure for the returned value. See XhqWse Web Service Methods for details.
	timeZone	The time zone in which both the input and output time is specified. This can be any valid Windows Time Zone on the machine (for example, UTC, Pacific Standard Time, and so forth). The full list of time zones can be found in the Windows registry at the following location: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones In case this parameter is NULL or empty, the Solution Server Time Zone is used. The string value for server time is: "XHQ Solution Server Time".
	samplingMode	The sampling mode for the series. It is a String that can have one of the following values: ACTUALS, ACTUALS_WITH_ADJACENT, FITS, TIMESLICED_MIN, TIMESLICED_MAX, TIMESLICED_AVG, TIMESLICED_INTERPOLATE. See Also: *Data Retrieval Modes for XHQ*
	noPoints	The number of points to return.
Return Value		A String array of VTQ structures.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path. • A path beginning with a period (.) indicates a path relative to this object. • All others are treated as member aliases or tag names.

Method	Description

`XhqWse.GetCollectionData`	Gets the Collection dataset result for a collection query
	`public string[][] GetCollectionData(`
	`string solutionHost,`
	`string path,`
	`string columns`
	`);`

Method		Description
Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	The path to a global or link collection.
	columns	A comma separated String of member names defining which members are mapped to columns. Supplying null, empty String or * results in all the columns being returned in "natural" order.
Return Value		A String array of array representing the collection data. The first String array represents the collection column and the second one represents the collection row data.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path to a link collection. • A path beginning with a period (.) indicates a path relative to a link collection. • All others are treated as global collections.

Method		Description

`XhqWse.GetCollectionDataFilter`		Gets the Collection dataset result for a collection query.
		`public string[][] GetCollectionDataFilter(`
		`  string solutionHost,`
		`  string path,`
		`  string columns,`
		`  string whereClause,`
		`  string orderByClause`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	The path to a global or link collection.
	columns	A comma separated String of member names defining which members are mapped to columns. Supplying null, empty String or * results in all the columns being returned in "natural" order.
	whereClause	A SQL WHERE clause that specifies the selection criteria; do not supply the key word WHERE at the beginning of the string.
	orderByClause	A SQL ORDER BY clause that specifies the selection ordering; do not supply the key word ORDER BY at the beginning of the string.
Return Value		A String array of array representing the collection data. The first String array represents the collection column and the second one represents the collection row data.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path to a link collection.

Method	Description
	• A path beginning with a period (.) indicates a path relative to a link collection. • All others are treated as global collections.

Method		Description

`XhqWse.GetCollectionDataOrderBy`		Gets the Collection dataset result for a collection query.
		`public string[][] GetCollectionDataOrderBy(`
		`  string solutionHost,`
		`  string path,`
		`  string columns,`
		`  string orderByClause`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	The path to a global or link collection.
	columns	A comma separated String of member names defining which members are mapped to columns. Supplying null, empty String or * results in all the columns being returned in "natural" order.
	orderByClause	A SQL ORDER BY clause that specifies the selection ordering; do not supply the key word ORDER BY at the beginning of the string.
Return Value		A String array of array representing the collection data. The first String array represents the collection column and the second one represents the collection row data.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path to a link collection. • A path beginning with a period (.) indicates a path relative to a link collection. • All others are treated as global collections.

Method		Description

`XhqWse.GetCollectionDataWhere`		Gets the Collection dataset result for a collection query.
		`public string[][] GetCollectionDataWhere(`
		`  string solutionHost,`
		`  string path,`
		`  string columns,`
		`  string whereClause`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	The path to a global or link collection.

Method		Description
	columns	A comma separated String of member names defining which members are mapped to columns. Supplying null, empty String or * results in all the columns being returned in "natural" order.
	whereClause	A SQL WHERE clause that specifies the selection criteria; do not supply the key word WHERE at the beginning of the string.
Return Value		A String array of array representing the collection data. The first String array represents the collection column and the second one represents the collection row data.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path to a link collection. • A path beginning with a period (.) indicates a path relative to a link collection. • All others are treated as global collections.

Method		Description

`XhqWse.GetCollectionCount`		Gets the collection records count.
		`public int GetCollectionCount(`
		`  string solutionHost,`
		`  string path,`
		`  string whereClause`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	The path to a global or link collection.
	whereClause	A SQL WHERE clause that specifies the selection criteria; supplying the key word WHERE at the beginning of the string is optional.
Return Value		An integer representing the number of records in the collection.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path to a link collection. • A path beginning with a period (.) indicates a path relative to a link collection. • All others are treated as global collections.

Method	Description

`XhqWse.GetCollectionColumnAverage`	Gets the average value for a certain column.
	`public string[] GetCollectionColumnAverage(`
	`  string solutionHost,`

Method		Description

		`string path,`
		`string column,`
		`string whereClause`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	The path to a global or link collection.
	column	The name of the column of the collection for which the aggregate is being computed.
	whereClause	A SQL WHERE clause that specifies the selection criteria; supplying the key word WHERE at the beginning of the string is optional.
Return Value		A String array that has three elements: the first one is the aggregate value for the column, the second is the timestamp and the third one is the quality of the collection aggregate.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path to a link collection. • A path beginning with a period (.) indicates a path relative to a link collection. • All others are treated as global collections.

Method		Description

`XhqWse.GetCollectionColumnCount`		Gets the count for a certain column.
		`public string[] GetCollectionColumnCount(`
		`string solutionHost,`
		`string path,`
		`string column,`
		`string whereClause`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	The path to a global or link collection.
	column	The name of the column of the collection for which the aggregate is being computed.
	whereClause	A SQL WHERE clause that specifies the selection criteria; supplying the key word WHERE at the beginning of the string is optional.
Return Value		A String array that has three elements: the first one is the aggregate value for the column, the second is the timestamp and the third one is the quality of the collection aggregate.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path to a link collection.

Method	Description
	• A path beginning with a period (.) indicates a path relative to a link collection. • All others are treated as global collections.

Method		Description

`XhqWse.GetCollectionColumnMaximum`		Gets the maximum value for a certain column.
		`public string[] GetCollectionColumnMaximum(`
		`  string solutionHost,`
		`  string path,`
		`  string column,`
		`  string whereClause`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	The path to a global or link collection.
	column	The name of the column of the collection for which the aggregate is being computed.
	whereClause	A SQL WHERE clause that specifies the selection criteria; supplying the key word WHERE at the beginning of the string is optional.
Return Value		A String array that has three elements: the first one is the aggregate value for the column, the second is the timestamp and the third one is the quality of the collection aggregate.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path to a link collection. • A path beginning with a period (.) indicates a path relative to a link collection. • All others are treated as global collections.

Method		Description

`XhqWse.GetCollectionColumnMinimum`		Gets the Collection minimum value for a certain column.
		`public string[] GetCollectionColumnMinimum(`
		`  string solutionHost,`
		`  string path,`
		`  string column,`
		`  string whereClause`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	The path to a global or link collection.
	column	The name of the column of the collection for which the aggregate is

Method		Description
		being computed.
	whereClause	A SQL WHERE clause that specifies the selection criteria; supplying the key word WHERE at the beginning of the string is optional.
Return Value		A String array that has three elements: the first one is the aggregate value for the column, the second is the timestamp and the third one is the quality of the collection aggregate.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path to a link collection. • A path beginning with a period (.) indicates a path relative to a link collection. • All others are treated as global collections.

Method		Description

`XhqWse.GetCollectionColumnSum`		Gets the sum value for a certain column.
		`public string[] GetCollectionColumnSum(`
		`  string solutionHost,`
		`  string path,`
		`  string column,`
		`  string whereClause`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	The path to a global or link collection.
	column	The name of the column of the collection for which the aggregate is being computed.
	whereClause	A SQL WHERE clause that specifies the selection criteria; supplying the key word WHERE at the beginning of the string is optional.
Return Value		A String array that has three elements: the first one is the aggregate value for the column, the second is the timestamp and the third one is the quality of the collection aggregate.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path to a link collection. • A path beginning with a period (.) indicates a path relative to a link collection. • All others are treated as global collections.

Method		Description

XhqWse.InsertCollectionData		Performs an insert operation on a collection dataset.
		`public bool InsertCollectionData(`
		`  string solutionHost,`
		`  string collection,`
		`  string[] columns,`
		`  string[][] data`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	collection	The global collection name.
	columns	An array of member names defining which members are mapped to columns.
	data	A String array of array containing the update/insert collection data.
Return Value		Returns true if the insert operation has been successful. Otherwise returns false.
Remarks		N/A

Method		Description

XhqWse.ReplaceCollectionData		Performs an replace operation on a collection dataset.
		`public bool ReplaceCollectionData(`
		`  string solutionHost,`
		`  string collection,`
		`  string[] columns,`
		`  string[][] data`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	collection	The global collection name.
	columns	An array of member names defining which members are mapped to columns.
	data	A String array of array containing the update/insert collection data.
Return Value		Returns true if the replace operation has been successful. Otherwise returns false.
Remarks		N/A

Method	Description

XhqWse.UpdateCollectionData	Performs an update operation on a collection dataset.
	`public bool UpdateCollectionData(`
	`  string solutionHost,`
	`  string collection,`
	`  string[] columns,`

Method		Description
		`string[][] data` `);`
Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	collection	The global collection name.
	columns	An array of member names defining which members are mapped to columns.
	data	A String array of array containing the update/insert collection data.
Return Value		Returns true if update operation has been successful. Otherwise returns false.
Remarks		N/A

Method		Description

`XhqWse.DeleteCollectionData`		Performs a delete operation on a collection dataset.
		`public bool DeleteCollectionData(`
		`string solutionHost,`
		`string collection,`
		`string[] columns,`
		`string[][] data`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	collection	The global collection name.
	columns	An array of member names defining which members are mapped to columns.
	data	A String array of array containing the update/insert collection data.
Return Value		Returns true if the delete operation has been successful. Otherwise returns false.
Remarks		N/A

Method		Description

`XhqWse.GetSolutionRoot`		Gets the XHQ solution root name.
		`public string GetSolutionRoot(`
		`string solutionHost`
		`);`

Parameter:	*solutionHost*	The name of the machine that hosts the XHQ solution.
Return Value		A String representing the name of the solution root.
Remarks		N/A

Method		Description

`XhqWse.GetGlobalCollections`		Gets all the XHQ global collections names.
		`public string[] GetGlobalCollections(`
		`  string solutionHost`
		`);`

Parameter:	*solutionHost*	The name of the machine that hosts the XHQ solution.
Return Value		An array of type String representing the names of the global collections.
Remarks		N/A

Method		Description

`XhqWse.GetAllChildMembers`		Gets all the child members' names for a component specified by a path.
		`public string[] GetAllChildMembers(`
		`  string solutionHost,`
		`  string parentPath,`
		`  Boolean showHidden`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	parentPath	The path to a component.
	showHidden	A flag that indicates whether or not to return the members marked as "hidden" in the XHQ Workbench.
Return Value		A String array representing the object members names.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path to a link collection. • A path beginning with a period (.) indicates a path relative to a link collection.

Method		Description

`XhqWse.GetChildPrimitiveMembers`		Gets the child primitive members' names for a component specified by a path.
		`public string[] GetChildPrimitiveMembers(`
		`  string solutionHost,`
		`  string parentPath,`
		`  Boolean showHidden`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	parentPath	The path to a component.
	showHidden	A flag that indicates whether or not to return the members marked as "hidden" in the XHQ Workbench.

Method		Description
Return Value		A String array representing the object members names.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path to a link collection. • A path beginning with a period (.) indicates a path relative to a link collection.

Method		Description

`XhqWse.GetChildComponentMembers`		Gets the child component/object members' names for a component specified by a path.
		`public string[] GetChildComponentMembers(`
		`  string solutionHost,`
		`  string parentPath,`
		`  Boolean showHidden`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	parentPath	The path to a component.
	showHidden	A flag that indicates whether or not to return the members marked as "hidden" in the XHQ Workbench.
Return Value		A String array representing the object members names.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path to a link collection. • A path beginning with a period (.) indicates a path relative to a link collection.

Method		Description

`XhqWse.GetChildCollectionMembers`		Gets the child collection members' names for a component specified by a path.
		`public string[] GetChildCollectionMembers(`
		`  string solutionHost,`
		`  string parentPath,`
		`  Boolean showHidden`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	parentPath	The path to a component.
	showHidden	A flag that indicates whether or not to return the members marked as "hidden" in the XHQ Workbench.

Method	Description
Return Value	A String array representing the object members names.
Remarks	The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path to a link collection. • A path beginning with a period (.) indicates a path relative to a link collection.

Method		Description

`XhqWse.GetCollectionMembers`		Gets the members' names for a collection specified by a path.
		`public string[] GetCollectionMembers(`
		`  string solutionHost,`
		`  string path`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	The path to a collection.
Return Value		A String array representing the object members names.
Remarks		The following restrictions apply to the **`path`** parameter: • A path beginning with a double-colon (::) indicates an absolute path to a link collection. • A path beginning with a period (.) indicates a path relative to a link collection. • All others are treated as global collections.

Method		Description

`XhqWse.IsSolutionAccessible`		Checks if a connection to the Solution Server can be established.
		`public string[] IsSolutionAccessible(`
		`  string solutionHost,`
		`  string path`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	The path to a collection.
Return Value		Returns TRUE if a connection to the Solution Server has been established and false if the connection cannot be obtained.
Remarks		N/A

Method		Description

`XhqWse.GenerateSolutionTreeNode`		Generates the complete XHQ Solution tree (for a given node defined by an absolute path) as XML.
		`public XmlDocument GenerateSolutionTreeNode(`
		`  string solutionHost,`
		`  string path`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	The absolute path to a component (node).
Return Value		The XHQ Solution tree node, as XML.
Remarks		N/A

Method		Description

`XhqWse.GetPathListAccess`		Retrieves the READ and WRITE access permissions for an array of XHQ points (specified by paths or aliases/tags), for the given user.
		Important: This operation can only be executed by a user with Solution Admin permission. If the username is NULL or empty, the READ/WRITE access is returned for the current user. Paths must be absolute (fully qualified) or aliases. Absolute paths begin with double colons ("::"); otherwise, it is considered an alias.
		`public PathAccess[] GetpathListAccess(`
		`  string solutionHost,`
		`  string[] paths,`
		`  string username`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	paths	A string array of absolute paths or aliases/tag names.
	username	The username for which the paths READ/WRITE access is returned. If NULL or empty, the READ and WRITE access is returned for the current user
Return Value		An array of **`PathAccess`** objects containing the path, the read, and the write access for that path. **Note:** `PathAccess` is a structure having the Path, ReadAccess, and WriteAccess public fields.
Remarks		The following restrictions apply to the `paths` parameter: • A path beginning with a double colon (`"::"`) indicates an absolute path. • All others are treated as member aliases or tag names.

Method	Description
`XhqWse.CanReadMember`	Determines if the given global path or alias can be read for the current user.

Method		Description

		Important: Paths must be absolute (fully qualified) or aliases. Absolute paths begin with double colons ("::"); otherwise, it is considered an alias.
		`public bool CanReadMember(`
		`  string solutionHost,`
		`  string path`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Absolute path or alias.
Return Value		Returns TRUE if the path can be read; otherwise, returns FALSE.
Remarks		The following restrictions apply to the `paths` parameter: • A path beginning with a double colon ("`::`") indicates an absolute path. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.CanWriteMember`		Determines if the given global path or alias can be written for the current user.
		Important: Paths must be absolute (fully qualified) or aliases. Absolute paths begin with double colons ("::"); otherwise, it is considered an alias.
		`public bool CanWriteMember(`
		`  string solutionHost,`
		`  string path`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Absolute path or alias.
Return Value		Returns TRUE if the path can be written; otherwise, returns FALSE.
Remarks		The following restrictions apply to the `paths` parameter: • A path beginning with a double colon ("`::`") indicates an absolute path. • All others are treated as member aliases or tag names.

Method	Description

`XhqWse.GetReadableMembers`	Retrieves the list of readable member names for the given path, for the current user.
	Important: Paths must be absolute (fully qualified) or aliases. Absolute paths begin with double colons ("::"); otherwise, it is considered an alias.
	`public string[] GetReadableMembers(`
	`  string solutionHost,`

Method		Description
		`string path` `);`
Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Absolute path or alias.
Return Value		String array of readable member names for the given path.
Remarks		The following restrictions apply to the `paths` parameter: • A path beginning with a double colon (`"::"`) indicates an absolute path. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetWriteableMembers`		Retrieves the list of writeable member names for the given path, for the current user.
		Important: Paths must be absolute (fully qualified) or aliases. Absolute paths begin with double colons ("::"); otherwise, it is considered an alias.
		`public string[] GetWriteableMembers(`
		`string solutionHost,`
		`string path`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Absolute path or alias.
Return Value		String array of writeable member names for the given path.
Remarks		The following restrictions apply to the `paths` parameter: • A path beginning with a double colon (`"::"`) indicates an absolute path. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetUserRoleList`		Returns the list of XHQ Roles names for the specified user.
		`public string[] GetUserRoleList(`
		`string solutionHost,`
		`string username`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	username	The username for which the XHQ roles names are returned. If NULL or empty, the READ and WRITE access is returned for the current user.
Return Value		An array of strings representing the XHQ Roles names.
Remarks		N/A

Method		Description

`XhqWse.IsUserInRole`		Flag that indicates if the current user belongs or not to the specified role.
		`public bool IsUserInRole(`
		`  string solutionHost,`
		`  string username,`
		`  string roleName`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	username	The username for which the flag is returned. If NULL or empty, the flag is returned for the current user.
	roleName	The name of the role.
Return Value		Returns TRUE if the specified user belongs to the role; otherwise, returns FALSE.
Remarks		N/A

Method		Description

`XhqWse.GetRolesWithReadAccess`		Retrieves the XHQ Roles names that have read access to the given path.
		`public string[] GetRolesWithReadAccess(`
		`  string solutionHost,`
		`  string path`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Absolute path or alias.
Return Value		An array of strings representing the XHQ Roles names that have read access to the given path.
Remarks		The following restrictions apply to the `path` parameter: • A path beginning with a double colon (`"::"`) indicates an absolute path. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetRolesWithWriteAccess`		Retrieves the XHQ Roles names that have write access to the given path.
		`public string[] GetRolesWithWriteAccess(`
		`  string solutionHost,`
		`  string path`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	path	Absolute path or alias.

Method		Description
Return Value		An array of strings representing the XHQ Roles names that have write access to the given path.
Remarks		The following restrictions apply to the `path` parameter: • A path beginning with a double colon (`"::"`) indicates an absolute path. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetRolesProvidingReadAccess`		Retrieves the list of XHQ Roles names that allow the specified user read access to the given path.
		`public string[] GetRolesProvidingReadAccess(`
		`  string solutionHost,`
		`  string username,`
		`  string path`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	username	The username for which XHQ Roles names are returned. If NULL or empty, the roles are returned for the current user.
	path	Absolute path or alias.
Return Value		An array of strings representing the XHQ Roles names that have read access to the given path, for the specified user.
Remarks		The following restrictions apply to the `path` parameter: • A path beginning with a double colon (`"::"`) indicates an absolute path. • All others are treated as member aliases or tag names.

Method		Description

`XhqWse.GetRolesProvidingWriteAccess`		Retrieves the list of XHQ Roles names that allow the specified user write access to the given path.
		`public string[] GetRolesProvidingWriteAccess(`
		`  string solutionHost,`
		`  string username,`
		`  string path`
		`);`

Parameters:	*solutionHost*	The name of the machine that hosts the XHQ solution.
	username	The username for which XHQ Roles names are returned. If NULL or empty, the roles are returned for the current user.
	path	Absolute path or alias.

Method	Description
Return Value	An array of strings representing the XHQ Roles names that have write access to the given path, for the specified user.
Remarks	The following restrictions apply to the `path` parameter: • A path beginning with a double colon (`"::"`) indicates an absolute path. • All others are treated as member aliases or tag names.

Importing the XHQ Model

The XHQ Web Services includes three web methods that enable you to invoke the `xhqci` command **`importmodel`** asynchronously through XHQ Web Services. They are:

- `ImportModel`
- `ImportModelWithCredentials`
- `GetXhqciCommandOutput`

The asynchronous import in the Web Service involves two steps:

1. Uploading the import ZIP file and asynchronously starting the Model import process by invoking the `ImportModel()` or `ImportModelWithCredentials()` operations. This returns a unique key to the client.
2. Periodically invoking the `GetXhqciCommandOutput()` operation in the client application. The client then uses the unique key to periodically check the output status of the import model process.

Important Things to Note

- Both the uploaded ZIP import file and the import model operation output are temporarily written (stored) and read to-and-from the hard disk. Therefore, the IIS application pool identity owner, under which the web service runs, must be granted Read/Write/Create permissions for the `%XHQ_SERVER_REPOS%` directory. The `repos` directory includes the `temp` subfolder to which the ZIP and output files are stored.
- The XHQ Web Service application must be installed on IIS on the same machine that runs XHQ Server, because it uses the `xhqci_s` batch file locally to run the import command.
- No encryption is needed if the web service and the client are both in the intranet (LAN or WAN), or if SSL is used.
- When encryption is needed, however, the client must use the encryption provided by the Base64 encoding mechanism, as shown in *EXAMPLE: .NET Client Invoking the Web Service*. This is the only encryption type for passwords that is currently supported and decrypted by the XHQ Web Service.
- When uploading large ZIP import files onto the server using the web service, you must modify the size of the HTTP request (which has a default maximum size of 4 MB) in the `web.config` file. See the following example.

EXAMPLE: MODIFYING THE WEB.CONFIG FILE FOR LARGE ZIP IMPORT FILES

Given: A 20 MB .ZIP needs to be uploaded and imported.

Solution: The `web.config` file is located in the **`%XHQ_WEB_HOME%\webapps\xhqwse`** directory (which by default is `<systemdrive>:\Program Files (x86)\XHQ\XHQ Web Root\webapps\xhqwse`).

Modify the `web.config` file on both the web service client and the web service server by changing the following element:

```
<system.web>
     <httpRuntime maxRequestLength="20480 "/>
</system.web>
```

Web Service Methods for Importing

The **`XhqWse`** service includes the following methods.

ImportModel

Method		Description

`XhqWse.ImportModel`		Imports the ZIP file contents into the XHQ Workbench (model and views).
		`public string[] ImportModel(`
		`  byte fileStream,`
		`  string modelServerHost,`
		`  bool importViews,`
		`  bool importTemplates`
		`);`

Parameters:	*fileStream*	The ZIP file to be imported, as a byte array.
	modelServerHost	The name of the machine that hosts the XHQ Model. If NULL, the import is performed on the same XHQ Model that is hosted on the machine that hosts this web service.
	importViews	Boolean flag that specifies whether or not the views are also imported.
	importTemplates	Boolean flag that specifies whether or not the templates are also imported.
Return Value		A unique key, as a string, that is used to invoke the `GetXhqciCommandOutput` operation, which checks the import process output and status/
Remarks		N/A

ImportModelWithCredentials

Method		Description

`XhqWse.ImportModelWithCredentials`		Imports the ZIP file contents into the XHQ Workbench (model and views), using an encrypted password.
		`public string[] ImportModelWithCredentials(`
		`  byte fileStream,`
		`  string modelServerHost,`
		`  bool importViews,`
		`  bool importTemplates,`
		`  bool encryptedPassword,`
		`  string userName,`
		`  string password`
		`);`

Parameters:	*fileStream*	The ZIP file to be imported, as a byte array.
	modelServerHost	The name of the machine that hosts the XHQ Model. If NULL, the import is performed on the same XHQ Model that is hosted on the machine that hosts this web service.
	importViews	Boolean flag that specifies whether or not the views are also imported.
	importTemplates	Boolean flag that specifies whether or not the templates are also

Method		Description
		imported.
	encryptedPassword	Boolean flag that specifies whether the password is sent in clear text or is encrypted.
	userName	The username under which the `xhqci` command is run.
	password	The password under which the `xhqci` command is run.
Return Value		A unique key, as a string, that is used to invoke the `GetXhqciCommandOutput` operation, which checks the import process output and status.
Remarks		N/A

GetXhqciCommandOutput

Method		Description

`XhqWse.GetXhqciCommandOutput`		Asynchronously receives the import model command output/status text. The key returned by the `ImportModel()` or the `ImportModelWithCredentials`() operation is used to invoke this method periodically to check the status of the import.
		`public string[] GetXhqciCommandOutput(`
		`  string key,`
		`  int lineOffset`
		`);`

Parameters:	*key*	The unique key used by the web service to retrieve the xhqci command output.
	lineOffset	The starting line (from zero) from which the command output is read and returned to the client.
Return Value		The xhqci command output as an array of strings, with each string representing a new line in the output message. It returns NULL when there are no more lines to read from the process output (EOF).
Remarks		N/A

WSDL Definition, Input and Output Parameters

The WSDL definition and input and output parameters for the three methods are as follows:

```
<s:element name="ImportModel">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="fileStream" type="s:base64Binary" />
      <s:element minOccurs="0" maxOccurs="1" name="modelServerHost" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="importViews" type="s:boolean" />
      <s:element minOccurs="1" maxOccurs="1" name="importTemplates" type="s:boolean" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="ImportModelResponse">
```

```
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="ImportModelResult" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="ImportModelWithCredentials">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="fileStream" type="s:base64Binary" />
      <s:element minOccurs="0" maxOccurs="1" name="modelServerHost" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="importViews" type="s:boolean" />
      <s:element minOccurs="1" maxOccurs="1" name="importTemplates" type="s:boolean" />
      <s:element minOccurs="1" maxOccurs="1" name="encrytptedPassword" type="s:boolean"
/>
      <s:element minOccurs="0" maxOccurs="1" name="userName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="password" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="ImportModelWithCredentialsResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="ImportModelWithCredentialsResult"
type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetXhqciCommandOutput">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="key" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="lineOffset" type="s:int" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetXhqciCommandOutputResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetXhqciCommandOutputResult"
type="tns:ArrayOfString" />
    </s:sequence>
  </s:complexType>
</s:element>
```

Example: .NET Client Invoking the Web Service

Given: **`FileUpload1`** is a .NET Framework File Upload control used in an `.aspx` page to specify the input path for the import ZIP file on the client.

Solution: In the following example, the output of the `xhqci` command output and status is displayed in a different ASP.NET web page named `CommandOutput.aspx`.

```
protected void btnImport_Click(object sender, EventArgs e)
{
  try
  {
    Stream x = FileUpload1.FileContent;
    int fileLen = FileUpload1.PostedFile.ContentLength;
```

```
    Byte[] buffer = new Byte[x.Length];
    x.Read(buffer, 0, fileLen);
    XhqWse wsClient = new XhqWse();
    wsClient.PreAuthenticate = true;
    wsClient.Credentials = System.Net.CredentialCache.DefaultCredentials;

    //String[] list = wsClient.ImportModel(buffer, "grinds", true, true);
    String[] list = wsClient.ImportModelWithCredentials(buffer,
"grinds", true, true, true, "username", Encrypt("pwd"));

      Session["WsClient"] = wsClient;
      Session["OutputKey"] = key;
      Session["lineOffset"] = "0";
      Session["status"] = outputStatus;

      Response.Redirect("CommandOutput.aspx");
  }
  catch (Exception ex)
  {
     lblResult.Text = "WS Error: " + ex.Message;
  }
}
```

If **encryption** is needed, the following method can be used to encrypt the password, using UTF-8 string and Base64 encode.

```
//method to encrypt a string
private string Encrypt(string strPwd)
{
  try
  {
    byte[] bytEncData = new byte[strPwd.Length];
    bytEncData = System.Text.Encoding.UTF8.GetBytes(strPwd);
    string strEncodedData = Convert.ToBase64String(bytEncData);
    return strEncodedData;
  }
  catch (Exception ex)
  {
    throw new Exception("Error in base64Encode" + ex.Message);
  }
}
```

The `CommandOutput.aspx` web page invokes the web service `GetXhqciCommandOutput()` method periodically (for example, using the `window.setInterval()` JavaScript function). It then renders the `xhqci` command output in the web page asynchronously, using a Label control.

```
//periodically check the status of the xhqci command output by polling
private void showXhqciCommandOutput()
{
  XhqWse wsClient = (WsTestClient.xhqWse.XhqWse)Session["WsClient"];
  String key = (String)Session["OutputKey"];
  String lo = (String)Session["lineOffset"];
  ArrayList outputStatus = (ArrayList)Session["status"];

  int lineOffset = Convert.ToInt32(lo);

  if (wsClient != null && key != null)
  {
    String[] outStr = wsClient.GetXhqciCommandOutput(key, lineOffset);
```

```
    if (outStr != null)
    {
      foreach (String line in outStr)
      {
        outputStatus.Add(line);
      }
      String s = "";
      foreach (object line in outputStatus)
      {
        s = s + (string)line + "<br>";
      }
      output.Text = s;

      Session["lineOffset"] = "" + outputStatus.Count;
      Session["status"] = outputStatus;
    }
  }
}
```

Using the Administrative Web Service, XhqAdminWs

This WCF Web Service enables you to invoke a majority of xhqci commands over HTTP. Invoking `xhqci` commands over the web involves a design and solution based on an **asynchronous**, two-step approach in the Web Service. This allows for better performance and easier maintenance than the synchronous approach. The `XhqAdminWs` Web Service involves the following steps.

Step 1 Invoke an `XhqAdminWs` operation. This asynchronously starts the xhqci command process and returns a **unique key**.

Step 2 Use the unique key to periodically check the output and status of the process initiated in Step 1. This is done by periodically invoking the `GetXhqciCommandOutput()` operation in the client application.

Step 3 *This step is OPTIONAL*. If you invoke an export-type xhqci command that produces an XML or ZIP file (such as `exportmodel`, `exportsoln`, and so forth), then use the same unique key from Step 1 to invoke the `GetExportFileStream(string key)` method to return the file stream as a byte array. Do this after the `GetXhqciCommandOutput()` from Step 2 returns a NULL, which signals the completion of the export xhqci operation. The exported temporary file is permanently deleted (implicit deletion) from the location on the machine that hosts the web service.

Important Things to Note

- For `xhqci` import-type operations, the uploaded ZIP or XML files that are to be imported, as well as the `xhqci` command output, needs to be temporarily written (stored) and read to and from the hard disk. So, the IIS Application Pool identity owner, under which the web service runs, needs to be granted adequate permissions (read/write/create) for the `%XHQ_REPOS%` directory. Additionally, the domain account under which the `XhqAdminWs` service Application Pool runs must be an XHQ Admin account and have the "Load User Profile" attribute set to TRUE in the IIS 7.0 Application Pool configuration.
- The `XhqAdminWs` service must be installed on the same machine that runs the XHQ Server, because it needs to use the `xhqeci` batch file locally to run the `xhqci` commands.
- Password encoding is not needed if the web service and the client are both in the intranet (LAN or WAN), or if SSL is used. If encoding is desired for passwords, the client must use the encoding provided by Base64 encoding mechanism (as shown in *EXAMPLE XhqAdminWs Web Service*). Note, this is the only encryption type for the password that is currently supported and can be decrypted by the XHQ Web Service.
- When uploading large import files (XML or ZIP format) on the server using this web service, you need to modify the size of the HTTP Request (default max size is 4 MB) in the `web.config` file. For example, in order to support uploading and importing of ZIP files up to 20 MB, modify the `web.config` file, located in the **`%XHQ_WEB_HOME%\webapps\xhqws`** directory (which by default is `<systemdrive>:\Program Files (x86)\XHQ\XHQ Web Root\webapps\xhqws`), on both the web service client and the web service server, like this:

```
<system.web>
        <httpRuntime maxRequestLength="20480"/>
</system.web>
```

- In order to provide a convenient way to test all the web service operations, a web application named **`TestXhqWse`** is available. It is deployed locally on the same machine with the `XhqAdminWs` and the XHQ Server. This test web application can be accessed from either of the following URLs.
  - From the **XHQ Server**, `http://localhost/indx/testxhqwse/xhqcicommands.html`
  - From a **remote** machine, `http://<XHQ_SERVER_HOST>/indx/testxhqwse/xhqcicommands.html`

XhqAdminWs Web Methods QuickList

Web Method	Description
`AnsImport`	Imports the condition definitions from an XML file into ANS.
`BackupRepos`	Backs up the XHQ Solution repos online, without having to shutdown the server.
`DeleteCollections`	Deletes one or more global collections from the XHQ Solution.
`DeleteComponents`	Deletes one or more components from the XHQ Model.
`DistributeModel`	Distributes the model from the "xhqServerHost" to "targetHost" model host or solution realm.
`ExportConnectionGroups`	Exports connection groups into an XML file.
`ExportConnectionProcesses`	Exports connection processes into an XML file.
`ExportConnectors`	Exports connectors definitions into a ZIP file.
`ExportDeleteConnectionGroups`	Exports the connection groups that are to be deleted (marked for deletion through import) into an XML file.
`ExportDeleteConnectionProcesses`	Exports the connection processes that are to be deleted (marked for deletion through import) into an XML file.
`ExportDeleteModel`	Exports the XHQ Model (components, component members, and views) to be deleted into a ZIP file.
`ExportDeleteRolesMappings`	Exports the role mappings definitions that are to be deleted (marked for deletion through import) into an XML file.
`ExportModel`	Exports the XHQ Workbench (components and views) into a ZIP file.
`ExportRoles`	Exports the component role definitions from the XHQ Workbench into an XML file.
`ExportRolesMappings`	Exports the role mappings definitions into an XML file.
`ExportSolution`	Exports the selected XHQ Solution configurations into an XML file.
`ExportSolutionDefs`	Exports the solution definitions into an XML file.
`ExportUoM`	Exports all existing units of measure (UoM) and categories from XHQ into an XML file.
`ExportVTCViews`	Exports VTC Views into a ZIP file.
`GetExportFileStream`	Returns the file stream previously generated by one of the asynchronous export `xhqci` commands operations.
`GetXhqciCommandOutput`	Returns the current available output for an `xhqci` command using a key from the client and a line offset.
`ImportConnectionGroups`	Imports connection groups from an XML file.
`ImportConnectionProcesses`	Imports connection processes from an XML file.
`ImportConnectors`	Imports connectors definitions from a ZIP file.

Web Method	Description
`ImportModel`	Imports a ZIP file into XHQ Workbench (components and views).
`ImportRoles`	Imports component role definitions from an XML file.
`ImportRolesMappings`	Imports the role mappings definitions from an XML file
`ImportSolution`	Imports an XML file into the XHQ Solution.
`ImportSolutionDefs`	Imports solution definitions from an XML file.
`ImportUoM`	Imports units of measure (UoM) and categories from an XML file.
`ImportVTCViews`	Imports VTC Views from a ZIP file.
`LinkServer`	Links to a server in a distributed XHQ environment.
`PurgeVTCViews`	Purges all VTC Views from XHQ.
`RefreshRoles`	Starts the process of refreshing roles for an XHQ Server.
`RevertUoM`	Reverts XHQ to the default units of measure (UoM) and categories.
`StartConnector`	Starts a connection process.
`StopConnector`	Stops a connection process.
`UnlinkServer`	Unlinks a server in a distributed XHQ environment.
`UpdateSolution`	Updates the XHQ Solution to the latest component model by performing a UTL operation.

XhqAdminWs Web Service Methods

The following is a detailed list of operations for the **`XhqAdminWs`** Web Service.

Method		Description

`XhqAdminWs.ExportModel`		Exports the XHQ Workbench (components and views) into a ZIP file.
		`public string ExportModel(`
		`  string xhqServerHost,`
		`  string rootComponent,`
		`  bool exportChildren,`
		`  bool exportViews,`
		`  bool exportTemplates,`
		`  bool encodedPassword,`
		`  string userName,`
		`  string password`
		`);`

Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
	rootComponent	The name of the root component to export.
	exportChildren	Boolean flag that is processed only if a root parameter is given. If set to FALSE, only the root component is exported. If set to TRUE, then the root component, and all it's children, are exported.
	exportViews	Boolean flag that specifies whether or not the views are to be exported.
	exportTemplates	Boolean flag that specifies whether or not the templates are to be exported
	encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
	userName	The name of the user for which the xhqci command is run.
	password	The password for the user for which the xhqci command is run.
Return Value		A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the export process output and status. This key can also be used to invoke the `GetExportFileStream` operation to retrieve the actual exported ZIP file.
Remarks		N/A

Method	Description

`XhqAdminWs.ImportModel`	Imports a ZIP file into XHQ Workbench (components and views).
	`public string ImportModel(`
	`  byte[] importFileStream,`
	`  string xhqServerHost,`
	`  string rootComponent,`
	`  bool importChildren,`

Method		Description

		`bool importViews,`
		`bool importViewProps,`
		`bool importTemplates,`
		`bool rollback,`
		`bool append,`
		`bool delete,`
		`bool encodedPassword,`
		`string userName,`
		`string password`
		`);`

Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
	importFileStream	The ZIP file to be imported, as a byte array.
	rootComponent	The name of the root component to import.
	importChildren	Boolean flag that is processed only if a root parameter is given. If set to FALSE, only the root component is imported. If set to TRUE, then the root component, and all it's children, are imported.
	importViews	Boolean flag that specifies whether or not the views are to be imported.
	importTemplates	Boolean flag that specifies whether or not the templates are to be imported.
	rollback	Boolean flag that if set to TRUE sets the import mode to "rollback".
	append	Boolean flag that if set to TRUE sets the import mode to "append". Component members included in the model import file are either replaced (existing matching members) or added to the XHQ Workbench. Components that existed in the XHQ Workbench, but whose definition does not appear in the model file, are left alone. Rollbacks are not supported in this mode.
	delete	Boolean flag that if set to TRUE sets the import mode to "delete". If a component is in the model import file, but has no members, then the component is deleted. If a component is in the import file, but has members, then deletes are performed on the members included in the component definition in the model import file. Any members not specified in this case, are left alone. Components that existed in the XHQ Workbench, but are not in the model import file, are also left alone. Rollbacks are not supported in this mode.
	encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
	userName	The name of the user for which the xhqci command is run.
	password	The password for the user for which the xhqci command is run.
Return Value		A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the import process output and status.

Method	Description
Remarks	N/A

Method	Description

`XhqAdminWs.DeleteComponents`	Deletes one or more components from the XHQ Model.
	`public string DeleteComponents(`
	`  string xhqServerHost,`
	`  string[] deletedComponentsList,`
	`  bool encodedPassword,`
	`  string userName,`
	`  string password`
	`);`

Parameters: *xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
deletedComponentsList	String array containing the names (or class IDs) of the components to be deleted.
encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
userName	The name of the user for which the xhqci command is run.
password	The password for the user for which the xhqci command is run.
Return Value	A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the deletion process output and status.
Remarks	N/A

Method	Description

`XhqAdminWs.ExportDeleteModel`	Exports the XHQ Model (components, component members, and views) to be deleted into a ZIP file.
	`public string ExportDeleteModel(`
	`  string xhqServerHost,`
	`  string[] deletedComponentsList,`
	`  Dictionary<string, string> deletedMembersList,`
	`  Dictionary<string, string> deletedViewsList,`
	`  bool encodedPassword,`
	`  string userName,`
	`  string password`
	`);`

Parameters: *xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
deletedComponentsList	String array containing the names (or class IDs) of the components to be

Method	Description
	deleted in the export file.
deletedMembersList	Hash map (Dictionary) containing the names of the component members to be deleted. The key of the map is the component name that contains the members to be deleted, as a string, and the value is a string array containing the names of the members to be marked for deletion in the export file.
deletedViewsList	Hash map (Dictionary) containing the names of the component views to be deleted. The key of the map is the component name that contains the views to be deleted, as a string, and the value is a string array containing the names of the views to be marked for deletion in the export file.
encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
userName	The name of the user for which the xhqci command is run.
password	The password for the user for which the xhqci command is run.
Return Value	A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the export process output and status. This key can also be used to invoke the `GetExportFileStream` operation to retrieve the actual exported ZIP file.
Remarks	N/A

Method	Description

`XhqAdminWs.ExportSolution`	Exports the selected XHQ Solution configurations into an XML file.
	`public string ExportSolution(`
	`  string xhqServerHost,`
	`  string exportPathRoot,`
	`  byte[] pathsFileStream,`
	`  bool exportChildren,`
	`  bool exportCollections,`
	`  bool delete,`
	`  bool encodedPassword,`
	`  string userName,`
	`  string password`
	`);`

Parameters: *xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
exportPathRoot	The absolute path for the solution's export root. The path must begin with two colon characters (::). If NULL or empty, the default is the top-level solution node.
pathsFileStream	A byte array representing a file that has a single path of an object in the solution, per line. It can be any valid filename.
exportChildren	Boolean flag that specifies whether the descendants of the

Method	Description
	`exportPathRoot` are to be exported or not. If set to FALSE, only the root path configuration is exported. If set to TRUE, then the root path configuration, and all it's children, are exported.
exportCollections	Boolean flag that specifies whether the global collections are exported, or not.
delete	Boolean flag that specifies whether the NULL aliases, expressions, history, link collections, roles, and connection group configurations are to be exported with a `none` XML attribute, or not.
encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
userName	The name of the user for which the xhqci command is run.
password	The password for the user for which the xhqci command is run.
Return Value	A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the export process output and status. This key can also be used to invoke the `GetExportFileStream` operation to retrieve the actual exported XML file.
Remarks	N/A

Method	Description

`XhqAdminWs.ImportSolution`	Imports an XML file into the XHQ Solution.
	`public string ImportSolution(`
	`  byte[] importFileStream,`
	`  string xhqServerHost,`
	`  bool encodedPassword,`
	`  string userName,`
	`  string password`
	`);`

Parameters: *xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
importFileStream	The XML file to be imported, as a byte array.
encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
userName	The name of the user for which the xhqci command is run.
password	The password for the user for which the xhqci command is run.
Return Value	A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the import process output and status.
Remarks	N/A

Method	Description

`XhqAdminWs.UpdateSolution`	Updates the XHQ Solution to the latest component model by performing a UTL operation.
	`public string UpdateSolution(`
	`  string xhqServerHost,`
	`  bool encodedPassword,`
	`  string userName,`
	`  string password`
	`);`

Parameters: *xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
userName	The name of the user for which the xhqci command is run.
password	The password for the user for which the xhqci command is run.
Return Value	A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the update process output and status.
Remarks	N/A

Method	Description

`XhqAdminWs.DeleteCollections`	Deletes one or more global collections from the XHQ Solution.
	`public string DeleteCollections(`
	`  string xhqServerHost,`
	`  string[] deletedCollectionsList,`
	`  bool deleteTags,`
	`  bool encodedPassword,`
	`  string userName,`
	`  string password`
	`);`

Parameters: *xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
deletedCollectionsList	String array containing the names or numeric IDs (OIDs) of the global collections to be deleted.
deleteTags	Boolean flag that specified whether or not to delete Tags.
encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
userName	The name of the user for which the xhqci command is run.
password	The password for the user for which the xhqci command is run.

Method	Description
Return Value	A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the deletion process output and status.
Remarks	N/A

Method		Description

`XhqAdminWs.ExportSolutionDefs`		Exports the solution defintions into an XML file.
		`public string ExportSolutionDefs(`
		`  string xhqServerHost,`
		`  string[] solutionDefsList,`
		`  bool encodedPassword,`
		`  string userName,`
		`  string password`
		`);`

Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
	solutionDefsList	String array containing the names of the solutions to be exported. If NULL or empty (size 0), ALL existing solutions are exported.
	encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
	userName	The name of the user for which the xhqci command is run.
	password	The password for the user for which the xhqci command is run.
Return Value		A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the export process output and status. This key can also be used to invoke the `GetExportFileStream` operation to retrieve the actual exported XML file.
Remarks		N/A

Method		Description

`XhqAdminWs.ImportSolutionDefs`		Imports solution definitions from an XML file.
		`public string ImportSolutionDefs(`
		`  byte[] importFileStream,`
		`  string xhqServerHost,`
		`  string[] solutionDefsList,`
		`  bool encodedPassword,`
		`  string userName,`
		`  string password`
		`);`

Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts

Method	Description
	this web service (localhost).
importFileStream	The XML file to be imported, as a byte array.
solutionDefsList	String array containing the names of the solutions to be imported from the import XML file. If NULL or empty (size 0), ALL existing solutions are imported.
encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
userName	The name of the user for which the xhqci command is run.
password	The password for the user for which the xhqci command is run.
Return Value	A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the import process output and status.
Remarks	N/A

Method	Description

`XhqAdminWs.DistributeModel`	Distributes the model from the "xhqServerHost" to "targetHost" model host or solution realm.
	`public string[] DistributeModel(`
	`  string xhqServerHost,`
	`  string targetHostOrRealm,`
	`  boolUseRealm,`
	`  bool encodedPassword,`
	`  string userName,`
	`  string password`
	`);`

Parameters: *xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
targetHostOrRealm	The FQDN target host name to update the model from the "xhqServerHost" or a realm name of a solution to do model update.
useRealm	The flag that indicates the type for the model distribution. True if the model distribution is made to a solution realm; False if the model distribution is made to a target host model.
encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
userName	The name of the user for which the xhqci command is run.
password	The password for the user for which the xhqci command is run.
Return Value	A unique key, as a string, that can be used to invoke the 'GetXhqciCommandOutput' operation to check the export process output

Method	Description
	and status. This key can also be used to invoke the 'GetExportFileStream' operation to retrieve the actual exported XML file
Remarks	N/A

Method	Description

`XhqAdminWs.RefreshRoles`	Starts the process of refreshing roles for an XHQ Server. This operation is the equivalent of the `rolesadmin refresh` xhqci command.
	Important: This operation can only be executed by a user with Roles Admin permission.
	This is the only synchronous operation for this web service.
	`public string[] RefreshRoles(`
	`  string xhqServerHost,`
	`  bool encodedPassword,`
	`  string userName,`
	`  string password`
	`);`

Parameters: *xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
userName	The name of the user for which the xhqci command is run.
password	The password for the user for which the xhqci command is run.
Return Value	The xhqci `rolesadmin refresh` command output as an array of strings, with each string representing a new line in the output message.
Remarks	N/A

Method	Description

`XhqAdminWs.ExportRoles`	Exports the component role definitions from the XHQ Workbench into an XML file (that is separate from the XHQ Workbench export file). The export file includes all roles assigned for all components. The Model Admin role is implicitly included in the export file against all components. This means that even though the export file does not explicitly include the Model Admin role, it is automatically assumed that the role is assigned to all components.
	`public string ExportRoles(`
	`  string xhqServerHost,`
	`  bool encodedPassword,`
	`  string userName,`
	`  string password`
	`);`

Method		Description
Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
	encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
	userName	The name of the user for which the xhqci command is run.
	password	The password for the user for which the xhqci command is run.
Return Value		A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the export process output and status. This key can also be used to invoke the `GetExportFileStream` operation to retrieve the actual exported XML file.
Remarks		N/A

Method		Description

`XhqAdminWs.ImportRoles`		Imports component role definitions from an XML file. The import file contains all of the components and the role assignment for each component. The Model Admin role is implicitly assigned to all components. This means that even though the import file does not explicitly include the Model Admin role, it is automatically assumed that the role is assigned to all components. If an invalid role is encountered in the import file, then the role is ignored and the import process continues. If all roles for a given component are invalid, then the only role assigned to that component will be the Model Admin role.
		Important: You must have both "Role Assigning" and "Model Importing" rights to import component roles.
		`public string ImportRoles(`
		`  byte[] importFileStream,`
		`  string xhqServerHost,`
		`  bool encodedPassword,`
		`  string userName,`
		`  string password`
		`);`

Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
	importFileStream	The XML file to be imported, as a byte array.
	encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
	userName	The name of the user for which the xhqci command is run.
	password	The password for the user for which the xhqci command is run.

Method	Description
Return Value	A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the import process output and status.
Remarks	N/A

Method	Description

`Xhq AdminWs.ExportRolesMappings`	Exports the role mappings definitions into an XML file.
	`public string ExportRolesMappings(`
	`  string xhqServerHost,`
	`  string[] rolesList,`
	`  bool encodedPassword,`
	`  string userName,`
	`  string password`
	`);`

Parameters: *xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
rolesList	String array containing the names of the role mappings to be exported. If NULL, or empty (size 0), then ALL existing role mappings are exported.
encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
userName	The name of the user for which the xhqci command is run.
password	The password for the user for which the xhqci command is run.
Return Value	A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the export process output and status. This key can also be used to invoke the `GetExportFileStream` operation to retrieve the actual exported XML file.
Remarks	N/A

Method	Description

`Xhq AdminWs.ImportRolesMappings`	Imports the role mappings definitions from an XML file.
	`public string ImportRolesMappings(`
	`  byte[] importFileStream,`
	`  string xhqServerHost,`
	`  string[] rolesList,`
	`  bool encodedPassword,`
	`  string userName,`
	`  string password`
	`);`

Parameters: *xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts

Method		Description
		this web service (localhost).
	importFileStream	The XML file to be imported, as a byte array.
	rolesList	String array containing the names of the role mappings to be imported from the import XML file. If NULL, or empty (size 0), then ALL existing role mappings from the XML file are imported.
	encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
	userName	The name of the user for which the xhqci command is run.
	password	The password for the user for which the xhqci command is run.
Return Value		A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the import process output and status.
Remarks		N/A

Method		Description

`Xhq Admin Ws.ExportDeleteRolesMappings`		Exports the role mappings definitions that are to be deleted (marked for deletion through import) into an XML file.
		`public string ExportDeleteRolesMappings(`
		`  string xhqServerHost,`
		`  string[] deletedRolesList,`
		`  byte[] rolesFileStream,`
		`  bool encodedPassword,`
		`  string userName,`
		`  string password`
		`);`

Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
	deletedRolesList	String array containing the names of the role mappings to be marked for deletion in the export file.
	rolesFileStream	A byte array representing a file that has the name of a role mapping, per line. It can be any valid filename.
	encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
	userName	The name of the user for which the xhqci command is run.
	password	The password for the user for which the xhqci command is run.
Return Value		A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the export process output and status. This key can also be used to invoke the `GetExportFileStream` operation to retrieve the actual exported XML

Method	Description
	file.
Remarks	N/A

Method		Description

`XhqAdminWs.ExportConnectors`		Exports connectors definitions into a ZIP file.
		`public string ExportConnectors(`
		`  string xhqServerHost,`
		`  string[] connectorsList,`
		`  bool encodedPassword,`
		`  string userName,`
		`  string password`
		`);`

Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
	connectorsList	String array containing the names of the connectors to be exported. If NULL, or empty (size 0), then ALL existing connector definitions are exported.
	encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
	userName	The name of the user for which the xhqci command is run.
	password	The password for the user for which the xhqci command is run.
Return Value		A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the export process output and status. This key can also be used to invoke the `GetExportFileStream` operation to retrieve the actual exported ZIP file.
Remarks		N/A

Method		Description

`XhqAdminWs.ImportConnectors`		Imports connectors definitions from a ZIP file.
		`public string ImportConnectors(`
		`  byte[] importFileStream,`
		`  string xhqServerHost,`
		`  string[] connectorsList,`
		`  bool encodedPassword,`
		`  string userName,`
		`  string password`
		`);`

Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).

Method	Description
importFileStream	The XML file to be imported, as a byte array.
connectorsList	String array containing the names of the role mappings to be imported. If NULL, or empty (size 0), then ALL existing connector definitions from the XML file are imported.
encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
userName	The name of the user for which the xhqci command is run.
password	The password for the user for which the xhqci command is run.
Return Value	A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the import process output and status.
Remarks	N/A

Method	Description

`Xhq Admin Ws.ExportConnectionGroups`	Exports connection groups into an XML file.
	`public string ExportConnectionGroups(`
	`  string xhqServerHost,`
	`  string[] connectionGroupsList,`
	`  bool encodedPassword,`
	`  string userName,`
	`  string password`
	`);`

Parameters: *xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
connectionGroupsList	String array containing the names of the connection groups to be exported. If NULL, or empty (size 0), then ALL existing connection groups are exported.
encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
userName	The name of the user for which the xhqci command is run.
password	The password for the user for which the xhqci command is run.
Return Value	A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the export process output and status. This key can also be used to invoke the `GetExportFileStream` operation to retrieve the actual exported XML file.
Remarks	N/A

Method	Description

Xhq Admin Ws.ImportConnectionGroups	Imports connection groups from an XML file.
	`public string ImportConnectionGroups(`
	`  byte[] importFileStream,`
	`  string xhqServerHost,`
	`  string[] connectionGroupsList,`
	`  bool encodedPassword,`
	`  string userName,`
	`  string password`
	`);`

Parameters: *xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
importFileStream	The XML file to be imported, as a byte array.
connectionGroupsList	String array containing the names of the connection groups to be imported. If NULL, or empty (size 0), then ALL existing connection groups from the XML file are imported.
encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
userName	The name of the user for which the xhqci command is run.
password	The password for the user for which the xhqci command is run.
Return Value	A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the import process output and status.
Remarks	N/A

Method	Description

Xhq Admin Ws.ExportDeleteConnectionGroups	Exports the connection groups that are to be deleted (marked for deletion through import) into an XML file.
	`public string ExportDeleteConnectionGroups(`
	`  string xhqServerHost,`
	`  string[] deletedConnectionGroupsList,`
	`  bool encodedPassword,`
	`  string userName,`
	`  string password`
	`);`

Parameters: *xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
delectedConnectionGroupsList	String array containing the names of the connection groups to be deleted in the XML export file.
encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).

Method	Description
userName	The name of the user for which the xhqci command is run.
password	The password for the user for which the xhqci command is run.
Return Value	A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the export process output and status. This key can also be used to invoke the `GetExportFileStream` operation to retrieve the actual exported XML file.
Remarks	N/A

Method	Description

Xhq Admin Ws.ExportConnectionProcesses	Exports connection processes into an XML file.
	`public string ExportConnectionProcesses(`
	`  string xhqServerHost,`
	`  string[] connectionProcessesList,`
	`  bool encodedPassword,`
	`  string userName,`
	`  string password`
	`);`

Parameters: *xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
connectionProcessesList	String array containing the names of the connection processes to be exported. If NULL, or empty (size 0), then ALL existing connection processes are exported.
encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
userName	The name of the user for which the xhqci command is run.
password	The password for the user for which the xhqci command is run.
Return Value	A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the export process output and status. This key can also be used to invoke the `GetExportFileStream` operation to retrieve the actual exported XML file.
Remarks	N/A

Method	Description

Xhq Admin Ws.ImportConnectionProcesses	Imports connection processes from an XML file.
	`public string ImportConnectionProcesses(`
	`  byte[] importFileStream,`

Method		Description

		`string xhqServerHost,`
		`string[] connectionProcessesList,`
		`bool encodedPassword,`
		`string userName,`
		`string password`
		`);`

Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
	importFileStream	The XML file to be imported, as a byte array.
	connectionProcessesList	String array containing the names of the connection processes to be imported. If NULL, or empty (size 0), then ALL existing connection processes from the XML file are imported.
	encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
	userName	The name of the user for which the xhqci command is run.
	password	The password for the user for which the xhqci command is run.
Return Value		A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the import process output and status.
Remarks		N/A

Method		Description

`Xhq Admin Ws.ExportDeleteConnectionProcesses`		Exports the connection processes that are to be deleted (marked for deletion through import) into an XML file.
		`public string ExportDeleteConnectionProcesses(`
		`string xhqServerHost,`
		`string[] deletedConnectionProcessesList,`
		`bool encodedPassword,`
		`string userName,`
		`string password`
		`);`

Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
	delectedConnectionProcessesList	String array containing the names of the connection processes to be deleted in the XML export file.
	encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
	userName	The name of the user for which the xhqci command is run.

Method		Description
	password	The password for the user for which the xhqci command is run.
Return Value		A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the export process output and status. This key can also be used to invoke the `GetExportFileStream` operation to retrieve the actual exported XML file.
Remarks		N/A

Method		Description

`XhqAdminWs.ExportUoM`		Exports all existing units of measure (UoM) and categories from XHQ into an XML file.
		`public string ExportUoM(`
		`  string xhqServerHost,`
		`  bool encodedPassword,`
		`  string userName,`
		`  string password`
		`);`

Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
	encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
	userName	The name of the user for which the xhqci command is run.
	password	The password for the user for which the xhqci command is run.
Return Value		A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the export process output and status. This key can also be used to invoke the `GetExportFileStream` operation to retrieve the actual exported XML file.
Remarks		N/A

Method	Description

`XhqAdminWs.ImportUoM`	Imports units of measure (UoM) and categories from an XML file.
	`public string ImportUoM(`
	`  byte[] importFileStream,`
	`  string xhqServerHost,`
	`  bool replace,`
	`  bool encodedPassword,`
	`  string userName,`
	`  string password`
	`);`

Method		Description
Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
	importFileStream	The XML file to be imported, as a byte array.
	replace	Boolean flag that specifies if the units and categories are being replaced or not through import. If the replace flag is TRUE, the categories and units present in the current Model server are all deleted and replaced with the ones in the XML file. If the replace flag is FALSE, then the import operation performs a modification, renaming, or deletion for the categories and units that are present in the Model server.
	encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
	userName	The name of the user for which the xhqci command is run.
	password	The password for the user for which the xhqci command is run.
Return Value		A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the import process output and status.
Remarks		N/A

Method		Description

`XhqAdminWs.RevertUoM`		Reverts XHQ to the default units of measure (UoM) and categories.
		`public string RevertUoM(`
		`  string xhqServerHost,`
		`  bool encodedPassword,`
		`  string userName,`
		`  string password`
		`);`

Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
	encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
	userName	The name of the user for which the xhqci command is run.
	password	The password for the user for which the xhqci command is run.
Return Value		A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the reversion process output and status.
Remarks		N/A

Method		Description

`XhqAdminWs.ExportVTCViews`		Exports VTC Views into a ZIP file.
		`public string ExportVTCViews(`
		`  string xhqServerHost,`
		`  Dictionary<string, VTCView> reusableViewsList,`
		`  Dictionary<string, VTCView> singleViewsList,`
		`  int accessType,`
		`  bool encodedPassword,`
		`  string userName,`
		`  string password`
		`);`

Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
	reusableViewsList	Hash map (Dictionary) containing the names of the reusable VTC views to be exported. The key of the map is the component name, as a string, that contains the reusable VTC views; the value is a "VTCView" object array containing the reusable VTC views information (name, owner, deletion flag) associated with that XHQ component and that will be exported. If the "VTCView" array is NULL or empty for a component key, then ALL reusable views are exported for that component.
	singleViewsList	Hash map (Dictionary) containing the names of the single VTC views to be exported. The key of the map is the XHQ path, as a string, that contains the single VTC views; the value is a "VTCView" object array containing the single VTC views information (name, owner, deletion flag) associated with that XHQ path and that will be exported. If the "VTCView" array is NULL or empty for a path key, then ALL single views that are associated with that particular path are exported.
	accessType	Integer that specifies the view access type selector. It can have one of the following three values: • 0 - ALL VTC views (**both** private and published) are exported. • 1 - Only the **published** VTC views are exported. • 2 - Only the **private** VTC views are exported.
	encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
	userName	The name of the user for which the xhqci command is run.
	password	The password for the user for which the xhqci command is run.
Return Value		A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the export process output and status. This key can also be used to invoke the `GetExportFileStream` operation to retrieve the actual exported ZIP file.
Remarks		N/A

Method		Description

`XhqAdminWs.ImportVTCViews`		Imports VTC Views from a ZIP file.
		`public string ImportVTCViews(`
		`  byte[] importFileStream,`
		`  string xhqServerHost,`
		`  bool encodedPassword,`
		`  string userName,`
		`  string password`
		`);`

Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
	importFileStream	The ZIP file to be imported, as a byte array.
	encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
	userName	The name of the user for which the xhqci command is run.
	password	The password for the user for which the xhqci command is run.
Return Value		A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the import process output and status.
Remarks		N/A

Method		Description

`XhqAdminWs.PurgeVTCViews`		Purges all VTC Views from XHQ.
		`public string PurgeVTCViews(`
		`  string xhqServerHost,`
		`  bool encodedPassword,`
		`  string userName,`
		`  string password`
		`);`

Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
	encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
	userName	The name of the user for which the xhqci command is run.
	password	The password for the user for which the xhqci command is run.
Return Value		A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the view purge process output and status.
Remarks		N/A

Method		Description

Xhq AdminWs.GetXhqciCommandOutput		Returns the current available output for an `xhqci` command using a key from the client and a line offset. This method is to be used to asynchronously receive the `xhqci` command output/status text. Use the key returned by the asynchronous xhqci operations (for example, `ImportModel()`, `ExportModel()`, and so forth) to invoke this method periodically, to check the status of the import.
		`public string[] GetXhqciCommandOutput(`
		`  string key,`
		`  int lineOffset`
		`);`

Parameters:	*key*	The key used by the web service to retrieve the `xhqci` command output.
	lineOffset	The starting line (from 0) from which the command output will be read and returned to the client.
Return Value		The `xhqci` import command output as an array of strings, each string representing a new line in the output message. It will return NULL when there are no more lines to read from the process output (EOF).
Remarks		N/A

Method		Description

Xhq AdminWs.GetExportFileStream		Returns the file stream previously generated by one of the asynchronous export `xhqci` commands operations (for example, `ExportModel`, `ExportSolution`) as a byte array. This operation needs to be invoked after the `GetXhqciCommandOutput()` returns NULL, signaling the completion of the export `xhqci` operation.
		Upon completion of this method the exported temporary file will be permanently deleted (implicit deletion) from the location on the machine that hosts the web service.
		`public byte[] GetExportFileStream(`
		`  string key`
		`);`

Parameters:	*key*	The key used by the web service to retrieve the exported file stream.
Return Value		The export file stream, as a byte array.
Remarks		N/A

Method	Description

XhqAdminWs.AnsImport	Imports the condition definitions from an XML file into XHQ ANS.
	`public string[] AnsImport(`
	`  byte[] importFileStream,`
	`  string ansServerHost,`

Method		Description

		`bool autoCreateLists,`
		`bool updateDefsById,`
		`bool importSubsc,`
		`bool importLOConfigs,`
		`bool encodedPassword,`
		`string userName,`
		`string password`
		`);`

Parameters:	*importFileStream*	The XML file to be imported, as a byte array.
	ansServerHost	The name of the machine that hosts the ANS Server. If NULL the update operation will be performed on the same XHQ ANS Server that is hosted on the machine that hosts this web service (localhost).
	autoCreateLists	Boolean flag that indicates if to automatically create definition and subscription lists or not.
	updateDefsById	Boolean flag that indicates if to update definitions and subscriptions using definition IDs or not.
	importSubsc	Boolean flag that specifies if ANS subscriptions are also to be imported or not.
	importLOConfigs	Boolean flag that specifies if Lost Opportunity configurations are also to be imported or not.
	encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
	userName	The name of the user for which the xhqci command is run.
	password	The password for the user for which the xhqci command is run.
Return Value		A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the ANS import process output and status.
Remarks		N/A

Method		Description

`XhqAdminWs.BackupRepos`		Online backup of the XHQ Solution repos, without having to shutdown the server.
		`public string BackupRepos(`
		`string xhqServerHost,`
		`bool createZipFile,`
		`bool encodedPassword,`
		`string userName,`
		`string password`
		`);`

Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts

Method	Description
	this web service (localhost).

createZipFile	A flag that indicates if the output of the online backup for the repos is going to be a ZIP file saved on the web service client machine or a directory saved in the "data" folder on the XHQ Server machine, for example:
	`C:\XHQ\data\backups\repos_backups\<unique time id>_<machine name>`
	If **true**, then the backup is saved as a ZIP file in the same location on the XHQ Server machine, for example:
	`C:\XHQ\data\backups\repos_backups\<unique time id>_<machine name>.zip`

encodedPassword	A Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
userName	The name of the user for which the xhqci command is run.
password	The password for the user for which the xhqci command is run.
Return Value	A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the online backup process output and status. This key can also be used to invoke the `GetExportFileStream` operation to retrieve the actual backed up repos ZIP file.
Remarks	N/A

Method	Description

`XhqAdminWs.LinkServer`	Links to a server in a distributed XHQ environment.
	`public string LinkServer(`
	`  string xhqServerHost,`
	`  string linkPath,`
	`  string linkServer,`
	`  bool encodedPassword,`
	`  string userName,`
	`  string password`
	`);`

Parameters: *xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
linkPath	The path for the parent component.
linkServer	The server name for link.
encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
userName	The name of the user for which the xhqci command is run.

Method		Description
	password	The password for the user for which the xhqci command is run.
Return Value		A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the server linking process output and status.
Remarks		N/A

Method		Description

`XhqAdminWs.UnlinkServer`		Unlinks a server in a distributed XHQ environment.
		`public string UnlinkServer(`
		`  string xhqServerHost,`
		`  string linkPath,`
		`  bool encodedPassword,`
		`  string userName,`
		`  string password`
		`);`

Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
	linkPath	The path for the parent component.
	encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
	userName	The name of the user for which the xhqci command is run.
	password	The password for the user for which the xhqci command is run.
Return Value		A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the unlinking process output and status.
Remarks		N/A

Method		Description

`XhqAdminWs.StartConnector`		Starts a connection process.
		`public string StartConnector(`
		`  string xhqServerHost,`
		`  string connectorName,`
		`  bool encodedPassword,`
		`  string userName,`
		`  string password`
		`);`

Parameters:	*xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).

Method	Description
connectorName	The name of the connector to be started.
encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
userName	The name of the user for which the xhqci command is run.
password	The password for the user for which the xhqci command is run.
Return Value	A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the connector starting/stopping process output and status.
Remarks	N/A

Method	Description

`XhqAdminWs.StopConnector`	Stops a connection process.
	`public string StopConnector(`
	`  string xhqServerHost,`
	`  string connectorName,`
	`  bool encodedPassword,`
	`  string userName,`
	`  string password`
	`);`

Parameters: *xhqServerHost*	The name of the machine that hosts the XHQ Server. If NULL, the export is performed on the same XHQ Server that is hosted on the machine that hosts this web service (localhost).
connectorName	The name of the connector to be stopped.
encodedPassword	Boolean flag that specifies if the password is sent as clear text or encoded (using Base64).
userName	The name of the user for which the xhqci command is run.
password	The password for the user for which the xhqci command is run.
Return Value	A unique key, as a string, that can be used to invoke the `GetXhqciCommandOutput` operation to check the connector starting/stopping process output and status.
Remarks	N/A

EXAMPLE: XHQADMINWS WEB SERVICE

The following code snippet is an example of a .NET client (consumer) that invokes the web service **`ImportModel`** operation.

```
    protected void btnImport_Click(object sender, EventArgs e)
        {

   try
            {
                Stream x = ImportFileUpload.FileContent;
                int fileLen = ImportFileUpload.PostedFile.ContentLength;
                byte[] buffer = new byte[x.Length];
                x.Read(buffer, 0, fileLen);

                XHQAdminWebServiceClient wsClient = new XHQAdminWebServiceClient();
                string key = wsClient.ImportModel(buffer,
                                                  xhqServerHost,
                                                  root,
                                                  importChildren,
                                                  importViews,
                                                  importViewProps,
                                                  importTemplates,
                                                  rollback,
                                                  append,
                                                  delete,
                                                  encodedPassword,
                                                  username,
                                                  password);

                Session["WsClient_" + key] = wsClient;
                Session["lineOffset_" + key] = "0";
                Session["status_" + key] = outputStatus;

                Response.Redirect("../CommandOutput.aspx?key=" + key);
            }
            catch (Exception ex)
            {
                lblResult.Text = "WS Error: " + ex.Message;
            }
        }
```

In this case, **`FileUpload1`** is a .NET Framework File Upload control used in an ASP page to specify the input path for the import ZIP file on the client. The output of the `xhqci` command output and status is displayed on the screen in a different ASP.Net web page, `CommandOutput.aspx`.

If encoding is needed for security purposes, the following method can be used to encode the password using UTF-8 string and base64 encode.

```
    //method to encrypt a string
        private string Encode(string strPwd)
        {
            try
            {
                byte[] bytEncData = new byte[strPwd.Length];
                bytEncData = System.Text.Encoding.UTF8.GetBytes(strPwd);
                string strEncodedData = Convert.ToBase64String(bytEncData);
                return strEncodedData;
            }
```

```
            catch (Exception ex)
            {
                throw new Exception("Error in base64Encode" + ex.Message);
            }
        }
```

The `CommandOutput.aspx` web page invokes the web service **`GetXhqciCommandOutput()`** method periodically (using, for instance, the `window.setInterval()` JavaScript function) and renders the `xhqci` command output in the web page asynchronously, using a Label control:

```
        //periodically check the status of the xhqci command output by polling
        private void showXhqciCommandOutput()
        {
            XhqWse wsClient = (WsTestClient.xhqWse.XhqWse)Session["WsClient"];
            String key = (String)Session["OutputKey"];
            String lo = (String)Session["lineOffset"];
            ArrayList outputStatus = (ArrayList)Session["status"];

            int lineOffset = Convert.ToInt32(lo);

            if (wsClient != null && key != null)
            {
                String[] outStr = wsClient.GetXhqciCommandOutput(key, lineOffset);

                if (outStr != null)
                {
                    foreach (String line in outStr)
                    {
                        outputStatus.Add(line);
                    }

                    String s = "";
                    foreach (object line in outputStatus)
                    {
                        s = s + (string)line + "<br>";
                    }
                    output.Text = s;

                    Session["lineOffset"] = "" + outputStatus.Count;
                    Session["status"] = outputStatus;
                }
            }
```

Using the ANS Web Service, AnsWs

The ANS Web Service allows third party developers to easily create, update, read, and delete ANS configurations, as well as give them access to various types of ANS reports. The ANS Webs Service is **`AnsWs`**. It is a WCF-based web service and provides both a SOAP interface as well as a REST interface for accessing its operations.

The ANS Web Service allows full CRUD (Create/Read/Update/Delete)-style operations performed by the XHQ Alert Server on the ANS configurations.

The **URL to access** the web service is:

- `http://[%XHQ_SERVER%]/indx/answs/`

  or
- `http://[%XHQ_SERVER%]/indx/answs/AnsWs.svc`

The **WSDL** has the following URL address:

- `http://[%XHQ_SERVER%]/indx/answs/AnsWs.svc?wsdl`

Example:

`http://calc.corp.indxhq.com/indx/answs/AnsWs.svc?wsdl`

The client programming invocation pattern, for the web service to access various ANS configuration functions, follows these 3 basic steps:

Step 1 Create a new instance of the `AnsWs` proxy object.

Step 2 Call the specific `AnsWs` method, or group of methods, using the web service proxy object constructed at step 1.

Step 3 Disconnect by closing the proxy client.

Here's a simple client example:

```
ConditionDef[] defList = new ConditionDef[3];
//populate the defList array
//Invoke the XHQ Web Service
AnsServiceClient client = new AnsServiceClient();
Response[] resp = client.CreateConditionDefinitions("localhost", defList);
// Always close the client.
client.Close();
foreach(Response r in resp)
{
                Console.WriteLine(" ");
                Console.WriteLine("Def Name: " + r.Name);
                Console.WriteLine("Def ID: " + r.ID);
                Console.WriteLine("Error Msg: " + r.ErrorMessage);
}
```

AnsWs Web Methods QuickList

Method	Description
CreateCodes	Creates new ANS codes (e.g. categories) using bulk processing.
CreateConditionDefinitionLists	Creates new ANS condition definition lists using bulk processing.

Method	Description
CreateConditionDefinitions	Creates new ANS condition definitions using bulk processing.
CreateListSections	Creates new ANS list sections using bulk processing.
CreateNotificationRoutings	Creates new ANS notification routings using bulk processing.
CreateSubscriptionLists	Creates new ANS subscription lists using bulk processing.
CreateSubscriptions	Creates new ANS subscriptions using bulk processing.
DeleteCodes	Deletes existing ANS codes using bulk processing.
DeleteConditionDefinitionLists	Deletes existing ANS condition definition lists using bulk processing.
DeleteConditionDefinitions	Deletes existing ANS condition definitions using bulk processing.
DeleteListSections	Deletes existing ANS definition lists sections using bulk processing.
DeleteNotificationRoutings	Deletes existing ANS notification routings using bulk processing.
DeleteSubscriptionLists	Deletes existing ANS subscription lists using bulk processing.
DeleteSubscriptions	Deletes existing ANS subscriptions using bulk processing.
GetCodesById	Retrieves a collection of codes (e.g. categories) given their correspondent IDs using bulk processing.
GetCodesByName	Retrieves list of ANS codes (e.g. categories) given their correspondent names and code types, using bulk
GetConditionDefinitionsById	Retrieves a list of ANS condition definitions given their correspondent IDs using bulk processing.
GetConditionDefinitionsByList	Retrieves a list of all ANS condition definitions for a given definition list, using bulk processing.
GetConditionDefinitionsByName	Retrieves a list of ANS condition definitions given their names using bulk processing.
GetDefinitionListsById	Retrieves a collection of ANS definition lists given their correspondent IDs using bulk processing.
GetDefinitionListsByName	Retrieves a collection of ANS definition lists given their names using bulk processing.
GetListSectionsById	Retrieves a collection of ANS definition list sections given their correspondent IDs and the definition list IDs.
GetListSectionsByList	Retrieves all ANS sections for a given definition list, using bulk processing.
GetListSectionsByName	Retrieves a collection of ANS definition list sections given their correspondent names and the definition list names, using bulk processing.
GetMyNotificationRoutings	Retrieves all ANS notification routings for the caller user, using bulk processing.
GetNewConditionDef	Utility method that creates new, default initialized,

Method	Description
	ConditionDef object. It is recommended to call this method before calling CreateConditionDefinitions or UpdateConditionDefinitions methods.
GetNewSubscription	Utility method that creates new, default initialised, Subscription object. It is recommended to call this method before calling CreateSubscriptions or UpdateSubscriptions methods.
GetNewSubscriptionList	Utility method that creates new, default initialized, SubscriptionList object. It is recommended to call this method before calling CreateSubscriptionLists or UpdateSubscriptionLists methods.
GetNotificationRoutingsById	Retrieves a collection of ANS notification routings given their correspondent IDs using
GetSubscriptionListsById	Retrieves a collection of ANS subscription lists given their correspondent IDs using bulk
GetSubscriptionListsByName	Retrieves a collection of ANS subscription lists given their names using bulk processing
GetSubscriptionsById	Retrieves a collection of ANS condition subscriptions given their correspondent IDs using bulk processing.
GetSubscriptionsByList	Retrieves all ANS subscriptions for a given subscription list, using bulk processing.
IsAnsAccessible	Checks if a connection to the ANS Server can be established.
Ping	Pings the WCF web service by trying to establish a connection to the ANS Server. This will load the Client API DLLs, so the next subsequent calls to the service will be much faster. This method is to be used as the first call to the service, as a "warm up" for the WCF service (e.g. from within the Keep Alive (WAKA) service).
UpdateCodes	Edits (updates) existing ANS codes (e.g. categories) using bulk processing.

Method	Description
UpdateConditionDefinitionLists	Edits (updates) existing ANS condition definition lists using bulk processing.
UpdateConditionDefinitions	Edits (updates) existing ANS condition definitions using bulk processing.
UpdateListSections	Edits (updates) existing ANS list sections using bulk processing.
UpdateNotificationRoutings	Edits (updates) existing ANS notification routings using bulk processing.
UpdateSubscriptionLists	Edits (updates) existing ANS subscription lists using bulk processing.
UpdateSubscriptions	Edits (updates) existing ANS subscription using bulk processing.

AnsWs Web Service Methods

The following are detailed descriptions for each **`AnsWs`** Web Service method, grouped by functionality.

Method		Description
`Ping`		Pings the WCF web service by trying to establish a connection to the ANS Server. This will load the Client API DLLs, so the next subsequent calls to the service will be much faster. This method is to be used as the first call to the service, as a "warm up" for the WCF service (e.g. from within the Keep Alive (WAKA) service). `bool Ping(string ansHost)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
Return Value		Returns True if connections to the ANS Server have been established and false if the connections cannot be obtained.

Method		Description
`IsAnsAccessible`		Checks if a connection to the ANS Server can be established. `bool IsAnsAccessible(string ansHost)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
Return Value		Returns True if a connection to the ANS Server has been established and false if the connection cannot be obtained.

Method	Description
`GetNewConditionDef`	Utility method that creates a new, default initialized, ConditionDef object. It is recommended to call this method before calling CreateConditionDefinitions or UpdateConditionDefinitions methods. `public ConditionDef GetNewConditionDef()`
Return Value	Returns a new, default initialized, ConditionDef.

Method	Description
`GetNewSubscriptionList`	Utility method that creates a new, default initialized, Subscription object. It is recommended to call this method before calling CreateSubscriptions or UpdateSubscriptions methods. `public Subscription GetNewSubscription()`
Return Value	Returns a new, default initialized, Subscription.

Method	Description
`GetNewSubscription`	Utility method that creates new, default initialized, Subscription

Method		Description
		object. It is recommended to call this method before calling CreateSubscriptions or UpdateSubscriptions methods `public Subscription GetNewSubscription()`
Return Value		Returns a new, default initialized, Subscription.

Method		Description
`CreateConditionDefinitions`		Creates new ANS condition definitions using bulk processing. `Response[] CreateConditionDefinitions(string ansHost, ConditionDef[] defList)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	defList	The list of ConditionDef alerts to be created.
Return Value		Returns a bulk value response in the form of Response.

Method		Description
`UpdateConditionDefinitions`		Edits (updates) existing ANS condition definitions using bulk processing. `Response[] UpdateConditionDefinitions(string ansHost, ConditionDef[] defList)`
Parameters:	ansHost	The name of the machine that hosts the ANS.
	defList	The list of ConditionDef alerts to be updated.
Return Value		Returns a bulk value response in the form of Response.

Method		Description
`GetConditionDefinitionsById`		Retrieves a list of ANS condition definitions given their correspondent IDs using bulk processing. `ResponseEx<ConditionDef>[] GetConditionDefinitionsById(string ansHost, int[] defIdList)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	defIdList	The list of condition definition IDs to be retrieved.
Return Value		Returns a bulk response in the form of a ConditionDef list.

Method	Description
`GetConditionDefinitionsByName`	Retrieves a list of ANS condition definitions given their names using bulk processing.

Method		Description
		`ResponseEx<ConditionDef>[] GetConditionDefinitionsByName(string ansHost, string[] defNameList)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	defNameList	The list of condition definition names to be retrieved.
Return Value		Returns a bulk response in the form of a ConditionDef list.

Method		Description
`GetConditionDefinitionsByList`		Retrieves a list of all ANS condition definitions for a given definition list, using bulk processing. `ResponseEx<ConditionDef>[] GetConditionDefinitionsByList(string ansHost, string defListName)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	defNameList	The list of condition definition names to be retrieved.
Return Value		Returns a bulk response in the form of a ConditionDef list.

Method		Description
`DeleteConditionDefinitions`		Deletes existing ANS condition definitions using bulk processing. `SimpleResponse[] DeleteConditionDefinitions (string ansHost, int[] defIds)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	defIds	The list of condition definition IDs to be deleted
Return Value		Returns a bulk value response in the form of SimpleResponse.

Method		Description
`CreateConditionDefinitionLists`		Creates new ANS condition definition lists using bulk processing. `Response[] CreateConditionDefinitionLists(string ansHost, ConditionDefList[] defList)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	defList	The list of ConditionDefList lists to be created.
Return Value		Returns a bulk value response in the form of Response.

Method		Description
`UpdateConditionDefinitionLists`		Edits (updates) existing ANS condition definition lists using bulk processing. `Response[] UpdateConditionDefinitionLists(string ansHost, ConditionDefList[] defList)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	defList	The list of ConditionDefList lists to be updated.
Return Value		Returns a bulk value response in the form of Response.

Method		Description
`GetDefinitionListsById`		Retrieves a collection of ANS definition lists given their correspondent IDs using bulk processing. `ResponseEx<ConditionDefList>[] GetDefinitionListsById(string ansHost, int[] defListIds)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	defListIds	The array of definition list IDs to be retrieved.
Return Value		Returns a bulk response in the form of a ConditionDefList list.

Method		Description
`GetDefinitionListsByName`		Retrieves a collection of ANS definition lists given their names using bulk processing. `ResponseEx<ConditionDefList>[] GetDefinitionListsByName(string ansHost, string[] defListNames)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	defListNames	The array of definition list names to be retrieved
Return Value		Returns a bulk response in the form of a ConditionDefList list.

Method		Description
`DeleteConditionDefinitionLists`		Deletes existing ANS condition definition lists using bulk processing. `SimpleResponse[] DeleteConditionDefinitionLists (string ansHost, int[] listIds)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	listIds	Array of definition lists IDs.
Return Value		Returns a bulk value response in the form of SimpleResponse.

Method		Description
`CreateListSections`		Creates new ANS list sections using bulk processing. `Response[] CreateListSections(string ansHost, ListSection[] sectionList)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	sectionList	The list of ListSection sections to be created.
Return Value		Returns a bulk value response in the form of Response.

Method		Description
`UpdateListSections`		Updates existing ANS list sections using bulk processing. `Response[] UpdateListSections(string ansHost, ListSection[] sectionList)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	sectionList	The list of ListSection sections to be updated.
Return Value		Returns a bulk value response in the form of Response.

Method		Description
`GetListSectionsById`		Retrieves a collection of ANS definition list sections given their correspondent IDs and the definition list IDs, using bulk processing. `ResponseEx<ListSection>[] GetListSectionsById (string ansHost, int[] sectionIds, int[] listIds)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	sectionIds	The array of section IDs to be retrieved.
	listIds	The array of definition list IDs.
Return Value		Returns a bulk response in the form of a ListSection list.

Method		Description
`GetListSectionsByName`		Retrieves a collection of ANS definition list sections given their correspondent names and the definition list names, using bulk processing. `ResponseEx<ListSection>[] GetListSectionsByName (string ansHost, string[] sectionNames, string[] listNames)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.

Method		Description
	sectionNames	The array of section names to be retrieved.
	listNames	The array of definition list names.
Return Value		Returns a bulk response in the form of a ListSection list.

Method		Description
`GetListSectionsByList`		Retrieves all ANS sections for a given definition list, using bulk processing. `ResponseEx<ListSection>[] GetListSectionsByList (string ansHost, string defListName)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	defListName	The definition list name
Return Value		Returns a bulk response in the form of a ListSection list

Method		Description
`DeleteListSections`		Deletes existing ANS definition lists sections using bulk processing. `SimpleResponse[] DeleteListSections(string ansHost, int[] sectionIds)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	sectionIds	Array of definition list section IDs.
Return Value		Returns a bulk value response in the form of SimpleResponse.

Method		Description
`CreateSubscriptions`		Creates new ANS subscriptions using bulk processing. `Response[] CreateSubscriptions(string ansHost, Subscription[] subList)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	subscList	The list of Subscriptions to be created.
Return Value		Returns a bulk value response in the form of Response.

Method	Description
`UpdateSubscriptions`	Updates existing ANS subscriptions using bulk processing. `Response[] UpdateSubscriptions(string ansHost,`

Method		Description
		`Subscription[] subList)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	subscList	The list of Subscriptions to be updated.
Return Value		Returns a bulk value response in the form of Response

Method		Description
`GetSubscriptionsById`		Retrieves a collection of ANS condition subscriptions given their correspondent IDs using bulk processing. `ResponseEx<Subscription>[] GetSubscriptionsById (string ansHost, int[] subscIds)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	subscIds	The array of subscription IDs to be retrieved.
Return Value		Returns a bulk response in the form of a Subscription list.

Method		Description
`GetSubscriptionsByList`		Retrieves all ANS subscriptions for a given subscription list, using bulk processing. `ResponseEx<Subscription>[] GetSubscriptionsByList (string ansHost, string subscListName)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	subscListName	The Subscription List name.
Return Value		Returns a bulk response in the form of a Subscription list.

Method		Description
`DeleteSubscriptions`		Deletes existing ANS subscriptions using bulk processing. `SimpleResponse[] DeleteSubscriptions(string ansHost, int[] subsIds)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	subsIds	Array of subscription IDs.
Return Value		Returns a bulk value response in the form of SimpleResponse

Method	Description
`CreateSubscriptionLists`	Creates new ANS subscription lists using bulk processing. `Response[] CreateSubscriptionLists(string ansHost, SubscriptionList[] subscList)`

Method		Description
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	subscList	The list of Subscription Lists to be created.
Return Value		Returns a bulk value response in the form of Response

Method		Description
`UpdateSubscriptionLists`		Updates existing ANS subscription lists using bulk processing. `Response[] UpdateSubscriptionLists(string ansHost, SubscriptionList[] subscList)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	subscList	The list of Subscription Lists to be updated.
Return Value		Returns a bulk value response in the form of Response.

Method		Description
`GetSubscriptionListsById`		Retrieves a collection of ANS subscription lists given their correspondent IDs using bulk processing. `ResponseEx<SubscriptionList>[] GetSubscriptionListsById(string ansHost, int[] subsListIds)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	subsListIds	The array of subscription list IDs to be retrieved.
Return Value		Returns a bulk response in the form of a Subscription List.

Method		Description
`GetSubscriptionListsByName`		Retrieves a collection of ANS subscription lists given their names using bulk processing. `ResponseEx<SubscriptionList>[] GetSubscriptionListsByName(string ansHost, string [] subsListNames)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	subsListNames	
Return Value		Returns a bulk response in the form of a ConditionDefList list.

Method		Description
`DeleteSubscriptionLists`		Deletes existing ANS subscription lists using bulk processing. `SimpleResponse[] DeleteSubscriptionLists(string`

Method		Description
		`ansHost, int[] listIds)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	listIds	Array of subscription lists IDs.
Return Value		Returns a bulk value response in the form of SimpleResponse.

Method		Description
`CreateCodes`		Creates new ANS codes (e.g. categories) using bulk processing. `Response[] CreateCodes(string ansHost, Code[] codeList)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	codeList	The list of "Code" object codes to be created.
Return Value		Returns a bulk value response in the form of Response

Method		Description
`UpdateCodes`		Updates existing ANS codes (e.g. categories) using bulk processing. `Response[] UpdateCodes(string ansHost, Code[] codeList)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	codeList	The list of "Code" object codes to be updated
Return Value		Returns a bulk value response in the form of Response.

Method		Description
`GetCodesById`		Retrieves a collection of codes (e.g. categories) given their correspondent IDs using bulk processing. `ResponseEx<Code>[] GetCodesById(string ansHost, int[] codeIdList)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	codeIdList	The array of code IDs to be retrieved
Return Value		Returns a bulk response in the form of a Code list

Method	Description
`GetCodesByName`	Retrieves list of ANS codes (e.g. categories) given their correspondent names and code types, using bulk processing.

Method		Description
		`ResponseEx<Code>[] GetCodesByName(string ansHost, string[] codeNames, string[] codeTypes)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	codeNames	The array of code names to be retrieved.
	codeTypes	The array of code types names. A code name can only have two possible string values: "ANS_CONDITION_CATEGORY" or "INTRVL_TYPE". These values are NOT case sensitive.
Return Value		Returns a bulk response in the form of a Code list.

Method		Description
`DeleteCodes`		Deletes existing ANS codes using bulk processing. `SimpleResponse[] DeleteCodes(string ansHost, int [] codeIds)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	codeIds	Array of codes IDs
Return Value		Returns a bulk value response in the form of SimpleResponse.

Method		Description
`CreateNotificationRoutings`		Creates new ANS notification routings using bulk processing. `Response[] CreateNotificationRoutings(string ansHost, NotificationRouting[] routeList)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	routeList	The list of Notification Routings to be created.
Return Value		Returns a bulk value response in the form of Response

Method		Description
`UpdateNotificationRoutings`		Updates existing ANS notification routings using bulk processing. `Response[] UpdateNotificationRoutings(string ansHost, NotificationRouting[] routeList)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	routeList	The list of Notification Routings to be updated.
Return Value		Returns a bulk value response in the form of Response.

Method		Description
`GetNotificationRoutingsById`		Retrieves a collection of ANS notification routings given their correspondent IDs using bulk processing. `ResponseEx<NotificationRouting>[] GetNotificationRoutingsById(string ansHost, int[] routeIdList)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	routeIdList	The array of notification routing IDs to be retrieved.
Return Value		Returns a bulk response in the form of a Notification Routing list.

Method		Description
`GetMyNotificationRoutings`		Retrieves all ANS notification routings for the caller user, using bulk processing. `ResponseEx<NotificationRouting>[] GetMyNotificationRoutings(string ansHost)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
Return Value		Returns a bulk response in the form of a Notification Routing list.

Method		Description
`DeleteNotificationRoutings`		Deletes existing ANS notification routings using bulk processing. `SimpleResponse[] DeleteNotificationRoutings (string ansHost, int[] routeIds)`
Parameters:	*ansHost*	The name of the machine that hosts the ANS.
	routeIds	Array of route IDs.
Return Value		Returns a bulk value response in the form of SimpleResponse.

REST-compliant XHQ Web Services

In addition to the SOAP endpoint, the WCF XHQ Web Service, XhqWse, also supports REST calls that return both XML or POX (Plain Old XML), and JSON (JavaScript Object Notation).

JSON is used with JavaScript and AJAX (or mobile) calls.

The endpoint addresses for the XhqWse Web Service are as follows.

Endpoint Addresses

Endpoint Type	Address
SOAP	http://[XhqWse_HOST]/indx/xhqwse/XhqWse.svc
REST, returning POX (XML)	http://[XhqWse_HOST] //indx/xhqwse/XhqWse.svc/rest/pox
REST, returning JSON	http://[XhqWse_HOST] //indx/xhqwse/XhqWse.svc/rest/json

You will not have to modify the WS invocation client programs or regenerate the proxies used to invoke the Web Service operations through its SOAP endpoint.

REST Operations

This section provides syntax examples for supported REST (or RESTful) operations.

This first set of RESTful operations are "**`HTTP GET`**"-style operations that are supported by both POX/XML and JSON (with the exception of the `GenerateSolutionTreeNode` operation, which is only supported by POX).

HTTP GET-style of Operations

Ping	IsSolutionAccessible	GetPointCurrentData	GetPointAtValue
GetPointActualValue	GetPointAverageValue	GetPointCountValue	GetPointDeltaValue
GetPointFirstValue	GetPointLastValue	GetPointIntegralValue	GetPointMaximumValue
GetPointMinimumValue	GetPointSumValue	GetTimeSeriesData	GetCollectionData
GetCollectionDataWhere	GetCollectionDataOrderBy	GetCollectionDataFilter	GetCollectionDataXml
GetCollectionDataWhereXml	GetCollectionDataOrderByXml	GetCollectionDataFilterXml	GetCollectionDataJson
GetCollectionDataWhereJson	GetCollectionDataOrderByJson	GetCollectionDataFilterJson	GetCollectionColumnAverage
GetCollectionColumnCount	GetCollectionColumnMaximum	GetCollectionColumnMinimum	GetCollectionColumnSum
GetCollectionCount	GetGlobalCollections	GetSolutionRoot	GetChildPrimitiveMembers
GetChildComponentMembers	GetChildCollectionMembers	GetAllChildMembers	WritePointValue
CanReadMember	CanWriteMember	GetReadableMembers	GetWriteableMembers
GetUserRoleList	IsUserInRole	GetRolesWithReadAccess	GetRolesWithWriteAccess
GetRolesProvidingReadAccess	GetRolesProvidingWriteAccess	GenerateSolutionTreeNode	

This second set of RESTful operations have complex input parameters (for example, String arrays) and are therefore exposed through "**`HTTP POST`**" for both POX/XML and JSON

HTTP POST-style of Operations

GetPathListAccess	InsertCollectionData	UpdateCollectionData	ReplaceCollectionData
DeleteCollectionData	GetBulkCurrentValues	GetBulkAtValues	GetBulkActualValues
GetBulkAverageValues	GetBulkCountValues	GetBulkDeltaValues	GetBulkFirstValues
GetBulkLastValues	GetBulkIntegralValues	GetBulkMaximumValues	GetBulkMinimumValues
GetBulkSumValues			

See the *.NET client code example* for an HTTP POST request.

Ping

Type	Request
POX	http://[XhqWse_HOST]/indx/xhqwse/XhqWse.svc/rest/`pox`/Ping?solutionHost={solutionHost} *Example:* http://localhost/indx/xhqwse/XhqWse.svc/rest/pox/Ping?solutionHost=grinds
JSON	http://[XhqWse_HOST]/indx/xhqwse/XhqWse.svc/rest/`json`/Ping?solutionHost={solutionHost} *Example:* http://localhost/indx/xhqwse/XhqWse.svc/rest/json/Ping?solutionHost=grinds

IsSolutionAccessible

Type	Request
POX	http://[XhqWse_HOST]/indx/xhqwse/XhqWse.svc/rest/`pox`/IsSolutionAccessible?solutionHost={solutionHost} *Example:* http://localhost/indx/xhqwse/XhqWse.svc/rest/pox/IsSolutionAccessible?solutionHost=grinds
JSON	http://[XhqWse_HOST]/indx/xhqwse/XhqWse.svc/rest/`json`/IsSolutionAccessible?solutionHost={solutionHost} *Example:* http://localhost/indx/xhqwse/XhqWse.svc/rest/json/IsSolutionAccessible?solutionHost=grinds

GetPointCurrentData

The **units** and **timeZone** parameters are optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetPointCurrentData ?solutionHost={solutionHost}&path={path}&units={units}&timeZone={timeZone} *Example:* http://localhost/indx/xhqwse/XhqWse.svc/rest/pox/GetPointCurrentData ?solutionHost=grinds&path=sim&timeZone=Pacific Standard Time The POX (XML) response is similar to: <VTQ xmlns="http://schemas.datacontract.org/2004/07/Indx.Xhq.Wse" xmlns:i="http://www.w3.org/2001/XMLSchema-instance"> <Description>Tide level</Description> <Quality>GOOD</Quality> <Time>2012-08-09 13:06:30.396</Time> <Units>kg</Units> <Value>7.97752</Value> </VTQ>
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetPointCurrentData ?solutionHost={solutionHost}&path={path}&units={units}&timeZone={timeZone} *Example:* http://localhost/indx/xhqwse/XhqWse.svc/rest/json/GetPointCurrentData ?solutionHost=grinds&path=sim&timeZone=Pacific Standard Time The JSON response is similar to: {"d":{"__type":"VTQ:#Indx.Xhq.Wse","Description":"Tide level","Quality":"GOOD","Time":"2012-08-09 13:08:00.396","Units":"kg","Value":"3.7160227"}}

GetPointAtValue

The **units** and **timeZone** parameters are optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetPointAtValue ?solutionHost={solutionHost}&path={path}&atTime={atTime}&units={units}&timeZone={timeZone}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetPointAtValue ?solutionHost={solutionHost}&path={path}&atTime={atTime}&units={units}&timeZone={timeZone}

GetPointActualValue

The **units** and **timeZone** parameters are optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetPointActualValue ?solutionHost={solutionHost}&path={path}& actualTime={ actualTime}&units={units}&timeZone= {timeZone}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetPointActualValue ?solutionHost={solutionHost}&path={path}& actualTime={ actualTime}&units={units}&timeZone= {timeZone}

GetPointAverageValue

The **units** and **timeZone** parameters are optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetPointAverageValue ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &units={units}&timeZone={timeZone}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetPointAverageValue ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &units={units}&timeZone={timeZone}

GetPointCountValue

The **units** and **timeZone** parameters are optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetPointCountValue ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &units={units}&timeZone={timeZone}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetPointCountValue ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &units={units}&timeZone={timeZone}

GetPointDeltaValue

The **units** and **timeZone** parameters are optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetPointDeltaValue ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &units={units}&timeZone={timeZone}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetPointDeltaValue ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &units={units}&timeZone={timeZone}

GetPointFirstValue

The **units** and **timeZone** parameters are optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetPointFirstValue ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &units={units}&timeZone={timeZone}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetPointFirstValue ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &units={units}&timeZone={timeZone}

GetPointLastValue

The **units** and **timeZone** parameters are optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetPointLastValue ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &units={units}&timeZone={timeZone}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetPointLastValue ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &units={units}&timeZone={timeZone}

GetPointIntegralValue

The **units** and **timeZone** parameters are optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetPointIntegralValue ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &units={units}&timeZone={timeZone}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetPointIntegralValue ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &units={units}&timeZone={timeZone}

GetPointMaximumValue

The **units** and **timeZone** parameters are optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetPointMaximumValue ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &units={units}&timeZone={timeZone}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetPointMaximumValue ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &units={units}&timeZone={timeZone}

GetPointMinimumValue

The **units** and **timeZone** parameters are optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetPointMinimumValue ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &units={units}&timeZone={timeZone}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetPointMinimumValue ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &units={units}&timeZone={timeZone}

GetPointSumValue

The **units** and **timeZone** parameters are optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetPointSumValue ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime}

Type	Request
	&units={units}&timeZone={timeZone}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetPointSumValue ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &units={units}&timeZone={timeZone}

GetTimeSeriesData

The **units** and **timeZone** parameters are optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetTimeSeriesData ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &timeZone={timeZone}&samplingMode={samplingMode}&noPoints={noPoints}&units={units}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetTimeSeriesData ?solutionHost={solutionHost}&path={path}&startTime={startTime}&endTime={endTime} &timeZone={timeZone}&samplingMode={samplingMode}&noPoints={noPoints}&units={units}

GetCollectionData

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetCollectionData ?solutionHost={solutionHost}&path={path}&columns={`comma separated list of column names`} &rowsCount={rowsCount}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetCollectionData ?solutionHost={solutionHost}&path={path}&columns={`comma separated list of column names`} &rowsCount={rowsCount}

GetCollectionDataWhere

The **whereClause** parameter is optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetCollectionDataWhere ?solutionHost={solutionHost}&path={path}&columns={`comma separated list of column names`} &whereClause={whereClause}&rowsCount={rowsCount}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetCollectionDataWhere ?solutionHost={solutionHost}&path={path}&columns={`comma separated list of column names`} &whereClause={whereClause}&rowsCount={rowsCount}

GetCollectionDataOrderBy

The **orderByClause** parameter is optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetCollectionDataOrderBy ?solutionHost={solutionHost}&path={path}&columns={`comma separated list of column names`} & orderByClause={orderByClause}&rowsCount={rowsCount}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetCollectionDataOrderBy ?solutionHost={solutionHost}&path={path}&columns={`comma separated list of column names`} &orderByClause={orderByClause}&rowsCount={rowsCount}

GetCollectionDataFilter

The **whereClause** and **orderByClause** parameters are optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetCollectionDataFilter ?solutionHost={solutionHost}&path={path}&columns={`comma separated list of column names`} &whereClause={whereClause}&orderByClause={orderByClause}&rowsCount={rowsCount}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetCollectionDataFilter ?solutionHost={solutionHost}&path={path}&columns={`comma separated list of column names`} &whereClause={whereClause}&orderByClause={orderByClause}&rowsCount={rowsCount}

GetCollectionDataXml

The **Path** parameter is the name of the XHQ Global collection or the absolute path for a Link collection.

The method can also be invoked using the XhqWse SOAP endpoint and it will always return pure XML (POX).

Type	Request
XML POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetCollectionDataXml ?solutionHost={solutionHost}&path={path}&columns={`comma separated list of column names, or * for all columns`} &rowsCount={rowsCount}

GetCollectionDataJson

The **Path** parameter is the name of the XHQ Global collection or the absolute path for a Link collection.

Type	Request
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetCollectionDataJson ?solutionHost={solutionHost}&path={path}&columns={`comma separated list of column names, or * for all columns`} &rowsCount={rowsCount}

GetCollectionDataWhereXml

The **whereClause** parameter is optional and can be missing from the query string.

The **Path** parameter is the name of the XHQ Global collection or the absolute path for a Link collection.

The method can also be invoked using the XhqWse SOAP endpoint and it will always return pure XML (POX).

Type	Request
XML POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetCollectionDataWhereXml ?solutionHost={solutionHost}&path={path}&columns={`comma separated list of column names, or * for all columns`} &whereClause={whereClause}&rowsCount={rowsCount}

GetCollectionDataWhereJson

The **whereClause** parameter is optional and can be missing from the query string.

The **Path** parameter is the name of the XHQ Global collection or the absolute path for a Link collection.

Type	Request
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetCollectionDataWhereJson ?solutionHost={solutionHost}&path={path}&columns={`comma separated list of column names, or * for all columns`} &whereClause={whereClause}&rowsCount={rowsCount}

GetCollectionDataOrderByXml

The **orderByClause** parameter is optional and can be missing from the query string.

The **Path** parameter is the name of the XHQ Global collection or the absolute path for a Link collection.

The method can also be invoked using the XhqWse SOAP endpoint and it will always return pure XML (POX).

Type	Request
XML POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetCollectionDataOrderByXml ?solutionHost={solutionHost}&path={path}&columns={`comma separated list of column names, or * for all columns`} &orderByClause={orderByClause}&rowsCount={rowsCount}

GetCollectionDataOrderByJson

The **orderByClause** parameter is optional and can be missing from the query string.

The **Path** parameter is the name of the XHQ Global collection or the absolute path for a Link collection.

Type	Request
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetCollectionDataOrderByJson ?solutionHost={solutionHost}&path={path}&columns={`comma separated list of column names, or * for all columns`} &orderByClause={orderByClause}&rowsCount={rowsCount}

GetCollectionDataFilterXml

The **whereClause** and **orderByClause** parameters are optional and can be missing from the query string.

The **Path** parameter is the name of the XHQ Global collection or the absolute path for a Link collection.

The method can also be invoked using the XhqWse SOAP endpoint and it will always return pure XML (POX).

Type	Request
XML POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetCollectionDataFilterXml ?solutionHost={solutionHost}&path={path}&columns={`comma separated list of column names, or * for all columns`} &whereClause={whereClause}&orderByClause={orderByClause}&rowsCount={rowsCount}

GetCollectionDataFilterJson

The **whereClause** and **orderByClause** parameters are optional and can be missing from the query string.

The **Path** parameter is the name of the XHQ Global collection or the absolute path for a Link collection.

Type	Request
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetCollectionDataFilterJson ?solutionHost={solutionHost}&path={path}&columns={`comma separated list of column names, or * for all columns`} &whereClause={whereClause}&orderByClause={orderByClause}&rowsCount={rowsCount}

GetCollectionColumnAverage

The **whereClause** parameter is optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetCollectionColumnAverage ?solutionHost={solutionHost}&path={path}&column={column}&whereClause={whereClause}

Type	Request
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetCollectionColumnAverage ?solutionHost={solutionHost}&path={path}&column={column}&whereClause={whereClause}

GetCollectionColumnCount

The **whereClause** parameter is optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetCollectionColumnCount ?solutionHost={solutionHost}&path={path}&column={column}&whereClause={whereClause}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetCollectionColumnCount ?solutionHost={solutionHost}&path={path}&column={column}&whereClause={whereClause}

GetCollectionColumnMaximum

The **whereClause** parameter is optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetCollectionColumnMaximum ?solutionHost={solutionHost}&path={path}&column={column}&whereClause={whereClause}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetCollectionColumnMaximum ?solutionHost={solutionHost}&path={path}&column={column}&whereClause={whereClause}

GetCollectionColumnMinimum

The **whereClause** parameter is optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetCollectionColumnMinimum ?solutionHost={solutionHost}&path={path}&column={column}&whereClause={whereClause}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetCollectionColumnMinimum ?solutionHost={solutionHost}&path={path}&column={column}&whereClause={whereClause}

GetCollectionColumnSum

The **whereClause** parameter is optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetCollectionColumnSum ?solutionHost={solutionHost}&path={path}&column={column}&whereClause={whereClause}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetCollectionColumnSum ?solutionHost={solutionHost}&path={path}&column={column}&whereClause={whereClause}

GetCollectionCount

The **whereClause** parameter is optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetCollectionCount ?solutionHost={solutionHost}&path={path}&whereClause={whereClause}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetCollectionCount ?solutionHost={solutionHost}&path={path}&whereClause={whereClause}

GetGlobalCollections

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetGlobalCollections ?solutionHost={solutionHost}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetGlobalCollections ?solutionHost={solutionHost}

GetSolutionRoot

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetSolutionRoot ?solutionHost={solutionHost}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetSolutionRoot ?solutionHost={solutionHost}

GetChildPrimitiveMembers

The **showHidden** parameter is optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetChildPrimitiveMembers ?solutionHost={solutionHost}&parentPath={parentPath}&showHidden={showHidden}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetChildPrimitiveMembers ?solutionHost={solutionHost}&parentPath={parentPath}&showHidden={showHidden}

GetChildComponentMembers

The **showHidden** parameter is optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetChildComponentMembers ?solutionHost={solutionHost}&parentPath={parentPath}&showHidden={showHidden}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetChildComponentMembers ?solutionHost={solutionHost}&parentPath={parentPath}&showHidden={showHidden}

GetChildCollectionMembers

The **showHidden** parameter is optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetChildCollectionMembers ?solutionHost={solutionHost}&parentPath={parentPath}&showHidden={showHidden}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetChildCollectionMembers ?solutionHost={solutionHost}&parentPath={parentPath}&showHidden={showHidden}

GetAllChildMembers

The **showHidden** parameter is optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetAllChildMembers ?solutionHost={solutionHost}&parentPath={parentPath}&showHidden={showHidden}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetAllChildMembers ?solutionHost={solutionHost}&parentPath={parentPath}&showHidden={showHidden}

WritePointValue

The **timeZone** parameter is optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/WritePointValue ?solutionHost={solutionHost}&path={path}&value={value}&time={time} &quality={quality}&timeZone={timeZone}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/ WritePointValue ?solutionHost={solutionHost}&path={path}&value={value}&time={time} &quality={quality}&timeZone={timeZone}

CanReadMember

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/CanReadMember ?solutionHost={solutionHost}&path={path}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/CanReadMember ?solutionHost={solutionHost}&path={path}

CanWriteMember

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/CanWriteMember ?solutionHost={solutionHost}&path={path}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/CanWriteMember ?solutionHost={solutionHost}&path={path}

GetReadableMembers

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetReadableMembers ?solutionHost={solutionHost}&path={path}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetReadableMembers ?solutionHost={solutionHost}&path={path}

GetWriteableMembers

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetWriteableMembers ?solutionHost={solutionHost}&path={path}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetWriteableMembers ?solutionHost={solutionHost}&path={path}

GetUserRoleList

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetUserRoleList ?solutionHost={solutionHost}&username={username}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetUserRoleList ?solutionHost={solutionHost}&username={username}

IsUserInRole

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/IsUserInRole ?solutionHost={solutionHost}&username={username}&roleName={roleName}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/IsUserInRole ?solutionHost={solutionHost}&username={username}&roleName={roleName}

GetRolesWithReadAccess

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetRolesWithReadAccess ?solutionHost={solutionHost}&path={path}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetRolesWithReadAccess ?solutionHost={solutionHost}&path={path}

GetRolesWithWriteAccess

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetRolesWithWriteAccess ?solutionHost={solutionHost}&path={path}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetRolesWithWriteAccess

Type	Request
	?solutionHost={solutionHost}&path={path}

GetRolesProvidingReadAccess

The **username** parameter is optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetRolesProvidingReadAccess ?solutionHost={solutionHost}&path={path}&username={username}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetRolesProvidingReadAccess ?solutionHost={solutionHost}&path={path}&username={username}

GetRolesProvidingWriteAccess

The **username** parameter is optional and can be missing from the query string.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GetRolesProvidingWriteAccess ?solutionHost={solutionHost}&path={path}&username={username}
JSON	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/json/GetRolesProvidingWriteAccess ?solutionHost={solutionHost}&path={path}&username={username}

GenerateSolutionTreeNode

There is no JSON request for this operation.

Type	Request
POX	http://[XhqWse_HOST]//indx/xhqwse/XhqWse.svc/rest/pox/GenerateSolutionTreeNode ?solutionHost={solutionHost}&path={path}

GetPathListAccess

Syntax: PathAccess[] GetPathListAccess(string solutionHost, string[] paths, string username);

InsertCollectionData

Syntax: bool InsertCollectionData(string solutionHost, string collection, string[] columns, string[][] data);

UpdateCollectionData

Syntax: bool UpdateCollectionData(string solutionHost, string collection, string[] columns, string[][] data);

ReplaceCollectionData

Syntax: bool ReplaceCollectionData(string solutionHost, string collection, string[] columns, string[][] data);

DeleteCollectionData

Syntax: bool DeleteCollectionData(string solutionHost, string collection, string[] columns, string[][] data);

GetBulkCurrentValues

Syntax: VTQ[] GetBulkCurrentValues(string solutionHost, string[] paths, string units, string timeZone);

GetBulkAtValues

Syntax: VTQ[] GetBulkAtValues(string solutionHost, string[] paths, string atTime, string units, string timeZone);

GetBulkActualValues

Syntax: VTQ[] GetBulkActualValues(string solutionHost, string[] paths, string actualTime, string units, string timeZone);

GetBulkAverageValues

Syntax: VTQ[] GetBulkAverageValues(string solutionHost, string[] paths, string startTime, string endTime, string units, string timeZone);

GetBulkCountValues

Syntax: VTQ[] GetBulkCountValues(string solutionHost, string[] paths, string startTime, string endTime, string units, string timeZone);

GetBulkDeltaValues

Syntax: VTQ[] GetBulkDeltaValues(string solutionHost, string[] paths, string startTime, string endTime, string units, string timeZone);

GetBulkFirstValues

Syntax: VTQ[] GetBulkFirstValues(string solutionHost, string[] paths, string startTime, string endTime, string units, string timeZone);

GetBulkLastValues

Syntax: VTQ[] GetBulkLastValues(string solutionHost, string[] paths, string startTime, string endTime, string units, string timeZone);

GetBulkIntegralValues

Syntax: VTQ[] GetBulkIntegralValues(string solutionHost, string[] paths, string startTime, string endTime, string units, string timeZone);

GetBulkMaximumValues

Syntax: VTQ[] GetBulkMaximumValues(string solutionHost, string[] paths, string startTime, string endTime, string units, string timeZone);

GetBulkMinimumValues

Syntax: VTQ[] GetBulkMinimumValues(string solutionHost, string[] paths, string startTime, string endTime, string units, string timeZone);

GetBulkSumValues

Syntax: VTQ[] GetBulkSumValues(string solutionHost, string[] paths, string startTime, string endTime, string units, string timeZone);

EXAMPLE FOR AN HTTP POST REQUEST

The following is a .NET client code example for an HTTP POST request for the **`GetBulkCurrentValues`** operation.

```
private void postInvoke ()
{
string uri = "http://localhost/indx/xhqwse/XhqWse.svc/rest/pox/";
//string uri = "http://localhost/indx/xhqwse/XhqWse.svc/rest/json/";
WebServiceHost host = new WebServiceHost(typeof(XhqWse), new Uri(uri));
WebHttpBinding binding = new WebHttpBinding();
binding.Security.Mode = WebHttpSecurityMode.TransportCredentialOnly;
binding.Security.Transport.ClientCredentialType = HttpClientCredentialType.Windows;
WebChannelFactory<IXhqService> factory = new WebChannelFactory<IXhqService>(binding, new
Uri(uri));
```

```
factory.Credentials.Windows.AllowedImpersonationLevel =
System.Security.Principal.TokenImpersonationLevel.Delegation;
IXhqService proxy = factory.CreateChannel();
proxy.GetBulkCurrentValues("grinds", new string[] { "sim1", "sim2" }, null, null);
((IClientChannel)proxy).Close();
factory.Close();
}
```

The HTTP POST request to the POX is similar to the following:

```
POST http://grinds/indx/xhqwse/XhqWse.svc/rest/pox/GetBulkCurrentValues HTTP/1.1
Content-Type: application/xml; charset=utf-8
Host: grinds
Content-Length: 431
Expect: 100-continue
<GetBulkCurrentValues
xmlns="http://webservices.xhq.siemens.com"><solutionHost>grinds</solutionHost><paths
xmlns:a="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
xmlns:i="http://www.w3.org/2001/XMLSchema-
instance"><a:string>sim1</a:string><a:string>sim2</a:string></paths><units i:nil="true"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance"/><timeZone i:nil="true"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance"/>
</GetBulkCurrentValues>
```

The POX (XML) response is similar to the following:

```
<ArrayOfVTQ xmlns="http://schemas.datacontract.org/2004/07/Indx.Xhq.Wse"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance"><VTQ><Description>Tide
level</Description><Quality>GOOD</Quality><Time>2012-08-10
22:24:09.396</Time><Units>kg</Units><Value>7.029199</Value></VTQ><VTQ><Description>Tide
level</Description><Quality>GOOD</Quality><Time>2012-08-10
22:24:09.396</Time><Units>kg</Units><Value>9.059199</Value></VTQ>
</ArrayOfVTQ>
```

For the JSON endpoint, an HTTP POST request is similar to the following:

```
POST http://grinds/indx/xhqwse/XhqWse.svc/rest/json/GetBulkCurrentValues HTTP/1.1
Content-Type: application/xml; charset=utf-8
Host: grinds
Content-Length: 431
Expect: 100-continue
<GetBulkCurrentValues
xmlns="http://webservices.xhq.siemens.com"><solutionHost>grinds</solutionHost><paths
xmlns:a="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
xmlns:i="http://www.w3.org/2001/XMLSchema-
instance"><a:string>sim1</a:string>"><a:string>sim2</a:string></paths><units i:nil="true"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance"/><timeZone i:nil="true"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance"/>
</GetBulkCurrentValues>
```

The JSON response is similar to the following:

```
{"d":[{"__type":"VTQ:#Indx.Xhq.Wse","Description":"Tide
level","Quality":"GOOD","Time":"2012-08-10
22:26:00.396","Units":"kg","Value":"0.57453954"},{"__
type":"VTQ:#Indx.Xhq.Wse","Description":"Tide level","Quality":"GOOD","Time":"2012-08-10
22:26:00.396","Units":"kg","Value":"3.7845456"}]}.
```

Testing the Web Service

The XHQ Web Service has a test client/consumer web application called **`TestXhqWse`** that serves two purposes:

- It allows the complete testing of all the Web Service operations;
- It provides examples and sample code that shows how to properly implement client consumers for the Web Service and invoke its web methods.

The XHQ Web Service test client has been implemented in .NET as a web application using ASP.NET pages to consume the XHQ WCF service. Each ASP.NET page corresponds to a Web Service method. The pages are grouped in folders corresponding to the Web Service functional blocks:

- Point values
- Bulk values
- Time-series
- Collection data
- XHQ metadata
- XHQ solution connection

The test client application must be installed on the same machine where the XHQ Web Service is installed.

Retreiving Large Collection Data

When using `TestXhqWse` to retrieve large collection data, the following error may occur:

```
The maximum message size quota for incoming messages (65536) has been exceeded. To
increase the quota, use the MaxReceivedMessageSize property on the appropriate binding
element.
```

In this case, do the following.

To increase the maximum message size quota

1. Go to the location specified by the environment variable **`%XHQ_WEB_HOME%\webapps\testxhqwse`** (which is by default is `<systemdrive>:\Program Files\XHQ\XHQ Web Root\webapps\testxhqwse`).
2. Open **`web.config`** in a text editor.
3. Find the following code fragment.

   Line numbers are displayed to provide clarity.

```
109     <basicHttpBinding>
110      <binding name="BasicHttpBinding_XHQ Web Service" closeTimeout="00:01:00"
111       openTimeout="00:01:00" receiveTimeout="00:10:00" sendTimeout="00:01:00"
112       allowCookies="false" bypassProxyOnLocal="false" hostNameComparisonMode="StrongWildcard"
113       maxBufferSize="65536" maxBufferPoolSize="524288" maxReceivedMessageSize="65536"
114       messageEncoding="Text" textEncoding="utf-8" transferMode="Buffered"
115       useDefaultWebProxy="true">
116       <readerQuotas maxDepth="32" maxStringContentLength="8192" maxArrayLength="16384"
117        maxBytesPerRead="4096" maxNameTableCharCount="16384" />
118       <security mode="TransportCredentialOnly">
119         <transport clientCredentialType="Windows" proxyCredentialType="None" realm=""/>
120         <message clientCredentialType="UserName" algorithmSuite="Default"/>
121       </security>
122      </binding>
123     </basicHttpBinding>
124    </bindings>
```

4. Locate **`maxBufferSize`** and replace the default value 65536 (bytes) with a value (in bytes) that is suitable to your needs (based on the size of the data returned by the Web Service).
5. Locate **`maxReceivedMessageSize`** and replace the default value 65536 bytes with the **same value** you entered for the `maxBufferSize` in Step 4.

   The maxBufferSize and maxReceivedMessageSize settings must be set to the same value.

6. **Save** `web.config`.

Configuring SSL (HTTPS) Access

SSL Access for the XhqWse Web Service

By default, XHQ installs the XhqWse web service with only HTTP enabled. The HTTP configuration for the WCF service can be found in the `web.config` file.

The code example below is a section from the `web.config` file that contains the HTTP (port 80) configuration. The highlighted text shows where changes need to be made to allow for SSL (HTTPS).

```
<system.serviceModel>
    <bindings>
      <basicHttpBinding>
        <binding name="XhqWseBinding" maxReceivedMessageSize="4194304">
          <security mode="TransportCredentialOnly">
            <transport clientCredentialType="Windows"/>
          </security>
          <readerQuotas
             maxArrayLength="4194304"
        />
        </binding>
      </basicHttpBinding>
</bindings>
    <services>
      <service behaviorConfiguration="Indx.Xhq.Wse.Service1Behavior"
name="Indx.Xhq.Wse.XhqWse">
        <endpoint address="" binding="basicHttpBinding"
bindingConfiguration="XhqWseBinding" contract="Indx.Xhq.Wse.IXhqService">
          <identity>
            <dns value="localhost"/>
          </identity>
        </endpoint>
      </service>
    </services>
    <behaviors>
      <serviceBehaviors>
        <behavior name="Indx.Xhq.Wse.Service1Behavior">
          <!-- To avoid disclosing metadata information, set the value below to false and
remove the metadata endpoint above before deployment -->
          <serviceMetadata httpGetEnabled="true"/>
          <!-- To receive exception details in faults for debugging purposes, set the
value below to true.  Set to false before deployment to avoid disclosing exception
information -->
          <serviceDebug includeExceptionDetailInFaults="true"/>
        </behavior>
      </serviceBehaviors>
    </behaviors>
  </system.serviceModel>
</configuration>
```

To enable the XhqWse web service to work with SSL (HTTPS)

1. Open the **`web.config`** file for the XhqWse Web Service, located in the **`%XHQ_WEB_HOME%\webapps\xhqwse`** directory (which by default is `<systemdrive>:\Program Files (x86)\XHQ\XHQ Web Root\webapps\xhqwse`).
2. Find the **security mode tag** and change the value to **`Transport`**.

```
<security mode="Transport">
```

3. Find the **endpoint address tag** and set the value using the FQDN for the server that hosts the WCF service.

```
<endpoint address="https://<MACHINE_FQDN>:8887/indx/xhqwse/XhqWse.svc"
```

4. OPTIONAL
   Find the **serviceMetadata tag** and make it accessible through HTTPS by adding the attribute `httpsGetEnabled="true"`.

```
<serviceMetadata httpGetEnabled="false" httpsGetEnabled="true"/>
```

This results in the following HTTPS configuration:

```
<system.serviceModel>
    <bindings>
      <basicHttpBinding>
        <binding name="XhqWseBinding" maxReceivedMessageSize="4194304">
          <security mode="Transport">
            <transport clientCredentialType="Windows"/>
          </security>
          <readerQuotas
             maxArrayLength="4194304"
        />
        </binding>
      </basicHttpBinding>
</bindings>
    <services>
      <service behaviorConfiguration="Indx.Xhq.Wse.Service1Behavior"
name="Indx.Xhq.Wse.XhqWse">
        <endpoint address="https://corp.acme.com:8887/indx/xhqwse/XhqWse.svc"
binding="basicHttpBinding" bindingConfiguration="XhqWseBinding"
contract="Indx.Xhq.Wse.IXhqService">
          <identity>
            <dns value="localhost"/>
          </identity>
        </endpoint>
      </service>
    </services>
    <behaviors>
      <serviceBehaviors>
        <behavior name="Indx.Xhq.Wse.Service1Behavior">
          <!-- To avoid disclosing metadata information, set the value below to false
and remove the metadata endpoint above before deployment -->
          <serviceMetadata httpGetEnabled="false" httpsGetEnabled="true"/>
          <!-- To receive exception details in faults for debugging purposes, set the
value below to true.  Set to false before deployment to avoid disclosing exception
information -->
          <serviceDebug includeExceptionDetailInFaults="true"/>
        </behavior>
      </serviceBehaviors>
    </behaviors>
  </system.serviceModel>
</configuration>
```

The XhqWse clients, including the Microsoft WCFTestClient utility, will also need to use the security mode Transport and the new endpoint address in their given configuration files.

For example:

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<configuration>
    <system.serviceModel>
        <bindings>
            <basicHttpBinding>
                <binding name="BasicHttpBinding_XHQ Web Service"
closeTimeout="00:01:00"
                    openTimeout="00:01:00" receiveTimeout="00:10:00"
sendTimeout="00:01:00"
                    allowCookies="false" bypassProxyOnLocal="false"
hostNameComparisonMode="StrongWildcard"
                    maxBufferSize="65536" maxBufferPoolSize="524288"
maxReceivedMessageSize="65536"
                    messageEncoding="Text" textEncoding="utf-8"
transferMode="Buffered"
                    useDefaultWebProxy="true">
                    <readerQuotas maxDepth="32"
maxStringContentLength="8192" maxArrayLength="16384"
                        maxBytesPerRead="4096" maxNameTableCharCount="16384"
/>
                    <security mode="Transport">
                        <transport clientCredentialType="Windows"
proxyCredentialType="None"
                            realm="" />
                        <message clientCredentialType="UserName"
algorithmSuite="Default" />
                    </security>
                </binding>
            </basicHttpBinding>
        </bindings>
        <client>
            <endpoint
address="https://corp.acme.com:8887/indx/xhqwse/XhqWse.svc"
                binding="basicHttpBinding"
bindingConfiguration="BasicHttpBinding_XHQ Web Service"
                contract="XHQWebService" name="BasicHttpBinding_XHQ Web
Service" />
        </client>
    </system.serviceModel>
</configuration>
```

The TestXhqwse sample web application only works with HTTP and, by default, is not configured to support SSL.

Web References

The following link shows how to configure SSL for IIS 7.0:

http://learn.iis.net/page.aspx/144/how-to-set-up-ssl-on-iis-7/

The following links shows how to configure and use self generated test certificates for client testing the SSL support if a real certificate, issued by a trusted authority (for example, VeriSign), is not available:

http://www.codeproject.com/KB/WCF/WCFSSL.aspx

http://msdn.microsoft.com/en-us/library/ff648498.aspx

SSL Access for the XHQ Applet and Web Applications

As with the `XhqWse` web service, XHQ installs by default the XHQ Applet and web applications (XHQ Alert Notification System (XHQ ANS), Target Management (TM), eLogs, and View Stats) with only HTTP (port 80) enabled. In order to enable SSL (HTTPS on port 443) for the XHQ Applet and these web applications, you must first configure the SSL certificate for the `indx` virtual directory (`https://localhost/indx`).

To configure the SSL certificate for the indx virtual directory

1. Open the **Internet Information Services (IIS) Manager**.
2. Navigate to the **`indx` virtual directory**.
3. Under **IIS**, click **SSL Settings**.
   The "SSL Settings" panel appears.
4. In the SSL Settings panel, check **Require SSL**.
   This applies the SSL settings to all XHQ applications (which include the XHQ Applet, ANS, View Stats, and the Performance Management (PM) web applications).
5. Update the View Stats, PM Administration, TM, and eLogs links for HTTPS (SSL) for the Web Application Keep Alive (WAKA) service in the **`XHQWebAppKeepAlive.xml`** file.

   See the topic, *HTTPS SSL and HTTP Links for WAKA*, for a View Stats WAKA configuration example.

HTTPS (SSL) and HTTP Links for WAKA

The link for HTTPS (SSL) for the Web Application Keep Alive (WAKA) service in the **`XHQWebAppKeepAlive.xml`** file is as follows.

For XhqWse:

```
<Site name="XhqWse">
    <Settings>
      <Setting name="FastInterval" value="2" />
      <Setting name="Interval" value="10" />
      <Setting name="Debug" value="true" />
    </Settings>
    <Paths>
<Path name="XhqWse Service"
value="https://localhost/indx/xhqwse/Default.aspx?useSSL=true" />
    </Paths>
</Site>
```

Depending on how the SSL certificate has been issued (for which machine name, and so forth), localhost may need to be replaced with the FQDN.

The link for HTTP for WAKA in the **`XHQWebAppKeepAlive.xml`** file is:

```
<Site name="XhqWse">
    <Settings>
      <Setting name="FastInterval" value="2" />
      <Setting name="Interval" value="10" />
      <Setting name="Debug" value="true" />
    </Settings>
    <Paths>
<Path name="XhqWse Service"      value="http://localhost/indx/xhqwse/Default.aspx" />
    </Paths>
</Site>
```

For ViewStats:

```
<Site name="ViewStats">
    <Settings>
      <Setting name="FastInterval" value="2" />
      <Setting name="Interval" value="20" />
      <Setting name="Debug" value="true" />
    </Settings>
    <Paths>
      <Path name="ViewStats Home Page"
value="https://localhost/indx/viewstats/Default.aspx" />
      <Path name="ViewStats Main Page"
value="https://localhost/indx/viewstats/MainPage.aspx" />
    </Paths>
</Site>
```

Depending on how the SSL certificate has been issued (for which machine name, and so forth), `localhost` may need to be replaced with the FQDN.

3 | XHQ Web Interface

The XHQ Web Interface is a RESTful web API service that is mainly designed for low bandwidth clients (such as mobile applications using 3G/4G and WiFi to connect), WAN, and high latency LAN clients. The data returned by the XHQ Web Interface is light and is in either JSON or POX (Plain Old XML) format.

Using the XHQ Web Interface is the recommended method of developing mobile iOS and Android applications that access XHQ data.

The XHQ Web Interface, **`xhqwi`**, supports the following XHQ Web Interface modules

- XHQ Admin
- XHQ ANS
- XHQ Core
- XHQ eLogs
- XHQ RDS
- XHQ Target Management

Objects persisted and modified using the **`xhqwi`** interface will have the "WI_" prefix in the Audit tables and views of the XHQ Core instance database.

The XHQ Web Interface, requires HTTP or HTTPS access and has the following base URL address:

```
http://<XHQ Web Interface Webapp Hostname>/indx/xhqwi/default.aspx
```

Example:

```
http://comp1/indx/xhqwi/default.aspx
```

Accessing Online Help

The **customized online help** (detailed) specifically for this service can be accessed at this URL:

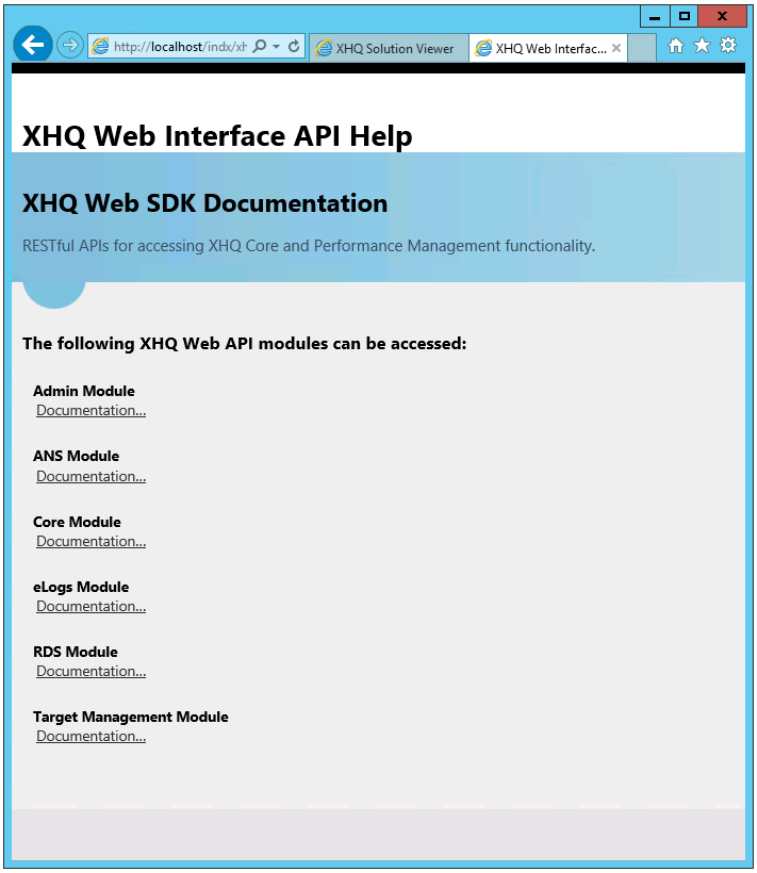
```
http://<XHQ Web Interface Webapp Hostname>/indx/xhqwi/help
```

Example 1:

```
http://comp1/indx/xhqwi/help
```

Example 2:

```
http://localhost/indx/xhqwi/help
```



Customized Online Help

In addition to the customized online help for the XHQ Web Interface service, you can also access the **Swagger online help**, which is an alternative way to access online documentation for this service. The Swagger online help can be accessed at this URL:

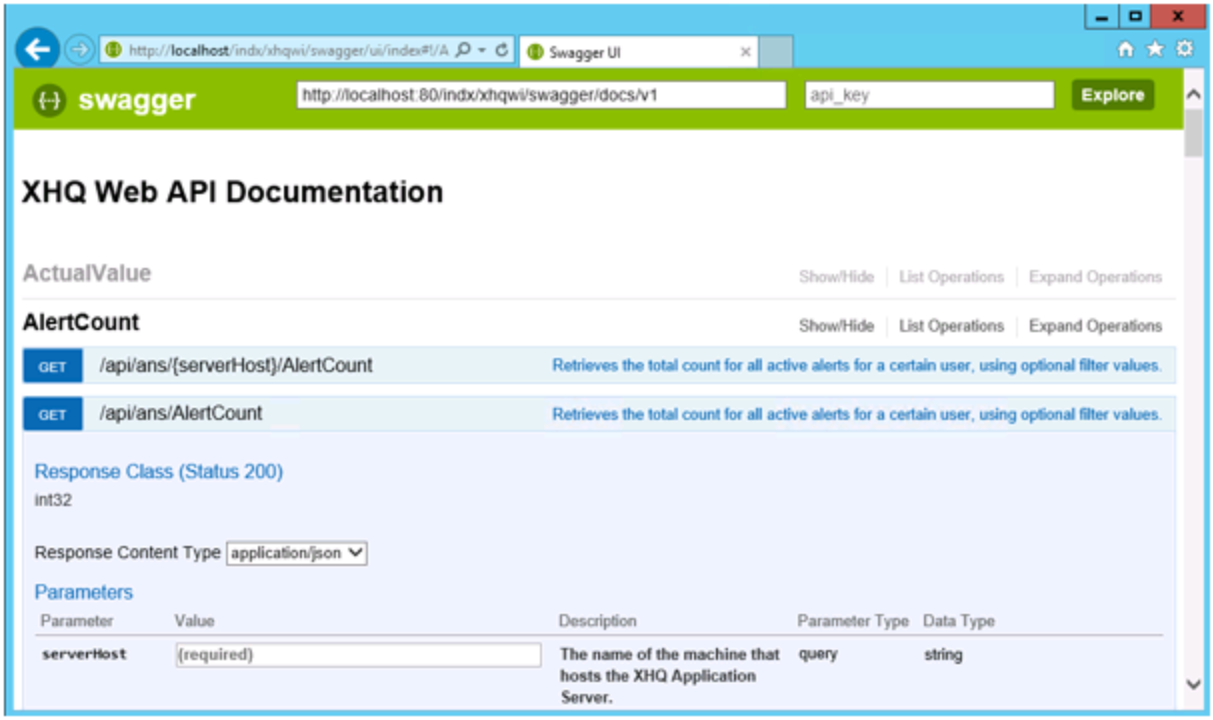
```
http://<XHQ Web Interface Webapp Hostname>/indx/xhqwi/swagger
```

Example 1:

```
http://acme1/indx/xhqwi/swagger
```

Example 2:

```
http://localhost/indx/xhqwi/swagger
```



Example of Swagger Online Help

XHQ Admin Module Methods

RESTful API calls for accessing XHQ Admin functionality.

A **detailed** description of each method is available from the following URL:

`http://<XHQ Web Interface Webapp Hostname>/indx/xhqwi/help`

Example:

`http://localhost/indx/xhqwi/help`.

Audit

API	Description
GET api/admin/{serverHost}/Audit?roleId={roleId}	Retrieves Retrieves a complete history of permissions changes for a certain role for audit trail purposes.
GET api/admin/Audit?serverHost={serverHost}&roleId={roleId}	Retrieves Retrieves a complete history of permissions changes for a certain role for audit trail purposes.

LimitClasses

API	Description
GET api/admin/{serverHost}/LimitClasses/{id}	Retrieves a Tolerance Limit Class.
GET api/admin/{serverHost}/LimitClasses	Retrieves the complete list of all Tolerance Limit Classes.
POST api/admin/{serverHost}/LimitClasses	Creates a new Tolerance Limit Class.
PUT api/admin/{serverHost}/LimitClasses	Edits (updates) an existing Tolerance Limit Class.
DELETE api/admin/{serverHost}/LimitClasses/{id}	Deletes an existing Limit Class.
GET api/admin/LimitClasses/{id}?serverHost={serverHost}	Retrieves a Tolerance Limit Class.
GET api/admin/LimitClasses?serverHost={serverHost}	Retrieves the complete list of all Tolerance Limit Classes.
POST api/admin/LimitClasses?serverHost={serverHost}	Creates a new Tolerance Limit Class.
PUT api/admin/LimitClasses?serverHost={serverHost}	Edits (updates) an existing Tolerance Limit Class.
DELETE api/admin/LimitClasses/{id}?serverHost={serverHost}	Deletes an existing Limit Class.

Paths

API	Description
GET api/admin/{serverHost}/Paths?roleId={roleId}&app={app}	Retrieves all paths for a certain security role.
PUT api/admin/{serverHost}/Paths?roleId={roleId}&app={app}	Edits (updates) the list of XHQ paths for a certain XHQ Role.
GET api/admin/Paths?serverHost={serverHost}&roleId=	Retrieves all paths for a certain security role.

API	Description
{roleId}&app={app}	
PUT api/admin/Paths?serverHost={serverHost}&roleId={roleId}&app={app}	Edits (updates) the list of XHQ paths for a certain XHQ Role.

Perm

API	Description
PUT api/admin/{serverHost}/Perm?roleId={roleId}&permType={permType}	Updates the list of permissions for a certain XHQ defined Role.
PUT api/admin/Perm?serverHost={serverHost}&roleId={roleId}&permType={permType}	Updates the list of permissions for a certain XHQ defined Role.

ReasonCodes

API	Description
GET api/admin/{serverHost}/ReasonCodes/{id}	Retrieves a Reason Code by its ID.
GET api/admin/{serverHost}/ReasonCodes	Retrieves all Reason Codes defined in the system. This call is to be used to build the Reason Code Tree in the client UI.
POST api/admin/{serverHost}/ReasonCodes	Creates a new Association Type in eLogs.
PUT api/admin/{serverHost}/ReasonCodes	Edits (updates) an existing Reason Code.
DELETE api/admin/{serverHost}/ReasonCodes/{id}	Deletes an existing Reason Code.
GET api/admin/ReasonCodes/{id}?serverHost={serverHost}	Retrieves a Reason Code by its ID.
GET api/admin/ReasonCodes?serverHost={serverHost}	Retrieves all Reason Codes defined in the system. This call is to be used to build the Reason Code Tree in the client UI.
POST api/admin/ReasonCodes?serverHost={serverHost}	Creates a new Association Type in eLogs.
PUT api/admin/ReasonCodes?serverHost={serverHost}	Edits (updates) an existing Reason Code.
DELETE api/admin/ReasonCodes/{id}?serverHost={serverHost}	Deletes an existing Reason Code.

Severities

API	Description
GET api/admin/{serverHost}/Severities/{id}	Retrieves a severity.
GET api/admin/{serverHost}/Severities	Retrieves the complete list of all severities.
POST api/admin/{serverHost}/Severities	Creates a new severity.
PUT api/admin/{serverHost}/Severities	Edits (updates) an existing severity.
DELETE api/admin/{serverHost}/Severities/{id}	Deletes an existing severity.

API	Description
GET api/admin/Severities/{id}?serverHost={serverHost}	Retrieves a severity.
GET api/admin/Severities?serverHost={serverHost}	Retrieves the complete list of all severities.
POST api/admin/Severities?serverHost={serverHost}	Creates a new severity.
PUT api/admin/Severities?serverHost={serverHost}	Edits (updates) an existing severity.
DELETE api/admin/Severities/{id}?serverHost={serverHost}	Deletes an existing severity.

SubTypes

API	Description
GET api/admin/{serverHost}/SubTypes/{id}?edit={edit}	Retrieves a sub-type.
GET api/admin/{serverHost}/SubTypes?app={app}	Retrieves all available Sub-Types for a certain XHQ PM Application type.
GET api/admin/{serverHost}/SubTypes?roleId={roleId} &app={app}	Retrieves the selected Sub-Type IDs for a certain security role.
GET api/admin/{serverHost}/SubTypes	Retrieves the complete list of all Sub-Types for all supported applications (ANS, TM, eLogs).
PUT api/admin/{serverHost}/SubTypes?roleId={roleId} &app={app}	Edits (updates) the list of Sub-Types for a certain XHQ Role.
POST api/admin/{serverHost}/SubTypes	Creates a new Sub-Type.
DELETE api/admin/{serverHost}/SubTypes/{id}	Deletes an existing Sub-Type.
GET api/admin/SubTypes/{id}?serverHost={serverHost} &edit={edit}	Retrieves a sub-type.
GET api/admin/SubTypes?serverHost={serverHost}&app= {app}	Retrieves all available Sub-Types for a certain XHQ PM Application type.
GET api/admin/SubTypes?serverHost={serverHost} &roleId={roleId}&app={app}	Retrieves the selected Sub-Type IDs for a certain security role.
GET api/admin/SubTypes?serverHost={serverHost}	Retrieves the complete list of all Sub-Types for all supported applications (ANS, TM, eLogs).
PUT api/admin/SubTypes?serverHost={serverHost} &roleId={roleId}&app={app}	Edits (updates) the list of Sub-Types for a certain XHQ Role.
POST api/admin/SubTypes?serverHost={serverHost}	Creates a new Sub-Type.
DELETE api/admin/SubTypes/{id}?serverHost={serverHost}	Deletes an existing Sub-Type.

Types

API	Description
GET api/admin/{serverHost}/Types/{id}?edit={edit}	Retrieves a type.
GET api/admin/{serverHost}/Types?app={app}	Retrieves all available Types for a certain XHQ PM

API	Description
	Application type.
GET api/admin/{serverHost}/Types?roleId={roleId}&app={app}	Retrieves the selected Type IDs for a certain security role.
GET api/admin/{serverHost}/Types	Retrieves the complete list of all Types for all supported applications (ANS, TM, eLogs).
GET api/admin/{serverHost}/Types?user={user}&app={app}	Retrieves all permissions for a given user, including accessible types and sub-types.
PUT api/admin/{serverHost}/Types?roleId={roleId}&app={app}	Edits (updates) the list of types for a certain XHQ Role.
PUT api/admin/{serverHost}/Types	Edits (updates) the list of types and sub-types.
POST api/admin/{serverHost}/Types	Creates a new Type.
DELETE api/admin/{serverHost}/Types/{id}	Deletes an existing Type.
GET api/admin/Types/{id}?serverHost={serverHost}&edit={edit}	Retrieves a type.
GET api/admin/Types?serverHost={serverHost}&app={app}	Retrieves all available Types for a certain XHQ PM Application type.
GET api/admin/Types?serverHost={serverHost}&roleId={roleId}&app={app}	Retrieves the selected Type IDs for a certain security role.
GET api/admin/Types?serverHost={serverHost}	Retrieves the complete list of all Types for all supported applications (ANS, TM, eLogs).
GET api/admin/Types?serverHost={serverHost}&user={user}&app={app}	Retrieves all permissions for a given user, including accessible types and sub-types.
PUT api/admin/Types?serverHost={serverHost}&roleId={roleId}&app={app}	Edits (updates) the list of types for a certain XHQ Role.
PUT api/admin/Types?serverHost={serverHost}	Edits (updates) the list of types and sub-types.
POST api/admin/Types?serverHost={serverHost}	Creates a new Type.
DELETE api/admin/Types/{id}?serverHost={serverHost}	Deletes an existing Type.

XHQ ANS Module Methods

RESTful API calls for accessing XHQ ANS functionality.

A **detailed** description of each method is available from the following URL:

`http://<XHQ Web Interface Webapp Hostname>/indx/xhqwi/help`

Example:

`http://localhost/indx/xhqwi/help`.

AnchorViews

API	Description
GET api/ans/{serverHost}/AnchorViews?anchorPath={anchorPath}	Retrieves all view names for a given anchor path.
GET api/ans/AnchorViews?serverHost={serverHost}&anchorPath={anchorPath}	Retrieves all view names for a given anchor path.

AnsId

API	Description
GET api/ans/{serverHost}/AnsId	Retrieves the ANS application ID as a string.
GET api/ans/AnsId?serverHost={serverHost}	Retrieves the ANS application ID as a string.

ASD

API	Description
GET api/ans/{serverHost}/ASD?name={name}&listType={listType}&list={list}&anchor={anchor}&active={active}&category={category}&sevMin={sevMin}&sortCol={sortCol}&sortOrder={sortOrder}	Retrieves a list of ANS Condition States, which can be used for an Alert Status Display UI, for instance.
GET api/ans/{serverHost}/ASD?license={license}	Checks if there is a valid ANS License present.
GET api/ans/ASD?serverHost={serverHost}&name={name}&listType={listType}&list={list}&anchor={anchor}&active={active}&category={category}&sevMin={sevMin}&sortCol={sortCol}&sortOrder={sortOrder}	Retrieves a list of ANS Condition States, which can be used for an Alert Status Display UI, for instance.
GET api/ans/ASD?serverHost={serverHost}&license={license}	Checks if there is a valid ANS License present.

Backfill

API	Description
POST api/ans/{serverHost}/Backfill?startTime={startTime}&endTime={endTime}	Backfills the excursion logs for a given condition definition.
POST api/ans/Backfill?serverHost={serverHost}&startTime={startTime}&endTime={endTime}	Backfills the excursion logs for a given condition definition.

CodeKey

API	Description
GET api/ans/{serverHost}/CodeKey/{id}	Retrieves an ANS code key (e.g. category or interval) given its correspondent ID.
GET api/ans/{serverHost}/CodeKey?codeType={codeType}	Retrieves a list of code keys of a certain type (e.g. category or interval) from ANS.
POST api/ans/{serverHost}/CodeKey	Creates a new Code Key in ANS.
PUT api/ans/{serverHost}/CodeKey	Updates (edits) an existing Code Key in ANS.
DELETE api/ans/{serverHost}/CodeKey/{id}	Deletes an existing ANS code key with the given ID.
GET api/ans/CodeKey/{id}?serverHost={serverHost}	Retrieves an ANS code key (e.g. category or interval) given its correspondent ID.
GET api/ans/CodeKey?serverHost={serverHost} &codeType={codeType}	Retrieves a list of code keys of a certain type (e.g. category or interval) from ANS.
POST api/ans/CodeKey?serverHost={serverHost}	Creates a new Code Key in ANS.
PUT api/ans/CodeKey?serverHost={serverHost}	Updates (edits) an existing Code Key in ANS.
DELETE api/ans/CodeKey/{id}?serverHost={serverHost}	Deletes an existing ANS code key with the given ID.

CondDef

API	Description
GET api/ans/{serverHost}/CondDef/{id}	Retrieves an ANS condition definition given its correspondent ID.
GET api/ans/{serverHost}/CondDef?defList={defList}	Retrieves a list of all ANS condition definitions for a given definition list.
GET api/ans/{serverHost}/CondDef?listId={listId}&sortCol={sortCol}&sortDir={sortDir}	Retrieves a sorted list of the definition items in a particular list along with miscellaneous properties of each.
POST api/ans/{serverHost}/CondDef	Creates a new condition definition in ANS.
PUT api/ans/{serverHost}/CondDef	Edits (updates) an existing ANS condition definition.
DELETE api/ans/{serverHost}/CondDef/{id}	Deletes an existing ANS condition definition with the given ID.
GET api/ans/CondDef/{id}?serverHost={serverHost}	Retrieves an ANS condition definition given its correspondent ID.
GET api/ans/CondDef?serverHost={serverHost}&defList={defList}	Retrieves a list of all ANS condition definitions for a given definition list.
GET api/ans/CondDef?serverHost={serverHost}&listId={listId}&sortCol={sortCol}&sortDir={sortDir}	Retrieves a sorted list of the definition items in a particular list along with miscellaneous properties of each.

API	Description
POST api/ans/CondDef?serverHost={serverHost}	Creates a new condition definition in ANS.
PUT api/ans/CondDef?serverHost={serverHost}	Edits (updates) an existing ANS condition definition.
DELETE api/ans/CondDef/{id}?serverHost={serverHost}	Deletes an existing ANS condition definition with the given ID.

DefList

API	Description
GET api/ans/{serverHost}/DefList/{id}	Retrieves an ANS definition list given its correspondent ID.
GET api/ans/{serverHost}/DefList?name={name}	Retrieves an ANS definition list given its name.
GET api/ans/{serverHost}/DefList?ownerOnly={ownerOnly}&user={user}	Retrieves all ANS definition lists for the calling or for the specified user.
POST api/ans/{serverHost}/DefList	Creates a new definition list in ANS.
PUT api/ans/{serverHost}/DefList	Edits (updates) an existing ANS definition list.
DELETE api/ans/{serverHost}/DefList/{id}	Deletes an existing ANS definition list with the given ID.
GET api/ans/DefList/{id}?serverHost={serverHost}	Retrieves an ANS definition list given its correspondent ID.
GET api/ans/DefList?serverHost={serverHost}&name={name}	Retrieves an ANS definition list given its name.
GET api/ans/DefList?serverHost={serverHost}&ownerOnly={ownerOnly}&user={user}	Retrieves all ANS definition lists for the calling or for the specified user.
POST api/ans/DefList?serverHost={serverHost}	Creates a new definition list in ANS.
PUT api/ans/DefList?serverHost={serverHost}	Edits (updates) an existing ANS definition list.
DELETE api/ans/DefList/{id}?serverHost={serverHost}	Deletes an existing ANS definition list with the given ID.

ExcursionLog

API	Description
GET api/ans/{serverHost}/ExcursionLog?defId={defId}&name={name}&anchor={anchor}&category={category}&listType={listType}&list={list}&minCount={minCount}&minDurationPct={minDurationPct}&startTime={startTime}&endTime={endTime}&reportType={reportType}&sortCol={sortCol}&sortOrder={sortOrder}	Retrieves a list of ANS of excursion objects giving some filter criteria, for an Excursion Log display UI, for instance.
GET api/ans/{serverHost}/ExcursionLog?maxDays={maxDays}	Deletes/purges all ANS excursion logs that are older than a certain date..
GET api/ans/ExcursionLog?serverHost={serverHost}&defId={defId}&name={name}&anchor={anchor}	Retrieves a list of ANS of excursion objects giving some filter criteria, for an Excursion Log display UI, for instance.

API	Description
&category={category}&listType={listType}&list={list}&minCount={minCount}&minDurationPct={minDurationPct}&startTime={startTime}&endTime={endTime}&reportType={reportType}&sortCol={sortCol}&sortOrder={sortOrder}	
GET api/ans/ExcursionLog?serverHost={serverHost}&maxDays={maxDays}	Deletes/purges all ANS excursion logs that are older than a certain date..

Export

API	Description
GET api/ans/{serverHost}/Export?type={type}&force={force}	Starts a DB Export operation on the Solution server.
GET api/ans/{serverHost}/Export?type={type}	Retrieves the current DB Export operation status from the Solution server.
GET api/ans/{serverHost}/Export?defList={defList}&exportDefs={exportDefs}&exportLoConfigs={exportLoConfigs}&exportAssocSubsc={exportAssocSubsc}&exportMySubsc={exportMySubsc}	Export a list of ANS condition definitions into XML.
GET api/ans/{serverHost}/Export?rc={rc}	Export all Reason Codes used in Lost Opportunity into XML.
GET api/ans/Export?serverHost={serverHost}&type={type}&force={force}	Starts a DB Export operation on the Solution server.
GET api/ans/Export?serverHost={serverHost}&type={type}	Retrieves the current DB Export operation status from the Solution server.
GET api/ans/Export?serverHost={serverHost}&defList={defList}&exportDefs={exportDefs}&exportLoConfigs={exportLoConfigs}&exportAssocSubsc={exportAssocSubsc}&exportMySubsc={exportMySubsc}	Export a list of ANS condition definitions into XML.
GET api/ans/Export?serverHost={serverHost}&rc={rc}	Export all Reason Codes used in Lost Opportunity into XML.

Import

API	Description
POST api/ans/{serverHost}/Import?type={type}&autoCreateLists={autoCreateLists}&updateDefsById={updateDefsById}&updateRcById={updateRcById}&importSubscriptions={importSubscriptions}&importLoConfigs={importLoConfigs}	Import a list of ANS condition definitions or Reason Codes or deletes condition definitions from an XML file located on the client.
POST api/ans/{serverHost}/Import?filePath={filePath}&type={type}&autoCreateLists={autoCreateLists}	Import a list of ANS condition definitions or Reason Codes or deletes condition definitions from an XML file

API	Description
&updateDefsById={updateDefsById}&updateRcById={updateRcById}&importSubscriptions={importSubscriptions}&importLoConfigs={importLoConfigs}	located on the server.
POST api/ans/Import?serverHost={serverHost}&type={type}&autoCreateLists={autoCreateLists}&updateDefsById={updateDefsById}&updateRcById={updateRcById}&importSubscriptions={importSubscriptions}&importLoConfigs={importLoConfigs}	Import a list of ANS condition definitions or Reason Codes or deletes condition definitions from an XML file located on the client.
POST api/ans/Import?serverHost={serverHost}&filePath={filePath}&type={type}&autoCreateLists={autoCreateLists}&updateDefsById={updateDefsById}&updateRcById={updateRcById}&importSubscriptions={importSubscriptions}&importLoConfigs={importLoConfigs}	Import a list of ANS condition definitions or Reason Codes or deletes condition definitions from an XML file located on the server.

Interval

API	Description
GET api/ans/{serverHost}/Interval/{id}	Retrieves an ANS interval given its correspondent ID.
GET api/ans/{serverHost}/Interval	Retrieves all ANS intervals for the calling user.
POST api/ans/{serverHost}/Interval	Creates a new interval in ANS.
PUT api/ans/{serverHost}/Interval	Edits (updates) an existing interval in ANS.
DELETE api/ans/{serverHost}/Interval/{id}	Deletes an existing ANS interval with the given ID.
GET api/ans/Interval/{id}?serverHost={serverHost}	Retrieves an ANS interval given its correspondent ID.
GET api/ans/Interval?serverHost={serverHost}	Retrieves all ANS intervals for the calling user.
POST api/ans/Interval?serverHost={serverHost}	Creates a new interval in ANS.
PUT api/ans/Interval?serverHost={serverHost}	Edits (updates) an existing interval in ANS.
DELETE api/ans/Interval/{id}?serverHost={serverHost}	Deletes an existing ANS interval with the given ID.

ListSection

API	Description
GET api/ans/{serverHost}/ListSection/{id}?listId={listId}	Retrieves an ANS definition list section given its correspondent ID and parent list ID.
GET api/ans/{serverHost}/ListSection?section={section}&defList={defList}	Retrieves an ANS definition list section given its correspondent name and parent list name.
GET api/ans/{serverHost}/ListSection?defList={defList}	Retrieves all ANS sections for a given definition list.

API	Description
POST api/ans/{serverHost}/ListSection	Creates a new list section in ANS.
PUT api/ans/{serverHost}/ListSection	Edits (updates) an existing ANS list section.
DELETE api/ans/{serverHost}/ListSection/{id}	Deletes an existing ANS definition list section with the given ID.
GET api/ans/ListSection/{id}?serverHost={serverHost} &listId={listId}	Retrieves an ANS definition list section given its correspondent ID and parent list ID.
GET api/ans/ListSection?serverHost={serverHost} &section={section}&defList={defList}	Retrieves an ANS definition list section given its correspondent name and parent list name.
GET api/ans/ListSection?serverHost={serverHost} &defList={defList}	Retrieves all ANS sections for a given definition list.
POST api/ans/ListSection?serverHost={serverHost}	Creates a new list section in ANS.
PUT api/ans/ListSection?serverHost={serverHost}	Edits (updates) an existing ANS list section.
DELETE api/ans/ListSection/{id}?serverHost={serverHost}	Deletes an existing ANS definition list section with the given ID.

LO

API	Description
GET api/ans/{serverHost}/LO	Checks if there is a valid LO License present.
GET api/ans/LO?serverHost={serverHost}	Checks if there is a valid LO License present.

Locale

API	Description
GET api/ans/Locale	Retrieves the Locale language settings from the HTTP Request object.

Notif

API	Description
GET api/ans/{serverHost}/Notif	Updates the notification handlers with latest settings from the ANS property files.
GET api/ans/{serverHost}/Notif?uaction={uaction} &email={email}&authEmail={authEmail}&soap={soap}	Updates the notification handlers state.
GET api/ans/Notif?serverHost={serverHost}	Updates the notification handlers with latest settings from the ANS property files.
GET api/ans/Notif?serverHost={serverHost}&uaction= {uaction}&email={email}&authEmail={authEmail}&soap= {soap}	Updates the notification handlers state.

NotifRouting

API	Description
GET api/ans/{serverHost}/NotifRouting/{id}	Retrieves an ANS notification routing given its correspondent ID.
GET api/ans/{serverHost}/NotifRouting	Retrieves a list of all ANS notification routings for the caller user.
POST api/ans/{serverHost}/NotifRouting	Creates a new notification routing in ANS.
PUT api/ans/{serverHost}/NotifRouting	Updates (edits) an existing notification routing in ANS.
DELETE api/ans/{serverHost}/NotifRouting/{id}	Deletes an existing ANS notification routing with the given ID.
GET api/ans/NotifRouting/{id}?serverHost={serverHost}	Retrieves an ANS notification routing given its correspondent ID.
GET api/ans/NotifRouting?serverHost={serverHost}	Retrieves a list of all ANS notification routings for the caller user.
POST api/ans/NotifRouting?serverHost={serverHost}	Creates a new notification routing in ANS.
PUT api/ans/NotifRouting?serverHost={serverHost}	Updates (edits) an existing notification routing in ANS.
DELETE api/ans/NotifRouting/{id}?serverHost={serverHost}	Deletes an existing ANS notification routing with the given ID.

Perm

API	Description
GET api/ans/{serverHost}/Perm/{id}	Retrieves an ANS permission given its correspondent ID.
GET api/ans/{serverHost}/Perm	Retrieves all ANS permissions for the calling user.
GET api/ans/{serverHost}/Perm?user={user}	Retrieves all permissions for a given user.
GET api/ans/Perm/{id}?serverHost={serverHost}	Retrieves an ANS permission given its correspondent ID.
GET api/ans/Perm?serverHost={serverHost}	Retrieves all ANS permissions for the calling user.
GET api/ans/Perm?serverHost={serverHost}&user={user}	Retrieves all permissions for a given user.

Role

API	Description
GET api/ans/{serverHost}/Role/{id}	Retrieves an XHQ Role given its correspondent ID.
GET api/ans/{serverHost}/Role	Retrieves all XHQ Roles for the calling user.
GET api/ans/{serverHost}/Role?defId={defId} &allowedRoles={allowedRoles}	Retrieves the security roles for a certain condition definition.
PUT api/ans/{serverHost}/Role	Edits (updates) an existing role in XHQ.
POST api/ans/{serverHost}/Role?defId={defId}&roleId=	Adds a new allowed security role for a certain condition

API	Description
{roleId}	definition.
DELETE api/ans/{serverHost}/Role?defId={defId}&roleId={roleId}	Removes an allowed security role for a certain condition definition.
GET api/ans/Role/{id}?serverHost={serverHost}	Retrieves an XHQ Role given its correspondent ID.
GET api/ans/Role?serverHost={serverHost}	Retrieves all XHQ Roles for the calling user.
GET api/ans/Role?serverHost={serverHost}&defId={defId}&allowedRoles={allowedRoles}	Retrieves the security roles for a certain condition definition.
PUT api/ans/Role?serverHost={serverHost}	Edits (updates) an existing role in XHQ.
POST api/ans/Role?serverHost={serverHost}&defId={defId}&roleId={roleId}	Adds a new allowed security role for a certain condition definition.
DELETE api/ans/Role?serverHost={serverHost}&defId={defId}&roleId={roleId}	Removes an allowed security role for a certain condition definition.

Schedule

API	Description
GET api/ans/{serverHost}/Schedule/{id}	Retrieves an ANS schedule given its correspondent ID.
GET api/ans/{serverHost}/Schedule	Retrieves all ANS schedules for the calling user.
POST api/ans/{serverHost}/Schedule	Creates a new schedule in ANS.
PUT api/ans/{serverHost}/Schedule	Edits (updates) an existing schedule in ANS.
DELETE api/ans/{serverHost}/Schedule/{id}	Deletes an existing ANS schedule with the given ID.
GET api/ans/Schedule/{id}?serverHost={serverHost}	Retrieves an ANS schedule given its correspondent ID.
GET api/ans/Schedule?serverHost={serverHost}	Retrieves all ANS schedules for the calling user.
POST api/ans/Schedule?serverHost={serverHost}	Creates a new schedule in ANS.
PUT api/ans/Schedule?serverHost={serverHost}	Edits (updates) an existing schedule in ANS.
DELETE api/ans/Schedule/{id}?serverHost={serverHost}	Deletes an existing ANS schedule with the given ID.

SubscriptionItem

API	Description
GET api/ans/{serverHost}/SubscriptionItem/{id}	Retrieves an ANS subscription item given its correspondent ID.
GET api/ans/{serverHost}/SubscriptionItem?subList={subList}	Retrieves a list of all ANS subscription items for a given subscription list.
GET api/ans/{serverHost}/SubscriptionItem?defId={defId}	Retrieves a list of all subscribers' names for a given condition definition ID..
POST api/ans/{serverHost}/SubscriptionItem	Creates a new subscription item in ANS.

API	Description
PUT api/ans/{serverHost}/SubscriptionItem	Edits (updates) an existing ANS subscription item.
DELETE api/ans/{serverHost}/SubscriptionItem/{id}	Deletes an existing ANS subscription with the given ID.
GET api/ans/SubscriptionItem/{id}?serverHost={serverHost}	Retrieves an ANS subscription item given its correspondent ID.
GET api/ans/SubscriptionItem?serverHost={serverHost}&subList={subList}	Retrieves a list of all ANS subscription items for a given subscription list.
GET api/ans/SubscriptionItem?serverHost={serverHost}&defId={defId}	Retrieves a list of all subscribers' names for a given condition definition ID..
POST api/ans/SubscriptionItem?serverHost={serverHost}	Creates a new subscription item in ANS.
PUT api/ans/SubscriptionItem?serverHost={serverHost}	Edits (updates) an existing ANS subscription item.
DELETE api/ans/SubscriptionItem/{id}?serverHost={serverHost}	Deletes an existing ANS subscription with the given ID.

SubscriptionList

API	Description
GET api/ans/{serverHost}/SubscriptionList/{id}	Retrieves an ANS subscription list given its correspondent ID.
GET api/ans/{serverHost}/SubscriptionList?name={name}	Retrieves an ANS subscription list given its name.
GET api/ans/{serverHost}/SubscriptionList	Retrieves all ANS subscription lists for the calling user.
POST api/ans/{serverHost}/SubscriptionList	Creates a new subscription list in ANS.
PUT api/ans/{serverHost}/SubscriptionList	Edits (updates) an existing subscription list in ANS.
DELETE api/ans/{serverHost}/SubscriptionList/{id}	Deletes an existing ANS subscription list with the given ID.
GET api/ans/SubscriptionList/{id}?serverHost={serverHost}	Retrieves an ANS subscription list given its correspondent ID.
GET api/ans/SubscriptionList?serverHost={serverHost}&name={name}	Retrieves an ANS subscription list given its name.
GET api/ans/SubscriptionList?serverHost={serverHost}	Retrieves all ANS subscription lists for the calling user.
POST api/ans/SubscriptionList?serverHost={serverHost}	Creates a new subscription list in ANS.
PUT api/ans/SubscriptionList?serverHost={serverHost}	Edits (updates) an existing subscription list in ANS.
DELETE api/ans/SubscriptionList/{id}?serverHost={serverHost}	Deletes an existing ANS subscription list with the given ID.

Template

API	Description
GET api/ans/{serverHost}/Template/{id}	Retrieves an ANS schedule template given its

API	Description
	correspondent ID.
GET api/ans/{serverHost}/Template	Retrieves all ANS templates for the calling user.
POST api/ans/{serverHost}/Template	Creates a new schedule template in ANS.
PUT api/ans/{serverHost}/Template	Edits (updates) an existing schedule template in ANS.
DELETE api/ans/{serverHost}/Template/{id}	Deletes an existing ANS template with the given ID.
GET api/ans/Template/{id}?serverHost={serverHost}	Retrieves an ANS schedule template given its correspondent ID.
GET api/ans/Template?serverHost={serverHost}	Retrieves all ANS templates for the calling user.
POST api/ans/Template?serverHost={serverHost}	Creates a new schedule template in ANS.
PUT api/ans/Template?serverHost={serverHost}	Edits (updates) an existing schedule template in ANS.
DELETE api/ans/Template/{id}?serverHost={serverHost}	Deletes an existing ANS template with the given ID.

User

API	Description
GET api/ans/{serverHost}/User/{id}	Retrieves an ANS User given its correspondent ID.
GET api/ans/{serverHost}/User	Retrieves all ANS Users.
GET api/ans/{serverHost}/User?cur={cur}	Retrieves the current user name.
POST api/ans/{serverHost}/User	Creates a new user in ANS.
PUT api/ans/{serverHost}/User	Edits (updates) an existing user in ANS.
DELETE api/ans/{serverHost}/User/{id}	Deletes an existing ANS user with the given ID.
GET api/ans/User/{id}?serverHost={serverHost}	Retrieves an ANS User given its correspondent ID.
GET api/ans/User?serverHost={serverHost}	Retrieves all ANS Users.
GET api/ans/User?serverHost={serverHost}&cur={cur}	Retrieves the current user name.
POST api/ans/User?serverHost={serverHost}	Creates a new user in ANS.
PUT api/ans/User?serverHost={serverHost}	Edits (updates) an existing user in ANS.
DELETE api/ans/User/{id}?serverHost={serverHost}	Deletes an existing ANS user with the given ID.

XHQ Core Module Methods

RESTful API calls for accessing XHQ Core functionality.

A **detailed** description of each method is available from the following URL:

`http://<XHQ Web Interface Webapp Hostname>/indx/xhqwi/help`

Example:

`http://localhost/indx/xhqwi/help`.

The calls, or methods, fall into five categories:

- Access
- Collections
- Metadata
- Points
- Utility

Access Calls

RESTful API calls for accessing XHQ security access calls functionality.

CanReadMember

API	Description
GET api/core/{serverHost}/Access/CanReadMember?path={path}	Determines if the given global path or alias can be read, for the current user. Paths must be absolute (fully qualified) or aliases. Absolute paths begin with "::" marker, otherwise it is considered an alias.

CanWriteMember

API	Description
GET api/core/{serverHost}/Access/CanWriteMember?path={path}	Determines if the given global path or alias can be written, for the current user. Paths must be absolute (fully qualified) or aliases. Absolute paths begin with "::" marker, otherwise it is considered an alias.

PathList

API	Description
POST api/core/{serverHost}/Access/PathList?username={username}	Retrieves the read and write access permissions for an array of XHQ points (specified by paths or aliases/tags), for the specified user. This operation can only be executed by a user with Solution Admin permission. If the username is NULL or empty the read/write access will be returned for the current user. Paths must be absolute (fully qualified) or aliases. Absolute paths begin with "::" marker, otherwise it is considered an alias.

ReadableMembers

API	Description
GET api/core/{serverHost} /Access/ReadableMembers?path={path}	Retrieves the list of readable member names for the given path, for the current user. Paths must be absolute (fully qualified) or aliases. Absolute paths begin with "::" marker, otherwise it is considered an alias.

ReadPathRoles

API	Description
GET api/core/{serverHost}/Access/ReadPathRoles?path={path}	Retrieves the XHQ Roles names that have read access to the given path.
GET api/core/{serverHost} /Access/ReadPathRoles?username={username}&path={path}	Retrieves the list of XHQ Roles names that allow the specified user read access to the given path.

UserRoles

API	Description
GET api/core/{serverHost}/Access/UserRoles?roleName={roleName}&username={username}	Flag that indicates if the specified user belongs or not to the specified role.
GET api/core/{serverHost}/Access/UserRoles?username={username}	Returns the list of XHQ Roles names for the specified user.

WriteableMembers

API	Description
GET api/core/{serverHost} /Access/WriteableMembers?path={path}	Retrieves the list of writeable member names for the given path, for the current user. Paths must be absolute (fully qualified) or aliases. Absolute paths begin with "::" marker, otherwise it is considered an alias.

WritePathRoles

API	Description
GET api/core/{serverHost}/Access/WritePathRoles?path={path}	Retrieves the XHQ Roles names that have write access to the given path.
GET api/core/{serverHost} /Access/WritePathRoles?username={username}&path={path}	Retrieves the list of XHQ Roles names that allow the specified user write access to the given path.

Collections Calls

RESTful API calls for accessing XHQ Collections calls functionality.

ColumnAverage

API	Description
GET api/core/{serverHost} /Collections/ColumnAverage?path={path} &column={column}&whereClause={whereClause}	Gets the Collection average value for a certain column.

ColumnCount

API	Description
GET api/core/{serverHost} /Collections/ColumnCount?path={path} &column={column}&whereClause={whereClause}	Gets the Collection count value for a certain column.

ColumnMaximum

API	Description
GET api/core/{serverHost} /Collections/ColumnMaximum?path={path}&column= {column}&whereClause={whereClause}	Gets the Collection maximum value for a certain column.

ColumnMinimum

API	Description
GET api/core/{serverHost} /Collections/ColumnMinimum?path={path}&column= {column}&whereClause={whereClause}	Gets the Collection minimum value for a certain column.

ColumnSum

API	Description
GET api/core/{serverHost}/Collections/ColumnSum?path= {path}&column={column}&whereClause={whereClause}	Gets the Collection sum value for a certain column.

Records

API	Description
GET api/core/{serverHost}/Collections/Records?path= {path}&columns={columns}&startIndex={startIndex} &rowsCount={rowsCount}&whereClause={whereClause} &orderByClause={orderByClause}	Gets a Linked Collection data set result formatted as JSON for a link collection query.
GET api/core/{serverHost}/Collections/Records?path= {path}&xml={xml}&columns={columns}&startIndex= {startIndex}&rowsCount={rowsCount}&whereClause= {whereClause}&orderByClause={orderByClause}	Gets a Linked Collection data set result formatted as XML for a link collection query.

RecordsCount

API	Description
GET api/core/{serverHost} /Collections/RecordsCount?path={path}&whereClause= {whereClause}	Gets the Collection records count.

Metadata Calls

RESTful API calls for accessing XHQ Metadata calls functionality.

AllMembers

API	Description
GET api/core/{serverHost} /Metadata/AllMembers?parentPath={parentPath} &showHidden={showHidden}	Gets all the child members' names for a component specified by a path.

CollectionMembers

API	Description
GET api/core/{serverHost} /Metadata/CollectionMembers?parentPath={parentPath} &showHidden={showHidden}	Gets the child collection members' names for a component specified by a path.
GET api/core/{serverHost} /Metadata/CollectionMembers?parentPath={parentPath} &infoType={infoType}	Gets the child collection members' info for a component specified by a path.

ComponentMembers

API	Description
GET api/core/{serverHost} /Metadata/ComponentMembers?parentPath={parentPath} &showHidden={showHidden}	Gets the child component members' names for a component specified by a path.

GlobalCollections

API	Description
GET api/core/{serverHost}/Metadata/GlobalCollections	Gets all the XHQ global collections names.

PrimitiveMembers

API	Description
GET api/core/{serverHost} /Metadata/PrimitiveMembers?parentPath={parentPath} &showHidden={showHidden}	Gets the child primitive members' names for a component specified by a path.

SolutionRoot

API	Description
GET api/core/{serverHost}/Metadata/SolutionRoot	Gets the XHQ solution root name.

SolutionTreeNode

API	Description
GET api/core/{serverHost} /Metadata/SolutionTreeNode?path={path}	Generates one level of the XHQ Solution tree as XML for a certain node (defined by an absolute path).

Points Calls

RESTful API calls for accessing XHQ Point calls functionality.

ActualValue

API	Description
GET api/core/{serverHost}/Points/ActualValue?path={path} &actualTime={actualTime}&units={units}&timeZone= {timeZone}	Gets the VTQ values for a single point/primitive at a specified moment in time (ACTUAL).
POST api/core/{serverHost} /Points/Bulk/ActualValue?actualTime={actualTime}&units= {units}&timeZone={timeZone}	Gets the VTQ data for a group/bulk of points/primitives at a specified moment in time (AT) as a VTQ array.

AtValue

API	Description
GET api/core/{serverHost}/Points/AtValue?path={path} &atTime={atTime}&units={units}&timeZone={timeZone}	Gets the VTQ values for a single point/primitive at a specified moment in time (AT).
POST api/core/{serverHost}/Points/Bulk/AtValue?atTime= {atTime}&units={units}&timeZone={timeZone}	Gets the VTQ data for a group/bulk of points/primitives at a specified moment in time (AT) as a VTQ array.

AverageValue

API	Description
GET api/core/{serverHost}/Points/AverageValue?path= {path}&startTime={startTime}&endTime={endTime}&units= {units}&timeZone={timeZone}	Gets the VTQ average value for a single point/primitive calculated for a time range.
POST api/core/{serverHost} /Points/Bulk/AverageValue?startTime={startTime} &endTime={endTime}&units={units}&timeZone= {timeZone}	Gets the VTQ average data for a group/bulk of points/primitives for a given time range, as a VTQ list.

CountValue

API	Description
GET api/core/{serverHost}/Points/CountValue?path={path}&startTime={startTime}&endTime={endTime}&units={units}&timeZone={timeZone}	Gets the VTQ count value for a single point/primitive calculated for a time range.
POST api/core/{serverHost}/Points/Bulk/CountValue?startTime={startTime}&endTime={endTime}&units={units}&timeZone={timeZone}	Gets the VTQ count data for a group/bulk of points/primitives for a given time range, as a VTQ list.

CurrentValue

API	Description
GET api/core/{serverHost}/Points/CurrentValue?path={path}&units={units}&timeZone={timeZone}	Gets the current VTQ values for a single point/primitive.
PUT api/core/{serverHost}/Points/CurrentValue?path={path}&value={value}&time={time}&quality={quality}&timeZone={timeZone}	Updates the value in the XHQ Solution Server for the member specified in the request.
PUT api/core/{serverHost}/Points/CurrentValue?timeZone={timeZone}	Writes values directly to backend (e.g. PI historian) for the members specified in the request. Paths must be absolute (fully qualified) or aliases (e.g. tags). Absolute paths begin with "::" marker, otherwise it is considered an alias.
POST api/core/{serverHost}/Points/Bulk/CurrentValue?units={units}&timeZone={timeZone}	Gets the current VTQ data for a group/bulk of points/primitives as a VTQ array.

DeltaValue

API	Description
GET api/core/{serverHost}/Points/DeltaValue?path={path}&startTime={startTime}&endTime={endTime}&units={units}&timeZone={timeZone}	Gets the VTQ delta value for a single point/primitive calculated for a time range.
POST api/core/{serverHost}/Points/Bulk/DeltaValue?startTime={startTime}&endTime={endTime}&units={units}&timeZone={timeZone}	Gets the VTQ delta data for a group/bulk of points/primitives for a given time range, as a VTQ list.

FirstValue

API	Description
GET api/core/{serverHost}/Points/FirstValue?path={path}&startTime={startTime}&endTime={endTime}&units={units}&timeZone={timeZone}	Gets the VTQ first value for a single point/primitive calculated for a time range.
POST api/core/{serverHost}/Points/Bulk/FirstValue?startTime={startTime}&endTime={endTime}&units={units}&timeZone={timeZone}	Gets the VTQ first data for a group/bulk of points/primitives for a given time range, as a VTQ list.

IntegralValue

API	Description
GET api/core/{serverHost}/Points/IntegralValue?path={path}&startTime={startTime}&endTime={endTime}&units={units}&timeZone={timeZone}	Gets the VTQ integral value for a single point/primitive calculated for a time range.
POST api/core/{serverHost}/Points/Bulk/IntegralValue?startTime={startTime}&endTime={endTime}&units={units}&timeZone={timeZone}	Gets the VTQ integral data for a group/bulk of points/primitives for a given time range, as a VTQ list.

LastValue

API	Description
GET api/core/{serverHost}/Points/LastValue?path={path}&startTime={startTime}&endTime={endTime}&units={units}&timeZone={timeZone}	Gets the VTQ last value for a single point/primitive calculated for a time range.
POST api/core/{serverHost}/Points/Bulk/LastValue?startTime={startTime}&endTime={endTime}&units={units}&timeZone={timeZone}	Gets the VTQ last data for a group/bulk of points/primitives for a given time range, as a VTQ list.

MaximumValue

API	Description
GET api/core/{serverHost}/Points/MaximumValue?path={path}&startTime={startTime}&endTime={endTime}&units={units}&timeZone={timeZone}	Gets the VTQ maximum value for a single point/primitive calculated for a time range.
POST api/core/{serverHost}/Points/Bulk/MaximumValue?startTime={startTime}&endTime={endTime}&units={units}&timeZone={timeZone}	Gets the VTQ maximum data for a group/bulk of points/primitives for a given time range, as a VTQ list.

MinimumValue

API	Description
GET api/core/{serverHost}/Points/MinimumValue?path={path}&startTime={startTime}&endTime={endTime}&units={units}&timeZone={timeZone}	Gets the VTQ minimum value for a single point/primitive calculated for a time range.
POST api/core/{serverHost}/Points/Bulk/MinimumValue?startTime={startTime}&endTime={endTime}&units={units}&timeZone={timeZone}	Gets the VTQ minimum data for a group/bulk of points/primitives for a given time range, as a VTQ list.

SumValue

API	Description
GET api/core/{serverHost}/Points/SumValue?path={path}&startTime={startTime}&endTime={endTime}&units=	Gets the VTQ sum value for a single point/primitive calculated for a time range.

API	Description
{units}&timeZone={timeZone}	
POST api/core/{serverHost} /Points/Bulk/SumValue?startTime={startTime}&endTime= {endTime}&units={units}&timeZone={timeZone}	Gets the VTQ sum data for a group/bulk of points/primitives for a given time range, as a VTQ list.

TimeSeries

API	Description
GET api/core/{serverHost}/Points/TimeSeries?path={path} &startTime={startTime}&endTime={endTime}&timeZone= {timeZone}&samplingMode={samplingMode}&noPoints= {noPoints}&units={units}	Gets the Time Series data for a primitive specified through a path or alias as a string array of triplets Value Time Quality (VTQ) structure from the time range specified with the given characteristics.

Utility Calls

RESTful API calls for accessing XHQ utility calls functionality.

CheckSolutionAccess

API	Description
GET api/core/{serverHost}/CheckSolutionAccess	Checks if a connection to the XHQ Solution Server can be established.
GET api/core/CheckSolutionAccess?serverHost={serverHost}	Checks if a connection to the XHQ Solution Server can be established.

Collections

API	Description
GET api/core/{serverHost}/Collections?getAll={getAll}	Retrieves XHQ Global collection names and their corresponding IDs.
GET api/core/{serverHost}/Collections?colName= {colName}	Retrieves all primitive members names and their data types for a global collection.
GET api/core/{serverHost}/Collections?collectionName= {collectionName}&col1={col1}&col2={col2}&col1Val= {col1Val}&col2Val={col2Val}&maxRows={maxRows}	Retrieves the Collection data for elogs Associations.
GET api/core/Collections?serverHost={serverHost}&getAll= {getAll}	Retrieves XHQ Global collection names and their corresponding IDs.
GET api/core/Collections?serverHost={serverHost} &colName={colName}	Retrieves all primitive members names and their data types for a global collection.
GET api/core/Collections?serverHost={serverHost} &collectionName={collectionName}&col1={col1}&col2= {col2}&col1Val={col1Val}&col2Val={col2Val}&maxRows= {maxRows}	Retrieves the Collection data for elogs Associations.

GlobalProps

API	Description
GET api/core/{serverHost}/GlobalProps?name={name}	Retrieves XHQ Global Property values.
GET api/core/GlobalProps?serverHost={serverHost} &name={name}	Retrieves XHQ Global Property values.

License

API	Description
GET api/core/{serverHost}/License?type={type}	Checks if there is a valid License present for the indicated XHQ module type.
GET api/core/License?serverHost={serverHost}&type= {type}	Checks if there is a valid License present for the indicated XHQ module type.

Logger

API	Description
PUT api/core/{serverHost}/Logger?module={module} &logMessage={logMessage}	Logs a new app client message in the appropriate web app module log (e.g. eLogs, ANS).
PUT api/core/Logger?serverHost={serverHost}&module= {module}&logMessage={logMessage}	Logs a new app client message in the appropriate web app module log (e.g. eLogs, ANS).

Login

API	Description
GET api/core/Login	Authenticates the calling user

PathPicker

API	Description
GET api/core/{serverHost}/PathPicker?parentPath= {parentPath}&memberType={memberType}&canWrite= {canWrite}&appId={appId}	Retrieves a list of all child component names, descriptions and primitive member names for a certain XHQ parent component.
GET api/core/{serverHost}/PathPicker?nameFilter= {nameFilter}&startRow={startRow}&maxRows={maxRows}	Retrieves a list of tags/aliases from XHQ.
GET api/core/{serverHost}/PathPicker?path={path} &appId={appId}	Validates a XHQ path by checking if the user can read and write to that specific path for the specified PM App (e.g. eLogs, TM).
GET api/core/PathPicker?serverHost={serverHost} &parentPath={parentPath}&memberType={memberType} &canWrite={canWrite}&appId={appId}	Retrieves a list of all child component names, descriptions and primitive member names for a certain XHQ parent component.
GET api/core/PathPicker?serverHost={serverHost} &nameFilter={nameFilter}&startRow={startRow} &maxRows={maxRows}	Retrieves a list of tags/aliases from XHQ.
GET api/core/PathPicker?serverHost={serverHost}&path= {path}&appId={appId}	Validates a XHQ path by checking if the user can read and write to that specific path for the specified PM App

API	Description
	(e.g. eLogs, TM).

Ping

API	Description
GET api/core/{serverHost}/Ping	Pings the web service by trying to establish a connection to the XHQ Solution and Model Server. This will load the Client API (Solution and Model) DLLs, so the next subsequent calls to the service will be much faster. This method is to be used as the first call to the service, as a "warm up" for the Web API service (e.g. from within the Keep Alive (WAKA) service).
GET api/core/Ping?serverHost={serverHost}	Pings the web service by trying to establish a connection to the XHQ Solution and Model Server. This will load the Client API (Solution and Model) DLLs, so the next subsequent calls to the service will be much faster. This method is to be used as the first call to the service, as a "warm up" for the Web API service (e.g. from within the Keep Alive (WAKA) service).

XHQ eLogs Module Methods

RESTful API calls for accessing XHQ eLogs functionality.

A **detailed** description of each method is available from the following URL:

`http://<XHQ Web Interface Webapp Hostname>/indx/xhqwi/help`

Example:

`http://localhost/indx/xhqwi/help`.

AssocTypes

API	Description
GET api/eLogs/{serverHost}/AssocTypes/{id}	Retrieves an eLog Association Type.
GET api/eLogs/{serverHost}/AssocTypes?ordered={ordered}	Retrieves all Association Types for eLogs.
POST api/eLogs/{serverHost}/AssocTypes	Creates a new Association Type in eLogs.
PUT api/eLogs/{serverHost}/AssocTypes	Edits (updates) an existing Association Type in eLogs.
DELETE api/eLogs/{serverHost}/AssocTypes/{id}	Deletes an existing Association Type in eLogs.
GET api/eLogs/AssocTypes/{id}?serverHost={serverHost}	Retrieves an eLog Association Type.
GET api/eLogs/AssocTypes?serverHost={serverHost} &ordered={ordered}	Retrieves all Association Types for eLogs.
POST api/eLogs/AssocTypes?serverHost={serverHost}	Creates a new Association Type in eLogs.
PUT api/eLogs/AssocTypes?serverHost={serverHost}	Edits (updates) an existing Association Type in eLogs.
DELETE api/eLogs/AssocTypes/{id}?serverHost= {serverHost}	Deletes an existing Association Type in eLogs.

Conditions

API	Description
GET api/eLogs/{serverHost}/Conditions/{id}	Retrieves an eLog Routine Condition.
GET api/eLogs/{serverHost}/Conditions	Retrieves all Routine Conditions for eLogs.
POST api/eLogs/{serverHost}/Conditions	Creates a new Routine Condition in eLogs.
PUT api/eLogs/{serverHost}/Conditions	Edits (updates) an existing Routine Condition in eLogs.
DELETE api/eLogs/{serverHost}/Conditions/{id}	Deletes an existing Routine Condition from eLogs.
GET api/eLogs/Conditions/{id}?serverHost={serverHost}	Retrieves an eLog Routine Condition.
GET api/eLogs/Conditions?serverHost={serverHost}	Retrieves all Routine Conditions for eLogs.
POST api/eLogs/Conditions?serverHost={serverHost}	Creates a new Routine Condition in eLogs.
PUT api/eLogs/Conditions?serverHost={serverHost}	Edits (updates) an existing Routine Condition in eLogs.
DELETE api/eLogs/Conditions/{id}?serverHost={serverHost}	Deletes an existing Routine Condition from eLogs.

CondLists

API	Description
GET api/eLogs/{serverHost}/CondLists/{id}	Retrieves an eLog Routine Condition List.

API	Description
GET api/eLogs/{serverHost}/CondLists	Retrieves all Routine Condition Lists from eLogs.
POST api/eLogs/{serverHost}/CondLists	Creates a new Routine Condition List in eLogs.
PUT api/eLogs/{serverHost}/CondLists	Edits (updates) an existing Routine Condition List in eLogs.
DELETE api/eLogs/{serverHost}/CondLists/{id}	Deletes an existing Routine Condition List from eLogs.
GET api/eLogs/CondLists/{id}?serverHost={serverHost}	Retrieves an eLog Routine Condition List.
GET api/eLogs/CondLists?serverHost={serverHost}	Retrieves all Routine Condition Lists from eLogs.
POST api/eLogs/CondLists?serverHost={serverHost}	Creates a new Routine Condition List in eLogs.
PUT api/eLogs/CondLists?serverHost={serverHost}	Edits (updates) an existing Routine Condition List in eLogs.
DELETE api/eLogs/CondLists/{id}?serverHost={serverHost}	Deletes an existing Routine Condition List from eLogs.

CondTypes

API	Description
GET api/eLogs/{serverHost}/CondTypes/{id}	Retrieves an eLog Routine Condition Type.
GET api/eLogs/{serverHost}/CondTypes	Retrieves all Routine Condition Types for eLogs.
POST api/eLogs/{serverHost}/CondTypes	Creates a new Routine Condition Type in eLogs.
PUT api/eLogs/{serverHost}/CondTypes	Edits (updates) an existing Routine Condition Type in eLogs.
DELETE api/eLogs/{serverHost}/CondTypes/{id}	Deletes an existing Routine Condition Type in eLogs.
GET api/eLogs/CondTypes/{id}?serverHost={serverHost}	Retrieves an eLog Routine Condition Type.
GET api/eLogs/CondTypes?serverHost={serverHost}	Retrieves all Routine Condition Types for eLogs.
POST api/eLogs/CondTypes?serverHost={serverHost}	Creates a new Routine Condition Type in eLogs.
PUT api/eLogs/CondTypes?serverHost={serverHost}	Edits (updates) an existing Routine Condition Type in eLogs.
DELETE api/eLogs/CondTypes/{id}?serverHost={serverHost}	Deletes an existing Routine Condition Type in eLogs.

Defaults

API	Description
GET api/eLogs/{serverHost}/Defaults/{id}	Retrieves an eLog Default Set by ID.
GET api/eLogs/{serverHost}/Defaults	Retrieves all Default Sets for eLogs.
POST api/eLogs/{serverHost}/Defaults	Creates a new Default Set in eLogs.
PUT api/eLogs/{serverHost}/Defaults	Edits (updates) an existing Default Set in eLogs.
DELETE api/eLogs/{serverHost}/Defaults/{id}	Deletes an existing Default Set in eLogs.
GET api/eLogs/Defaults/{id}?serverHost={serverHost}	Retrieves an eLog Default Set by ID.
GET api/eLogs/Defaults?serverHost={serverHost}	Retrieves all Default Sets for eLogs.
POST api/eLogs/Defaults?serverHost={serverHost}	Creates a new Default Set in eLogs.
PUT api/eLogs/Defaults?serverHost={serverHost}	Edits (updates) an existing Default Set in eLogs.
DELETE api/eLogs/Defaults/{id}?serverHost={serverHost}	Deletes an existing Default Set in eLogs.

ExportConfigs

API	Description
GET api/eLogs/{serverHost} /ExportConfigs?exportLogConfigs={exportLogConfigs} &exportRoutineConfigs={exportRoutineConfigs} &exportReportConfigs={exportReportConfigs} &exportUserConfigs={exportUserConfigs}	Exports eLogs Admin configurations into XML.
GET api/eLogs/ExportConfigs?serverHost={serverHost} &exportLogConfigs={exportLogConfigs} &exportRoutineConfigs={exportRoutineConfigs} &exportReportConfigs={exportReportConfigs} &exportUserConfigs={exportUserConfigs}	Exports eLogs Admin configurations into XML.

GenericUsers

API	Description
GET api/eLogs/{serverHost}/GenericUsers	Retrieves all Generic Users for eLogs.
POST api/eLogs/{serverHost}/GenericUsers?userName={userName}	Creates a new Generic User in eLogs.
DELETE api/eLogs/{serverHost}/GenericUsers/{id}	Deletes an existing Generic User in eLogs.
GET api/eLogs/GenericUsers?serverHost={serverHost}	Retrieves all Generic Users for eLogs.
POST api/eLogs/GenericUsers?serverHost={serverHost} &userName={userName}	Creates a new Generic User in eLogs.
DELETE api/eLogs/GenericUsers/{id}?serverHost={serverHost}	Deletes an existing Generic User in eLogs.

ImportConfigs

API	Description
POST api/eLogs/{serverHost} /ImportConfigs?updateConfigsById={updateConfigsById} &importLogConfigs={importLogConfigs} &importRoutineConfigs={importRoutineConfigs} &importReportConfigs={importReportConfigs} &importUserConfigs={importUserConfigs}	Imports eLogs Admin configurations from XML.
POST api/eLogs/ImportConfigs?serverHost={serverHost} &updateConfigsById={updateConfigsById} &importLogConfigs={importLogConfigs} &importRoutineConfigs={importRoutineConfigs} &importReportConfigs={importReportConfigs} &importUserConfigs={importUserConfigs}	Imports eLogs Admin configurations from XML.

KeyWords

API	Description
GET api/eLogs/{serverHost}/KeyWords	Retrieves all Keywords for eLogs.
GET api/eLogs/KeyWords?serverHost={serverHost}	Retrieves all Keywords for eLogs.

LogClasses

API	Description
GET api/eLogs/{serverHost}/LogClasses	Retrieves all Loc Classes (Sub-Types) for the eLogs app
GET api/eLogs/LogClasses?serverHost={serverHost}	Retrieves all Loc Classes (Sub-Types) for the eLogs app

LogTypes

API	Description
GET api/eLogs/{serverHost}/LogTypes	Retrieves all Log Types for the eLogs app
GET api/eLogs/{serverHost}/LogTypes?codeType={codeType}	Retrieves a list of codes of a certain type from ANS.
GET api/eLogs/LogTypes?serverHost={serverHost}	Retrieves all Log Types for the eLogs app
GET api/eLogs/LogTypes?serverHost={serverHost}&codeType={codeType}	Retrieves a list of codes of a certain type from ANS.

MultiDefaults

API	Description
GET api/eLogs/{serverHost}/MultiDefaults/{id}	Retrieves an eLog Multiple Default Set by ID.
GET api/eLogs/{serverHost}/MultiDefaults	Retrieves all Multiple Default Sets for eLogs.
POST api/eLogs/{serverHost}/MultiDefaults	Creates a new Multiple Default Set in eLogs.
PUT api/eLogs/{serverHost}/MultiDefaults	Edits (updates) an existing Multiple Default Set in eLogs.
DELETE api/eLogs/{serverHost}/MultiDefaults/{id}	Deletes an existing Multiple Default Set in eLogs.
GET api/eLogs/MultiDefaults/{id}?serverHost={serverHost}	Retrieves an eLog Multiple Default Set by ID.
GET api/eLogs/MultiDefaults?serverHost={serverHost}	Retrieves all Multiple Default Sets for eLogs.
POST api/eLogs/MultiDefaults?serverHost={serverHost}	Creates a new Multiple Default Set in eLogs.
PUT api/eLogs/MultiDefaults?serverHost={serverHost}	Edits (updates) an existing Multiple Default Set in eLogs.
DELETE api/eLogs/MultiDefaults/{id}?serverHost={serverHost}	Deletes an existing Multiple Default Set in eLogs.

OperLogs

API	Description
GET api/eLogs/{serverHost}/OperLogs/{id}	Retrieves an Operational Log.
GET api/eLogs/{serverHost}/OperLogs?startDate={startDate}&endDate={endDate}&startDateStr={startDateStr}&endDateStr={endDateStr}&logTypeId={logTypeId}&logClassId={logClassId}&logPriorityId=	Retrieves a list of Operational Logs.

API	Description
{logPriorityId}&logStatusId={logStatusId}&xhqPath={xhqPath}&shortText={shortText}&longText={longText}&keyWord={keyWord}&assocDesc={assocDesc}&inactiveFlg={inactiveFlg}&expiredFlg={expiredFlg}	
POST api/eLogs/{serverHost}/OperLogs	Creates a new Operational Log in eLogs.
POST api/eLogs/{serverHost}/OperLogs?count={count}	Creates multiple Operational Logs in eLogs.
PUT api/eLogs/{serverHost}/OperLogs	Edits (updates) an existing Operational Log in XHQ.
PUT api/eLogs/{serverHost}/OperLogs/{id}?status={status}&inactive={inactive}&expired={expired}	Updates the status for an Operational Log in XHQ.
GET api/eLogs/OperLogs/{id}?serverHost={serverHost}	Retrieves an Operational Log.
GET api/eLogs/OperLogs?serverHost={serverHost}&startDate={startDate}&endDate={endDate}&startDateStr={startDateStr}&endDateStr={endDateStr}&logTypeId={logTypeId}&logClassId={logClassId}&logPriorityId={logPriorityId}&logStatusId={logStatusId}&xhqPath={xhqPath}&shortText={shortText}&longText={longText}&keyWord={keyWord}&assocDesc={assocDesc}&inactiveFlg={inactiveFlg}&expiredFlg={expiredFlg}	Retrieves a list of Operational Logs.
POST api/eLogs/OperLogs?serverHost={serverHost}	Creates a new Operational Log in eLogs.
POST api/eLogs/OperLogs?serverHost={serverHost}&count={count}	Creates multiple Operational Logs in eLogs.
PUT api/eLogs/OperLogs?serverHost={serverHost}	Edits (updates) an existing Operational Log in XHQ.
PUT api/eLogs/OperLogs/{id}?serverHost={serverHost}&status={status}&inactive={inactive}&expired={expired}	Updates the status for an Operational Log in XHQ.

ParamGroups

API	Description
GET api/eLogs/{serverHost}/ParamGroups/{id}	Retrieves an eLog Parameter Group.
GET api/eLogs/{serverHost}/ParamGroups	Retrieves all Parameter Groups for eLogs.
POST api/eLogs/{serverHost}/ParamGroups	Creates a new Routine Parameter Group in eLogs.
PUT api/eLogs/{serverHost}/ParamGroups	Edits (updates) an existing Routine Parameter Group in eLogs.
DELETE api/eLogs/{serverHost}/ParamGroups/{id}	Deletes an existing Routine Parameter Group from eLogs.
GET api/eLogs/ParamGroups/{id}?serverHost={serverHost}	Retrieves an eLog Parameter Group.
GET api/eLogs/ParamGroups?serverHost={serverHost}	Retrieves all Parameter Groups for eLogs.
POST api/eLogs/ParamGroups?serverHost={serverHost}	Creates a new Routine Parameter Group in eLogs.
PUT api/eLogs/ParamGroups?serverHost={serverHost}	Edits (updates) an existing Routine Parameter Group in eLogs.
DELETE api/eLogs/ParamGroups/{id}?serverHost={serverHost}	Deletes an existing Routine Parameter Group from eLogs.

ParamTypes

API	Description
GET api/eLogs/{serverHost}/ParamTypes/{id}	Retrieves an eLog Routine Parameter Type.
GET api/eLogs/{serverHost}/ParamTypes	Retrieves all Routine Parameter Types for eLogs.
POST api/eLogs/{serverHost}/ParamTypes	Creates a new Routine Parameter Type in eLogs.
PUT api/eLogs/{serverHost}/ParamTypes	Edits (updates) an existing Routine Parameter Type in eLogs.
DELETE api/eLogs/{serverHost}/ParamTypes/{id}	Deletes an existing Routine Parameter Type in eLogs.
GET api/eLogs/ParamTypes/{id}?serverHost={serverHost}	Retrieves an eLog Routine Parameter Type.
GET api/eLogs/ParamTypes?serverHost={serverHost}	Retrieves all Routine Parameter Types for eLogs.
POST api/eLogs/ParamTypes?serverHost={serverHost}	Creates a new Routine Parameter Type in eLogs.
PUT api/eLogs/ParamTypes?serverHost={serverHost}	Edits (updates) an existing Routine Parameter Type in eLogs.
DELETE api/eLogs/ParamTypes/{id}?serverHost={serverHost}	Deletes an existing Routine Parameter Type in eLogs.

Priorities

API	Description
GET api/eLogs/{serverHost}/Priorities	Retrieves all Priorities for the eLogs app
GET api/eLogs/Priorities?serverHost={serverHost}	Retrieves all Priorities for the eLogs app

Reports

API	Description
GET api/eLogs/{serverHost}/Reports/{id}	Retrieves an eLog Report Definition.
GET api/eLogs/{serverHost}/Reports	Retrieves all Reports Definitions for eLogs.
POST api/eLogs/{serverHost}/Reports	Creates a new Report Definition in eLogs.
PUT api/eLogs/{serverHost}/Reports	Edits (updates) an existing Report Definition in eLogs.
DELETE api/eLogs/{serverHost}/Reports/{id}	Deletes an existing Report Definition from eLogs.
GET api/eLogs/Reports/{id}?serverHost={serverHost}	Retrieves an eLog Report Definition.
GET api/eLogs/Reports?serverHost={serverHost}	Retrieves all Reports Definitions for eLogs.
POST api/eLogs/Reports?serverHost={serverHost}	Creates a new Report Definition in eLogs.
PUT api/eLogs/Reports?serverHost={serverHost}	Edits (updates) an existing Report Definition in eLogs.
DELETE api/eLogs/Reports/{id}?serverHost={serverHost}	Deletes an existing Report Definition from eLogs.

RoutineParameters

API	Description
GET api/eLogs/{serverHost}/RoutineParameters/{id}	Retrieves an eLog Routine Parameter.
GET api/eLogs/{serverHost}/RoutineParameters	Retrieves all Routine Parameters for eLogs.
POST api/eLogs/{serverHost}/RoutineParameters	Creates a new Routine Parameter in eLogs.

API	Description
PUT api/eLogs/{serverHost}/RoutineParameters	Edits (updates) an existing Routine Parameter in eLogs.
DELETE api/eLogs/{serverHost}/RoutineParameters/{id}	Deletes an existing Routine Parameter from eLogs.
GET api/eLogs/RoutineParameters/{id}?serverHost={serverHost}	Retrieves an eLog Routine Parameter.
GET api/eLogs/RoutineParameters?serverHost={serverHost}	Retrieves all Routine Parameters for eLogs.
POST api/eLogs/RoutineParameters?serverHost={serverHost}	Creates a new Routine Parameter in eLogs.
PUT api/eLogs/RoutineParameters?serverHost={serverHost}	Edits (updates) an existing Routine Parameter in eLogs.
DELETE api/eLogs/RoutineParameters/{id}?serverHost={serverHost}	Deletes an existing Routine Parameter from eLogs.

Routines

API	Description
GET api/eLogs/{serverHost}/Routines/{id}	Retrieves an eLog Routine.
GET api/eLogs/{serverHost}/Routines?paramGrpId={paramGrpId}	Retrieves all eLogs Routines for a certain Parameter Group.
GET api/eLogs/{serverHost}/Routines?reportId={reportId}	Retrieves all eLogs Routines for a certain Shift Report.
POST api/eLogs/{serverHost}/Routines	Creates multiple (one or many) Routines in eLogs.
GET api/eLogs/Routines/{id}?serverHost={serverHost}	Retrieves an eLog Routine.
GET api/eLogs/Routines?serverHost={serverHost}&paramGrpId={paramGrpId}	Retrieves all eLogs Routines for a certain Parameter Group.
GET api/eLogs/Routines?serverHost={serverHost}&reportId={reportId}	Retrieves all eLogs Routines for a certain Shift Report.
POST api/eLogs/Routines?serverHost={serverHost}	Creates multiple (one or many) Routines in eLogs.

ShiftReports

API	Description
GET api/eLogs/{serverHost}/ShiftReports/{id}	Retrieves an eLog Shift Report.
POST api/eLogs/{serverHost}/ShiftReports	Creates a new Shift Report in eLogs.
PUT api/eLogs/{serverHost}/ShiftReports	Edits (updates) an existing Shift Report in eLogs.
GET api/eLogs/ShiftReports/{id}?serverHost={serverHost}	Retrieves an eLog Shift Report.
POST api/eLogs/ShiftReports?serverHost={serverHost}	Creates a new Shift Report in eLogs.
PUT api/eLogs/ShiftReports?serverHost={serverHost}	Edits (updates) an existing Shift Report in eLogs.

Units

API	Description
GET api/eLogs/{serverHost}/Units/{id}	Retrieves an eLog Unit of Measurement (UOM).
GET api/eLogs/{serverHost}/Units	Retrieves all Unit of Measurements (UOM) for eLogs.
POST api/eLogs/{serverHost}/Units	Creates a new Unit of Measurements (UOM) in eLogs.
PUT api/eLogs/{serverHost}/Units	Edits (updates) an existing Unit of Measurements (UOM)

API	Description
	in eLogs.
DELETE api/eLogs/{serverHost}/Units/{id}	Deletes an existing Unit of Measurement (UOM) in eLogs.
GET api/eLogs/Units/{id}?serverHost={serverHost}	Retrieves an eLog Unit of Measurement (UOM).
GET api/eLogs/Units?serverHost={serverHost}	Retrieves all Unit of Measurements (UOM) for eLogs.
POST api/eLogs/Units?serverHost={serverHost}	Creates a new Unit of Measurements (UOM) in eLogs.
PUT api/eLogs/Units?serverHost={serverHost}	Edits (updates) an existing Unit of Measurements (UOM) in eLogs.
DELETE api/eLogs/Units/{id}?serverHost={serverHost}	Deletes an existing Unit of Measurement (UOM) in eLogs.

XHQ RDS Module Methods

RESTful API calls for accessing XHQ RDS functionality.

A **detailed** description of each method is available from the following URL:

`http://<XHQ Web Interface Webapp Hostname>/indx/xhqwi/help`

Example:

`http://localhost/indx/xhqwi/help`.

Logs

API	Description
GET api/rds/{serverHost}/Logs?startTime={startTime}&endTime={endTime}	Retrieves a list of all logs in RDS.
GET api/rds/Logs?serverHost={serverHost}&startTime={startTime}&endTime={endTime}	Retrieves a list of all logs in RDS.

SubsCol

API	Description
GET api/rds/{serverHost}/SubsCol/{id}	Retrieves a subscribed collection and its SQL info by its ID in RDS.
GET api/rds/{serverHost}/SubsCol	Retrieves a list of all subscribed collections in RDS.
POST api/rds/{serverHost}/SubsCol	Creates a new subscribed collection and its SQL info in RDS.
PUT api/rds/{serverHost}/SubsCol	Updates an existing subscribed collection and its SQL info in RDS.
DELETE api/rds/{serverHost}/SubsCol/{id}	Deletes a subscribed collection and its associated SQL info.
GET api/rds/SubsCol/{id}?serverHost={serverHost}	Retrieves a subscribed collection and its SQL info by its ID in RDS.
GET api/rds/SubsCol?serverHost={serverHost}	Retrieves a list of all subscribed collections in RDS.
POST api/rds/SubsCol?serverHost={serverHost}	Creates a new subscribed collection and its SQL info in RDS.
PUT api/rds/SubsCol?serverHost={serverHost}	Updates an existing subscribed collection and its SQL info in RDS.
DELETE api/rds/SubsCol/{id}?serverHost={serverHost}	Deletes a subscribed collection and its associated SQL info.

XHQ Target Management Module Methods

RESTful API calls for accessing XHQ Target Management functionality.

A **detailed** description of each method is available from the following URL:

`http://<XHQ Web Interface Webapp Hostname>/indx/xhqwi/help`

Example:

`http://localhost/indx/xhqwi/help`.

ExportKpis

API	Description
POST api/tm/{serverHost}/ExportKpis	Exports into XML format a list of KPIs from the App server, identified by their Ids.
POST api/tm/ExportKpis?serverHost={serverHost}	Exports into XML format a list of KPIs from the App server, identified by their Ids.

ImportKpis

API	Description
POST api/tm/{serverHost}/ImportKpis	Imports from XML format a list of KPIs in the App server.
POST api/tm/ImportKpis?serverHost={serverHost}	Imports from XML format a list of KPIs in the App server.

Kpi

API	Description
GET api/tm/{serverHost}/Kpi/{id}	Retrieves a complete key performance indicator (KPI).
GET api/tm/{serverHost}/Kpi?c={c}&name={name}&metric={metric}&anchorPath={anchorPath}&type={type}&subType={subType}&state={state}	Retrieves the total count for all KPIs for the calling user.
GET api/tm/{serverHost}/Kpi?pageSize={pageSize}&pageIndex={pageIndex}&orderByColumn={orderByColumn}&orderAsc={orderAsc}&name={name}&metric={metric}&anchorPath={anchorPath}&type={type}&subType={subType}&state={state}	Retrieves a list of key performance indicators (KPIs).
POST api/tm/{serverHost}/Kpi	Creates a new performance indicator (KPI).
PUT api/tm/{serverHost}/Kpi	Updates an existing performance indicator (KPI).
DELETE api/tm/{serverHost}/Kpi/{id}	Deletes an existing performance indicator.
GET api/tm/Kpi/{id}?serverHost={serverHost}	Retrieves a complete key performance indicator (KPI).
GET api/tm/Kpi?serverHost={serverHost}&c={c}&name={name}&metric={metric}&anchorPath={anchorPath}&type={type}&subType={subType}&state={state}	Retrieves the total count for all KPIs for the calling user.
GET api/tm/Kpi?serverHost={serverHost}&pageSize={pageSize}&pageIndex={pageIndex}&orderByColumn=	Retrieves a list of key performance indicators (KPIs).

API	Description
{orderByColumn}&orderAsc={orderAsc}&name={name}&metric={metric}&anchorPath={anchorPath}&type={type}&subType={subType}&state={state}	
POST api/tm/Kpi?serverHost={serverHost}	Creates a new performance indicator (KPI).
PUT api/tm/Kpi?serverHost={serverHost}	Updates an existing performance indicator (KPI).
DELETE api/tm/Kpi/{id}?serverHost={serverHost}	Deletes an existing performance indicator.

Limits

API	Description
GET api/tm/{serverHost}/Limits?piId={piId}	Retrieves a complete history (audit trail) for all the limits for a KPI .
GET api/tm/Limits?serverHost={serverHost}&piId={piId}	Retrieves a complete history (audit trail) for all the limits for a KPI .

Subscribers

API	Description
GET api/tm/{serverHost}/Subscribers?type={type}&subtype={subtype}&path={path}&useFilter={useFilter}	Retrieves the list of all subscribers users for a certain KPI limit.
GET api/tm/Subscribers?serverHost={serverHost}&type={type}&subtype={subtype}&path={path}&useFilter={useFilter}	Retrieves the list of all subscribers users for a certain KPI limit.

Targets

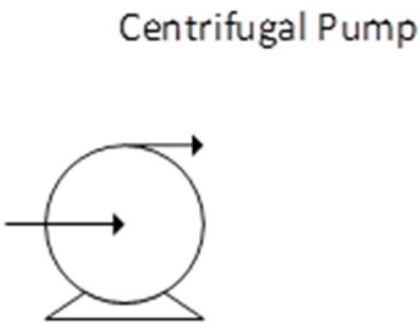
API	Description
GET api/tm/{serverHost}/Targets?piId={piId}&targetId={targetId}	Retrieves a complete target history (audit trail) for a KPI .
GET api/tm/Targets?serverHost={serverHost}&piId={piId}&targetId={targetId}	Retrieves a complete target history (audit trail) for a KPI .

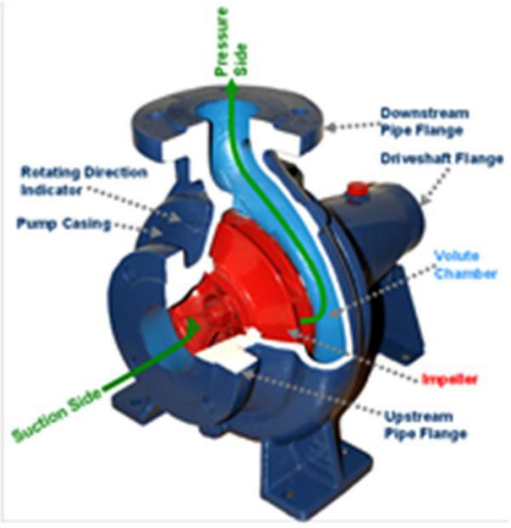
How to retrieve information from XHQ using the REST web services

From the given pump example, you will learn how to do the following tasks using the REST web services:

- Get tag data
- Get a list of assets
- Get attributes for a particular pump

Given: A set of pumps modeled in XHQ and with historical data.

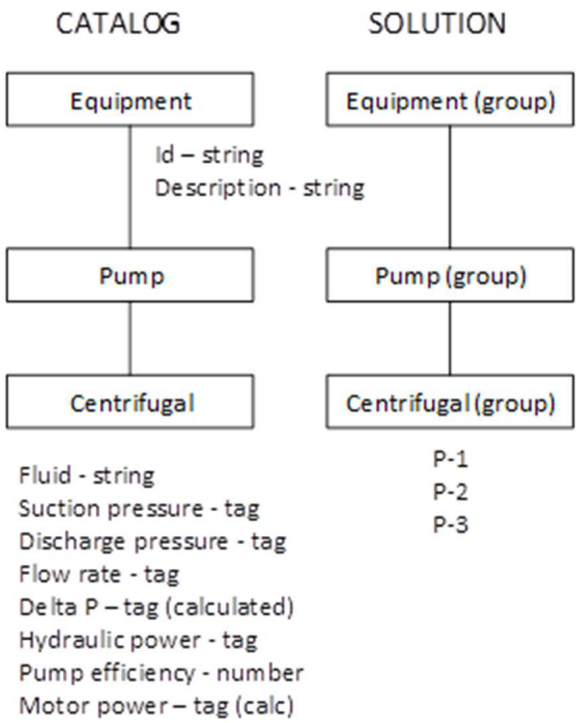




Pump Descriptions

Attribute	Pump 1	Pump 2	Pump 3
Pump Name	P-106A	P-201	P-319
Type	Centrifugal	Centrifugal	Centrifugal
Fluid	Water	Naphtha	Water
Specific Gravity	1.000	0.655	1.000
Flow Rate	79 t/hr 350 gpm	150 t/hr 700 gpm	1.1 t/hr 5 gpm
Suction Pressure	0.6 bar (a) -6 psig	1.36 bar (a) 5 psig	1.22 bar (a) 3 psig
Discharge Pressure	53 bar (a) 755 psig	6.18 bar 75 psig	5.67 bar 67.5 psia
Suction Temperature	38 °C 100 °F	65 °C 150 °F	20 °C 68 °F

Attribute	Pump 1	Pump 2	Pump 3
Hydraulic Power	116 kW 156 bhp	21.3 kW 28.9 bhp	0.14 kW 0.19 bhp
Pump Efficiency	75%	75%	50%
Power Delivered by Pump	155 kW 207 bhp	28.7 kW 38.5 bhp	0.28 W 0.38 bhp
Pipe Size	6 in	8 in	61in



For all the examples in this section, the "Basic Authentication" must be enabled in IIS for the xhqwi web service.

And in order for the JS and VBA examples to work, you need to replace the value for the "Authentication" header attribute with valid user credentials.

Get tag data

This example shows you how to get data for the pump discharge pressure, knowing the tag name. Three variations are displayed, using different tools and with different security needs.

Example 1: XHQ Web API call (CurrentValue)

GET/api/core/{serverHost}/Points/CurrentValue

Gets the current VTQ values for a single point/primitive.

Curl

```
curl -X GET --header 'Accept: application/json'
'http://vcagmobile/indx/xhqwi/api/core/vcagmobile/Points/CurrentValue?
path=%3A%3ASample.Equipment.Pumps.Centrifugal.P106A.FlowRate.rCurrentValue'
```

Request URL

```
http://vcagmobile/indx/xhqwi/api/core/vcagmobile/Points/CurrentValue?
path=%3A%3ASample.Equipment.Pumps.Centrifugal.P106A.FlowRate.rCurrentValue
```

Response Body


```
{
  "Value": "223.69614",
  "Time": "2017-05-15 20:11:14.527",
  "Quality": "GOOD",
  "Units": " ",
  "Description": "Sox Environmental"
}
```

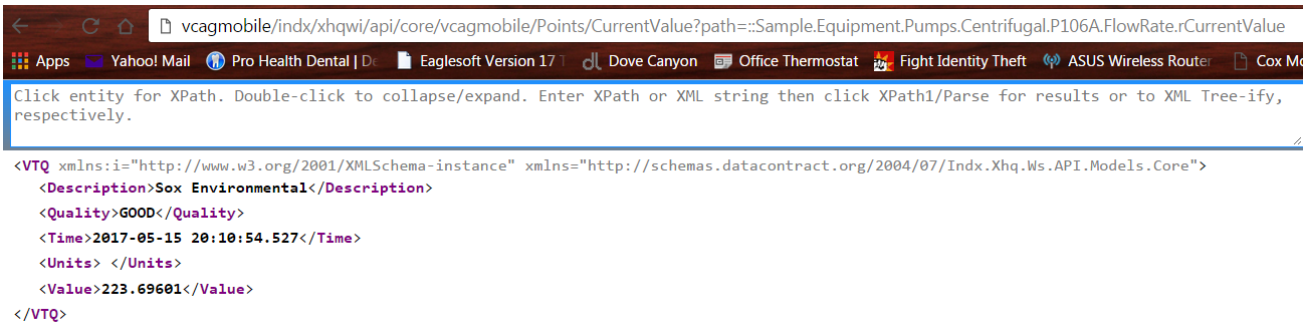

Response Code

```
200
```

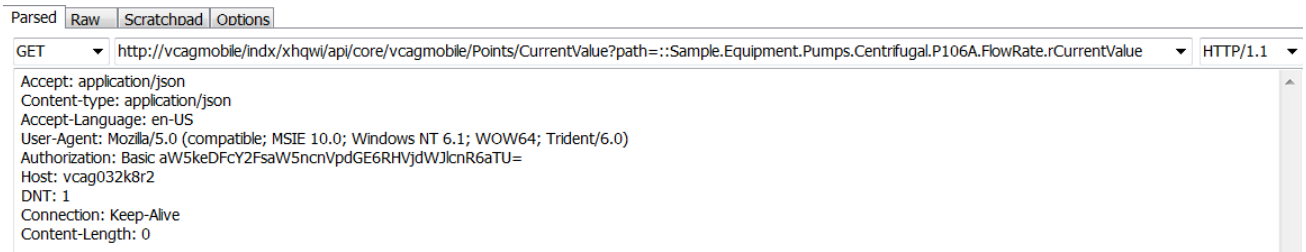
Response Headers


```
{
  "persistent-auth": "false",
  "pragma": "no-cache",
  "date": "Mon, 15 May 2017 20:11:28 GMT",
  "www-authenticate": "Negotiate
  oYG3MIG0oAMKAQChCwYJKoZIgvcSAQICooGfBIGcYIGZBgkqhkiG9xIBAgICAG
  +BiTCBhqADAgEFoQMCAQ+iejB4oAMCARKicQRvfPmnkaupkvH+srtfdlwQ+DtyHzihPVHqht89Dy8OvML/4
  Cb9/6WMfnP
  vSUm937/J8UQk5M8cvC3aWz62Z5Qavgt7jLbt33ZQzrTr5nUf3qZp/bSuDDzNpwcKoAz3Wucm2vdHa2HQhT
  jdGHqcFqXHnCLdk",
  "server": "Microsoft-IIS/7.5",
  "x-powered-by": "ASP.NET",
  "content-type": "application/json; charset=utf-8",
  "cache-control": "no-cache",
  "content-length": "117",
  "x-content-type-options": "nosniff",
```

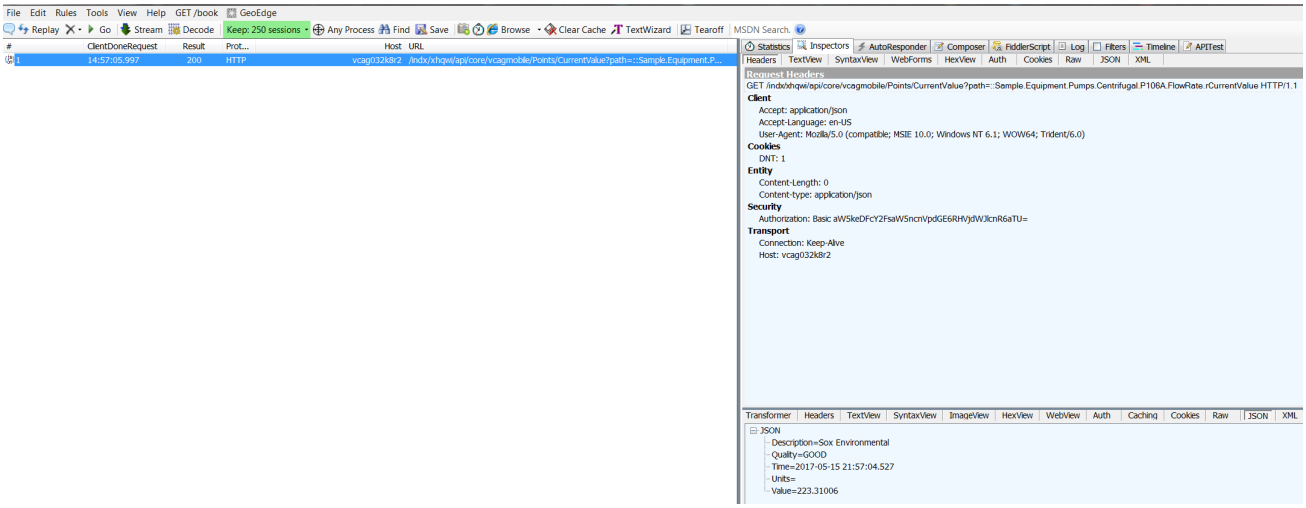
```
    "expires": "-1"
  }
```



Chrome Browser XML Response



HTTP Response in Fiddler



HTTP Response in Fiddler

Example 2: XHQ Web API call (TimeSeries)

`GET/api/core/{serverHost}/Points/TimeSeries`

Gets the Time Series data for a primitve specified through a path or alias as a string array of triplets Value Time Quality (VTQ) structure from the time range specified with the given characteristics.

Curl

```
curl -X GET --header 'Accept: application/json'
'http://vcagmobile/indx/xhqwi/api/core/vcagmobile/Points/TimeSeries?
path=%3A%3ASample.Equipment.Pumps.Centrifugal.P106A.FlowRate.rCurrentValue&startTime
=2017-05-14%2010%3A00%3A00.000&endTime=2017-05-15%2010%3A00%3A00.000&samplingMode
=TIMESLICED_AVG&noPoints=24'
```

Request URL

```
http://vcagmobile/indx/xhqwi/api/core/vcagmobile/Points/TimeSeries?path=%3A%3ASample.
Equipment.Pumps.Centrifugal.P106A.FlowRate.rCurrentValue&startTime=2017-05-
14%2010%3A00%3A00.000&endTime=2017-05-15%2010%3A00%3A00.000&samplingMode
=TIMESLICED_AVG&noPoints=24
```

Response Code

```
200
```

Response Headers


```
{
  "persistent-auth": "false",
  "pragma": "no-cache",
  "date": "Tue, 16 May 2017 19:01:41 GMT",
  "www-authenticate": "Negotiate
  oYG3MIG0oAMKAQChCwYJKoZIgvcSAQICooGfBIGcYIGZBgkqhkiG9xIBAgICAG+
  BiTCBhqADAgEFoQMCAQ+iejB4oAMCARKicQRvhD6MIMMGofJLDskt435dge6pcbG09WKlk4HJnQlT5xy3mi
  Xa9dIa3lg/
  JBlQXZN+FWmWH8XQfUe8dQgxKQ4u3m7WnbxBbykAT198lQp99B6WR9l7LiyZtlqy6mX3o3WkR94WBlAniLK
  JtDScfwOlvPnGJ",
  "server": "Microsoft-IIS/7.5",
  "x-powered-by": "ASP.NET",
  "content-type": "application/json; charset=utf-8",
  "cache-control": "no-cache",
  "content-length": "2829",
  "x-content-type-options": "nosniff",
  "expires": "-1"
}
```


Response Body


```
[
  {
    "Value": "231.93059",
    "Time": "2017-05-14 10:00:00.000",
    "Quality": "GOOD",
    "Units": " ",
    "Description": "Sox Environmental"
    },
    {
    "Value": "231.16803",
```

"Time": "2017-05-14 11:00:00.000",
"Quality": "GOOD",
"Units": " ",
"Description": "Sox Environmental"
},
{
"Value": "230.03737",
"Time": "2017-05-14 12:00:00.000",
"Quality": "GOOD",
"Units": " ",
"Description": "Sox Environmental"
},
{
"Value": "228.75035",
"Time": "2017-05-14 13:00:00.000",
"Quality": "GOOD",
"Units": " ",
"Description": "Sox Environmental"
},
{
"Value": "227.66469",
"Time": "2017-05-14 14:00:00.000",
"Quality": "GOOD",
"Units": " ",
"Description": "Sox Environmental"
},
{
"Value": "227.96533",
"Time": "2017-05-14 15:00:00.000",
"Quality": "GOOD",
"Units": " ",
"Description": "Sox Environmental"
},
{
"Value": "227.93565",
"Time": "2017-05-14 16:00:00.000",
"Quality": "GOOD",
"Units": " ",
"Description": "Sox Environmental"
},
{
"Value": "227.22646",

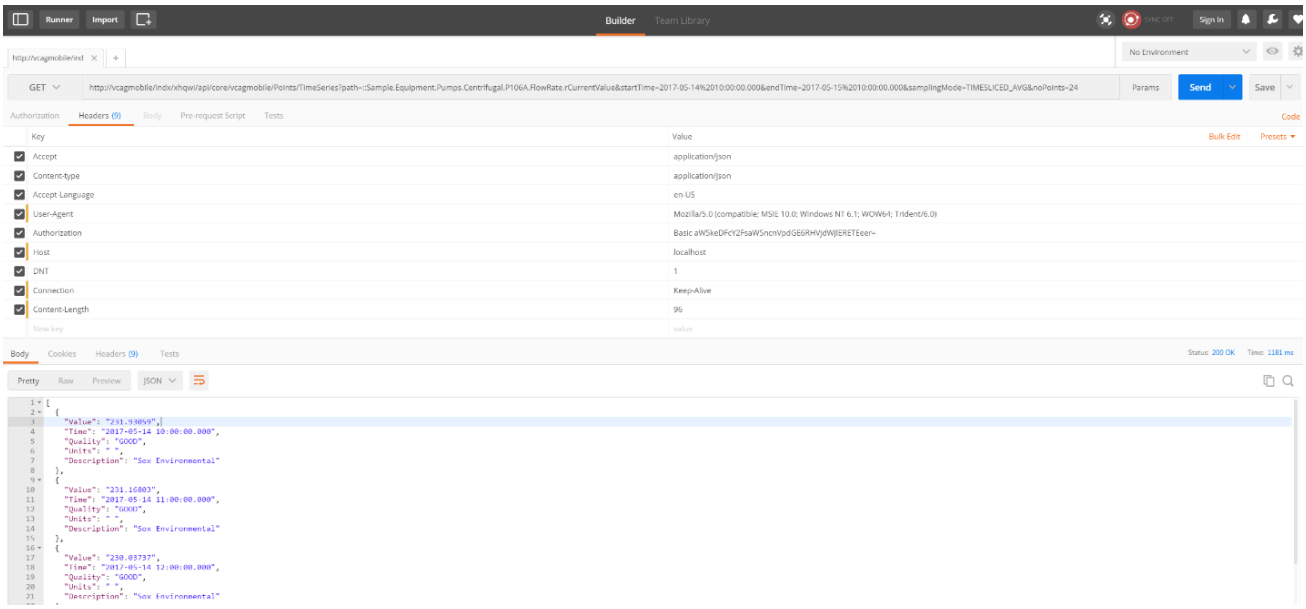
"Time": "2017-05-14 17:00:00.000",
"Quality": "GOOD",
"Units": " ",
"Description": "Sox Environmental"
},
{
"Value": "226.4638",
"Time": "2017-05-14 18:00:00.000",
"Quality": "GOOD",
"Units": " ",
"Description": "Sox Environmental"
},
{
"Value": "223.86302",
"Time": "2017-05-14 19:00:00.000",
"Quality": "GOOD",
"Units": " ",
"Description": "Sox Environmental"
},
{
"Value": "222.85477",
"Time": "2017-05-14 20:00:00.000",
"Quality": "GOOD",
"Units": " ",
"Description": "Sox Environmental"
},
{
"Value": "223.32388",
"Time": "2017-05-14 21:00:00.000",
"Quality": "GOOD",
"Units": " ",
"Description": "Sox Environmental"
},
{
"Value": "224.29916",
"Time": "2017-05-14 22:00:00.000",
"Quality": "GOOD",
"Units": " ",
"Description": "Sox Environmental"
},
{
"Value": "222.21317",

"Time": "2017-05-14 23:00:00.000",
"Quality": "GOOD",
"Units": " ",
"Description": "Sox Environmental"
},
{
"Value": "222.08551",
"Time": "2017-05-15 00:00:00.000",
"Quality": "GOOD",
"Units": " ",
"Description": "Sox Environmental"
},
{
"Value": "224.20691",
"Time": "2017-05-15 01:00:00.000",
"Quality": "GOOD",
"Units": " ",
"Description": "Sox Environmental"
},
{
"Value": "225.0067",
"Time": "2017-05-15 02:00:00.000",
"Quality": "GOOD",
"Units": " ",
"Description": "Sox Environmental"
},
{
"Value": "223.72154",
"Time": "2017-05-15 03:00:00.000",
"Quality": "GOOD",
"Units": " ",
"Description": "Sox Environmental"
},
{
"Value": "225.42622",
"Time": "2017-05-15 04:00:00.000",
"Quality": "GOOD",
"Units": " ",
"Description": "Sox Environmental"
},
{
"Value": "223.1285",

```
      "Time": "2017-05-15 05:00:00.000",
      "Quality": "GOOD",
      "Units": " ",
      "Description": "Sox Environmental"
      },
      {
      "Value": "222.37585",
      "Time": "2017-05-15 06:00:00.000",
      "Quality": "GOOD",
      "Units": " ",
      "Description": "Sox Environmental"
      },
      {
      "Value": "222.45024",
      "Time": "2017-05-15 07:00:00.000",
      "Quality": "GOOD",
      "Units": " ",
      "Description": "Sox Environmental"
      },
      {
      "Value": "223.5271",
      "Time": "2017-05-15 08:00:00.000",
      "Quality": "GOOD",
      "Units": " ",
      "Description": "Sox Environmental"
      },
      {
      "Value": "222.66367",
      "Time": "2017-05-15 09:00:00.000",
      "Quality": "GOOD",
      "Units": " ",
      "Description": "Sox Environmental"
    }
]
```

Postman Example

http://vcagmobile/indx/xhqwi/api/core/vcagmobile/Points/TimeSeries?path=::Sample.Equipment.Pumps.Centrifugal.P106A.FlowRate.rCurrentValue&startTime=2017-05-14%2010:00:00.000&endTime=2017-05-15%2010:00:00.000&samplingMode=TIMESLICED_AVG&noPoints=24



Get a list of assets

Gets a list of centrifugal pumps in some area of the solution.

In the following example, XHQ and user are in the same domain. Iterate through the list of pumps. Illustrate with JavaScript.

GET/api/core/{serverHost}/Metadata/ComponentMembers

Gets the child component members' names for a component specified by a path.

Curl

```
curl -X GET --header 'Accept: application/json'
'http://vcag032k8r2/indx/xhqwi/api/core/vcagmobile/Metadata/ComponentMembers?
parentPath=%3A%3ASample.Equipment.Pumps.Centrifugal&showHidden=false'
```

Request URL

```
http://vcag032k8r2/indx/xhqwi/api/core/vcagmobile/Metadata/ComponentMembers?
parentPath=%3A%3ASample.Equipment.Pumps.Centrifugal&showHidden=false
```

Response Body


```
[
  "P106A",
  "P201",
  "P319"
]
```


Response Code

```
200
```

Response Headers


```
{
  "persistent-auth": "false",
  "pragma": "no-cache",
  "date": "Wed, 17 May 2017 23:16:49 GMT",
  "www-authenticate": "Negotiate
  oYG3MIG0oAMKAQChCwYJKoZIgvcSAQICooGfBIGcYIGZBgkqhkiG9xIBAgICAG+
  BiTCBhqADAgEFoQMCAQ+iejB4oAMCARKicQRv3rX1YiD4JcE/P6e8fwgpzGXNT1gzflhsRC+2SFpOzJmKHW
  IzeHpgl/
  GI8cRgidMCVyT5YZyVLHzRoJwV/eI//tuzyEVXZT7mpFhntmSF8E8KFJuEJ40DPj5N9y81LpZlKtemX7Vpf
  2AOWyLu2bZ0",
  "server": "Microsoft-IIS/7.5",
  "x-powered-by": "ASP.NET",
  "content-type": "application/json; charset=utf-8",
  "cache-control": "no-cache",
  "content-length": "23",
  "x-content-type-options": "nosniff",
  "expires": "-1"
}
```

JavaScript Code Snippet

```
var pumpArea = "::Sample.Equipment.Pumps.Centrifugal"

$.ajax({
        method: "GET",
        url:
"http://localhost/indx/xhqwi/api/core/vcagmobile/Metadata/ComponentMembers?parentPath
="+pumpArea,
        headers: { "Accept": "application/json",
                   "Content-Type":"application/json"
                 },
    }).done(function(response) {
        // call successful, add response to the result div, formatted with <pre>
        $('#result').html('<pre>' + JSON.stringify(response, null, 5) + '</pre>');
    }).fail(function(e) {
        // call failed, show error in the result div
        $('#result').html('<span style="color: red">[' + e.status + '] ' +
e.responseText + '</span>');
 });
```

Complete HTML

The web API URL is introduced by the user in this case:

```
http://vcagmobile/indx/xhqwi/api/core/vcagmobile/Metadata/ComponentMembers?parentPath
=::Sample.Equipment.Pumps.Centrifugal
```

Due to browser security limitations, the JS examples code need to be run under a web app (or virtual directory) in IIS, on the same machine with “xhqwi”. And the web service name for the xqhwi web API URL call must always be “localhost” (this is due to CORS).

```
<html>
    <head>
        <title>Demo for JavaScript XHQ Web API call</title>
    </head>
    <body>
        <div id="form">
            Call link (GET method only): <br />
            <input type="text" id="call_url" style="width: 90%" /><br />
            <button id="call">Get Pump List</button>
        </div>

        <hr />

        <div id="result"></div>

        <!-- include jquery for ajax support -->
        <script type="text/javascript" src="js/jquery.js"></script>

        <!-- the script that makes the call on button click -->
        <script type="text/javascript">
            // start when all elements are loaded in the DOM
            $(document).ready(function() {
                // trigger this function when the call button is clicked
                $('#call').click(function() {
                    var callUrl = $('#call_url').val();

                    // execute only if call url is not an empty string
                    if(callUrl) {
                        $.ajax({
                            method: "GET",
                            url: callUrl,
                            headers: { "Accept": "application/json",
                                       "Content-Type":"application/json"
                                     },
                        }).done(function(response) {
                            // call successful, add response to the result div, formatted
with <pre>
                            $('#result').html('<pre>' + JSON.stringify(response, null, 5)
+ '</pre>');
                        }).fail(function(e) {
                            // call failed, show error in the result div
                            $('#result').html('<span style="color: red">[' + e.status +
'] ' + e.responseText + '</span>');
                        });
                    }
                });
            });
        </script>
    </body>
</html>
```

Get attributes for a particular pump

The following examples show you how to get a list of attributes and create a table that shows the attributes and values for a particular pump.

EXAMPLE 1

Given: The following VBA code for Excel 2013.

```
Sub XHQRESTAPICall()
  Dim xmlDoc As New MSXML2.DOMDocument
  Dim MyRequest As New WinHttp.WinHttpRequest
  Dim WrdArray() As String
  Dim AttributePath As String
  Dim Value As String
  Dim Units As String
  Const PumpPath As String = "::Sample.Equipment.Pumps.Centrifugal.P106A"

  Application.ScreenUpdating = False
  Application.Calculation = xlCalculationManual

  ' Set the table header
  Cells(3, 3) = "Pump P106A"
  Range("C3").Characters(1, 25).Font.Bold = True

  Cells(5, 3) = "Attributes"
  Range("C5").Characters(1, 25).Font.Bold = True
  Range("C5").Characters(1, 25).Font.Underline = True

  Cells(5, 6) = "Values"
  Range("F5").Characters(1, 25).Font.Bold = True
  Range("F5").Characters(1, 25).Font.Underline = True

  ' Set the GET request
  MyRequest.Open "GET",
  "http://localhost/indx/xhqwi/api/core/vcagmobile/Metadata/PrimitiveMembers?parentPath
  =" + PumpPath

  ' Set The HTTP request header attributes
  MyRequest.SetRequestHeader "Accept", "application/json"
  MyRequest.SetRequestHeader "Content-type", "application/json"
  MyRequest.SetRequestHeader "Authorization", "Basic dXNlcm5hbWU6cGFzc3dvcmQ="

  ' Send Request.
```

```
MyRequest.send

If (MyRequest.Status = 200) Then
  'And we get the response
  'MsgBox MyRequest.responseText

  WrdArray = Split(MyRequest.responseText, "[")
  WrdArray = Split(WrdArray(1), "]")
  WrdArray = Split(WrdArray(0), ",")

  i = 6
For Each s In WrdArray
  s = Replace(s, Chr(34), "")
  AttributePath = PumpPath + "." + s

  ' Set the GET request
  MyRequest.Open "GET",
  "http://localhost/indx/xhqwi/api/core/vcagmobile/Points/CurrentValue?path=" +
  AttributePath

  ' Set The HTTP request header attributes
  MyRequest.SetRequestHeader "Accept", "application/xml"
  MyRequest.SetRequestHeader "Content-type", "application/json"
  MyRequest.SetRequestHeader "Authorization", "Basic dXNlcm5hbWU6cGFzc3dvcmQ="

  ' Send Request.
  MyRequest.send

      If (MyRequest.Status = 200) Then
        'And we get the response
        xmlDoc.LoadXML (MyRequest.responseText)
        Value = xmlDoc.SelectSingleNode("/VTQ/Value").Text
        Units = xmlDoc.SelectSingleNode("/VTQ/Units").Text
      Else
        MsgBox "XHQ Web API Error: " + MyRequest.StatusText
      End If

  Cells(i, 3) = s
  Cells(i, 6) = Value + " " + Units
  i = i + 1
Next

  Cells.Replace What:="""", Replacement:="", LookAt:=xlPart
Else
```

```
  MsgBox "XHQ Web API Error: " + MyRequest.StatusText
End If

' Set the GET request
MyRequest.Open "GET",
"http://localhost/indx/xhqwi/api/core/vcagmobile/Metadata/ComponentMembers?parentPath=
::Sample.Equipment.Pumps.Centrifugal.P106A"

' Set The HTTP request header attributes
MyRequest.SetRequestHeader "Accept", "application/json"
MyRequest.SetRequestHeader "Content-type", "application/json"
MyRequest.SetRequestHeader "Authorization", "Basic dXNlcm5hbWU6cGFzc3dvcmQ="

' Send Request.
MyRequest.send

If (MyRequest.Status = 200) Then
  'And we get the response
  'MsgBox MyRequest.responseText

  WrdArray = Split(MyRequest.responseText, "[")
  WrdArray = Split(WrdArray(1), "]")
  WrdArray = Split(WrdArray(0), ",")

  i = 11
For Each s In WrdArray
  s = Replace(s, Chr(34), "")
  AttributePath = PumpPath + "." + s + ".rCurrentValue"

  ' Set the GET request
  MyRequest.Open "GET",
  "http://localhost/indx/xhqwi/api/core/vcagmobile/Points/CurrentValue?path=" +
  AttributePath

  ' Set The HTTP request header attributes
  MyRequest.SetRequestHeader "Accept", "application/xml"
  MyRequest.SetRequestHeader "Content-type", "application/json"
  MyRequest.SetRequestHeader "Authorization", "Basic dXNlcm5hbWU6cGFzc3dvcmQ="

  ' Send Request.
  MyRequest.send

        If (MyRequest.Status = 200) Then
              'And we get the response
```

```
                xmlDoc.LoadXML (MyRequest.responseText)
                Value = xmlDoc.SelectSingleNode("/VTQ/Value").Text
                Units = xmlDoc.SelectSingleNode("/VTQ/Units").Text
        Else
                MsgBox "XHQ Web API Error: " + MyRequest.StatusText
        End If

    Cells(i, 3) = s
    Cells(i, 6) = Value + " " + Units
    i = i + 1
    Next

        Cells.Replace What:="""", Replacement:="", LookAt:=xlPart
    Else
        MsgBox "XHQ Web API Error: " + MyRequest.StatusText
    End If

  Application.ScreenUpdating = True
  Application.Calculation = xlCalculationAutomatic
End Sub
```

The output looks like this when running the above VBA macro in an Excel 2013 spreadsheet:

	A	B	C	D	E	F	G	H
1								
2								
3			**Pump P106A**					
4								
5			**Attributes**			**Values**		
6			sDescription			Centrifugal Pump P-106A		
7			sID			P-106A		
8			sManufacturer					
9			sSerialNumber					
10			sType					
11			DischargePressure			755	psig	
12			FlowRate			350	gpm	
13			SuctionPressure			0.6	bar	
14								
15								

EXAMPLE 2

This example gets the average data for the last 24 hours for all attributes.

Given: The **pump path** is **`::Sample.Equipment.Pumps.Centrifugal.P106A`**.

JavaScript Code Snippet

```
var arr;
var path = '::Sample.Equipment.Pumps.Centrifugal.P106A.'
var value = '.rCurrentValue'
var callUrl = path;
// execute only if call url is not an empty string
if (callUrl) {
  $.ajax({
    method: "GET",
    url:
    "http://localhost/indx/xhqwi/api/core/vcagmobile/Metadata/ComponentMembers?parentPat
    h=" + callUrl,
    headers: {
           "Accept": "application/json",
           "Content-Type": "application/json",
           "Authorization": "Basic dXNlcm5hbWU6cGFzc3dvcmQ="
    },
  }).done(function (response) {
    arr = $.map(response, function (el) { return el });
    if (arr) {
           var out = '';
           var s = ''
           for (var i = 0; i < arr.length; i++) {
                  s = path + arr[i] + value
                  $.ajax({
                         method: "GET",
                         url:
    "http://localhost/indx/xhqwi/api/core/vcagmobile/Points/AverageValue?path=
    " + s + "&startTime=2017-05-14%2010%3A00%3A00.000&endTime=2017-05-
    15%2010%3A00%3A00.000",
                         headers: {
                                "Accept": "application/json",
                                "Content-Type": "application/json",
                                "Authorization": "Basic dXNlcm5hbWU6cGFzc3dvcmQ="
                         },
                  }).done(function (response) {
                  out = out + JSON.stringify(response, null, 5) + "\n"
                  // call successful, add response to the result div, formatted with
    <pre>
                  if (i == 3) {
                         $('#result').html('<pre>' + out + '</pre>');
                  }
                  }).fail(function (e) {
```

```
                // call failed, show error in the result div
                $('#result').html('<span style="color: red">[' + e.status + '] ' +
    e.responseText + '</span>');
                });
            }
    }
  }).fail(function (e) {
  // call failed, show error in the result div
  $('#result').html('<span style="color: red">[' + e.status + '] ' + e.responseText +
  '</span>');
  });
}
```

Complete HTML

```
<html>
<head>
<title>Demo for JavaScript XHQ Web API call</title>
</head>
<body>
  <div id="form">
    XHQ Pump path: <br />
    <input type="text" id="call_url" style="width: 90%" /><br />
    <button id="call">Get Pump Atrributes Average Values</button>
  </div>
  <hr />
  <div id="result"></div>
  <!-- include jquery for ajax support -->
  <script type="text/javascript" src="js/jquery.js"></script>
  <!-- the script that makes the call on button click -->
  <script type="text/javascript">
    var arr;
    // start when all elements are loaded in the DOM
    $(document).ready(function () {
            // trigger this function when the call button is clicked
            $('#call').click(function () {
                var callUrl = $('#call_url').val();
                var path = callUrl + '.'
                var value = '.rCurrentValue'
                // execute only if call url is not an empty string
                if (callUrl) {
                    $.ajax({
                        method: "GET",
```

```
                            url:
"http://localhost/indx/xhqwi/api/core/vcagmobile/Metadata/ComponentMembers?parentPat
h=" + callUrl,
                            headers: {
                                   "Accept": "application/json",
                                   "Content-Type": "application/json",
                                   "Authorization": "Basic dXNlcm5hbWU6cGFzc3dvcmQ="
                            },
                     }).done(function (response) {
                            arr = $.map(response, function (el) { return el });
                            if (arr) {
                                   var out = '';
                                   var s = ''
                                   for (var i = 0; i < arr.length; i++) {
                                          s = path + arr[i] + value
                                          $.ajax({
                                                 method: "GET",
                                                 url:
"http://localhost/indx/xhqwi/api/core/vcagmobile/Points/AverageValue?path=" + s +
"&startTime=2017-05-14%2010%3A00%3A00.000&endTime=2017-05-15%2010%3A00%3A00.000",
                                                 headers: {
                                                        "Accept": "application/json",
                                                        "Content-Type":
"application/json",
                                                        "Authorization": "Basic
dXNlcm5hbWU6cGFzc3dvcmQ="
                                                 },
                                          }).done(function (response) {
                                                 out = out + JSON.stringify(response,
null, 5) + "\n"
                                                 // call successful, add response to
the result div, formatted with <pre>
                                                 if (i == 3) {
                                                        $('#result').html('<pre>' +
out + '</pre>');
                                                 }
                                          }).fail(function (e) {
                                                 // call failed, show error in the
result div
                                                 $('#result').html('<span
style="color: red">[' + e.status + '] ' + e.responseText + '</span>');
                                          });
                                   }
                            }
                     }).fail(function (e) {
                            // call failed, show error in the result div
```

```
                                $('#result').html('<span style="color: red">[' + e.status
    + '] ' + e.responseText + '</span>');
                        });
                }
            });
    });
  </script>
</body>
</html>
```

EXAMPLE 3

The VBA code for Excel 2013 is identical with the one from Example 1 with one exception.

Instead of this line in the web API REST call for the current values:

```
MyRequest.Open "GET",
"http://localhost/indx/xhqwi/api/core/vcagmobile/Points/CurrentValue?path=" +
AttributePath
```

Use this line:

```
MyRequest.Open "GET",
"http://localhost/indx/xhqwi/api/core/vcagmobile/Points/CurrentValue?path=" +
AttributePath + "&units=psig"
```

The added exception is highlighted.

This is what the output looks like when running the VBA macro from this example in the Excel 2013 spreadsheet:

	A	B	C	D	E	F	G	H
1								
2								
3			**Pump P106A**					
4								
5			**Attributes**			**Values**		
6			sDescription			Centrifugal Pump P-106A		
7			sID			P-106A		
8			sManufacturer					
9			sSerialNumber					
10			sType					
11			DischargePressure			53 bar		
12			FlowRate			79 t/hr		
13			SuctionPressure			6 psig		
14								
15								

Using the XHQ Performance Management Mobile App

This topic shows you how to create and set-up a native Android mobile application for XHQ Performance Management (TM, eLogs, and ANS). In each subsection, snippets of code are provided as examples on how to configure certain features of the app.

This mobile app uses the XHQ Web Interface (Web API) and is packaged in the **`XHQ Mobile.apk`** file. The XHQ Web Interface is used for all calls to XHQ. Note, that all calls to the Web API service are not made on the Android main UI thread, but on a separate thread created especially for that task (such as using the "***android.os.AsyncTask***" class). The reason for this is to optimize performance.

REQUIREMENTS

- Currently, this app must be **sideloaded** from the XHQ installation media onto your Android device. It is not available on the Google Play store.
- This app is designed to run in **Android 4.0 and above**, and is optimized for Android tablet screens 7 inches and larger.
- Because the app uses the XHQ Web Interface to communicate and retrieve data to and from XHQ, it requires **WIFI** access from the Android device.

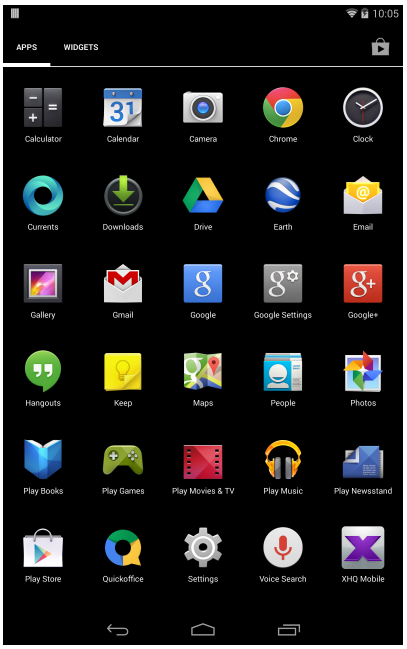
Installing the App

Because this app is not available on the Google Play store, you must **sideload** it (install it manually). First, make sure to enable installation from "Unknown" sources on your Android device. This option is usually found in the Security section of your device's System settings.

Next, copy the **`XHQ Mobile.apk`** file (from the XHQ installation media) onto your device, for example, at the root of your tablet's storage. Using a file manager app, find the `.apk` file and tap on it to initiate the install.

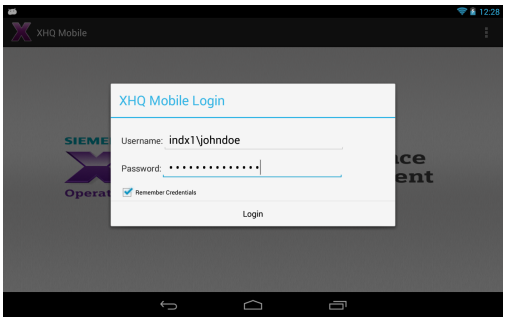
Launching the XHQ Mobile App

Once the app is installed onto your Android tablet, the "XHQ Mobile" icon appears on the Apps screen.



Apps Screen - XHQ Mobile Icon

Tap the icon, and the **XHQ Mobile Login** appears.



XHQ Mobile Login

Code Example 1: Authentication

This first code snippet example shows you how to authenticate a user during the log in process. If the **Remember Credentials** box is checked, then the user will no longer have to enter credentials during subsequent log ins.

The text highlighted in yellow is the actual call to the XHQ Web Interface.

See the topic, *The RestClient Class*, for the class used to call the XHQ Web Interface service in this example.

```
private class LoginTask extends AsyncTask<Void, Void, String> {
boolean err = false;
String server = null;
String username = null;
String pwd = null;
```

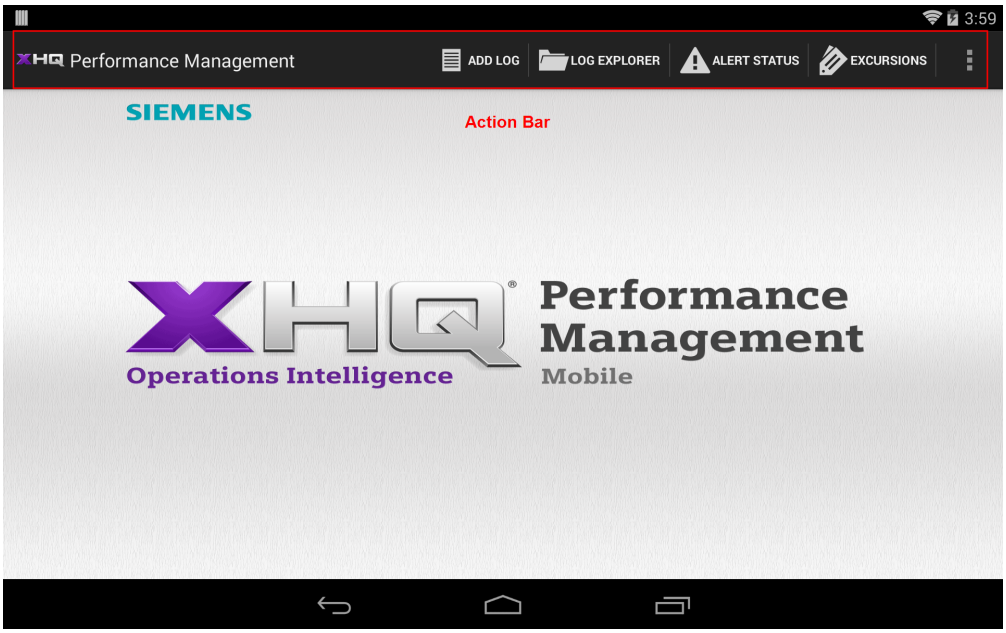
```
String credStr = null;
boolean persistFlg = false;
String url = null;
RestClient client = null;
private ProgressDialog dialog = new ProgressDialog(LoginActivity.this);

LoginTask(String server, String username, String pwd, boolean persistFlg)
{
this.server = server;
this.username = username;
this.pwd = pwd;
this.persistFlg = persistFlg;
credStr = username + ":" + pwd;
url = "http://" + server + "/indx/xhqwi/api/login";
}
@Override
protected void onPreExecute() {
dialog.setMessage("Authenticating User...");
dialog.show();
}

protected String doInBackground(Void... params) {
client = new RestClient(url);
client.AddHeader("Host", server);
client.AddHeader("Accept", "application/json");
client.AddHeader("Authorization", "Basic " + Base64.encodeToString(credStr.getBytes(),
Base64.NO_WRAP));
try {
client.Execute(RestClient.RequestMethod.GET);
} catch (Exception e) {
e.printStackTrace();
Log.e(ERROR_TAG, "Severe Error logging in XHQ Web API...Reason: " + e.getLocalizedMessage
());
Log.e(ERROR_TAG, "HTTP Return Code: " + client.getResponseCode());
Log.e(ERROR_TAG, "HTTP Error Message: " + client.getErrorMessage());
AlertDialog.Builder alertDialog = new AlertDialog.Builder(
LoginActivity.this);

alertDialog.setTitle("XHQ Web API Error");
alertDialog.setMessage("Severe Error quering XHQ Web API! Please see the logs for
details.");

// Setting OK Button
alertDialog.setPositiveButton("OK", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int which) {
dialog.cancel();
}
});

// Showing Alert Message
alertDialog.show();
}

//an HTTP error occurred
if(client.getResponseCode() != HttpStatus.SC_OK)
{
err = true;
Log.e(ERROR_TAG, "HTTP Error logging in XHQ Web API...");
Log.e(ERROR_TAG, "HTTP Return Code: " + client.getResponseCode());
Log.e(ERROR_TAG, "HTTP Error Message: " + client.getErrorMessage());
}
```

```
return client.getResponse();
}

protected void onPostExecute(String result) {

if (dialog.isShowing()) {
dialog.dismiss();
}

try
{
if(err)
{
Log.e(ERROR_TAG, "Error Reason: " + result);
AlertDialog.Builder alertDialog = new AlertDialog.Builder(
LoginActivity.this);
alertDialog.setTitle("XHQ API Login Error");
alertDialog.setMessage("Error authenticating user! Incorrect username or password! Try
logging again?");
// Setting OK Button
alertDialog.setPositiveButton("Yes", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int which) {
//try the Login again
LoginDialog d = new LoginDialog();
d.show(getFragmentManager(), "Login Dialog");
}
});
// Setting OK Button
alertDialog.setNegativeButton("No", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int which) {
dialog.cancel();
SharedPreferences prefs = PreferenceManager.getDefaultSharedPreferences
(LoginActivity.this);
    SharedPreferences.Editor editor = prefs.edit();
    editor.putBoolean("load_data", true);
    editor.commit();
LoginActivity.this.finish();
}
});

// Showing Alert Message
alertDialog.show();
}
}
catch (Exception e) {
Log.e(ERROR_TAG, "Error Reason: " + result);

AlertDialog.Builder alertDialog = new AlertDialog.Builder(
LoginActivity.this);

alertDialog.setTitle("XHQ Web API Error");
alertDialog.setMessage("Error logging into XHQ Web API. Reason: " + result
+ " Please see the logs for more details.");

// Setting OK Button
alertDialog.setPositiveButton("OK", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int which) {
dialog.cancel();
}
});
```

```
// Showing Alert Message
alertDialog.show();
e.printStackTrace();
}
}
}
```

The XHQ Mobile Home Screen

This is the **Home** screen. It is the main screen of the app, the central hub for accessing all other screens.



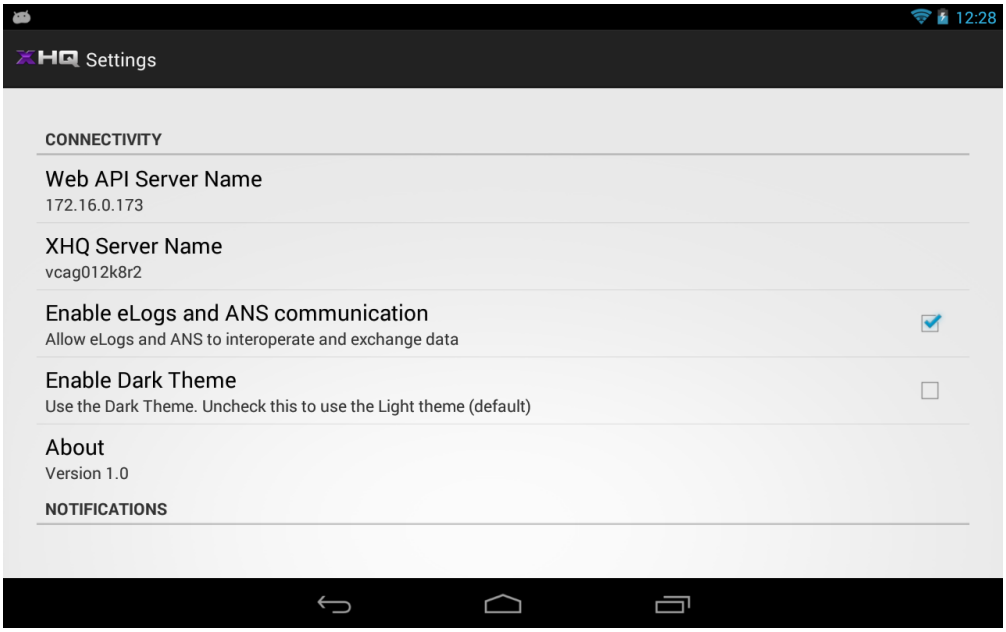
XHQ Mobile - XHQ Performance Management

From this screen the user can:

- Add a new log to eLogs;
- Browse and explore all eLog logs using various filter criteria;
- Browse and explore ANS alerts status using various filter criteria;
- Browse and explore all ANS' excursions using various filter criteria;
- Change the log in credentials;
- Modify the application settings and preferences;
- Exit the application.

The Settings Menu

To access **XHQ Settings** from the Home screen, tap the **Menu button** (three vertical dots located on the upper right-hand corner).



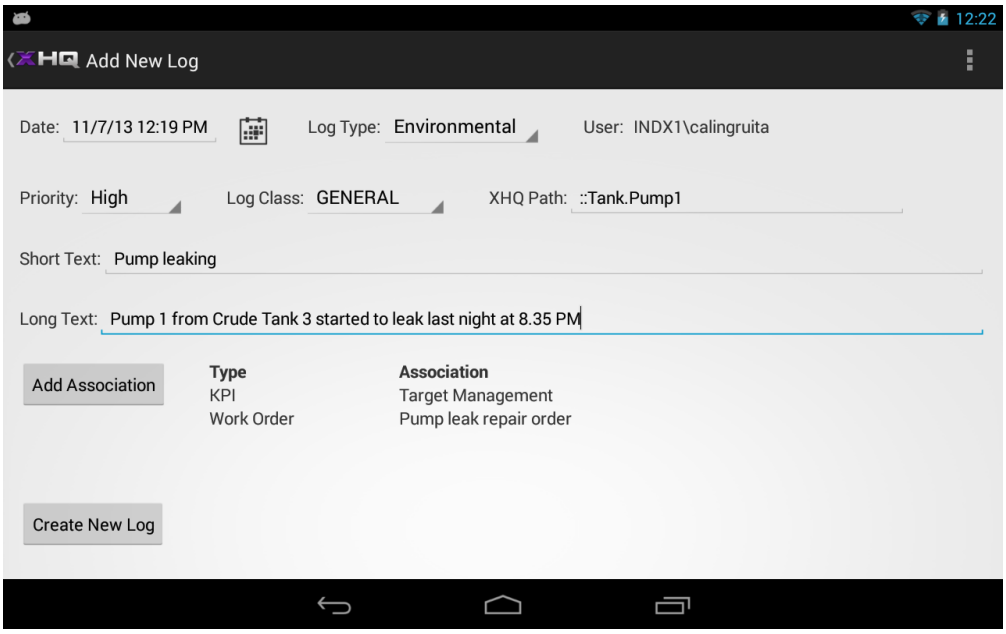
XHQ Settings

From the **Settings menu** of the app, you can:

- Access the Web API Server and the XHQ Server names;
- Enable eLogs and ANS to communicate with each other and exchange data;
- Select between two themes: The Light Theme is recommended for use during the day or well lit indoor areas; the Dark Theme for night or low-light situations.

Adding a Log

From the Home screen Action bar, tap **Add Log**.



Add New Log Screen

To set Date, click the calendar icon. This launches the Date/Time Picker. Once you've set the new log, tap **Create New Log**.

Code Example 2: Creating a New Log

This snippet example illustrates how to create a new log in eLogs using the XHQ Web Interface.

The text highlighted in yellow is the actual call to the XHQ Web Interface.

See the topic, *The RestClient Class*, for the class used to call the XHQ Web Interface service in this example.

```
private class CreateOperLogTask extends AsyncTask<OperLog, Void, String> {
boolean err = false;

SharedPreferences sharedPref = PreferenceManager.getDefaultSharedPreferences
(AddLogActivity.this);
String wsHost = sharedPref.getString("api_server_host", "localhost");
String appServerHost = sharedPref.getString("xhq_server_host", "localhost");
String username = sharedPref.getString("username", "");
String pwd = sharedPref.getString("pwd", "");
String credStr = username + ":" + pwd;
String url = "http://" + wsHost + "/indx/xhqwi/api/operlogs?serverhost=" + appServerHost;
private ProgressDialog dialog = new ProgressDialog(AddLogActivity.this);
RestClient client = new RestClient(url);
@Override
protected void onPreExecute() {
dialog.setMessage("Creating new Operational Log...");
dialog.show();
}
```

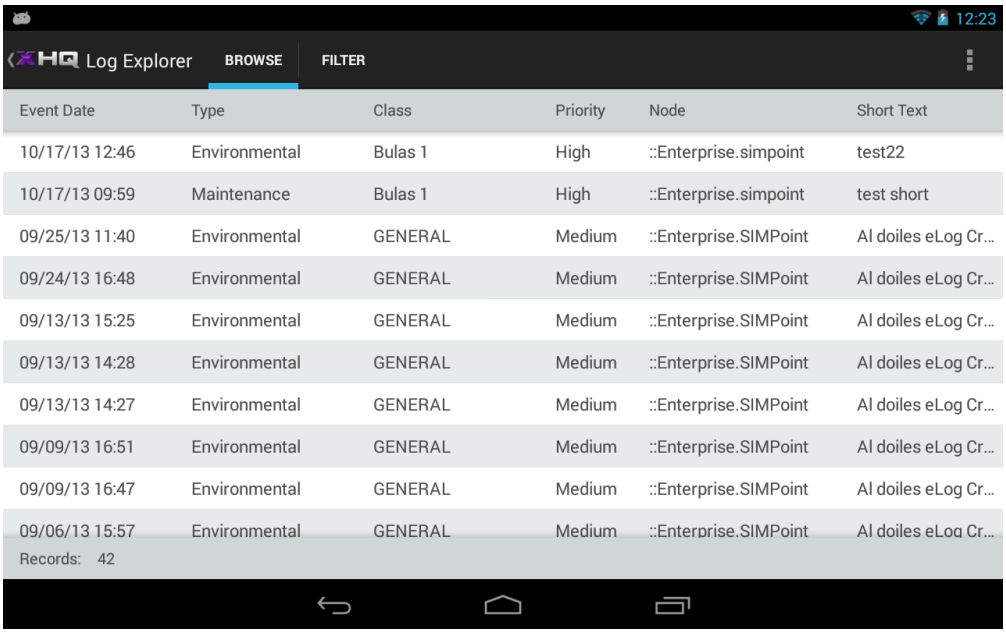
```
protected String doInBackground(OperLog... opLog) {
client.AddHeader("Host", wsHost);
client.AddHeader("Accept", "application/json");
client.AddHeader("Content-type", "application/json");
client.AddHeader("Authorization", "Basic " + Base64.encodeToString(credStr.getBytes(),
Base64.NO_WRAP));

try {
client.AddJSONHolder(getJsonObjectFromOperLog(opLog[0]));
client.Execute(RestClient.RequestMethod.POST);
} catch (Exception e) {
e.printStackTrace();
Log.e(ERROR_TAG, "Severe Error quering XHQ Web API...Reason: " + e.getLocalizedMessage
());
Log.e(ERROR_TAG, "HTTP Return Code: " + client.getResponseCode());
Log.e(ERROR_TAG, "HTTP Error Message: " + client.getErrorMessage());
AlertDialog.Builder alertDialog = new AlertDialog.Builder(
AddLogActivity.this);

alertDialog.setTitle("XHQ Web API Error");
alertDialog.setMessage("Severe Error quering XHQ Web API! Please see the logs for
details.");

// Setting OK Button
alertDialog.setPositiveButton("OK", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int which) {
dialog.cancel();
}
});

// Showing Alert Message
alertDialog.show();
}
//an HTTP error occurred
if(client.getResponseCode() != HttpStatus.SC_CREATED)
{
err = true;
Log.e(ERROR_TAG, "HTTP Error quering XHQ Web API...");
Log.e(ERROR_TAG, "HTTP Return Code: " + client.getResponseCode());
Log.e(ERROR_TAG, "HTTP Error Message: " + client.getErrorMessage());
}

return client.getResponse();
}

protected void onPostExecute(String result) {

if (dialog.isShowing()) {
dialog.dismiss();
}

try
{
if(!err)
{
Bundle bLogType = new Bundle();
Bundle bLogClass = new Bundle();
Bundle bPrio = new Bundle();
bLogType.putSerializable("LogTypesBundle", logTypesList);
bLogClass.putSerializable("LogClassesBundle", logClassesList);
```

```
bPrio.putSerializable("PrioritiesBundle", prioritiesList);
Intent intent = new Intent(AddLogActivity.this, LogExplorerActivity.class);
intent.putExtra("LogTypes", bLogType);
intent.putExtra("LogClasses", bLogClass);
intent.putExtra("Priorities", bPrio);
startActivity(intent);

JSONObject obj = new JSONObject(result);
Toast.makeText(AddLogActivity.this, "Successfully created a new Operational Log with ID:
"
+ obj.getInt("LogId"), Toast.LENGTH_SHORT).show();
Log.d(DEBUG_TAG, "Created new Operational Log with ID: " + obj.getInt("LogId"));
}
else
{
Log.e(ERROR_TAG, "Error Reason: " + result);
AlertDialog.Builder alertDialog = new AlertDialog.Builder(
AddLogActivity.this);

alertDialog.setTitle("XHQ Web API Error");
alertDialog.setMessage("Error quering XHQ Web API. HTTP Return Code: " +
client.getResponseCode()
+ ". Message Detail: " + client.getErrorMessage() + ". Reason: " + result
+ ". Please see the logs for more details.");

// Setting OK Button
alertDialog.setPositiveButton("OK", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int which) {
dialog.cancel();
}
});

// Showing Alert Message
alertDialog.show();
}
}
catch (JSONException e) {
Log.e(ERROR_TAG, "Error Reason: " + result);

AlertDialog.Builder alertDialog = new AlertDialog.Builder(
AddLogActivity.this);

alertDialog.setTitle("XHQ Web API Error");
alertDialog.setMessage("Error quering XHQ Web API. Reason: " + result
    + " Please see the logs for more details.");
// Setting OK Button
alertDialog.setPositiveButton("OK", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int which) {
dialog.cancel();
}
});

// Showing Alert Message
alertDialog.show();
e.printStackTrace();
}
}
private JSONObject getJsonObjectFromOperLog(OperLog opLog) throws JSONException
{
JSONObject holder = new JSONObject();
```

```
holder.put("LogId", opLog.getLogId());
holder.put("LogTypeId", opLog.getLogTypeId());
holder.put("LogClassId", opLog.getLogClassId());
holder.put("LogPriorityId", opLog.getLogPriorityId());
holder.put("LogStatusId", opLog.getLogStatusId());
holder.put("LogTextId", opLog.getLogTextId());
holder.put("LogSeq", opLog.getLogSeq());
holder.put("DateTime", opLog.getDateTime());
holder.put("XhqUser", opLog.getXhqUser());
holder.put("XhqPath", opLog.getXhqPath());
holder.put("LongText", opLog.getLongText());
holder.put("ShortText", opLog.getShortText());
holder.put("InactiveFlg", opLog.isInactiveFlg());
holder.put("ExpiredFlg", opLog.isExpiredFlg());
holder.putOpt("KeyWord", opLog.getKeyWord());
holder.putOpt("CustUser", opLog.getCustUser());

Set<String> assocTypes = assoc.keySet();
if(assocTypes != null && assocTypes.size() > 0)
{
JSONArray assocList = new JSONArray();

for(String type : assocTypes)
{
int typeId = -1;
String[] typesList = getResources().getStringArray(R.array.assoc_types_array);
String[] typesIdList = getResources().getStringArray(R.array.assoc_type_ids_array);

for(int i=0; i<typesList.length; i++)
{
if(type.equalsIgnoreCase(typesList[i]))
{
typeId = Integer.parseInt(typesIdList[i]);
break;
}
}
String desc = assoc.get(type);
JSONObject obj = new JSONObject();
obj.put("TypeId", typeId);
obj.put("Description", desc);
obj.put("NewFlg", false);

assocList.put(obj);
}
holder.putOpt("AssocList", assocList);
}
return holder;
}
}
```

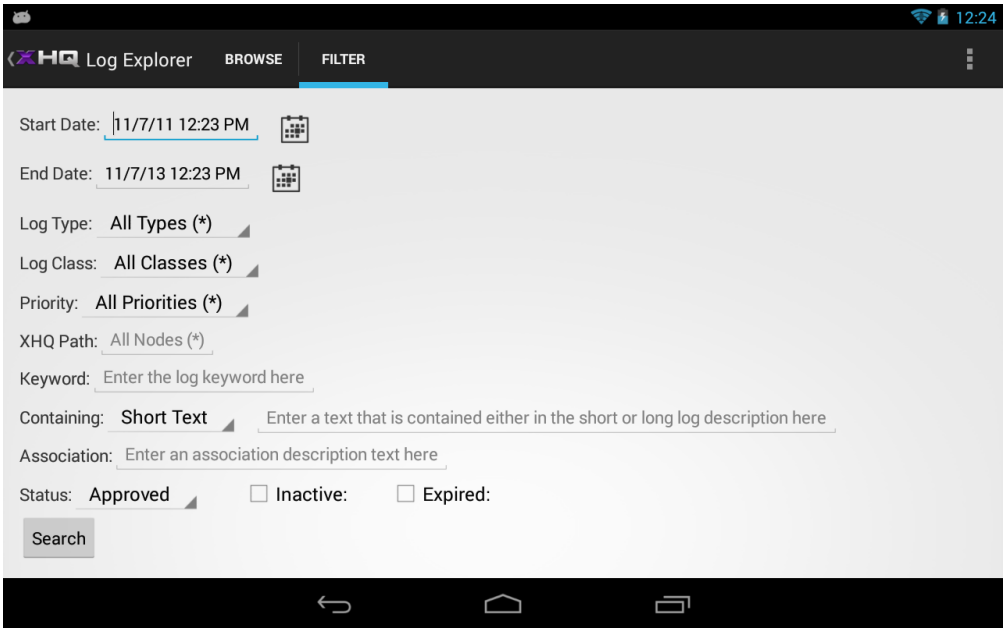
The Log Explorer

From the Home screen Action bar, tap **Log Explorer**.



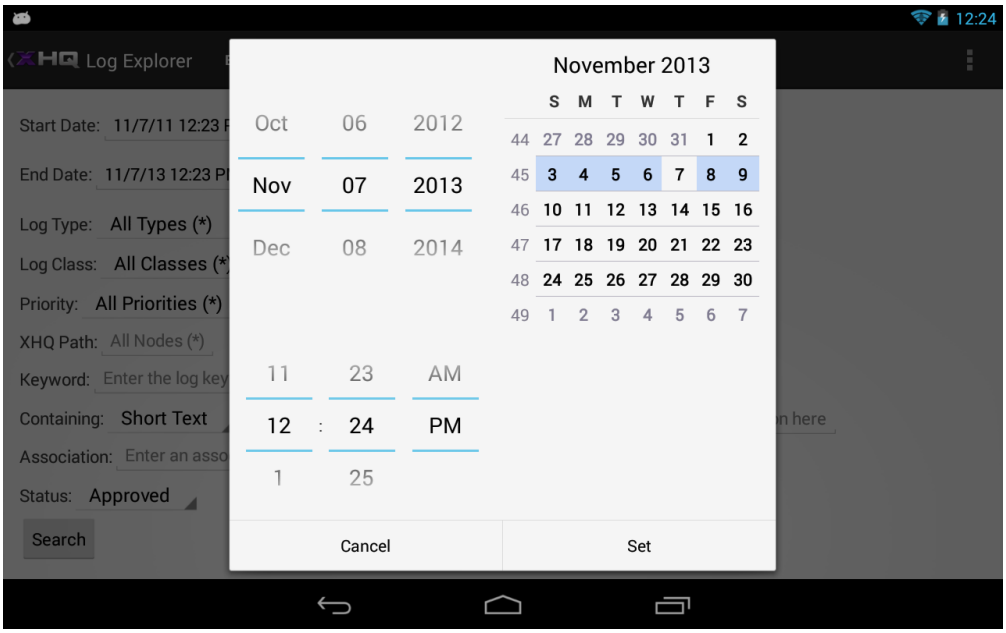
Log Explorer - Browse

From the Log Explorer Action bar, tap **Filter** to show only the log(s) you want to see.



Log Explorer - Filter

To set either the Start or End date, click the calendar icon. This launches the Date/Time Picker.



Log Explorer - Date and Time Picker

Once you've set the filter, tap **Search**.

Code Example 3: Getting Logs from eLogs

This code snippet example shows how to invoke the XHQ Web Interface in order to get a list of logs in eLogs.

The text highlighted in yellow is the actual call to the XHQ Web Interface.

See the topic, *The RestClient Class*, for the class used to call the XHQ Web Interface service in this example.

```
private class GetOperLogListTask extends AsyncTask<FilterCriteria, Void, String> {
boolean err = false;
boolean search = false;
RestClient client = null;

SharedPreferences sharedPref = PreferenceManager.getDefaultSharedPreferences
(LogExplorerActivity.this);
String wsHost = sharedPref.getString("api_server_host", "localhost");
String appServerHost = sharedPref.getString("xhq_server_host", "localhost");
String username = sharedPref.getString("username", "");
String pwd = sharedPref.getString("pwd", "");
String credStr = username + ":" + pwd;
String url = "http://" + wsHost + "/indx/xhqwi/api/operlogs?serverhost=" +
appServerHost;
private ProgressDialog dialog = new ProgressDialog(LogExplorerActivity.this);
GetOperLogListTask(boolean s)
{
search = s;
}
@Override
        protected void onPreExecute() {
            dialog.setMessage("Retrieving Operational Log List...");
            dialog.show();
        }
```

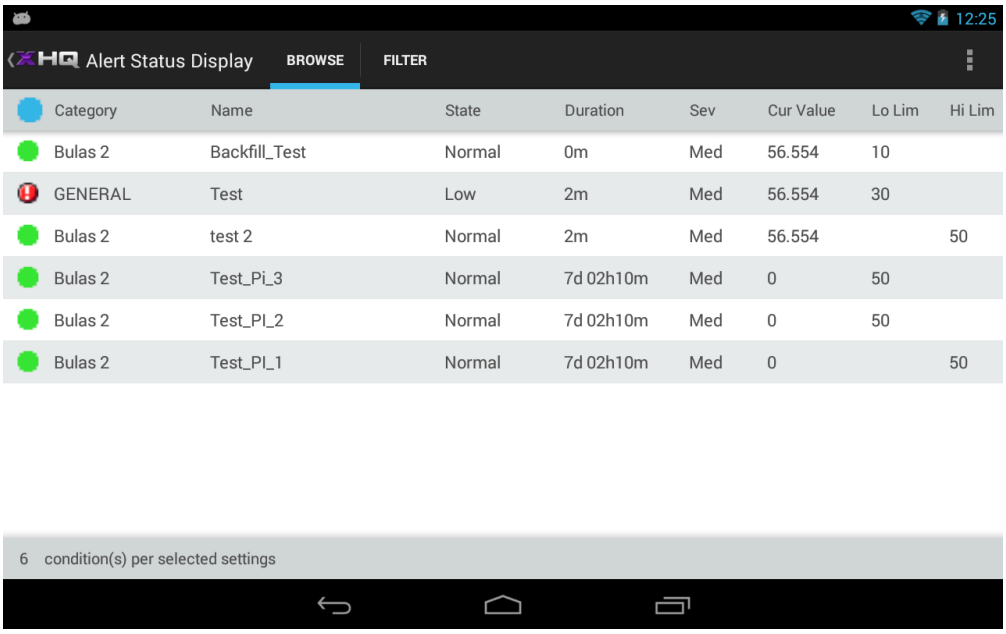
```
protected String doInBackground(FilterCriteria... fList) {
FilterCriteria filter = fList[0];
if(search)
{
long startDate = filter.getStartDate();
long endDate = filter.getEndDate();
int logTypeId = filter.getLogTypeId();
int logClassId = filter.getLogClassId();
int priorityId = filter.getLogPriorityId();
int statusId = filter.getLogStatusId();
String path = filter.getXhqPath();
String keyWord = filter.getKeyWord();
String shortText = filter.getShortText();
String longText = filter.getLongText();
String assocDesc = filter.getAssocDesc();
boolean inactiveFlg = filter.isInactiveFlg();
boolean expiredFlg = filter.isExpiredFlg();
if(startDate > 0)
url = url + "&startDate=" + startDate;
if(endDate > 0)
url = url + "&endDate=" + endDate;
if(logTypeId > Integer.MIN_VALUE)
url = url + "&logTypeId=" + logTypeId;
if(logClassId > Integer.MIN_VALUE)
url = url + "&logClassId=" + logClassId;
if(priorityId > Integer.MIN_VALUE)
url = url + "&logPriorityId=" + priorityId;
url = url + "&logStatusId=" + statusId;
if(path != null)
url = url + "&xhqPath=" + path;
if(keyWord != null)
url = url + "&keyWord=" + keyWord;
if(shortText != null)
url = url + "&shortText=" + shortText;
if(longText != null)
url = url + "&longText=" + longText;
if(assocDesc != null)
url = url + "&assocDesc=" + assocDesc;
url = url + "&inactiveFlg=" + inactiveFlg;
url = url + "&expiredFlg=" + expiredFlg;
}
else
{
url = url + "&startDate=" + filter.getStartDate()
      + "&endDate=" + filter.getEndDate();
}
client = new RestClient(url);

client.AddHeader("Host", wsHost);
client.AddHeader("Accept", "application/json");
client.AddHeader("Authorization", "Basic " + Base64.encodeToString(credStr.getBytes(),
Base64.NO_WRAP));

try {
client.Execute(RestClient.RequestMethod.GET);
} catch (Exception e) {
e.printStackTrace();
Log.e(ERROR_TAG, "Severe Error quering XHQ Web API...Reason: " + e.getLocalizedMessage
());
AlertDialog.Builder alertDialog = new AlertDialog.Builder(
```

```
LogExplorerActivity.this);

alertDialog.setTitle("XHQ Web API Error");
alertDialog.setMessage("Severe Error quering XHQ Web API! Please see the logs for
details.");

// Setting OK Button
alertDialog.setPositiveButton("OK", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int which) {
dialog.cancel();
}
});

// Showing Alert Message
alertDialog.show();
}
//an HTTP error occurred
if(client.getResponseCode() != HttpStatus.SC_OK)
{
err = true;
Log.e(ERROR_TAG, "HTTP Error quering XHQ Web API...");
Log.e(ERROR_TAG, "HTTP Return Code: " + client.getResponseCode());
Log.e(ERROR_TAG, "HTTP Error Message: " + client.getErrorMessage());
}

return client.getResponse();
}

protected void onPostExecute(String result) {
if (dialog.isShowing()) {
                dialog.dismiss();
            }
try
{
if(!err)
{
JSONArray items = new JSONArray(result);
if(items != null)
{
OperLog[] logList = new OperLog[items.length()];
for (int i = 0; i < items.length(); i++)
{
JSONObject obj = items.getJSONObject(i);
long evDate = obj.getLong("DateTime");
dateFormat.setTimeZone(TimeZone.getDefault());
String eventDateStr = dateFormat.format(new Date(evDate));
OperLog log = new OperLog();
log.setLogId(obj.getInt("LogId"));
log.setEventDateStr(eventDateStr);
log.setLogType(obj.getString("LogType"));
log.setLogClass(obj.getString("LogClass"));
log.setLogPriority(obj.getString("LogPriority"));
log.setXhqPath(obj.getString("XhqPath"));
log.setShortText(obj.getString("ShortText"));
    logList[i] = log;
}
Log.d(DEBUG_TAG, "Retrieved " + items.length() + " Operational Logs from eLogs App
Server.");
setListView(logList);
}
}
```

```
else
{
Log.e(ERROR_TAG, "Error Reason: " + result);
AlertDialog.Builder alertDialog = new AlertDialog.Builder(
LogExplorerActivity.this);

alertDialog.setTitle("XHQ Web API Error");
alertDialog.setMessage("Error quering XHQ Web API. HTTP Return Code: " +
client.getResponseCode()
+ ". Message Detail: " + client.getErrorMessage() + ". Reason: " + result
+ ". Please see the logs for more details.");

// Setting OK Button
alertDialog.setPositiveButton("OK", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int which) {
dialog.cancel();
}
});

// Showing Alert Message
alertDialog.show();
}
}
catch (JSONException e) {
Log.e(ERROR_TAG, "Error Reason: " + result);
AlertDialog.Builder alertDialog = new AlertDialog.Builder(
LogExplorerActivity.this);

alertDialog.setTitle("XHQ Web API Error");
alertDialog.setMessage("Error quering XHQ Web API. Reason: " + result
    + " Please see the logs for more details.");

// Setting OK Button
alertDialog.setPositiveButton("OK", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int which) {
dialog.cancel();
}
});
// Showing Alert Message
alertDialog.show();
e.printStackTrace();
}
}
}
```

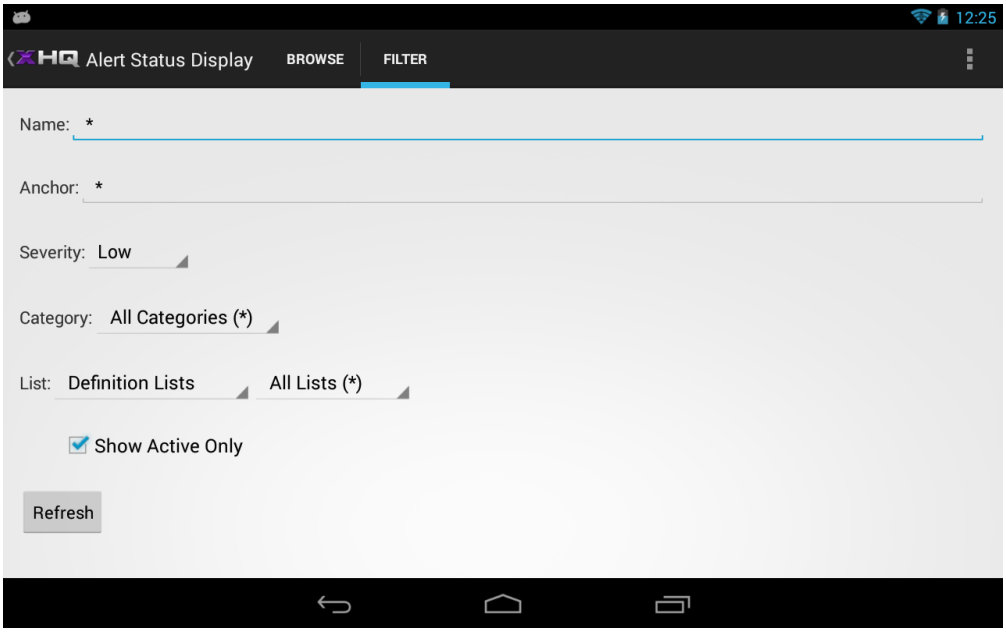
The ANS Alert Status

From the Home screen Action bar, tap **Alert Status**.



Alert Status Display - Browse

From the Alert Status Display Action bar, tap **Filter** to show only the alert(s) you want to see.



Alert Status Display - Filter

Once you've set the filter, tap **Refresh**.

Code Example 4: Retrieving ANS Alerts

This code snippet example shows hows to call the XHQ Web Interface and retrieve a list of filtered ANS alerts.

The text highlighted in yellow is the actual call to the XHQ Web Interface.

See the topic, *The RestClient Class*, for the class used to call the XHQ Web Interface service in this example.

```
private class GetASDListTask extends AsyncTask<ConditionStateFilter, Void, String> {
boolean err = false;
boolean refresh = false;
RestClient client = null;

SharedPreferences sharedPref = PreferenceManager.getDefaultSharedPreferences
(AsdActivity.this);
String wsHost = sharedPref.getString("api_server_host", "localhost");
String appServerHost = sharedPref.getString("xhq_server_host", "localhost");
String username = sharedPref.getString("username", "");
String pwd = sharedPref.getString("pwd", "");
String credStr = username + ":" + pwd;
String url = "http://" + wsHost + "/indx/xhqwi/api/ans/asd";
private ProgressDialog dialog = new ProgressDialog(AsdActivity.this);
GetASDListTask(boolean s)
{
refresh = s;
}
@Override
        protected void onPreExecute() {
            dialog.setMessage("Retrieving Alert Status...");
            dialog.show();
        }

protected String doInBackground(ConditionStateFilter... fList) {
ConditionStateFilter filter = fList[0];
client = new RestClient(url);

client.AddHeader("Host", wsHost);
client.AddHeader("Accept", "application/json");
client.AddHeader("Authorization", "Basic " + Base64.encodeToString(credStr.getBytes(),
Base64.NO_WRAP));
client.AddParam("serverhost", appServerHost);
if(filter != null && refresh)
{
String name = filter.getName();
String anchor = filter.getAnchor();
String sev = filter.getSevMin();
String category = filter.getCategory();
byte listType = filter.getListType();
String list = filter.getList();
boolean activeFlg = filter.isActiveFlg();
if(name != null)
client.AddParam("name", name);
if(anchor != null)
client.AddParam("anchor", anchor);
if(sev != null)
client.AddParam("sevMin", sev);
if(category != null)
client.AddParam("category", category);
client.AddParam("listType", "" + listType);
```

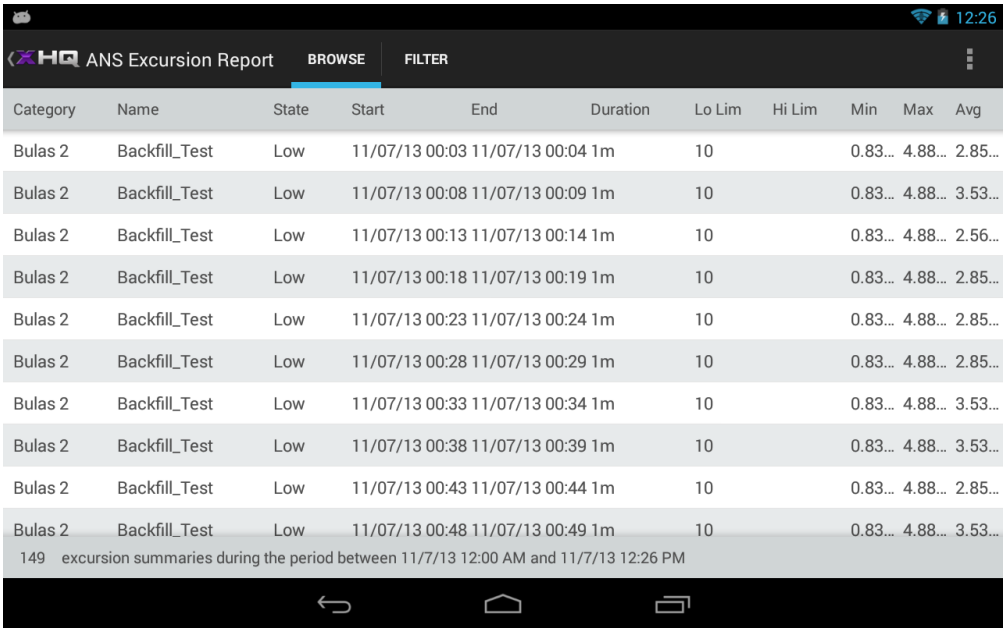
```
client.AddParam("list", list);
client.AddParam("active", "" + activeFlg);
}

try {
client.Execute(RestClient.RequestMethod.GET);
} catch (Exception e) {
e.printStackTrace();
Log.e(ERROR_TAG, "Severe Error quering XHQ Web API...Reason: " + e.getLocalizedMessage
());
Log.e(ERROR_TAG, "HTTP Return Code: " + client.getResponseCode());
Log.e(ERROR_TAG, "HTTP Error Message: " + client.getErrorMessage());
AlertDialog.Builder alertDialog = new AlertDialog.Builder(
AsdActivity.this);

alertDialog.setTitle("XHQ Web API Error");
alertDialog.setMessage("Severe Error quering XHQ Web API! Please see the logs for
details.");

// Setting OK Button
alertDialog.setPositiveButton("OK", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int which) {
dialog.cancel();
}
});

// Showing Alert Message
alertDialog.show();
}
//an HTTP error occurred
if(client.getResponseCode() != HttpStatus.SC_OK)
{
err = true;
Log.e(ERROR_TAG, "HTTP Error quering XHQ Web API...");
}

return client.getResponse();
}

protected void onPostExecute(String result) {
if (dialog.isShowing()) {
                dialog.dismiss();
            }
try
{
if(!err)
{
JSONArray items = new JSONArray(result);
if(items != null)
{
ConditionState[] csList = new ConditionState[items.length()];
for (int i = 0; i < items.length(); i++)
{
JSONObject obj = items.getJSONObject(i);
ConditionState cs = new ConditionState();
cs.setDefId(obj.getInt("DefId"));
cs.setName(obj.getString("Name"));
cs.setType(obj.getString("Type"));
cs.setActive(obj.getBoolean("Active"));
cs.setMasked(obj.getBoolean("Masked"));
cs.setTimeString(obj.getString("TimeString"));
```

```
cs.setTime(obj.getLong("Time"));
cs.setDuration(obj.getString("Duration"));
cs.setState(obj.getString("State"));
cs.setSeverity(obj.getString("Severity"));
cs.setAnchor(obj.getString("Anchor"));
cs.setView(obj.getString("View"));
cs.setValid(obj.getBoolean("Valid"));
cs.setEnabled(obj.getBoolean("Enabled"));
cs.setInfoText(obj.getString("ConditionInfo"));
cs.setCategory(obj.getString("Category"));
cs.setDetectorOk(obj.getBoolean("DetectorOk"));
cs.setUnits(obj.getString("Units"));
cs.setCurValue(obj.getString("CurValue"));
cs.setLowLimit(obj.getString("LowLimit"));
cs.setHighLimit(obj.getString("HighLimit"));
    csList[i] = cs;
}
Log.d(DEBUG_TAG, "Retrieved " + items.length() + " alerts status from the ANS Server.");
setListView(csList);
}
}
else
{
Log.e(ERROR_TAG, "Error Reason: " + result);
AlertDialog.Builder alertDialog = new AlertDialog.Builder(
AsdActivity.this);
alertDialog.setTitle("XHQ Web API Error");
alertDialog.setMessage("Error quering XHQ Web API. HTTP Return Code: " +
client.getResponseCode()
+ ". Message Detail: " + client.getErrorMessage() + ". Reason: " + result
+ ". Please see the logs for more details.");
// Setting OK Button
alertDialog.setPositiveButton("OK", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int which) {
dialog.cancel();
}
});
// Showing Alert Message
alertDialog.show();
}
}
catch (JSONException e) {
Log.e(ERROR_TAG, "Error Reason: " + result);
AlertDialog.Builder alertDialog = new AlertDialog.Builder(
AsdActivity.this);

alertDialog.setTitle("XHQ Web API Error");
alertDialog.setMessage("Error quering XHQ Web API. Reason: " + result
    + " Please see the logs for more details.");
// Setting OK Button
alertDialog.setPositiveButton("OK", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int which) {
dialog.cancel();
}
});

// Showing Alert Message
alertDialog.show();
e.printStackTrace();
}
}
```

```
}
```

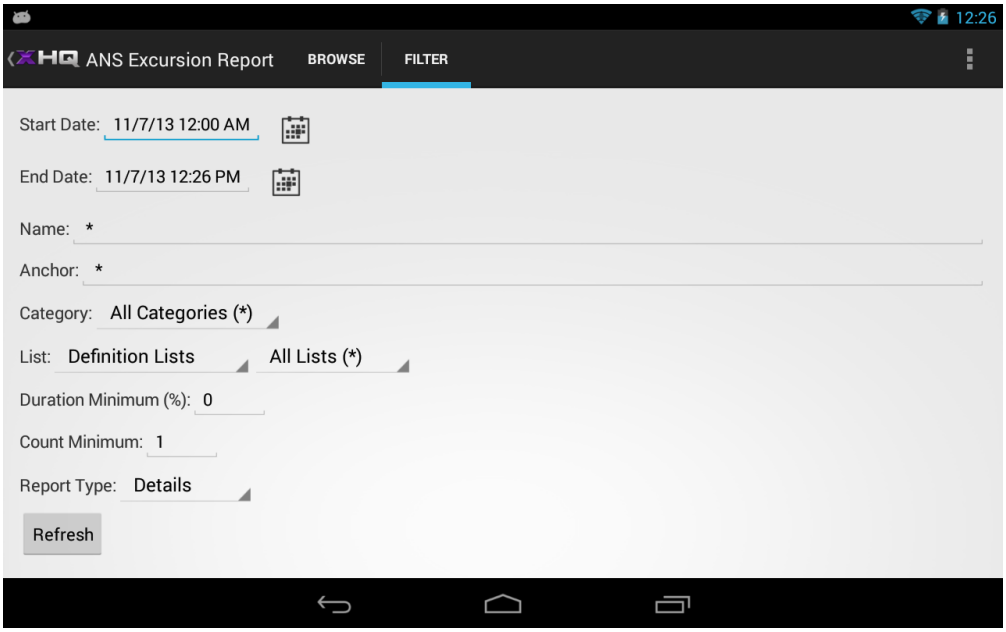
The ANS Excursion Report

From the Home screen Action bar, tap **Excursions**.



ANS Excursion Report - Browse

From the Alert Excursion Report Action bar, tap **Filter** to show only the excursion(s) you want to see.



ANS Excursion Report - Filter

To set either the Start or End date, click the calendar icon. This launches the Date/Time Picker. Once you've set the filter, tap **Refresh**.

Code Example 5: Retrieving ANS Excursions

This snippet code example shows how to call the XHQ Web Interface and retrieve a filtered list of ANS excursions.

The text highlighted in yellow is the actual call to the XHQ Web Interface.

See the topic, *The RestClient Class*, for the class used to call the XHQ Web Interface service in this example.

```
private class GetExcursionListTask extends AsyncTask<ExcursionFilter, Void, String> {
boolean err = false;
boolean refresh = false;
boolean isSummaryReport = true;
String sTime = "";
String eTime = "";
RestClient client = null;

SharedPreferences sharedPref = PreferenceManager.getDefaultSharedPreferences
(ExcursionReportActivity.this);
String wsHost = sharedPref.getString("api_server_host", "localhost");
String appServerHost = sharedPref.getString("xhq_server_host", "localhost");
String username = sharedPref.getString("username", "");
String pwd = sharedPref.getString("pwd", "");
String credStr = username + ":" + pwd;
String url = "http://" + wsHost + "/indx/xhqwi/api/excursionlog";
private ProgressDialog dialog = new ProgressDialog(ExcursionReportActivity.this);
GetExcursionListTask(boolean s)
{
refresh = s;
}
@Override
        protected void onPreExecute() {
            dialog.setMessage("Retrieving Excursion Log List...");
            dialog.show();
        }

protected String doInBackground(ExcursionFilter... fList) {
ExcursionFilter filter = fList[0];
client = new RestClient(url);

client.AddHeader("Host", wsHost);
client.AddHeader("Accept", "application/json");
client.AddHeader("Authorization", "Basic " + Base64.encodeToString(credStr.getBytes(),
Base64.NO_WRAP));
client.AddParam("serverhost", appServerHost);
if(refresh)
{
String name = filter.getName();
String anchor = filter.getAnchor();
String category = filter.getCategory();
byte listType = filter.getListType();
String list = filter.getList();
int minCount = filter.getMinCount();
int minDuration = filter.getMinDurationPct();
byte reportType = filter.getReportType();
long startTime = filter.getStartDate();
long endTime = filter.getEndDate();

if(reportType == ExcursionFilter.REPORT_DETAILS)
```

```
isSummaryReport = false;
if(name != null)
client.AddParam("name", name);
if(anchor != null)
client.AddParam("anchor", anchor);
if(category != null)
client.AddParam("category", category);
client.AddParam("listType", "" + listType);
client.AddParam("list", list);
client.AddParam("minCount", "" + minCount);
client.AddParam("minDurationPct", "" + minDuration);
client.AddParam("reportType", "" + reportType);
if(startTime > 0)
client.AddParam("startTime", "" + startTime);
if(endTime > 0)
client.AddParam("endTime", "" + endTime);
}
else
{
client.AddParam("startTime", "" + filter.getStartDate());
client.AddParam("endTime", "" + filter.getEndDate());
}
dateFormat.setTimeZone(TimeZone.getDefault());
sTime = dateFormat.format(new Date(filter.getStartDate()));
eTime = dateFormat.format(new Date(filter.getEndDate()));

try {
client.Execute(RestClient.RequestMethod.GET);
} catch (Exception e) {
e.printStackTrace();
Log.e(ERROR_TAG, "Severe Error quering XHQ Web API...Reason: " + e.getLocalizedMessage
());
Log.e(ERROR_TAG, "HTTP Return Code: " + client.getResponseCode());
Log.e(ERROR_TAG, "HTTP Error Message: " + client.getErrorMessage());
AlertDialog.Builder alertDialog = new AlertDialog.Builder(
ExcursionReportActivity.this);

alertDialog.setTitle("XHQ Web API Error");
alertDialog.setMessage("Severe Error quering XHQ Web API! Please see the logs for
details.");

// Setting OK Button
alertDialog.setPositiveButton("OK", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int which) {
dialog.cancel();
}
});

// Showing Alert Message
alertDialog.show();
}
//an HTTP error occurred
if(client.getResponseCode() != HttpStatus.SC_OK)
{
err = true;
Log.e(ERROR_TAG, "HTTP Error quering XHQ Web API...");
Log.e(ERROR_TAG, "HTTP Return Code: " + client.getResponseCode());
Log.e(ERROR_TAG, "HTTP Error Message: " + client.getErrorMessage());
}

return client.getResponse();
```

```
}

protected void onPostExecute(String result) {
if (dialog.isShowing()) {
                dialog.dismiss();
            }
try
{
if(!err)
{
JSONArray items = new JSONArray(result);
if(items != null)
{
Excursion[] excurList = new Excursion[items.length()];
for (int i = 0; i < items.length(); i++)
{
JSONObject obj = items.getJSONObject(i);
Excursion excur = new Excursion();
excur.setExcursionId(obj.getInt("ExcursionId"));
excur.setDefId(obj.getInt("DefId"));
excur.setName(obj.getString("Name"));
excur.setState(obj.getString("State"));
excur.setCategory(obj.getString("Category"));
excur.setCount(obj.getInt("Count"));
excur.setDurationSecs((float)obj.getDouble("DurationSecs"));
excur.setDurationPercent(obj.getInt("DurationPercent"));
excur.setDuration(obj.getString("Duration"));
excur.setQuantitySampled(obj.getInt("QuantitySampled"));
excur.setMinimum((float)obj.getDouble("Minimum"));
excur.setMaximum((float)obj.getDouble("Maximum"));
excur.setAverage((float)obj.getDouble("Average"));
excur.setOngoing(obj.getBoolean("Ongoing"));
excur.setUnits(obj.getString("Units"));
excur.setDefListName(obj.getString("DefListName"));
excur.setDefListSection(obj.getString("DefListSection"));
excur.setAnchor(obj.getString("Anchor"));
excur.setView(obj.getString("View"));
excur.setInfoText(obj.getString("ConditionInfo"));
excur.setStartTime(obj.getString("StartTimeStr"));
excur.setEndTime(obj.getString("EndTimeStr"));
excur.setStartTime(obj.getLong("StartTime"));
excur.setEndTime(obj.getLong("EndTime"));
excur.setCurValue(obj.getString("CurValue"));
excur.setLowLimit(obj.isNull("LowLimit") ? "" : obj.getString("LowLimit"));
excur.setHighLimit(obj.isNull("HighLimit") ? "" : obj.getString("HighLimit"));
excurList[i] = excur;
}
Log.d(DEBUG_TAG, "Retrieved " + items.length() + " Excursion Records from eLogs App
Server.");
String s = "excursion summaries during the period between " + sTime + " and " + eTime;
setListView(excurList, isSummaryReport, s);
}
}
else
{
Log.e(ERROR_TAG, "Error Reason: " + result);
AlertDialog.Builder alertDialog = new AlertDialog.Builder(
ExcursionReportActivity.this);

alertDialog.setTitle("XHQ Web API Error");
```

```
alertDialog.setMessage("Error quering XHQ Web API. HTTP Return Code: " +
client.getResponseCode()
+ ". Message Detail: " + client.getErrorMessage() + ". Reason: " + result
+ ". Please see the logs for more details.");
// Setting OK Button
alertDialog.setPositiveButton("OK", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int which) {
dialog.cancel();
}
});

// Showing Alert Message
alertDialog.show();
}
}
catch (JSONException e) {
Log.e(ERROR_TAG, "Error Reason: " + result);

AlertDialog.Builder alertDialog = new AlertDialog.Builder(
ExcursionReportActivity.this);

alertDialog.setTitle("XHQ Web API Error");
alertDialog.setMessage("Error quering XHQ Web API. Reason: " + result
    + " Please see the logs for more details.");

// Setting OK Button
alertDialog.setPositiveButton("OK", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int which) {
dialog.cancel();
}
});
// Showing Alert Message
alertDialog.show();
e.printStackTrace();
}
}
}
```

The RestClient Class

This code snippet is the **RestClient class** used to call the XHQ Web Interface service in the code examples for the XHQ Performance Management Mobile app.

```
package com.indx.xhq.mobile;

import java.io.*;
import java.net.URLEncoder;
import java.util.ArrayList;

import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.NameValuePair;
import org.apache.http.client.ClientProtocolException;
import org.apache.http.client.HttpClient;
import org.apache.http.client.entity.UrlEncodedFormEntity;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.client.methods.HttpUriRequest;
import org.apache.http.entity.StringEntity;
import org.apache.http.impl.client.DefaultHttpClient;
import org.apache.http.message.BasicHeader;
import org.apache.http.message.BasicNameValuePair;
import org.apache.http.protocol.HTTP;
import org.json.JSONException;
import org.json.JSONObject;

public class RestClient
{
    private ArrayList <NameValuePair> params;
    private ArrayList <NameValuePair> headers;

    private String url;
    private JSONObject jsonHolder = null;

    private int responseCode;
    private String message;

    private String response;

    public enum RequestMethod
    {
        GET,
        POST
    }

    public String getResponse() {
        return response;
    }

    public JSONObject getJSONResponse() throws JSONException
    {
return new JSONObject(response);
    }

    public String getXMLResponse() {
        return response;
    }
```

```
    public String getErrorMessage() {
        return message;
    }

    public int getResponseCode() {
        return responseCode;
    }

    public RestClient(String url)
    {
        this.url = url;
        params = new ArrayList<NameValuePair>();
        headers = new ArrayList<NameValuePair>();
    }

    public void AddParam(String name, String value)
    {
        params.add(new BasicNameValuePair(name, value));
    }

    public void AddHeader(String name, String value)
    {
        headers.add(new BasicNameValuePair(name, value));
    }

    public void AddJSONHolder(JSONObject holder)
    {
    jsonHolder = holder;
    }

    public void Execute(RequestMethod method) throws Exception
    {
        switch(method) {
            case GET:
            {
                //add parameters
                String combinedParams = "";
                if(!params.isEmpty()){
                    combinedParams += "?";
                    for(NameValuePair p : params)
                    {
                        String paramString = p.getName() + "=" + URLEncoder.encode
(p.getValue(), "UTF-8");
                        if(combinedParams.length() > 1)
                        {
                            combinedParams  +=  "&" + paramString;
                        }
                        else
                        {
                            combinedParams += paramString;
                        }
                    }
                }

                HttpGet request = new HttpGet(url + combinedParams);

                //add headers
                for(NameValuePair h : headers)
                {
                    request.addHeader(h.getName(), h.getValue());
```

```
            }

            executeRequest(request);
            break;
        }
        case POST:
        {
            HttpPost request = new HttpPost(url);

            //add headers
            for(NameValuePair h : headers)
            {
                request.addHeader(h.getName(), h.getValue());
            }

            if(!params.isEmpty()){
                request.setEntity(new UrlEncodedFormEntity(params, HTTP.UTF_8));
            }

            //this is to support a full JSON body HTTP POST request
            if(jsonHolder != null)
            {
                BasicHeader basicHeader = new BasicHeader(HTTP.CONTENT_TYPE,
                                          "application/json");
            StringEntity se = new StringEntity(jsonHolder.toString(), "UTF-8");
            se.setContentType(basicHeader);
            request.setEntity(se);
            }

            executeRequest(request);
            break;
        }
    }
}

private void executeRequest(HttpUriRequest request)
{
    HttpClient client = new DefaultHttpClient();

    HttpResponse httpResponse;

    try {
        httpResponse = client.execute(request);
        responseCode = httpResponse.getStatusLine().getStatusCode();
        message = httpResponse.getStatusLine().getReasonPhrase();

        HttpEntity entity = httpResponse.getEntity();

        if (entity != null) {

            InputStream instream = entity.getContent();
            response = convertStreamToString(instream);

            // Closing the input stream will trigger connection release
            instream.close();
        }

    } catch (ClientProtocolException e)  {
        client.getConnectionManager().shutdown();
        e.printStackTrace();
    } catch (IOException e) {
```

```
            client.getConnectionManager().shutdown();
            e.printStackTrace();
        }
    }

    private static String convertStreamToString(InputStream is) {

        BufferedReader reader = new BufferedReader(new InputStreamReader(is));
        StringBuilder sb = new StringBuilder();

        String line = null;
        try {
            while ((line = reader.readLine()) != null) {
                sb.append(line + "\n");
            }
        } catch (IOException e) {
            e.printStackTrace();
        } finally {
            try {
                is.close();
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
        return sb.toString();
    }
}
```

Whitelisting XHQ Servers

The `web.config` file includes a property that allows you to provide a comma-separated list of whitelisted XHQ Servers.

By default, this property is missing from the `web.config` file. In this case, the API service implicitly allows connections only to the local XHQ Server, localhost.

To specify a list of whitelisted XHQ Servers

1. Go to the **`%XHQ_WEB_HOME%\webapps\xhqwi`** directory (which by default is `<systemdrive>:\Program Files (x86)\XHQ\XHQ Web Root\webapps\xhqwi`) and open `web.config`.
2. Locate the `<appSettings>` tag and **add** the following property (highlighted in yellow).

```
<appSettings>
<add key="ServerHostWhitelist" value="xhqServer1, xhqServer2, xhqServer3" />
...
</appSettings>
```

   Where the **value** is a comma-delimited string of whitelisted XHQ Servers to connect to using the REST API. In other words, the `xhqwi` API allows access to these whitelisted servers.

3. **Save** `web.config`.

4 | XHQ BI Data Provider Web Service

The XHQ BI Data Provider web service is a RESTful web service that is designed for integrating XHQ with Business Intelligence (BI) tools such as Microsoft Power BI, Tableau, TIBCO SpotFire, and applications that can consume OData feed. The XHQ BI Data Provider web service can return data in either JSON or AtomPub (XML) format.

Refer to `http://www.odata.org/documentation/odata-version-3-0/odata-version-3-0-core-protocol/` for an overview of the OData protocol.

Physical Path

XHQ BI Data Provider web service is located in the `%XHQ_WEB_HOME%\webapps\xhqodata` folder. It works out-of-the-box and does not require any user configuration.

To connect to a remote XHQ Server or query XHQ global collections, see the topic, *Application Configuration*.

Virtual Directory Path

The XHQ BI Data Provider web service virtual directory is located in the `indx/xhqodata` node in IIS.

Application Pool

The XHQ BI Data Provider web service is served by a dedicated application pool called **`XhqxhqodataPool`**

IIS recycles the application pool at the specified "Regular Time Interval (minutes)" (Advanced Settings > Recycling group) irrespective of whether the application pool is idle or not. This setting will impact XHQ BI Data Provider clients, such as Tableau, that directly requests XHQ collection data. If you encounter a "404 Resource not found" error when requesting collections data directly, retry after visiting the XHQ BI Data Provider service root URL.

Security

The XHQ BI Data Provider web service uses integrated Windows authentication. The application runs under the security context of the user authenticated by IIS. The same user account is used to connect to the XHQ Server.

Application Configuration

The XHQ BI Data Provider web service looks for an application configuration file, **`AppConfig.xml`**, in the following locations and in the order specified.

1. `/indx/custom/xhqodata/AppConfig.xml`
2. `~/Config/AppConfig.xml`

The following table shows the configuration keys and its purpose.

Key	Description
`XhqServerConnectionString`	The value must be an XHQ ADO.NET connection string, containing the XHQ Server host name. The default value is “data source=localhost”.
`XhqServer`	The value must be an XHQ Server host name. The default value is “localhost”.
`ShowGlobalCollections`	The value must either be “True” or “False”. If the value is “True”, XHQ global collections are listed in the service document feed and users can request global collection data. The default is “False”. **WARNING**: Setting this value to “True” might impact performance.

XHQ BI Data Provider Home Page URL

XHQ BI Data Provider homepage URL is the landing page for the XHQ BI Data Provider web service.

The XHQ BI Data Provider web service, requires HTTP or HTTPS access and has the following base URL address:

```
http://<XHQ_SERVER>/indx/xhqodata
```

Example:

```
http://localhost/indx/xhqodata
```

URL Components

XHQ BI Data Provider web service URL has at most three significant parts: the **service root URL**, the **resource path**, and the **query options**.

The following are two example URLs broken down into their component parts:

```
http://localhost/indx/xhqodata/xhqodata.svc
_____________________________________________/
                    |
             Service root URL

http://localhost/indx/xhqodata/xhqodata.svc/XHQCollection/?$orderby=aColumn
_____________________________________/  ____________/  _____________/
                |                          |                |
        Service root URL                resource path    query options
```

The XHQ BI Data Provider Web Service Root (Service Document) URL

The service root URL identifies the root of XHQ BI Data Provider service. It can be accessed using the following URL:

```
http://<XHQ_SERVER>/indx/xhqodata/xhqodata.svc
```

The XHQ BI Data Provider web service root URL is used to request the **Service Document**. The Service Document provides a list of entity collections that can be queried from the service.

The Service Root URL serves two purposes. It generates the meta data for all the XHQ collections the user has access to and returns the service document. If you are accessing the XHQ BI Data Provider web service for the first time, you must first visit the service root URL before performing any other operations.

A valid BI Data Provider license is required to access the service root URL and the available resources.

The service document returned by the service root URL is in AtomPub (XML) format. The service document can be requested in JSON format using the following URL.

```
http://<XHQ_SERVER>/indx/xhqodata/xhqodata.svc/?$format=json
```

Metadata Document Request (Service Metadata Document) URL

The Service Metadata Document describes an XML representation of the data model and operations exposed by the OData service. It's similar to the schema definition of relational tables.

To request metadata document, use the following URL:

```
http://<XHQ_SERVER>/indx/xhqodata/xhqodata.svc/$metadata
```

Requesting Data Using Resource Path

The resource path section of the URL identifies the entity collection (XHQ collection) to be queried.

OData standards allow a URL requesting a collection (of entities), a single entity within a collection, and related collection. However, the XHQ BI Data Provider web service allows only one collection in the resource path.

For example, the following URL does not work even though it's a valid URL according to the OData standard.

```
http://[XHQ_SERVER]/indx/xhqodata/xhqodata.svc/Collection(1)
/RelatedCollection
```

To request the XHQ collection data from the XHQ BI Data Provider web service, use the following URL syntax:

```
http://<XHQ_SERVER>/indx/xhqodata/xhqodata.svc/<XHQ_Collection_Name>
```

For example, to request data for `::Sample.Area1.Inventory.colMaterials` link collection, the URL is:

```
http://<XHQ_SERVER>/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___
colMaterials
```

The dot (".") in the link collection path is replaced with triple underscores ("___").

For more information, see the topic, *Addressing Entities*.

Addressing Entities

The basic rules for requesting XHQ collection data using the resource path are as follows:

1. Do not use a double-colon (“::”) prefix in collection names.

   For example, to request data for a collection named “::acme_collection”, use the following URL:

   ```
   http://<XHQ_SERVER>/indx/xhqodata/xhqodata.svc/acme_collection
   ```

2. To request data for link and nested collections, replace the dot (“.”) in the path with triple underscores ("___").

   For example, to request data for a link collection “::Sample.Area1.Inventory.colMaterialCategories”, use the following URL:

   ```
   http://<XHQ_SERVER>/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___
   colMaterialCategories
   ```

Basic OData Support

The XHQ BI Data Provider web service supports a limited subset of the OData Version 3.0 Core Protocol, with limited support for querying collections of entities.

The following query options are supported.

- `$filter`
- `$select`
  All columns are returned. Equivalent to `SELECT * FROM “XHQ Collection”`.
- `$orderby`
- `$top`
- `$format`

Querying Collections

XHQ BI Data Provider services support querying XHQ collections. The target collection is specified through a URL, and query operations such as filter, projection, and sort are specified as query options. Filter query options are prefixed with the dollar sign (“$”) character.

The $filter System Query Option

The set of entities (rows/records) returned may be restricted through the use of the `$filter` query option.

For example:

```
http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___
colTMTanks/?$filter=sMaterial eq ‘LCO’
```

This returns all records from the `::Sample.Area1.Inventory.colTMTanks` link collection, whose `sMaterial` value is "LCO".

The following table lists a set of **built-in operations** supported by the `$filter` option.

Operator	Description	Examples
eq	Equality operator	http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___colTMTanks/?$filter=rMax eq 200 http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___colTMTanks/?$filter=rMax eq '200' http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___colTMTanks/?$filter=sMaterial eq 'LCO'
ne	Inequality operator	http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___colTMTanks/?$filter=rMax ne 200
gt	Greater-than operator	http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___colTMTanks/?$filter=rMax gt 200
lt	Less-than operator	http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___colTMTanks/?$filter=rMax lt 200
le	Less-than or equal-to operator.	http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___colTMTanks/?$filter=rMax le 200
ge	Greater-than or equal-to operator	http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___colTMTanks/?$filter=rMax ge 200
and	Logical-and operator	http://localhost/indx/xhqodata/xhqodata.svc/solution_trends_actuals?$filter=fully_qualified_name eq '13FC51' and point_count eq 100 and time_stamp ge datetime'2016-12-01T00:00:00' and time_stamp le datetime'2016-12-02T00:00:00'
or	Logical-or operator	http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___colTMTanks/?$filter=sMaterial eq 'LCO' or sMaterial eq 'LGO'

XHQ BI Data Provider web service supports the following **built-in functions** that can be used with the `$filter` query option.

Function	Example
bool startswith(string string, string prefixString)	http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___colTMTanks/?$filter=startswith(sMaterialCategory,'INTER')
bool endswith(string string, string suffixString)	http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___colTMTanks/?$filter=endswith(sMaterialCategory,'STOCK')
bool contains(string string, string searchString)	http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___colTMTanks/?$filter=contains(sMaterialCategory,'FEED')

The $select System Query Option

The `$select` System Query option requests that the service return only the properties (columns) requested by the client.

The value of the `$select` query option is a comma separated list of columns.

For example, the following request returns just the sMaterial, sMaterialCategory and rMax columns from the `::Sample.Area1.Inventory.colTMTanks` link collection:

```
http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___
colTMTanks/?$select=sMaterial,sMaterialCategory,rMax
```

The $orderby System Query Option

The `$orderby` System Query option specifies the order in which the entities (rows/records) are returned from the service.

The value of the `$orderby` query option contains a comma separated list of columns.

The coloumns name may include the suffix "asc" for ascending and "desc" for descending, separated by one or more spaces. If "asc" or "desc" is not specified, the service returns the data in ascending order.

For example, the following request returns just the sMaterialCategory,sMaterial and rMax columns with sMaterialCategory and sMaterial columns in ascending order and rMax colum in descending order:

```
http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___
colTMTanks/?$select=sMaterialCategory,sMaterial,rMax&$orderby=sMaterialCategory,sMater
ial,rMax desc
```

The $top System Query Option

The `$top` System Query option specifies that the service return only the first n records requested by the client, where n is a non-negative integer value.

For example, the following request returns just the sMaterial, sMaterialCategory and rMax columns from `::Sample.Area1.Inventory.colTMTanks` link collection:

```
http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___
colTMTanks/?$top=5
```

This returns only the first 5 records from the `::Sample.Area1.Inventory.colTMTanks` link collection.

The $format System Query Option

A request with the `$format` System Query option specifies that the response should use the media type specified by the query option.

If the `$format` query option is present, it takes precedence over the value(s) specified in the accept request header.

- If the value of the query option is "atom" or "xml", then the media type used in the response is "application/atom+xml".
- If the value of the query option is "json", then the media type used in the response is "application/json".
- If the `$format` query option is NOT specified, the service returns collection data in JSON format and the service document in AtomPub (XML) format.

Examples:

```
http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___
colTMTanks/?$format=xml
```

```
http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___
colTMTanks/?$format=atom
```

To request the service document in JSON format, use:

```
http://localhost/indx/xhqodata/xhqodata.svc?$format=json
```

Querying Solution Values and Solution Time-Series (Trend) Data

Users can request solution values and solution time-series data using “solution_values_*” and “solution_trend_*” collections. However, the “solution_*” collections require filters with specific properties (columns) in the request.

The following table provides the list of “solution_*”, “solution_values_*”, and “solution_trend_*” collections and required filters.

Collection	Required Filter Values
solution_instances	oid, class_name, or fully_qualified_name.
solution_members	fully_qualified_name or (parent_path + name), or class_name.
solution_trends_actuals	(fully_qualified_name or (parent_path + name)), time_stamp, and point_count.
solution_trends_actuals_with_adjacent	(fully_qualified_name or (parent_path + name)), time_stamp, and point_count.
solution_trends_fits	(fully_qualified_name or (parent_path + name)), time_stamp, and point_count.
solution_trends_timesliced_avg	(fully_qualified_name or (parent_path + name)), time_stamp, and point_count.
solution_trends_timesliced_interpolate	(fully_qualified_name or (parent_path + name)), time_stamp, and point_count.
solution_trends_timesliced_max	(fully_qualified_name or (parent_path + name)), time_stamp, and point_count.
solution_trends_timesliced_min	(fully_qualified_name or (parent_path + name)), time_stamp, and point_count.
solution_values_actual	fully_qualified_name or (parent_path + name), and time_stamp.
solution_values_at	fully_qualified_name or (parent_path + name), and time_stamp.
solution_values_avg	fully_qualified_name or (parent_path + name), and time_stamp.
solution_values_count	fully_qualified_name or (parent_path + name), and time_stamp.
solution_values_delta	fully_qualified_name or (parent_path + name), and time_stamp.
solution_values_first	fully_qualified_name or (parent_path + name), and time_stamp.
solution_values	fully_qualified_name or (parent_path + name), and time_stamp.
solution_values_integral	fully_qualified_name or (parent_path + name), and time_stamp.
solution_values_last	fully_qualified_name or (parent_path + name), and time_stamp.

Collection	Required Filter Values
solution_values_max	fully_qualified_name or (parent_path + name), and time_stamp.
solution_values_min	fully_qualified_name or (parent_path + name), and time_stamp.
solution_values_sum	fully_qualified_name or (parent_path + name), and time_stamp.

Business Intelligence (BI) Tools Integration

Business Intelligence (BI) tools such as Microsoft Power BI, TIBCO Spotfire, and Tableau, can request data from the XHQ BI Data Provider web service using their corresponding OData connectors.

Connectivity has been tested with the following versions of the BI tools.

BI Tool	Version
Microsoft Power BI Desktop	2.40.4554.463 64-bit (October 2016)
Tableau Desktop Professional Edition	10.2.2 (10200.17.0505.1445) 64-bit
TIBCO Spotfire	7.5.0.79
Qlik Web Connectors	2.1.0.19723
Microsoft Power Query for Excel	2.38.4491.181 32-bit

Refer to BI tool documentation for connecting to OData web service, creating analytics, reports, dashboards and visualizations.

The XHQ Server will return a "*402 Payment Required*" HTTP status code if the XHQ BI Data Provider is not licensed. Note, this status code is reserved for future use and its purpose and usage may change.

Microsoft Power BI Desktop

Refer to Microsoft Power BI Desktop documentation on how to connect to OData feeds.

`https://powerbi.microsoft.com/en-us/documentation/powerbi-desktop-connect-odata/`

Some of the solution values and solution time-series collections require time_stamp range as one of the filters. When you setup time_stamp range filter, make sure to select **Between** in the list of **Date/Time** filters.

In addition, parallel loading of multiple XHQ collections using XHQ BI Data Provider in Power BI results in a "500 Internal Server Error." If this occurs, from Power BI, go to the File menu > Options and Settings > Options > CURRENT FILE > Data Load. Uncheck the "Enable parallel loading of tables" setting.

Tableau Desktop Professional Edition

Refer to Tableau Desktop Professional Edition documentation on how to connect to OData feeds.

`http://onlinehelp.tableau.com/v10.1/pro/desktop/en-us/help.htm#examples_odata.html?Highlight=odata`

Tableau does not support OData feed containing service documents or service metadata documents. The request URL should contain the service root URL + the resource path at a minimum and query options if needed.

For example, to request the first 5 records from the `::Sample.Area1.Inventory.colTMTanks` link collection:

`http://localhost/indx/xhqodata/xhqodata.svc/Sample___Area1___Inventory___colTMTanks/?$top=5`

Before requesting data from XHQ BI Data Provider web service, ensure that the collection metadata has been generated by visiting the service root URL.

5 | Data Retrieval Modes for XHQ

Modes are used to define data retrieval of time series data. There are no modes for current value retrieval. Current values are always the most recent value retrieved from the data source. Hence, current values are not the value at the current time.

When the data source (historian) provides the desired mode, XHQ does not compute the values. It passes the appropriate parameters to the data source such that the data source computes the values.

Refer to the documentation of the particular historian for details about computations of aggregates.

This topic is limited to XHQ calculations when the data source cannot provide the desired modes.

API – Mode Groups

Group	Modes
TSVAL	Used in calls that return a single value given an input timestamp. • AT Interpolated value at the timestamp. • ACTUAL Nearest value at or before the timestamp. **Note:** The interpolation type uses the same approach that is referenced in the XHQ Reference Guide.
TSAGG	Used in calls that return a single aggregate value given an input time span (start and end times). • AVG The time-weighted average of the values in the time span. • MAX The maximum value of the items in the time span. • MIN The minimum value of the items in the time span.
TSMODE	Used in calls that return a series of values given an input time span (start and end times) and a number of points. Depending on the availability of raw values, the number of points returned may be different. • ACTUALS The actual (raw) values of the items in the time span. • ACTUALS_WITH_ADJACENTS The actual (raw) values of the items in the time span plus the value before the start and the value after the end of the time span. • TIMESLICED_FITS A data compressed set of up to 1000 values suitable for plotting in a trend chart. • TIMESLICED_AVG The time-weighted average of each time slice.

Group	Modes
	• TIMESLICED_INTERPOLATE The interpolated value for each time slice. • TIMESLICED_MAXIMUM The maximum value of the items in each time slice. • TIMESLICED_MINIMUM The minimum value of the items in each time slice.

About Time Slices

The time span between start and end is divided into equal *time slices* based on the number of points (API, Web Services, ADO.NET). The start time is used as the reference (start of the first slice). And the number of slices is equal to the number of points minus 1.

A value is returned for each of the timestamps that results from dividing the time span by the number of points. For example, if the time span is 5 minutes and the number of points is 6, then values will be returned for each of the following timestamps: start, start + 1min, start + 2min, start + 3min, start + 4min, and end. How the values are computed is explained in the topic, "Raw Values and Computation of Aggregates". So, a time slice is basically the interval between successive timestamps. Time-sliced aggregate values are reported with a timestamp that corresponds to the beginning of the slice.

Refer to the topic, *Historical Aggregate Methods*, located in the XHQ Reference Guide, for details on time-weighing.

Raw Values and Computation of Aggregates

Inside each time slice, aggregate values are computed as follows.

- Using all the raw points inside of the slice.
- And in the cases where time weighing is used, if there are no corresponding actual/raw value at the boundaries of the slice, an additional value is added to the start and the end of the slice using linear interpolation (between the adjacent before and the first raw value in the slice and between the adjacent after and the last raw value).

Aggregate values (TIMESLICED_*) are reported using the time stamp at the start of the slice.

Views and Server Side Expressions

Expressions only return a single value. And so, there is no equivalent to TSMODE.

The following **Historical Access Methods** are equivalent to the **TSVAL** counterparts:

- .at()
- .actual()

And the following **Historical Aggregate Methods** are equivalent to the **TSAGG** counterparts:

- .min()
- .max()
- .avg()
- .integral()
- .count()
- .sum()
- .first()
- .last()
- .delta()

Trends

Trends retrieve a time series of values, which is equivalent to TSMODE.

This mode:	Maps to this Trend retrieval:
TSMODE	
• TIMESLICED_FITS	FITS (which is the default)
• ACTUALS_WITH_ADJACENTS	RAW
• TIMESLICED_AVG	AVG
• TIMESLICED_MIN	MIN
• TIMESLICED_MAX	MAX
• TIMESLICED_INTERPOLATE	INTERPOLATE

Web Services

The following table maps the modes to their Web Services method counterparts.

This mode:	Maps to this Web Services method:
TSVAL	
• AT	Get*AtValue(s)
• ACTUAL	Get*ActualValue(s)
TSAGG	
• AVG	Get*AverageValue(s)
• MAX	Get*MaximumValue(s)
• MIN	Get*MinimumValue(s)
TSMODE	GetTimeSeriesData This method supports passing of any of the TSMODE modes. Refer to the topic, *Web Service Operations*.

* Can be Point or Bulk (that is, more than one XHQ primitive member or Tag).

Appendices

Section Contents

- A - For Performance Management
- B - XHQ Web Services WSDL Interface
- C - Database Views

A - For Performance Management

Performance Management Client API

The following methods are supported by the Application Server Client API.

getConnection

```
public static PerformanceManagement getConnection(java.lang.String pmHost)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Gets a shared instance of this class for interfacing with the XHQ Application Server on the specified server machine host. The application is responsible for managing the life cycle of this connection. However, when no references to the shared instance are present, the connection may be closed.

Parameters: `pmHost` - The host on which the Application server is running.

Returns: A shared instance.

Throws: `PMCommunicationException` - If there is a problem communicating with the XHQ App Server.

`com.indx.xhq.pm.clientapi.AppCommunicationException`

disconnect

```
public void disconnect()
```

Description: Disconnects the client from the XHQ Application Server. Any subsequent API calls fail. Any asynchronous updates are stopped.

createPerformanceIndicator

```
public com.indx.xhq.pm.clientapi.AppServerReply createPerformanceIndicator
(com.indx.xhq.pm.clientapi.PerformanceIndicator pi)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Creates a new performance indicator.

Parameters: `userId` – The XHQ User ID.

`pi` - Performance Indicator object.

Returns: The newly created Performance Indicator, contained in the "object" field of the "AppServerReply".

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

updatePerformanceIndicator

```
public com.indx.xhq.pm.clientapi.AppServerReply updatePerformanceIndicator
(com.indx.xhq.pm.clientapi.PerformanceIndicator pi)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Updates an existing performance indicator.

Parameters: `userId` – The XHQ User ID.

`pi` - Performance Indicator object.

Returns: The updated Performance Indicator, contained in the "object" field of the "AppServerReply".

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

deletePerformanceIndicator

```
public com.indx.xhq.pm.clientapi.AppServerReply deletePerformanceIndicator(int piId)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Deletes an existing performance indicator.

Parameters: `userId` – The XHQ User ID.

`piId` - PI ID

Returns: A Boolean flag indicating whether the delete operation has been successful, contained in the "successFlag" field of the "AppServerReply

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getPerformanceIndicator

```
public com.indx.xhq.pm.clientapi.AppServerReply getPerformanceIndicator(int piId)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Retrieves a complete performance indicator.

Parameters: `userId` – The XHQ User ID.

`piId` - PI ID

Returns: The Performance Indicator, contained in the "object" field of the "AppServerReply".

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getPerformanceIndicatorList

```
public com.indx.xhq.pm.clientapi.AppServerReply getPerformanceIndicatorList
(java.lang.String whereClause, java.lang.String orderByClause,
int pageSize, int pageIndex)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Retrieves a filtered performance indicator list.

Parameters: `userId` – The XHQ User ID.

`whereClause` – A WHERE clause that specifies selection criteria; supplying "WHERE" at

the beginning of the string is optional.

`orderByClause` - An ORDER BY clause that specifies selection criteria; supplying "ORDER BY" at the beginning of the string is optional.

`pageSize` - An integer that specifies the number of PIs per page, for pagination purposes.

`pageIndex` - An integer that specifies the page index number, for pagination purposes; it starts from 0 (zero).

Returns: An "AppServerReply" object that has the following fields: a Boolean successFlag field that indicates if the operation has been successful or not; a String message that contains an error message, if any; a field containing the actual Exception object, if any; an Object field containing an array of PIs.

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getPiCount

```
public com.indx.xhq.pm.clientapi.AppServerReply getPiCount(java.lang.String whereClause)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Retrieves the count for all PIs for a certain user.

Parameters: `userId` – The XHQ User ID.

`whereClause` - A WHERE clause that specifies selection criteria; supplying "WHERE" at the beginning of the string is optional.

Returns: The PI count as an integer in the "object" field of the "AppServerReply".

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getPiNames

```
public com.indx.xhq.pm.clientapi.AppServerReply getPiNames(java.lang.String whereClause,
java.lang.String orderByClause)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Retrieves a filtered list of PI names from the database.

Parameters: `whereClause` - A WHERE clause that specifies selection criteria; each column must be prefixed with "p."; supplying "WHERE" at the beginning of the string is optional.

`orderByClause` - An ORDER BY clause that specifies selection criteria; each column must be prefixed with "p."; supplying "ORDER BY" at the beginning of the string is optional.

Returns: The PI names String array contained in the "object" field of the "AppServerReply".

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getPiMetrics

```
public com.indx.xhq.pm.clientapi.AppServerReply getPiMetrics(int metricType,
java.lang.String whereClause, java.lang.String orderByClause)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Retrieves a filtered list of PI metrics from the database.

Parameters: `metricType` - An integer that specified the desired type of the return metrics; Can have one the following four values specified in the Metric class: TYPE_ALIAS, TYPE_MEMBER, TYPE_EXPRESSION, TYPE_ALL.

`whereClause` - A WHERE clause that specifies selection criteria; supplying "WHERE" at the beginning of the string is optional.

`orderByClause` - An ORDER BY clause that specifies selection criteria; supplying "ORDER BY" at the beginning of the string is optional.

Returns: The PI metrics String array contained in the "object" field of the "AppServerReply".

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getAppCodesList

```
public com.indx.xhq.pm.clientapi.AppServerReply getAppCodesList(int userId,
int appCodeId, java.lang.String codeType)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Retrieves a list of codes of a certain type (for example, PI Types, PI Priorities) for a certain application (such as ANS or TM or eLogs).

Parameters: `userId` – The XHQ User ID.

`appCodeId` – The application key.

`codeType` – The code type.

Returns: An array of CodeName objects contained in the "object" field of the "AppServerReply".

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getCode

```
public com.indx.xhq.pm.clientapi.AppServerReply getCode(int userId, int codeId)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Retrieves the complete code information.

Parameters: `userId` – The XHQ User ID.

`codeId` – The code ID.

Returns: A CodeName object contained in the "object" field of the "AppServerReply".

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

setAppCodeAccess

```
public com.indx.xhq.pm.clientapi.AppServerReply setAppCodeAccess(int userId,
com.indx.xhq.pm.clientapi.CodeName[ ] codes)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Sets the code access for all available applications (such as ANS and TM).

Parameters: `userId` – The XHQ User ID.

`codes` – The codes array.

Returns: A Boolean flag indicating whether the delete operation has been successful; contained in the "successFlag" field of the "AppServerReply".

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

setCode

```
public com.indx.xhq.pm.clientapi.AppServerReply setCode(int userId,
com.indx.xhq.pm.clientapi.CodeName code)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Creates a new code or updates an existing one.

Parameters: `userId` – The XHQ User ID.

`code` – The complete Target Management code information.

Returns: A Boolean flag indicating whether the delete operation has been successful; contained in the "successFlag" field of the "AppServerReply".

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getTarget

```
public com.indx.xhq.pm.clientapi.AppServerReply getTarget(int piId)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Retrieves a performance indicator's Target.

Parameters: `piId` - PI ID

Returns: The Target, contained in the "object" field of the "AppServerReply".

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

updateTarget

```
public com.indx.xhq.pm.clientapi.AppServerReply updateTarget(int piId,
com.indx.xhq.pm.clientapi.Target target)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Updates an existing Target.

Parameters: `piId` - PI ID

`target` - The PI Target.

Returns: The updated Target, contained in the "object" field of the "AppServerReply".

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getTargetAuditTrail

```
public com.indx.xhq.pm.clientapi.AppServerReply getTargetAuditTrail(int piId,
int targetId)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Retrieves a complete target history (audit trail) for a certain PI.

Parameters: `piId` - PI ID

`targetId` - Target ID

Returns: An array of Target objects, contained in the "object" field of the "AppServerReply".

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getLimitsAuditTrail

```
public com.indx.xhq.pm.clientapi.AppServerReply getLimitsAuditTrail(int piId)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Retrieves a complete history (audit trail) for all the limits of a certain PI.

Parameters: `piId` – PI ID

Returns: An array of arrays of Limit objects, contained in the "object" field of the "AppServerReply"; the first dimension of the multidimensional array indicates the number of distinct limits for the PI, while the second dimension indicates how many different versions we have for each individual limit.

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getComponentMembers

```
public com.indx.xhq.pm.clientapi.AppServerReply getComponentMembers
(java.lang.String parentPath)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Retrieves a list of all child component and primitive member names for a certain XHQ parent component.

Parameters: `parentPath` - The parent component path. A null path or a "::" path indicates the top level of the solution tree.

Returns: An array of 2 arrays of Strings contained in the "object" field of the "AppServerReply". The first array contains the list of all child component names, while the second array contains the list of all child primitive member names.

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getTagAliases

```
public com.indx.xhq.pm.clientapi.AppServerReply getTagAliases()
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Retrieves a list of all tag aliases in an existing XHQ Solution.

Parameters: N/A

Returns: An array of 3 String arrays contained in the "object" field of the "AppServerReply". The first arrays contains the tag name, the second one the tag description and the third one the tag units.

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

importPerformanceIndicators

```
public AppServerReply importPerformanceIndicators(byte[ ] xmlStream)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Imports from XML format a list of PIs in the App server.

Parameters: `xmlStream` - A byte array representing a XML chunk read from client file and needed to be parsed.

Returns: A Boolean flag indicating the success of the operation

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

exportPerformanceIndicators

```
public AppServerReply exportPerformanceIndicators(int[ ] piIds)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Exports into XML format a list of PIs from the App server, identified by their ID.

Parameters: `piIds` - An integer array containing the ids of the PIs to be exported.

Returns: A byte array `representing` the XML ready to be written to a file on the client.

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

authenticateUser

```
public com.indx.xhq.pm.clientapi.AppServerReply authenticateUser
(java.lang.String username)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Authenticates the user based on his/her OS username and returns back user info like user Id and user permissions.

Parameters: `username` – The OS username.

Returns: A UserInfo object contained in the "object" field of the "AppServerReply".

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getRoleList

```
public com.indx.xhq.pm.clientapi.AppServerReply getRoleList(int userId)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Gets the list of all XHQ-defined Roles.

Parameters: `userId` – The User ID.

Returns: An array of Role objects contained in the "object" field of the "AppServerReply".

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getPermissionList

```
public com.indx.xhq.pm.clientapi.AppServerReply getPermissionList(int userId,
java.lang.String permType)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Gets the list of all XHQ-defined Permissions for a certain type of app (such as TM, eLogs, or ANS).

Parameters: `userId` – The User ID.

`permType` – A string indicating the permissions/application type.

Returns: An array of CodeName objects containing the permission Id and name.

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getRolePermissions

```
public com.indx.xhq.pm.clientapi.AppServerReply getRolePermissions(int userId, int
roleId, java.lang.String permType)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Gets the list of all permissions for a given XHQ-defined role.

Parameters: `userId` – The User ID.

`roleId` – The role ID.

`permType` – A string indicating the application permission type (for example, 'TM_PERMISSION', 'ANS_PERMISSION').

Returns: An array of integer representing the permission Ids for the specified role.

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getRoleCategories

```
public com.indx.xhq.pm.clientapi.AppServerReply getRoleCategories(int userId, int roleId,
int appId, java.lang.String catType)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Gets the list of all XHQ-defined Permissions for a certain type of app (such as TM, eLogs, or ANS).

Parameters: `userId` – The User ID.

`roleId` – The role ID.

`appId` – The application ID.

`catType` – A string indicating the category type (for example, 'ANS_CONDITION_CATEGORY').

Returns: An array of integer representing the permission Ids for the specified role.

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getRolePaths

```
public com.indx.xhq.pm.clientapi.AppServerReply getRolePaths(int userId, int roleId, int
appId)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Gets the list of XHQ paths (nodes) for a certain XHQ-defined Role and for a certain application (such as TM, eLogs, or Shift Reports).

Parameters: `userId` – The User ID.

`roleId` – The role ID.

`appId` – The application ID.

Returns: An array of strings representing the paths/nodes for the specified role; it can include the (*) wildcard in the path syntax. The following syntax is supported for each path:

- An XHQ absolute path with no wildcards (i.e. "::Enterprise.SIMPoint.CurrentValue");
- A single * wildcard indicates that the user can write to any path;
- An XHQ absolute path that ends with a wildcard like ".*" (i.e. "::Enterprise.Area1.*"), meaning the user can write to the "::Enterprise.Area1" node and any of its children nodes, all the way to the leafs of that tree area;
- An XHQ absolute path that contains a certain area in between wild cards (i.e. "*.Area1.*"), meaning the user can write to the any path that has an "Area1" node.

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

updateRolePermissions

```
public com.indx.xhq.pm.clientapi.AppServerReply updateRolePermissions(int userId, int
roleId, java.lang.String permType,
int[] permList)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Updates the list of permissions for a given XHQ-defined Role.

Parameters: `userId` – The User ID.

`roleId` – The role ID.

`permType` – A string indicating the application permission type (for example, 'TM_ PERMISSION', 'ANS_PERMISSION').

`permList` – A byte array representing the role permissions list.

Returns: A Boolean flag indicating the success or failure of the update operation.

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

updateRoleCategories

```
public com.indx.xhq.pm.clientapi.AppServerReply updateRoleCategories(int userId, int
roleId, int appId, java.lang.String catType, int[] catList)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Updates the list of categories for a given XHQ-defined Role.

Parameters: `userId` – The User ID.

`roleId` – The role ID.

`appId` – The application ID.

`catType` – A string indicating the category type (for example, 'ANS_CONDITION_CATEGORY').

`catList` – A byte array representing the role categories list.

Returns: A Boolean flag indicating the success or failure of the update operation.

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

updateRolePaths

```
public com.indx.xhq.pm.clientapi.AppServerReply updateRolePaths(int userId, int roleId,
int appId, java.lang.String[] pathList)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Updates the list of XHQ paths (nodes) for a certain XHQ-defined Role and for a certain application (such as TM, eLogs, or Shift Reports).

Parameters: `userId` – The User ID.

`roleId` – The role ID.

`appId` – The application ID.

`pathList` – A string array representing the role paths/nodes list; it can include the (*) wildcard in the path syntax. The following syntax is supported for each path:

- An XHQ absolute path with no wildcards (i.e. "::Enterprise.SIMPoint.CurrentValue");
- A single * wildcard indicates that the user can write to any path;
- An XHQ absolute path that ends with a wildcard like ".*" (i.e. "::Enterprise.Area1.*"), meaning the user can write to the "::Enterprise.Area1" node and any of its children nodes, all the way to the leafs of that tree area;
- An XHQ absolute path that contains a certain area in between wild cards (i.e. "*.Area1.*"), meaning the user can write to the any path that has an "Area1" node.

Returns: A Boolean flag indicating the success or failure of the update operation.

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getUserList

```
public com.indx.xhq.pm.clientapi.AppServerReply getUserList(int userId)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Gets the list of all TM users.

Parameters: `userId` – The User ID.

Returns: An array of UserInfo objects contained in the "object" field of the "AppServerReply".

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

getOwnerPiList

```
public com.indx.xhq.pm.clientapi.AppServerReply getOwnerPiList(int userId,
int piOwnerId)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Gets the list of all PIs that a certain user owns.

Parameters: `userId` – The User ID.

`piOwnerId` – The User ID for the owner of the PI list.

Returns: An array of CodeName objects containing the PI ID and PI name.

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

changePiOwner

```
public com.indx.xhq.pm.clientapi.AppServerReply changePiOwner(int userId,
int piOwnerId, int[] piIds)
throws com.indx.xhq.pm.clientapi.AppCommunicationException
```

Description: Transfers the ownership of the PI list to another user.

Parameters: `userId` – The User ID.

`piOwnerId` – The new User ID for the owner of the PI list.

`piIds` – An integer array containing the IDs of the PIs to be exported.

Returns: A Boolean flag, indicating whether the PI change of ownership operation has been successful, contained in the "successFlag" field of the "AppServerReply".

Throws: `com.indx.xhq.pm.clientapi.AppCommunicationException`

Web Services Methods for Performance Management

The following methods enable you to browse, filter, bulk load, and bulk update Performance Indicators in XHQ Performance Management.

getPerformanceIndicatorList

```
public WebPi[] getPerformanceIndicatorList(String appServerHost, String whereClause,
String orderByClause, int pageSize,int pageIndex)
```

Description: Retrieves a filtered list of performance indicators.

getPerformanceIndicator

```
public WebPi getPerformanceIndicator(String appServerHost, int piId)
```

Description: Retrieves a complete performance indicator.

getPerformanceIndicatorCount

```
public int getPerformanceIndicatorCount(String appServerHost, String whereClause)
```

Description: Retrieves the count of all KPIs defined in the system.

bulkLoadPerformanceIndicators

```
public byte[] bulkLoadPerformanceIndicators(String appServerHost, int[] piIds)
```

Description: Bulk loads a number of PIs based on their IDs.

bulkUpdatePerformanceIndicators

```
public int[] bulkUpdatePerformanceIndicators(String appServerHost, byte[] xmlStream)
```

Description: Bulk updates a number of PIs.

To access the Performance Management Web Services

Go to **http://<WS_MACHINE_HOST>/indx/pmws/TmWs.asmx**.

Where **<WS_MACHINE_HOST>** is the machine where XHQ Server and the XHQ SDK is installed.

For detailed information on each method, go to **http://<WS_MACHINE_HOST>/indx/pmws/TmWs.asmx?WSDL**.

EXAMPLE 1: BULK LOAD AND GET THE XML FOR PIS USING THE EXCEL ADD-IN

1. Call the **`getPerformanceIndicatorList()`** web method.
   This gets (returns) the PI IDs. You can filter or use pagination, or opt to return all PIs.
2. Next, call the **`bulkLoadPerformanceIndicators()`** method.
   This returns the export XML, using the PI IDs returned in step 1 as input. In turn, these can be mapped and represented in the Excel spreadsheet.

EXAMPLE 2: BULK UPDATE OR SAVE THE CHANGES USING IMPORT XML

1. Call the **`bulkUpdatePerformanceIndicators()`** method.
2. Pass the XML byte stream as an input parameter.

B - XHQ Web Services WSDL Interface

WSDL for XhqWse Web Service

WSDL Location: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl
Target Namespace: http://tempuri.org/

Services	Bindings	Port Types	Messages	Types
Xhq Wse	BasicHttpBinding_XHQ_x0020_Web_x0020_Service	XHQ_x0020_Web_x0020_Service	XHQ_x0020_Web_x0020_Service_DeleteCollectionData_InputMessage	anyType
			XHQ_x0020_Web_x0020_Service_DeleteCollectionData_OutputMessage	anyURI
			XHQ_x0020_Web_x0020_Service_GetAllChildMembers_InputMessage	ArrayOfArrayOfstring
			XHQ_x0020_Web_x0020_Service_GetAllChildMembers_OutputMessage	ArrayOfArrayOfstring
			XHQ_x0020_Web_x0020_Service_GetBulkActualValues_InputMessage	ArrayOfstring
			XHQ_x0020_Web_x0020_Service_GetBulkActualValues_OutputMessage	ArrayOfstring
			XHQ_x0020_Web_x0020_Service_GetBulkAtValues_InputMessage	ArrayOfVTQ
			XHQ_x0020_Web_x0020_Service_GetBulkAtValues_OutputMessage	ArrayOfVTQ
			XHQ_x0020_Web_x0020_Service_GetBulkAverageValues_InputMessage	base64Binary
			XHQ_x0020_Web_x0020_Service_GetBulkAverageValues_OutputMessage	boolean
			XHQ_x0020_Web_x0020_Service_GetBulkCountValues_InputMessage	byte
			XHQ_x0020_Web_x0020_Service_GetBulkCountValues_OutputMessage	char
			XHQ_x0020_Web_x0020_Service_GetBulkCurrentValues_InputMessage	char
			XHQ_x0020_Web_x0020_Service_GetBulkCurrentValues_OutputMessage	dateTime
			XHQ_x0020_Web_x0020_Service_GetBulkDeltaValues_InputMessage	decimal

Services	Bindings	Port Types	Messages	Types
			XHQ_x0020_Web_x0020_Service_GetBulkDeltaValues_OutputMessage	DeleteCollectionData
			XHQ_x0020_Web_x0020_Service_GetBulkFirstValues_InputMessage	DeleteCollectionDataResponse
			XHQ_x0020_Web_x0020_Service_GetBulkFirstValues_OutputMessage	double
			XHQ_x0020_Web_x0020_Service_GetBulkIntegralValues_InputMessage	duration
			XHQ_x0020_Web_x0020_Service_GetBulkIntegralValues_OutputMessage	duration
			XHQ_x0020_Web_x0020_Service_GetBulkLastValues_InputMessage	float
			XHQ_x0020_Web_x0020_Service_GetBulkLastValues_OutputMessage	GetAllChildMembers
			XHQ_x0020_Web_x0020_Service_GetBulkMaximumValues_InputMessage	GetAllChildMembersResponse
			XHQ_x0020_Web_x0020_Service_GetBulkMaximumValues_OutputMessage	GetBulkActualValues
			XHQ_x0020_Web_x0020_Service_GetBulkMinimumValues_InputMessage	GetBulkActualValuesResponse
			XHQ_x0020_Web_x0020_Service_GetBulkMinimumValues_OutputMessage	GetBulkAtValues
			XHQ_x0020_Web_x0020_Service_GetBulkSumValues_InputMessage	GetBulkAtValuesResponse
			XHQ_x0020_Web_x0020_Service_GetBulkSumValues_OutputMessage	GetBulkAverageValues
			XHQ_x0020_Web_x0020_Service_GetChildCollectionMembers_InputMessage	GetBulkAverageValuesResponse
			XHQ_x0020_Web_x0020_Service_GetChildCollectionMembers_OutputMessage	GetBulkCountValues
			XHQ_x0020_Web_x0020_Service_GetChildComponentMembers_InputMessage	GetBulkCountValuesResponse
			XHQ_x0020_Web_x0020_Service_GetChildComponentMembers_OutputMessage	GetBulkCurrentValues
			XHQ_x0020_Web_x0020_Service_	GetBulkCurrentValuesResponse

Services	Bindings	Port Types	Messages	Types
			GetChildPrimitiveMembers_InputMessage	
			XHQ_x0020_Web_x0020_Service_GetChildPrimitiveMembers_OutputMessage	GetBulkDeltaValues
			XHQ_x0020_Web_x0020_Service_GetCollectionColumnAverage_InputMessage	GetBulkDeltaValuesResponse
			XHQ_x0020_Web_x0020_Service_GetCollectionColumnAverage_OutputMessage	GetBulkFirstValues
			XHQ_x0020_Web_x0020_Service_GetCollectionColumnCount_InputMessage	GetBulkFirstValuesResponse
			XHQ_x0020_Web_x0020_Service_GetCollectionColumnCount_OutputMessage	GetBulkIntegralValues
			XHQ_x0020_Web_x0020_Service_GetCollectionColumnMaximum_InputMessage	GetBulkIntegralValuesResponse
			XHQ_x0020_Web_x0020_Service_GetCollectionColumnMaximum_OutputMessage	GetBulkLastValues
			XHQ_x0020_Web_x0020_Service_GetCollectionColumnMinimum_InputMessage	GetBulkLastValuesResponse
			XHQ_x0020_Web_x0020_Service_GetCollectionColumnMinimum_OutputMessage	GetBulkMaximumValues
			XHQ_x0020_Web_x0020_Service_GetCollectionColumnSum_InputMessage	GetBulkMaximumValuesResponse
			XHQ_x0020_Web_x0020_Service_GetCollectionColumnSum_OutputMessage	GetBulkMinimumValues
			XHQ_x0020_Web_x0020_Service_GetCollectionCount_InputMessage	GetBulkMinimumValuesResponse
			XHQ_x0020_Web_x0020_Service_GetCollectionCount_OutputMessage	GetBulkSumValues
			XHQ_x0020_Web_x0020_Service_GetCollectionData_InputMessage	GetBulkSumValuesResponse

Services	Bindings	Port Types	Messages	Types
			XHQ_x0020_Web_x0020_Service_GetCollectionData_OutputMessage	GetChildCollectionMembers
			XHQ_x0020_Web_x0020_Service_GetCollectionDataFilter_InputMessage	GetChildCollectionMembersResponse
			XHQ_x0020_Web_x0020_Service_GetCollectionDataFilter_OutputMessage	GetChildComponentMembers
			XHQ_x0020_Web_x0020_Service_GetCollectionDataOrderBy_InputMessage	GetChildComponentMembersResponse
			XHQ_x0020_Web_x0020_Service_GetCollectionDataOrderBy_OutputMessage	GetChildPrimitiveMembers
			XHQ_x0020_Web_x0020_Service_GetCollectionDataWhere_InputMessage	GetChildPrimitiveMembersResponse
			XHQ_x0020_Web_x0020_Service_GetCollectionDataWhere_OutputMessage	GetCollectionColumnAverage
			XHQ_x0020_Web_x0020_Service_GetGlobalCollections_InputMessage	GetCollectionColumnAverageResponse
			XHQ_x0020_Web_x0020_Service_GetGlobalCollections_OutputMessage	GetCollectionColumnCount
			XHQ_x0020_Web_x0020_Service_GetPointActualValue_InputMessage	GetCollectionColumnCountResponse
			XHQ_x0020_Web_x0020_Service_GetPointActualValue_OutputMessage	GetCollectionColumnMaximum
			XHQ_x0020_Web_x0020_Service_GetPointAtValue_InputMessage	GetCollectionColumnMaximumResponse
			XHQ_x0020_Web_x0020_Service_GetPointAtValue_OutputMessage	GetCollectionColumnMinimum
			XHQ_x0020_Web_x0020_Service_GetPointAverageValue_InputMessage	GetCollectionColumnMinimumResponse
			XHQ_x0020_Web_x0020_Service_GetPointAverageValue_OutputMessage	GetCollectionColumnSum
			XHQ_x0020_Web_x0020_Service_GetPointCountValue_InputMessage	GetCollectionColumnSumResponse
			XHQ_x0020_Web_x0020_Service_GetPointCountValue_OutputMessage	GetCollectionCount
			XHQ_x0020_Web_x0020_Service_GetPointCurrentData_InputMessage	GetCollectionCountResponse
			XHQ_x0020_Web_x0020_Service_	GetCollectionData

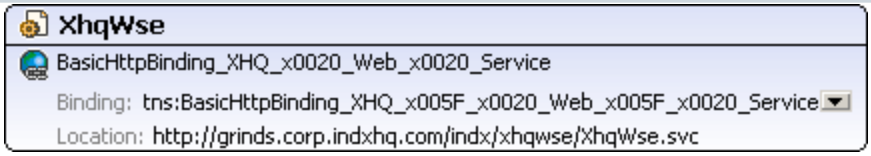
Services	Bindings	Port Types	Messages	Types
			GetPointCurrentData_OutputMessage	
			XHQ_x0020_Web_x0020_Service_GetPointDeltaValue_InputMessage	GetCollectionDataFilter
			XHQ_x0020_Web_x0020_Service_GetPointDeltaValue_OutputMessage	GetCollectionDataFilterResponse
			XHQ_x0020_Web_x0020_Service_GetPointFirstValue_InputMessage	GetCollectionDataOrderBy
			XHQ_x0020_Web_x0020_Service_GetPointFirstValue_OutputMessage	GetCollectionDataOrderByResponse
			XHQ_x0020_Web_x0020_Service_GetPointIntegralValue_InputMessage	GetCollectionDataResponse
			XHQ_x0020_Web_x0020_Service_GetPointIntegralValue_OutputMessage	GetCollectionDataWhere
			XHQ_x0020_Web_x0020_Service_GetPointLastValue_InputMessage	GetCollectionDataWhereResponse
			XHQ_x0020_Web_x0020_Service_GetPointLastValue_OutputMessage	GetGlobalCollections
			XHQ_x0020_Web_x0020_Service_GetPointMaximumValue_InputMessage	GetGlobalCollectionsResponse
			XHQ_x0020_Web_x0020_Service_GetPointMaximumValue_OutputMessage	GetPointActualValue
			XHQ_x0020_Web_x0020_Service_GetPointMinimumValue_InputMessage	GetPointActualValueResponse
			XHQ_x0020_Web_x0020_Service_GetPointMinimumValue_OutputMessage	GetPointAtValue
			XHQ_x0020_Web_x0020_Service_GetPointSumValue_InputMessage	GetPointAtValueResponse
			XHQ_x0020_Web_x0020_Service_GetPointSumValue_OutputMessage	GetPointAverageValue
			XHQ_x0020_Web_x0020_Service_GetSolutionRoot_InputMessage	GetPointAverageValueResponse
			XHQ_x0020_Web_x0020_Service_GetSolutionRoot_OutputMessage	GetPointCountValue
			XHQ_x0020_Web_x0020_Service_GetTimeSeriesData_InputMessage	GetPointCountValueResponse
			XHQ_x0020_Web_x0020_Service_GetTimeSeriesData_OutputMessage	GetPointCurrentData
			XHQ_x0020_Web_x0020_Service_	GetPointCurrentDataResponse

Services	Bindings	Port Types	Messages	Types
			InsertCollectionData_InputMessage	
			XHQ_x0020_Web_x0020_Service_InsertCollectionData_OutputMessage	GetPointDeltaValue
			XHQ_x0020_Web_x0020_Service_IsSolutionAccessible_InputMessage	GetPointDeltaValueResponse
			XHQ_x0020_Web_x0020_Service_IsSolutionAccessible_OutputMessage	GetPointFirstValue
			XHQ_x0020_Web_x0020_Service_ReplaceCollectionData_InputMessage	GetPointFirstValueResponse
			XHQ_x0020_Web_x0020_Service_ReplaceCollectionData_OutputMessage	GetPointIntegralValue
			XHQ_x0020_Web_x0020_Service_UpdateCollectionData_InputMessage	GetPointIntegralValueResponse
			XHQ_x0020_Web_x0020_Service_UpdateCollectionData_OutputMessage	GetPointLastValue
			XHQ_x0020_Web_x0020_Service_WritePointValue_InputMessage	GetPointLastValueResponse
			XHQ_x0020_Web_x0020_Service_WritePointValue_OutputMessage	GetPointMaximumValue
				GetPointMaximumValueResponse
				GetPointMinimumValue
				GetPointMinimumValueResponse
				GetPointSumValue
				GetPointSumValueResponse
				GetSolutionRoot
				GetSolutionRootResponse
				GetTimeSeriesData
				GetTimeSeriesDataResponse
				guid
				guid
				InsertCollectionData
				InsertCollectionDataResponse
				int
				IsSolutionAccessible

Services	Bindings	Port Types	Messages	Types
				IsSolutionAccessibleResponse
				long
				QName
				ReplaceCollectionData
				ReplaceCollectionDataResponse
				short
				string
				unsignedByte
				unsignedInt
				unsignedLong
				unsignedShort
				UpdateCollectionData
				UpdateCollectionDataResponse
				VTQ
				VTQ
				WritePointValue
				WritePointValueResponse

service XhqWse

Diagram:



Ports: BasicHttpBinding_XHQ_x0020_Web_x0020_Service

Binding tns:BasicHttpBinding_XHQ_x005F_x0020_Web_x005F_x0020_Service

Extensibility <soap:address location="http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc"/>

Source:

```
<wsdl:service name="XhqWse">
  <wsdl:port name="BasicHttpBinding_XHQ_x0020_Web_x0020_Service" binding="tns:BasicHttpBinding_
XHQ_x005F_x0020_Web_x005F_x0020_Service">
    <soap:address location="http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc"/>
  </wsdl:port>
</wsdl:service>
```

binding BasicHttpBinding_XHQ_x0020_Web_x0020_Service

Type:	i0:XHQ_x005F_x0020_Web_x005F_x0020_Service
Extensibility:	<soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
Operations:	**GetCollectionColumnMaximum** Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetCollectionColumnMaximum" style="document"/> Input <soap:body use="literal"/> Output <soap:body use="literal"/> **GetCollectionColumnMinimum** Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetCollectionColumnMinimum" style="document"/> Input <soap:body use="literal"/> Output <soap:body use="literal"/> **GetCollectionColumnSum** Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetCollectionColumnSum" style="document"/> Input <soap:body use="literal"/> Output <soap:body use="literal"/> **GetCollectionCount** Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetCollectionCount" style="document"/> Input <soap:body use="literal"/> Output <soap:body use="literal"/> **GetGlobalCollections** Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetGlobalCollections" style="document"/>

Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetSolutionRoot

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetSolutionRoot" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetChildPrimitiveMembers

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetChildPrimitiveMembers" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetChildComponentMembers

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetChildComponentMembers" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetChildCollectionMembers

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetChildCollectionMembers" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetAllChildMembers

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetAllChildMembers" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

WritePointValue

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/WritePointValue" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

InsertCollectionData

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/InsertCollectionData" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

UpdateCollectionData

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/UpdateCollectionData" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

ReplaceCollectionData

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/ReplaceCollectionData" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

DeleteCollectionData

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/DeleteCollectionData" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

IsSolutionAccessible

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/IsSolutionAccessible" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetPointCurrentData

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetPointCurrentData" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetPointAtValue

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetPointAtValue" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetPointActualValue

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetPointActualValue" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetPointAverageValue

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetPointAverageValue" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetPointCountValue

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetPointCountValue" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetPointDeltaValue

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetPointDeltaValue" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetPointFirstValue

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetPointFirstValue" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetPointIntegralValue

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetPointIntegralValue" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetPointLastValue

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetPointLastValue" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetPointMaximumValue

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetPointMaximumValue" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetPointMinimumValue

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetPointMinimumValue" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetPointSumValue

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetPointSumValue" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetBulkCurrentValues

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetBulkCurrentValues" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetBulkAtValues

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetBulkAtValues" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetBulkActualValues

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetBulkActualValues" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetBulkAverageValues

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetBulkAverageValues" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetBulkCountValues

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetBulkCountValues" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetBulkDeltaValues

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetBulkDeltaValues" style="document"/>

Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetBulkFirstValues

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetBulkFirstValues" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetBulkLastValues

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetBulkLastValues" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetBulkIntegralValues

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetBulkIntegralValues" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetBulkMaximumValues

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetBulkMaximumValues" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetBulkMinimumValues

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetBulkMinimumValues" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetBulkSumValues

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetBulkSumValues" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetTimeSeriesData

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetTimeSeriesData" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetCollectionData

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetCollectionData" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetCollectionDataWhere

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetCollectionDataWhere" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetCollectionDataOrderBy

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetCollectionDataOrderBy" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetCollectionDataFilter

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetCollectionDataFilter" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetCollectionColumnAverage

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetCollectionColumnAverage" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

GetCollectionColumnCount

Extensibility <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetCollectionColumnCount" style="document"/>
Input <soap:body use="literal"/>
Output <soap:body use="literal"/>

Source:

```
<wsdl:binding name="BasicHttpBinding_XHQ_x0020_Web_x0020_Service" type="i0:XHQ_x005F_x0020_Web_x005F_x0020_Service">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
  <wsp:PolicyReference URI="#BasicHttpBinding_XHQ_x0020_Web_x0020_Service_policy"/>
```

```
  <wsdl:operation name="GetCollectionColumnMaximum">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionColumnMaximum" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetCollectionColumnMinimum">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionColumnMinimum" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetCollectionColumnSum">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionColumnSum" style="document"/>
```

```
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetCollectionCount">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionCount" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetGlobalCollections">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetGlobalCollections" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
```

```
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetSolutionRoot">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetSolutionRoot" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetChildPrimitiveMembers">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetChildPrimitiveMembers" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetChildComponentMembers">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetChildComponentMembers" style="document"/>
```

```
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetChildCollectionMembers">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetChildCollectionMembers" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetAllChildMembers">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetAllChildMembers" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
```

```
  <wsdl:operation name="WritePointValue">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/WritePointValue" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="InsertCollectionData">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/InsertCollectionData" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="UpdateCollectionData">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/UpdateCollectionData" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
```

```
  </wsdl:operation>
  <wsdl:operation name="ReplaceCollectionData">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/ReplaceCollectionData" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="DeleteCollectionData">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/DeleteCollectionData" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="IsSolutionAccessible">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/IsSolutionAccessible" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
```

```
  <wsdl:operation name="GetPointCurrentData">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointCurrentData" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetPointAtValue">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointAtValue" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetPointActualValue">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointActualValue" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetPointAverageValue">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointAverageValue" style="document"/>
```

```
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetPointCountValue">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointCountValue" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetPointDeltaValue">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointDeltaValue" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetPointFirstValue">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointFirstValue" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
```

```
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetPointIntegralValue">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointIntegralValue" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetPointLastValue">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointLastValue" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetPointMaximumValue">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointMaximumValue" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetPointMinimumValue">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointMinimumValue" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
```

```
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetPointSumValue">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointSumValue" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetBulkCurrentValues">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkCurrentValues" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetBulkAtValues">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkAtValues" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetBulkActualValues">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkActualValues" style="document"/>
    <wsdl:input>
```

```
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetBulkAverageValues">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkAverageValues" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetBulkCountValues">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkCountValues" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetBulkDeltaValues">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkDeltaValues" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetBulkFirstValues">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkFirstValues" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
```

```
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetBulkLastValues">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkLastValues" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetBulkIntegralValues">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkIntegralValues" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetBulkMaximumValues">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkMaximumValues" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetBulkMinimumValues">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkMinimumValues" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
```

```
  <wsdl:operation name="GetBulkSumValues">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkSumValues" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetTimeSeriesData">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetTimeSeriesData" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetCollectionData">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionData" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetCollectionDataWhere">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionDataWhere" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetCollectionDataOrderBy">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionDataOrderBy" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
```

```
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetCollectionDataFilter">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionDataFilter" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetCollectionColumnAverage">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionColumnAverage" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetCollectionColumnCount">
    <soap:operation soapAction="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionColumnCount" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
</wsdl:binding>
```

porttype XHQ_x0020_Web_x0020_Service

Imported from:	http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0
Operations:	**GetCollectionColumnMaximum** Input tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnMaximum_InputMessage Output tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnMaximum_OutputMessage **GetCollectionColumnMinimum** Input tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnMinimum_InputMessage Output tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnMinimum_OutputMessage

GetCollectionColumnSum

Input tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnSum_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnSum_OutputMessage

GetCollectionCount

Input tns:XHQ_x0020_Web_x0020_Service_GetCollectionCount_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetCollectionCount_OutputMessage

GetGlobalCollections

Input tns:XHQ_x0020_Web_x0020_Service_GetGlobalCollections_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetGlobalCollections_OutputMessage

GetSolutionRoot

Input tns:XHQ_x0020_Web_x0020_Service_GetSolutionRoot_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetSolutionRoot_OutputMessage

GetChildPrimitiveMembers

Input tns:XHQ_x0020_Web_x0020_Service_GetChildPrimitiveMembers_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetChildPrimitiveMembers_OutputMessage

GetChildComponentMembers

Input tns:XHQ_x0020_Web_x0020_Service_GetChildComponentMembers_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetChildComponentMembers_OutputMessage

GetChildCollectionMembers

Input tns:XHQ_x0020_Web_x0020_Service_GetChildCollectionMembers_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetChildCollectionMembers_OutputMessage

GetAllChildMembers

Input tns:XHQ_x0020_Web_x0020_Service_GetAllChildMembers_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetAllChildMembers_OutputMessage

WritePointValue

Input tns:XHQ_x0020_Web_x0020_Service_WritePointValue_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_WritePointValue_OutputMessage

InsertCollectionData

Input tns:XHQ_x0020_Web_x0020_Service_InsertCollectionData_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_InsertCollectionData_OutputMessage

UpdateCollectionData

Input tns:XHQ_x0020_Web_x0020_Service_UpdateCollectionData_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_UpdateCollectionData_OutputMessage

ReplaceCollectionData

Input tns:XHQ_x0020_Web_x0020_Service_ReplaceCollectionData_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_ReplaceCollectionData_OutputMessage

DeleteCollectionData

Input tns:XHQ_x0020_Web_x0020_Service_DeleteCollectionData_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_DeleteCollectionData_OutputMessage

IsSolutionAccessible

Input tns:XHQ_x0020_Web_x0020_Service_IsSolutionAccessible_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_IsSolutionAccessible_OutputMessage

GetPointCurrentData

Input tns:XHQ_x0020_Web_x0020_Service_GetPointCurrentData_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetPointCurrentData_OutputMessage

GetPointAtValue

Input tns:XHQ_x0020_Web_x0020_Service_GetPointAtValue_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetPointAtValue_OutputMessage

GetPointActualValue

Input tns:XHQ_x0020_Web_x0020_Service_GetPointActualValue_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetPointActualValue_OutputMessage

GetPointAverageValue

Input tns:XHQ_x0020_Web_x0020_Service_GetPointAverageValue_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetPointAverageValue_OutputMessage

GetPointCountValue

Input tns:XHQ_x0020_Web_x0020_Service_GetPointCountValue_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetPointCountValue_OutputMessage

GetPointDeltaValue

Input tns:XHQ_x0020_Web_x0020_Service_GetPointDeltaValue_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetPointDeltaValue_OutputMessage

GetPointFirstValue

Input tns:XHQ_x0020_Web_x0020_Service_GetPointFirstValue_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetPointFirstValue_OutputMessage

GetPointIntegralValue

Input tns:XHQ_x0020_Web_x0020_Service_GetPointIntegralValue_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetPointIntegralValue_OutputMessage

GetPointLastValue

Input tns:XHQ_x0020_Web_x0020_Service_GetPointLastValue_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetPointLastValue_OutputMessage

GetPointMaximumValue

Input tns:XHQ_x0020_Web_x0020_Service_GetPointMaximumValue_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetPointMaximumValue_OutputMessage

GetPointMinimumValue

Input tns:XHQ_x0020_Web_x0020_Service_GetPointMinimumValue_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetPointMinimumValue_OutputMessage

GetPointSumValue

Input tns:XHQ_x0020_Web_x0020_Service_GetPointSumValue_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetPointSumValue_OutputMessage

GetBulkCurrentValues

Input tns:XHQ_x0020_Web_x0020_Service_GetBulkCurrentValues_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetBulkCurrentValues_OutputMessage

GetBulkAtValues

Input tns:XHQ_x0020_Web_x0020_Service_GetBulkAtValues_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetBulkAtValues_OutputMessage

GetBulkActualValues

Input tns:XHQ_x0020_Web_x0020_Service_GetBulkActualValues_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetBulkActualValues_OutputMessage

GetBulkAverageValues

Input tns:XHQ_x0020_Web_x0020_Service_GetBulkAverageValues_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetBulkAverageValues_OutputMessage

GetBulkCountValues

Input tns:XHQ_x0020_Web_x0020_Service_GetBulkCountValues_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetBulkCountValues_OutputMessage

GetBulkDeltaValues

Input tns:XHQ_x0020_Web_x0020_Service_GetBulkDeltaValues_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetBulkDeltaValues_OutputMessage

GetBulkFirstValues

Input tns:XHQ_x0020_Web_x0020_Service_GetBulkFirstValues_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetBulkFirstValues_OutputMessage

GetBulkLastValues

Input tns:XHQ_x0020_Web_x0020_Service_GetBulkLastValues_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetBulkLastValues_OutputMessage

GetBulkIntegralValues

Input tns:XHQ_x0020_Web_x0020_Service_GetBulkIntegralValues_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetBulkIntegralValues_OutputMessage

GetBulkMaximumValues

Input tns:XHQ_x0020_Web_x0020_Service_GetBulkMaximumValues_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetBulkMaximumValues_OutputMessage

GetBulkMinimumValues

Input tns:XHQ_x0020_Web_x0020_Service_GetBulkMinimumValues_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetBulkMinimumValues_OutputMessage

GetBulkSumValues

Input tns:XHQ_x0020_Web_x0020_Service_GetBulkSumValues_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetBulkSumValues_OutputMessage

GetTimeSeriesData

Input tns:XHQ_x0020_Web_x0020_Service_GetTimeSeriesData_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetTimeSeriesData_OutputMessage

GetCollectionData

Input tns:XHQ_x0020_Web_x0020_Service_GetCollectionData_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetCollectionData_OutputMessage

GetCollectionDataWhere

Input tns:XHQ_x0020_Web_x0020_Service_GetCollectionDataWhere_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetCollectionDataWhere_OutputMessage

GetCollectionDataOrderBy

Input tns:XHQ_x0020_Web_x0020_Service_GetCollectionDataOrderBy_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetCollectionDataOrderBy_OutputMessage

GetCollectionDataFilter

Input tns:XHQ_x0020_Web_x0020_Service_GetCollectionDataFilter_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetCollectionDataFilter_OutputMessage

GetCollectionColumnAverage

Input tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnAverage_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnAverage_OutputMessage

GetCollectionColumnCount

Input tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnCount_InputMessage
Output tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnCount_OutputMessage

Source:

```
<wsdl:portType name="XHQ_x0020_Web_x0020_Service">
  <wsdl:operation name="GetCollectionColumnMaximum">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnMaximum_
InputMessage" wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionColumnMaximum"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnMaximum_
OutputMessage" wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionColumnMaximumResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetCollectionColumnMinimum">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnMinimum_
InputMessage" wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionColumnMinimum"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnMinimum_
OutputMessage" wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
```

Service/GetCollectionColumnMinimumResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetCollectionColumnSum">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnSum_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionColumnSum"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnSum_
OutputMessage" wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionColumnSumResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetCollectionCount">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionCount_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionCount"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionCount_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionCountResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetGlobalCollections">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetGlobalCollections_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetGlobalCollections"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetGlobalCollections_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetGlobalCollectionsResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetSolutionRoot">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetSolutionRoot_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetSolutionRoot"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetSolutionRoot_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_

Service/GetSolutionRootResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetChildPrimitiveMembers">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetChildPrimitiveMembers_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetChildPrimitiveMembers"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetChildPrimitiveMembers_
OutputMessage" wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetChildPrimitiveMembersResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetChildComponentMembers">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetChildComponentMembers_
InputMessage" wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetChildComponentMembers"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetChildComponentMembers_
OutputMessage" wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetChildComponentMembersResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetChildCollectionMembers">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetChildCollectionMembers_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetChildCollectionMembers"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetChildCollectionMembers_
OutputMessage" wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetChildCollectionMembersResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetAllChildMembers">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetAllChildMembers_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetAllChildMembers"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetAllChildMembers_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetAllChildMembersResponse"/>

```
  </wsdl:operation>
  <wsdl:operation name="WritePointValue">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_WritePointValue_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/WritePointValue"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_WritePointValue_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/WritePointValueResponse"/>
  </wsdl:operation>
  <wsdl:operation name="InsertCollectionData">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_InsertCollectionData_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/InsertCollectionData"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_InsertCollectionData_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/InsertCollectionDataResponse"/>
  </wsdl:operation>
  <wsdl:operation name="UpdateCollectionData">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_UpdateCollectionData_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/UpdateCollectionData"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_UpdateCollectionData_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/UpdateCollectionDataResponse"/>
  </wsdl:operation>
  <wsdl:operation name="ReplaceCollectionData">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_ReplaceCollectionData_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/ReplaceCollectionData"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_ReplaceCollectionData_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/ReplaceCollectionDataResponse"/>
  </wsdl:operation>
  <wsdl:operation name="DeleteCollectionData">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_DeleteCollectionData_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
```

Service/DeleteCollectionData"/>
<wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_DeleteCollectionData_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/DeleteCollectionDataResponse"/>
</wsdl:operation>
<wsdl:operation name="IsSolutionAccessible">
<wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_IsSolutionAccessible_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/IsSolutionAccessible"/>
<wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_IsSolutionAccessible_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/IsSolutionAccessibleResponse"/>
</wsdl:operation>
<wsdl:operation name="GetPointCurrentData">
<wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetPointCurrentData_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointCurrentData"/>
<wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetPointCurrentData_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointCurrentDataResponse"/>
</wsdl:operation>
<wsdl:operation name="GetPointAtValue">
<wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetPointAtValue_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetPointAtValue"/>
<wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetPointAtValue_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointAtValueResponse"/>
</wsdl:operation>
<wsdl:operation name="GetPointActualValue">
<wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetPointActualValue_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointActualValue"/>
<wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetPointActualValue_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointActualValueResponse"/>
</wsdl:operation>
<wsdl:operation name="GetPointAverageValue">

```
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetPointAverageValue_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointAverageValue"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetPointAverageValue_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointAverageValueResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetPointCountValue">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetPointCountValue_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointCountValue"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetPointCountValue_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointCountValueResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetPointDeltaValue">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetPointDeltaValue_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointDeltaValue"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetPointDeltaValue_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointDeltaValueResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetPointFirstValue">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetPointFirstValue_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointFirstValue"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetPointFirstValue_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointFirstValueResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetPointIntegralValue">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetPointIntegralValue_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointIntegralValue"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetPointIntegralValue_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointIntegralValueResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetPointLastValue">
```

```
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetPointLastValue_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointLastValue"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetPointLastValue_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointLastValueResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetPointMaximumValue">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetPointMaximumValue_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointMaximumValue"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetPointMaximumValue_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointMaximumValueResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetPointMinimumValue">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetPointMinimumValue_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointMinimumValue"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetPointMinimumValue_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointMinimumValueResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetPointSumValue">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetPointSumValue_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointSumValue"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetPointSumValue_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetPointSumValueResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetBulkCurrentValues">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetBulkCurrentValues_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkCurrentValues"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetBulkCurrentValues_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkCurrentValuesResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetBulkAtValues">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetBulkAtValues_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetBulkAtValues"/>
```

```
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetBulkAtValues_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkAtValuesResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetBulkActualValues">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetBulkActualValues_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkActualValues"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetBulkActualValues_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkActualValuesResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetBulkAverageValues">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetBulkAverageValues_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkAverageValues"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetBulkAverageValues_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkAverageValuesResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetBulkCountValues">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetBulkCountValues_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkCountValues"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetBulkCountValues_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkCountValuesResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetBulkDeltaValues">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetBulkDeltaValues_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkDeltaValues"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetBulkDeltaValues_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkDeltaValuesResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetBulkFirstValues">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetBulkFirstValues_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkFirstValues"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetBulkFirstValues_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkFirstValuesResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetBulkLastValues">
```

```
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetBulkLastValues_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkLastValues"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetBulkLastValues_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkLastValuesResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetBulkIntegralValues">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetBulkIntegralValues_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkIntegralValues"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetBulkIntegralValues_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkIntegralValuesResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetBulkMaximumValues">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetBulkMaximumValues_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkMaximumValues"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetBulkMaximumValues_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkMaximumValuesResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetBulkMinimumValues">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetBulkMinimumValues_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkMinimumValues"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetBulkMinimumValues_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkMinimumValuesResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetBulkSumValues">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetBulkSumValues_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkSumValues"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetBulkSumValues_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetBulkSumValuesResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetTimeSeriesData">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetTimeSeriesData_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetTimeSeriesData"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetTimeSeriesData_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetTimeSeriesDataResponse"/>
  </wsdl:operation>
```

```
  <wsdl:operation name="GetCollectionData">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionData_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_Service/GetCollectionData"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionData_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionDataResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetCollectionDataWhere">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionDataWhere_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionDataWhere"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionDataWhere_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionDataWhereResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetCollectionDataOrderBy">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionDataOrderBy_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionDataOrderBy"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionDataOrderBy_
OutputMessage" wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionDataOrderByResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetCollectionDataFilter">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionDataFilter_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionDataFilter"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionDataFilter_OutputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionDataFilterResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetCollectionColumnAverage">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnAverage_
InputMessage" wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionColumnAverage"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnAverage_
OutputMessage" wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionColumnAverageResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetCollectionColumnCount">
    <wsdl:input message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnCount_InputMessage"
wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionColumnCount"/>
    <wsdl:output message="tns:XHQ_x0020_Web_x0020_Service_GetCollectionColumnCount_
OutputMessage" wsaw:Action="http://webservices.xhq.siemens.com/XHQ_x0020_Web_x0020_
Service/GetCollectionColumnCountResponse"/>
  </wsdl:operation>
</wsdl:portType>
```

message XHQ_x0020_Web_x0020_Service_DeleteCollectionData_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:DeleteCollectionData

Used by: Operation DeleteCollectionData in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_DeleteCollectionData_InputMessage">
  <wsdl:part name="parameters" element="tns:DeleteCollectionData"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_DeleteCollectionData_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:DeleteCollectionDataResponse

Used by: Operation DeleteCollectionData in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_DeleteCollectionData_OutputMessage">
  <wsdl:part name="parameters" element="tns:DeleteCollectionDataResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetAllChildMembers_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetAllChildMembers

Used by: Operation GetAllChildMembers in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetAllChildMembers_InputMessage">
```

```
  <wsdl:part name="parameters" element="tns:GetAllChildMembers"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetAllChildMembers_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetAllChildMembersResponse

Used by: Operation GetAllChildMembers in PortType XHQ_x0020_Web_x0020_Service

Source:

```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetAllChildMembers_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetAllChildMembersResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkActualValues_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkActualValues

Used by: Operation GetBulkActualValues in PortType XHQ_x0020_Web_x0020_Service

Source:

```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkActualValues_InputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkActualValues"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkActualValues_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkActualValuesResponse

Used Operation GetBulkActualValues in PortType XHQ_x0020_Web_x0020_Service

by:

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkActualValues_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkActualValuesResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkAtValues_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkAtValues

Used by: Operation GetBulkAtValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkAtValues_InputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkAtValues"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkAtValues_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkAtValuesResponse

Used by: Operation GetBulkAtValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkAtValues_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkAtValuesResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkAverageValues_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkAverageValues

Used by: Operation GetBulkAverageValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkAverageValues_InputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkAverageValues"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkAverageValues_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkAverageValuesResponse

Used by: Operation GetBulkAverageValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkAverageValues_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkAverageValuesResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkCountValues_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkCountValues

Used by: Operation GetBulkCountValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkCountValues_InputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkCountValues"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkCountValues_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkCountValuesResponse

Used by: Operation GetBulkCountValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkCountValues_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkCountValuesResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkCurrentValues_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkCurrentValues

Used by: Operation GetBulkCurrentValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkCurrentValues_InputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkCurrentValues"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkCurrentValues_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkCurrentValuesResponse

Used by: Operation GetBulkCurrentValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkCurrentValues_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkCurrentValuesResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkDeltaValues_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkDeltaValues

Used by: Operation GetBulkDeltaValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkDeltaValues_InputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkDeltaValues"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkDeltaValues_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkDeltaValuesResponse

Used by: Operation GetBulkDeltaValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkDeltaValues_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkDeltaValuesResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkFirstValues_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkFirstValues

Used by: Operation GetBulkFirstValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkFirstValues_InputMessage">
```

```
  <wsdl:part name="parameters" element="tns:GetBulkFirstValues"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkFirstValues_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkFirstValuesResponse

Used by: Operation GetBulkFirstValues in PortType XHQ_x0020_Web_x0020_Service

Source:

```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkFirstValues_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkFirstValuesResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkIntegralValues_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkIntegralValues

Used by: Operation GetBulkIntegralValues in PortType XHQ_x0020_Web_x0020_Service

Source:

```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkIntegralValues_InputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkIntegralValues"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkIntegralValues_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkIntegralValuesResponse

Used Operation GetBulkIntegralValues in PortType XHQ_x0020_Web_x0020_Service

by:

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkIntegralValues_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkIntegralValuesResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkLastValues_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkLastValues

Used by: Operation GetBulkLastValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkLastValues_InputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkLastValues"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkLastValues_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkLastValuesResponse

Used by: Operation GetBulkLastValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkLastValues_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkLastValuesResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkMaximumValues_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkMaximumValues

Used by: Operation GetBulkMaximumValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkMaximumValues_InputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkMaximumValues"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkMaximumValues_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkMaximumValuesResponse

Used by: Operation GetBulkMaximumValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkMaximumValues_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkMaximumValuesResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkMinimumValues_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkMinimumValues

Used by: Operation GetBulkMinimumValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkMinimumValues_InputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkMinimumValues"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkMinimumValues_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkMinimumValuesResponse

Used by: Operation GetBulkMinimumValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkMinimumValues_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkMinimumValuesResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkSumValues_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkSumValues

Used by: Operation GetBulkSumValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkSumValues_InputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkSumValues"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetBulkSumValues_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetBulkSumValuesResponse

Used by: Operation GetBulkSumValues in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetBulkSumValues_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetBulkSumValuesResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetChildCollectionMembers_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetChildCollectionMembers

Used by: Operation GetChildCollectionMembers in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetChildCollectionMembers_InputMessage">
  <wsdl:part name="parameters" element="tns:GetChildCollectionMembers"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetChildCollectionMembers_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetChildCollectionMembersResponse

Used by: Operation GetChildCollectionMembers in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetChildCollectionMembers_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetChildCollectionMembersResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetChildComponentMembers_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetChildComponentMembers

Used by: Operation GetChildComponentMembers in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetChildComponentMembers_InputMessage">
```

```
  <wsdl:part name="parameters" element="tns:GetChildComponentMembers"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetChildComponentMembers_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetChildComponentMembersResponse

Used by: Operation GetChildComponentMembers in PortType XHQ_x0020_Web_x0020_Service

Source:

```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetChildComponentMembers_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetChildComponentMembersResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetChildPrimitiveMembers_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetChildPrimitiveMembers

Used by: Operation GetChildPrimitiveMembers in PortType XHQ_x0020_Web_x0020_Service

Source:

```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetChildPrimitiveMembers_InputMessage">
  <wsdl:part name="parameters" element="tns:GetChildPrimitiveMembers"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetChildPrimitiveMembers_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetChildPrimitiveMembersResponse

Used by: Operation GetChildPrimitiveMembers in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetChildPrimitiveMembers_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetChildPrimitiveMembersResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionColumnAverage_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionColumnAverage

Used by: Operation GetCollectionColumnAverage in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionColumnAverage_InputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionColumnAverage"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionColumnAverage_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionColumnAverageResponse

Used by: Operation GetCollectionColumnAverage in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionColumnAverage_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionColumnAverageResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionColumnCount_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionColumnCount

Used Operation GetCollectionColumnCount in PortType XHQ_x0020_Web_x0020_Service

by:

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionColumnCount_InputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionColumnCount"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionColumnCount_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionColumnCountResponse

Used by: Operation GetCollectionColumnCount in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionColumnCount_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionColumnCountResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionColumnMaximum_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionColumnMaximum

Used by: Operation GetCollectionColumnMaximum in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionColumnMaximum_InputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionColumnMaximum"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionColumnMaximum_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionColumnMaximumResponse

Used by: Operation GetCollectionColumnMaximum in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionColumnMaximum_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionColumnMaximumResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionColumnMinimum_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionColumnMinimum

Used by: Operation GetCollectionColumnMinimum in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionColumnMinimum_InputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionColumnMinimum"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionColumnMinimum_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionColumnMinimumResponse

Used by: Operation GetCollectionColumnMinimum in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionColumnMinimum_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionColumnMinimumResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionColumnSum_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionColumnSum

Used by: Operation GetCollectionColumnSum in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionColumnSum_InputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionColumnSum"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionColumnSum_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionColumnSumResponse

Used by: Operation GetCollectionColumnSum in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionColumnSum_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionColumnSumResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionCount_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionCount

Used by: Operation GetCollectionCount in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionCount_InputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionCount"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionCount_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionCountResponse

Used by: Operation GetCollectionCount in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionCount_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionCountResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionData_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionData

Used by: Operation GetCollectionData in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionData_InputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionData"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionData_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionDataResponse

Used by: Operation GetCollectionData in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionData_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionDataResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionDataFilter_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionDataFilter

Used by: Operation GetCollectionDataFilter in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionDataFilter_InputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionDataFilter"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionDataFilter_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionDataFilterResponse

Used by: Operation GetCollectionDataFilter in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionDataFilter_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionDataFilterResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionDataOrderBy_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionDataOrderBy

Used by: Operation GetCollectionDataOrderBy in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionDataOrderBy_InputMessage">
```

```
  <wsdl:part name="parameters" element="tns:GetCollectionDataOrderBy"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionDataOrderBy_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionDataOrderByResponse

Used by: Operation GetCollectionDataOrderBy in PortType XHQ_x0020_Web_x0020_Service

Source:

```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionDataOrderBy_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionDataOrderByResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionDataWhere_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionDataWhere

Used by: Operation GetCollectionDataWhere in PortType XHQ_x0020_Web_x0020_Service

Source:

```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionDataWhere_InputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionDataWhere"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetCollectionDataWhere_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetCollectionDataWhereResponse

Used Operation GetCollectionDataWhere in PortType XHQ_x0020_Web_x0020_Service

by:

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetCollectionDataWhere_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetCollectionDataWhereResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetGlobalCollections_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetGlobalCollections

Used by: Operation GetGlobalCollections in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetGlobalCollections_InputMessage">
  <wsdl:part name="parameters" element="tns:GetGlobalCollections"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetGlobalCollections_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetGlobalCollectionsResponse

Used by: Operation GetGlobalCollections in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetGlobalCollections_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetGlobalCollectionsResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointActualValue_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointActualValue

Used by: Operation GetPointActualValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointActualValue_InputMessage">
  <wsdl:part name="parameters" element="tns:GetPointActualValue"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointActualValue_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointActualValueResponse

Used by: Operation GetPointActualValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointActualValue_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetPointActualValueResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointAtValue_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointAtValue

Used by: Operation GetPointAtValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointAtValue_InputMessage">
  <wsdl:part name="parameters" element="tns:GetPointAtValue"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointAtValue_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointAtValueResponse

Used by: Operation GetPointAtValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointAtValue_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetPointAtValueResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointAverageValue_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointAverageValue

Used by: Operation GetPointAverageValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointAverageValue_InputMessage">
  <wsdl:part name="parameters" element="tns:GetPointAverageValue"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointAverageValue_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointAverageValueResponse

Used by: Operation GetPointAverageValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointAverageValue_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetPointAverageValueResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointCountValue_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointCountValue

Used by: Operation GetPointCountValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointCountValue_InputMessage">
  <wsdl:part name="parameters" element="tns:GetPointCountValue"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointCountValue_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointCountValueResponse

Used by: Operation GetPointCountValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointCountValue_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetPointCountValueResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointCurrentData_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointCurrentData

Used by: Operation GetPointCurrentData in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointCurrentData_InputMessage">
```

```
  <wsdl:part name="parameters" element="tns:GetPointCurrentData"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointCurrentData_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointCurrentDataResponse

Used by: Operation GetPointCurrentData in PortType XHQ_x0020_Web_x0020_Service

Source:

```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointCurrentData_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetPointCurrentDataResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointDeltaValue_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointDeltaValue

Used by: Operation GetPointDeltaValue in PortType XHQ_x0020_Web_x0020_Service

Source:

```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointDeltaValue_InputMessage">
  <wsdl:part name="parameters" element="tns:GetPointDeltaValue"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointDeltaValue_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointDeltaValueResponse

Used Operation GetPointDeltaValue in PortType XHQ_x0020_Web_x0020_Service

by:

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointDeltaValue_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetPointDeltaValueResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointFirstValue_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointFirstValue

Used by: Operation GetPointFirstValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointFirstValue_InputMessage">
  <wsdl:part name="parameters" element="tns:GetPointFirstValue"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointFirstValue_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointFirstValueResponse

Used by: Operation GetPointFirstValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointFirstValue_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetPointFirstValueResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointIntegralValue_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointIntegralValue

Used by: Operation GetPointIntegralValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointIntegralValue_InputMessage">
  <wsdl:part name="parameters" element="tns:GetPointIntegralValue"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointIntegralValue_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointIntegralValueResponse

Used by: Operation GetPointIntegralValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointIntegralValue_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetPointIntegralValueResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointLastValue_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointLastValue

Used by: Operation GetPointLastValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointLastValue_InputMessage">
  <wsdl:part name="parameters" element="tns:GetPointLastValue"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointLastValue_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointLastValueResponse

Used by: Operation GetPointLastValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointLastValue_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetPointLastValueResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointMaximumValue_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointMaximumValue

Used by: Operation GetPointMaximumValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointMaximumValue_InputMessage">
  <wsdl:part name="parameters" element="tns:GetPointMaximumValue"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointMaximumValue_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointMaximumValueResponse

Used by: Operation GetPointMaximumValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointMaximumValue_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetPointMaximumValueResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointMinimumValue_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointMinimumValue

Used by: Operation GetPointMinimumValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointMinimumValue_InputMessage">
  <wsdl:part name="parameters" element="tns:GetPointMinimumValue"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointMinimumValue_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointMinimumValueResponse

Used by: Operation GetPointMinimumValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointMinimumValue_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetPointMinimumValueResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointSumValue_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointSumValue

Used by: Operation GetPointSumValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointSumValue_InputMessage">
```

```
  <wsdl:part name="parameters" element="tns:GetPointSumValue"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetPointSumValue_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetPointSumValueResponse

Used by: Operation GetPointSumValue in PortType XHQ_x0020_Web_x0020_Service

Source:

```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetPointSumValue_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetPointSumValueResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetSolutionRoot_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetSolutionRoot

Used by: Operation GetSolutionRoot in PortType XHQ_x0020_Web_x0020_Service

Source:

```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetSolutionRoot_InputMessage">
  <wsdl:part name="parameters" element="tns:GetSolutionRoot"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetSolutionRoot_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetSolutionRootResponse

Used Operation GetSolutionRoot in PortType XHQ_x0020_Web_x0020_Service

by:

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetSolutionRoot_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetSolutionRootResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetTimeSeriesData_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetTimeSeriesData

Used by: Operation GetTimeSeriesData in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetTimeSeriesData_InputMessage">
  <wsdl:part name="parameters" element="tns:GetTimeSeriesData"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_GetTimeSeriesData_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:GetTimeSeriesDataResponse

Used by: Operation GetTimeSeriesData in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_GetTimeSeriesData_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetTimeSeriesDataResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_InsertCollectionData_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:InsertCollectionData

Used by: Operation InsertCollectionData in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_InsertCollectionData_InputMessage">
  <wsdl:part name="parameters" element="tns:InsertCollectionData"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_InsertCollectionData_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:InsertCollectionDataResponse

Used by: Operation InsertCollectionData in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_InsertCollectionData_OutputMessage">
  <wsdl:part name="parameters" element="tns:InsertCollectionDataResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_IsSolutionAccessible_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:IsSolutionAccessible

Used by: Operation IsSolutionAccessible in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_IsSolutionAccessible_InputMessage">
  <wsdl:part name="parameters" element="tns:IsSolutionAccessible"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_IsSolutionAccessible_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:IsSolutionAccessibleResponse

Used by: Operation IsSolutionAccessible in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_IsSolutionAccessible_OutputMessage">
  <wsdl:part name="parameters" element="tns:IsSolutionAccessibleResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_ReplaceCollectionData_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:ReplaceCollectionData

Used by: Operation ReplaceCollectionData in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_ReplaceCollectionData_InputMessage">
  <wsdl:part name="parameters" element="tns:ReplaceCollectionData"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_ReplaceCollectionData_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:ReplaceCollectionDataResponse

Used by: Operation ReplaceCollectionData in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_ReplaceCollectionData_OutputMessage">
  <wsdl:part name="parameters" element="tns:ReplaceCollectionDataResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_UpdateCollectionData_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:UpdateCollectionData

Used by: Operation UpdateCollectionData in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_UpdateCollectionData_InputMessage">
  <wsdl:part name="parameters" element="tns:UpdateCollectionData"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_UpdateCollectionData_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:UpdateCollectionDataResponse

Used by: Operation UpdateCollectionData in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_UpdateCollectionData_OutputMessage">
  <wsdl:part name="parameters" element="tns:UpdateCollectionDataResponse"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_WritePointValue_InputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:WritePointValue

Used by: Operation WritePointValue in PortType XHQ_x0020_Web_x0020_Service

Source:
```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_WritePointValue_InputMessage">
```

```
  <wsdl:part name="parameters" element="tns:WritePointValue"/>
</wsdl:message>
```

message XHQ_x0020_Web_x0020_Service_WritePointValue_OutputMessage

Imported from: http://grinds.corp.indxhq.com/indx/xhqwse/XhqWse.svc?wsdl=wsdl0

Parts: parameters
element tns:WritePointValueResponse

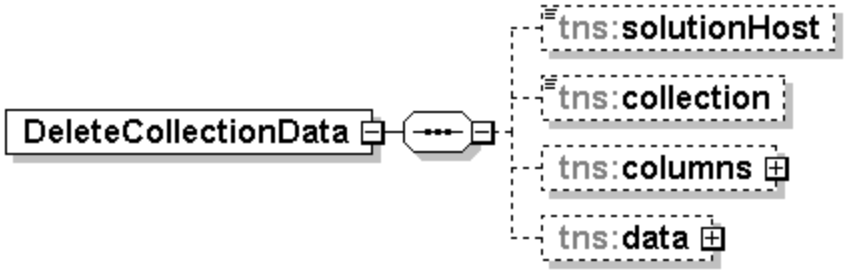
Used by: Operation WritePointValue in PortType XHQ_x0020_Web_x0020_Service

Source:

```
<wsdl:message name="XHQ_x0020_Web_x0020_Service_WritePointValue_OutputMessage">
  <wsdl:part name="parameters" element="tns:WritePointValueResponse"/>
</wsdl:message>
```

element DeleteCollectionData

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:collection tns:columns tns:data

Source:

```
<xsd:element name="DeleteCollectionData">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="collection" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="columns" type="q15:ArrayOfstring" nillable="true" minOccurs="0"/>
      <xsd:element name="data" type="q16:ArrayOfArrayOfstring" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element DeleteCollectionDataResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

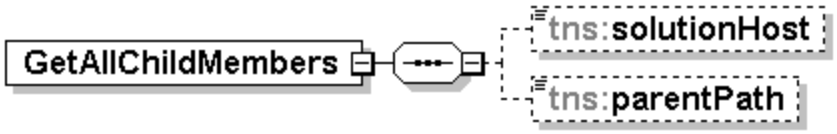
Children: tns:DeleteCollectionDataResult

Source:

```
<xsd:element name="DeleteCollectionDataResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="DeleteCollectionDataResult" type="xs:boolean" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetAllChildMembers

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:parentPath

Source:

```
<xsd:element name="GetAllChildMembers">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="parentPath" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetAllChildMembersResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

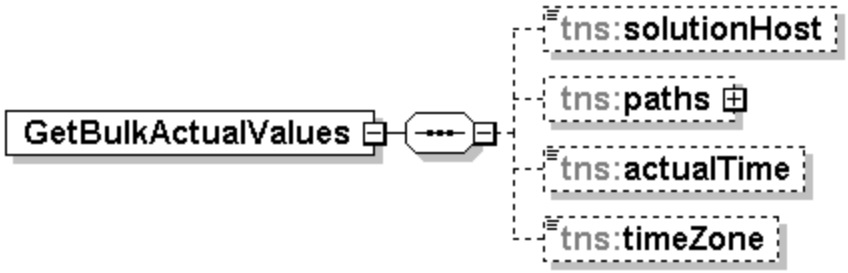
Children: tns:GetAllChildMembersResult

Source:

```
<xsd:element name="GetAllChildMembersResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetAllChildMembersResult" type="q8:ArrayOfstring" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkActualValues

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:paths tns:actualTime tns:timeZone

Source:

```
<xsd:element name="GetBulkActualValues">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="paths" type="q33:ArrayOfstring" nillable="true" minOccurs="0"/>
      <xsd:element name="actualTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkActualValuesResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:GetBulkActualValuesResult

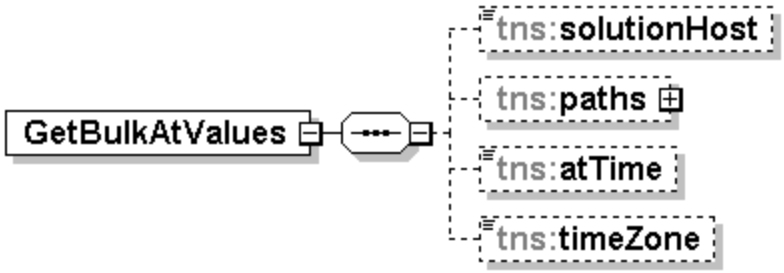
Source:

```
<xsd:element name="GetBulkActualValuesResponse">
  <xsd:complexType>
    <xsd:sequence>
```

```
    <xsd:element name="GetBulkActualValuesResult" type="q34:ArrayOfVTQ" nillable="true"
minOccurs="0"/>
   </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkAtValues

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:paths tns:atTime tns:timeZone

Source:

```
<xsd:element name="GetBulkAtValues">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="paths" type="q31:ArrayOfstring" nillable="true" minOccurs="0"/>
      <xsd:element name="atTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkAtValuesResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:GetBulkAtValuesResult

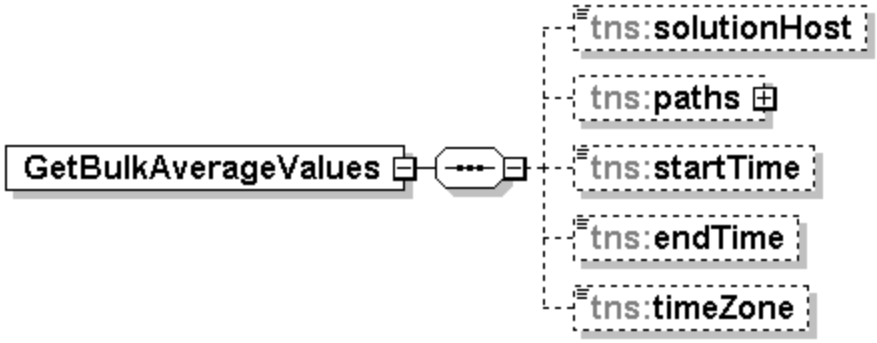
Source:

```
<xsd:element name="GetBulkAtValuesResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetBulkAtValuesResult" type="q32:ArrayOfVTQ" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
```

```
  </xsd:complexType>
</xsd:element>
```

element GetBulkAverageValues

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:paths tns:startTime tns:endTime tns:timeZone

Source:

```
<xsd:element name="GetBulkAverageValues">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="paths" type="q35:ArrayOfstring" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkAverageValuesResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:GetBulkAverageValuesResult

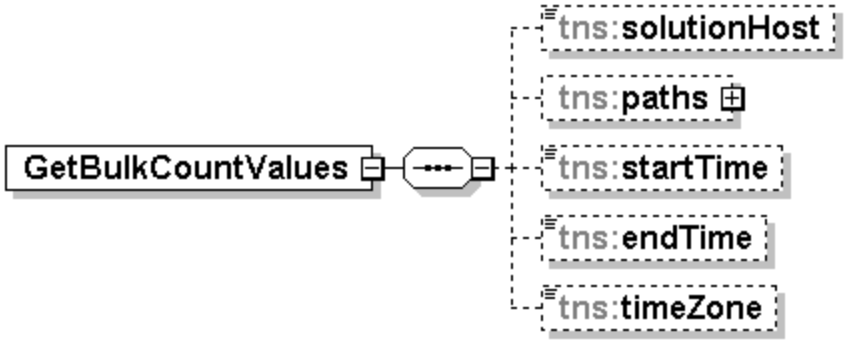
Source:

```
<xsd:element name="GetBulkAverageValuesResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetBulkAverageValuesResult" type="q36:ArrayOfVTQ" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
```

```
  </xsd:complexType>
</xsd:element>
```

element GetBulkCountValues

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:paths tns:startTime tns:endTime tns:timeZone

Source:

```
<xsd:element name="GetBulkCountValues">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="paths" type="q37:ArrayOfstring" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkCountValuesResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:GetBulkCountValuesResult

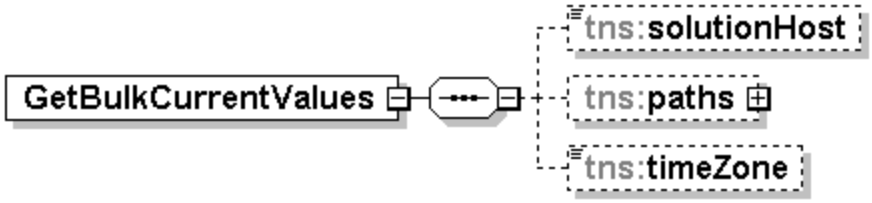
Source:

```
<xsd:element name="GetBulkCountValuesResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetBulkCountValuesResult" type="q38:ArrayOfVTQ" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
```

```
  </xsd:complexType>
</xsd:element>
```

element GetBulkCurrentValues

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:paths tns:timeZone

Source:

```
<xsd:element name="GetBulkCurrentValues">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="paths" type="q29:ArrayOfstring" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkCurrentValuesResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

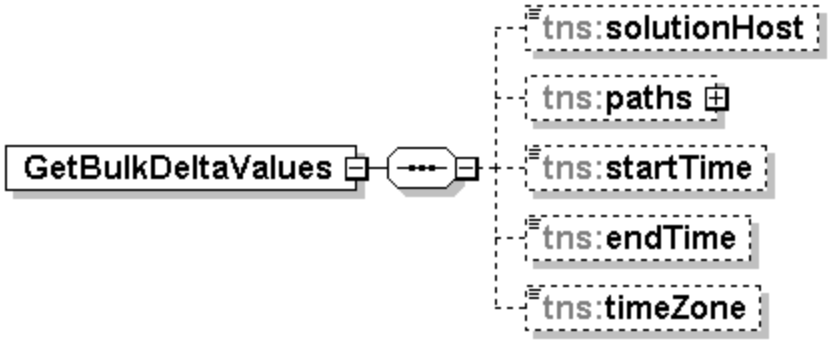
Children: tns:GetBulkCurrentValuesResult

Source:

```
<xsd:element name="GetBulkCurrentValuesResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetBulkCurrentValuesResult" type="q30:ArrayOfVTQ" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkDeltaValues

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:paths tns:startTime tns:endTime tns:timeZone

Source:

```
<xsd:element name="GetBulkDeltaValues">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="paths" type="q39:ArrayOfstring" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkDeltaValuesResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

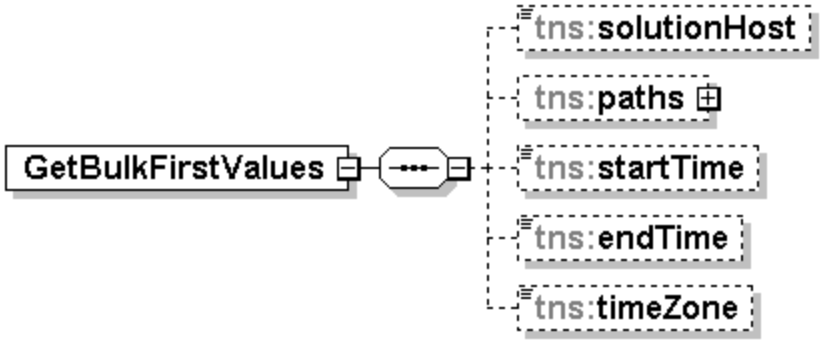
Children: tns:GetBulkDeltaValuesResult

Source:

```
<xsd:element name="GetBulkDeltaValuesResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetBulkDeltaValuesResult" type="q40:ArrayOfVTQ" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkFirstValues

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:paths tns:startTime tns:endTime tns:timeZone

Source:

```
<xsd:element name="GetBulkFirstValues">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="paths" type="q41:ArrayOfstring" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkFirstValuesResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

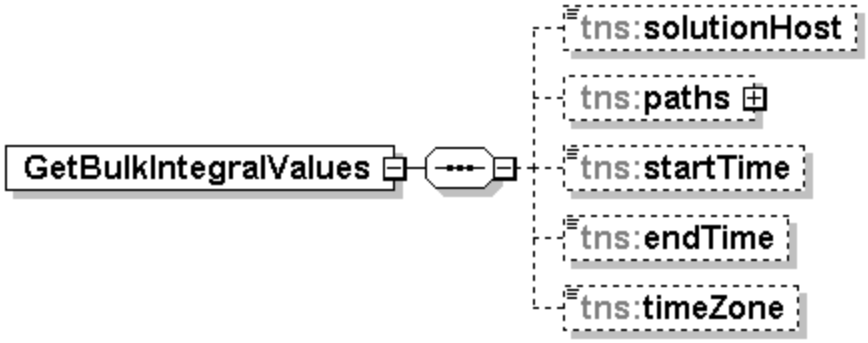
Children: tns:GetBulkFirstValuesResult

Source:

```
<xsd:element name="GetBulkFirstValuesResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetBulkFirstValuesResult" type="q42:ArrayOfVTQ" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkIntegralValues

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:paths tns:startTime tns:endTime tns:timeZone

Source:

```
<xsd:element name="GetBulkIntegralValues">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="paths" type="q45:ArrayOfstring" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkIntegralValuesResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

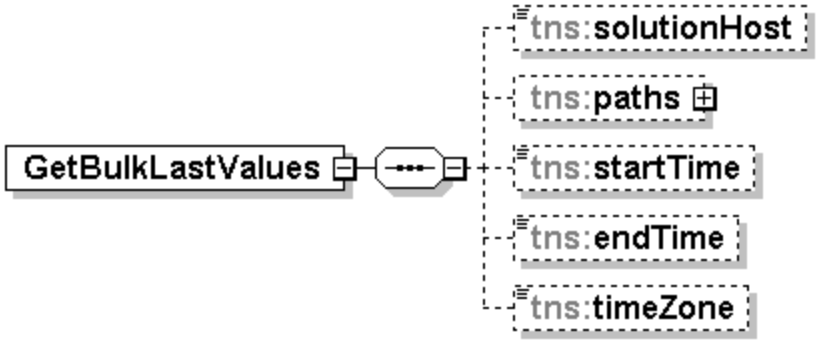
Children: tns:GetBulkIntegralValuesResult

Source:

```
<xsd:element name="GetBulkIntegralValuesResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetBulkIntegralValuesResult" type="q46:ArrayOfVTQ" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkLastValues

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:paths tns:startTime tns:endTime tns:timeZone

Source:

```
<xsd:element name="GetBulkLastValues">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="paths" type="q43:ArrayOfstring" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkLastValuesResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

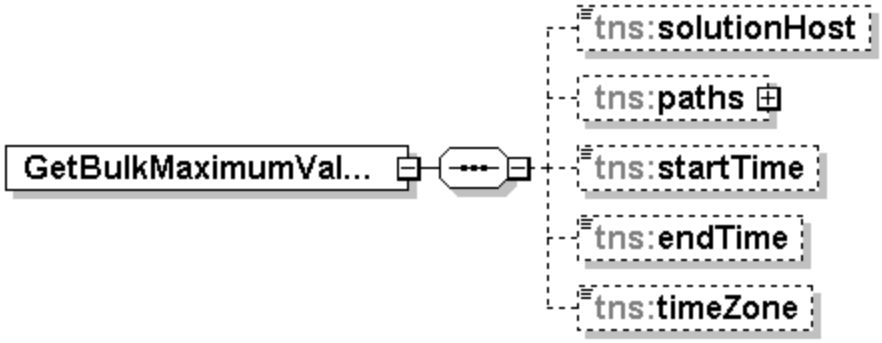
Children: tns:GetBulkLastValuesResult

Source:

```
<xsd:element name="GetBulkLastValuesResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetBulkLastValuesResult" type="q44:ArrayOfVTQ" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkMaximumValues

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:paths tns:startTime tns:endTime tns:timeZone

Source:

```
<xsd:element name="GetBulkMaximumValues">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="paths" type="q47:ArrayOfstring" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkMaximumValuesResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

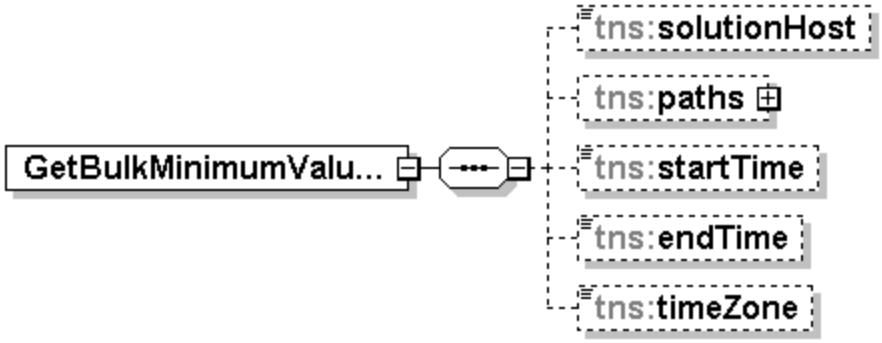
Children: tns:GetBulkMaximumValuesResult

Source:

```
<xsd:element name="GetBulkMaximumValuesResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetBulkMaximumValuesResult" type="q48:ArrayOfVTQ" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkMinimumValues

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:paths tns:startTime tns:endTime tns:timeZone

Source:

```
<xsd:element name="GetBulkMinimumValues">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="paths" type="q49:ArrayOfstring" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkMinimumValuesResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

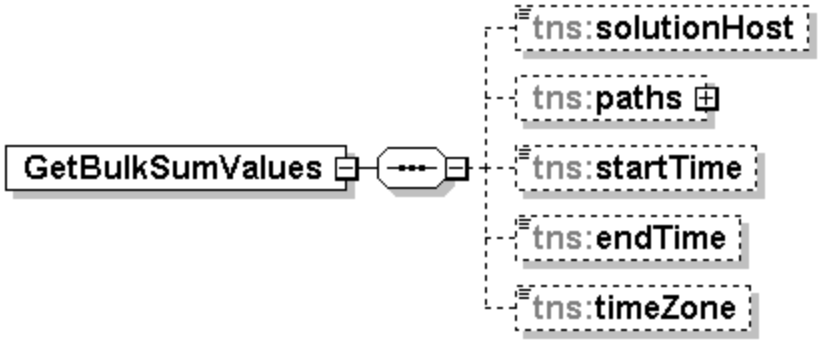
Children: tns:GetBulkMinimumValuesResult

Source:

```
<xsd:element name="GetBulkMinimumValuesResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetBulkMinimumValuesResult" type="q50:ArrayOfVTQ" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkSumValues

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:paths tns:startTime tns:endTime tns:timeZone

Source:

```
<xsd:element name="GetBulkSumValues">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="paths" type="q51:ArrayOfstring" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetBulkSumValuesResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

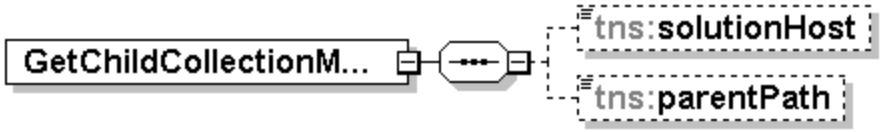
Children: tns:GetBulkSumValuesResult

Source:

```
<xsd:element name="GetBulkSumValuesResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetBulkSumValuesResult" type="q52:ArrayOfVTQ" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetChildCollectionMembers

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:parentPath

Source:

```
<xsd:element name="GetChildCollectionMembers">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="parentPath" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetChildCollectionMembersResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

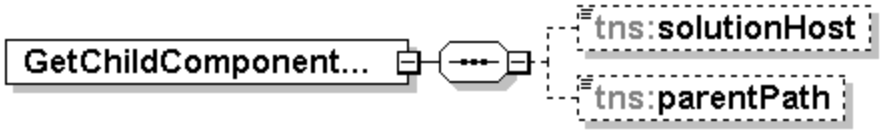
Children: tns:GetChildCollectionMembersResult

Source:

```
<xsd:element name="GetChildCollectionMembersResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetChildCollectionMembersResult" type="q7:ArrayOfstring" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetChildComponentMembers

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:parentPath

Source:
```
<xsd:element name="GetChildComponentMembers">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="parentPath" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetChildComponentMembersResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

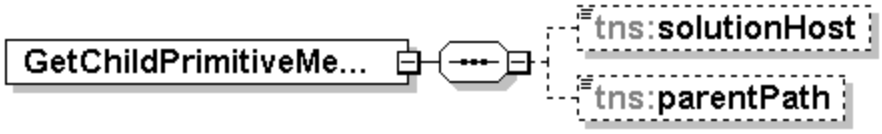
Properties: content complex

Children: tns:GetChildComponentMembersResult

Source:
```
<xsd:element name="GetChildComponentMembersResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetChildComponentMembersResult" type="q6:ArrayOfstring" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetChildPrimitiveMembers

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:parentPath

Source:
```
<xsd:element name="GetChildPrimitiveMembers">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="parentPath" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
```

```
</xsd:element>
```

element GetChildPrimitiveMembersResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

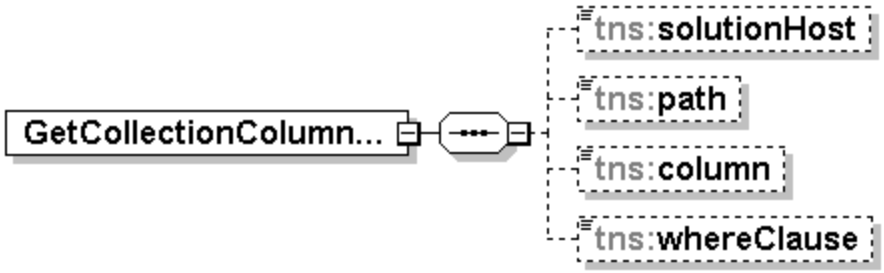
Children: tns:GetChildPrimitiveMembersResult

Source:

```
<xsd:element name="GetChildPrimitiveMembersResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetChildPrimitiveMembersResult" type="q5:ArrayOfstring" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetCollectionColumnAverage

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:column tns:whereClause

Source:

```
<xsd:element name="GetCollectionColumnAverage">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="column" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="whereClause" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetCollectionColumnAverageResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

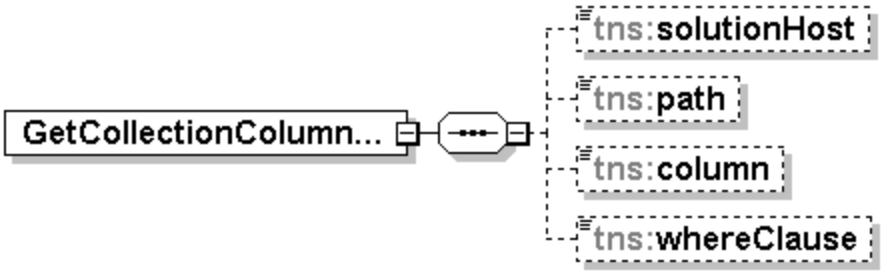
Children: tns:GetCollectionColumnAverageResult

Source:

```
<xsd:element name="GetCollectionColumnAverageResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetCollectionColumnAverageResult" type="q58:ArrayOfstring" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetCollectionColumnCount

Diagram:



namespace http://webservices.xhq.siemens.com

properties content complex

children tns:solutionHost tns:path tns:column tns:whereClause

source

```
<xsd:element name="GetCollectionColumnCount">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="column" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="whereClause" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetCollectionColumnCountResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

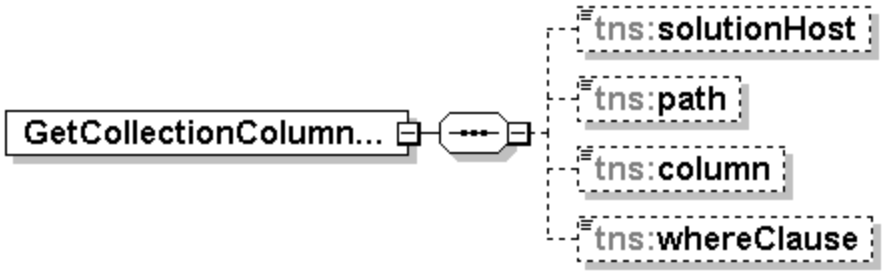
Children: tns:GetCollectionColumnCountResult

Source:

```
<xsd:element name="GetCollectionColumnCountResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetCollectionColumnCountResult" type="q59:ArrayOfstring" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetCollectionColumnMaximum

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:column tns:whereClause

Source:

```
<xsd:element name="GetCollectionColumnMaximum">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="column" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="whereClause" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetCollectionColumnMaximumResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

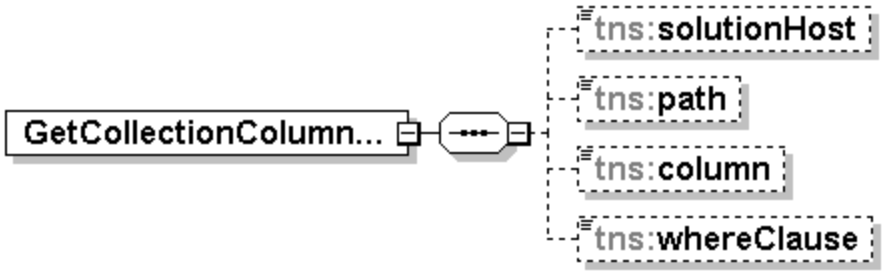
Children: tns:GetCollectionColumnMaximumResult

Source:

```
<xsd:element name="GetCollectionColumnMaximumResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetCollectionColumnMaximumResult" type="q1:ArrayOfstring"
nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetCollectionColumnMinimum

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:column tns:whereClause

Source:

```
<xsd:element name="GetCollectionColumnMinimum">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="column" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="whereClause" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetCollectionColumnMinimumResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

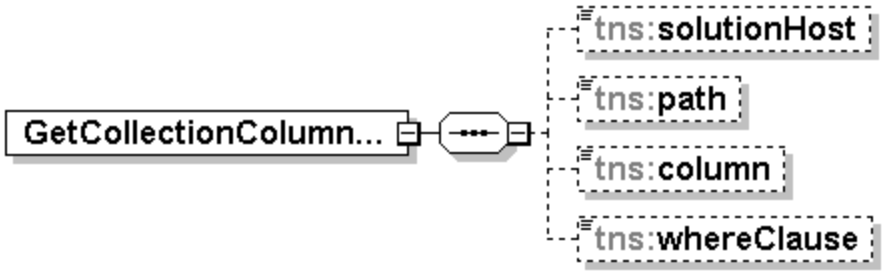
Children: tns:GetCollectionColumnMinimumResult

Source:

```
<xsd:element name="GetCollectionColumnMinimumResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetCollectionColumnMinimumResult" type="q2:ArrayOfstring"
nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetCollectionColumnSum

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:column tns:whereClause

Source:

```
<xsd:element name="GetCollectionColumnSum">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="column" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="whereClause" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetCollectionColumnSumResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

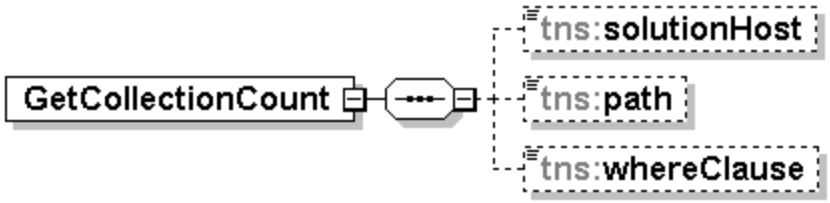
Children: tns:GetCollectionColumnSumResult

Source:

```
<xsd:element name="GetCollectionColumnSumResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetCollectionColumnSumResult" type="q3:ArrayOfstring" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetCollectionCount

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:whereClause

Source:

```
<xsd:element name="GetCollectionCount">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="whereClause" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetCollectionCountResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

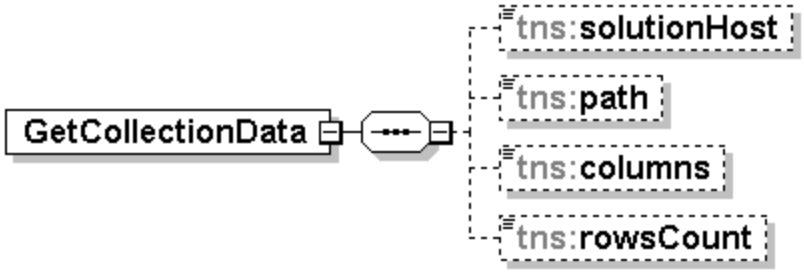
Properties: content complex

Children: tns:GetCollectionCountResult

Source:
```
<xsd:element name="GetCollectionCountResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetCollectionCountResult" type="xs:int" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetCollectionData

Diagram:



Namespace: http://webservices.xhq.siemens.com

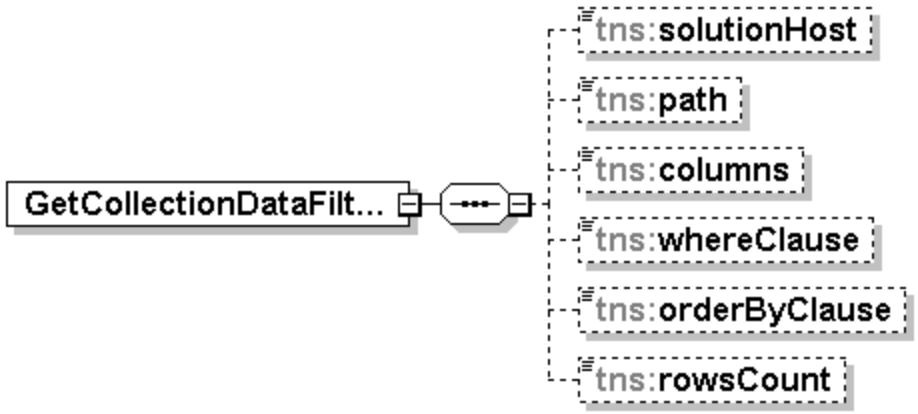
Properties: content complex

Children: tns:solutionHost tns:path tns:columns tns:rowsCount

Source:
```
<xsd:element name="GetCollectionData">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="columns" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="rowsCount" type="xs:int" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetCollectionDataFilter

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:columns tns:whereClause tns:orderByClause tns:rowsCount

Source:

```
<xsd:element name="GetCollectionDataFilter">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="columns" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="whereClause" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="orderByClause" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="rowsCount" type="xs:int" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetCollectionDataFilterResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:GetCollectionDataFilterResult

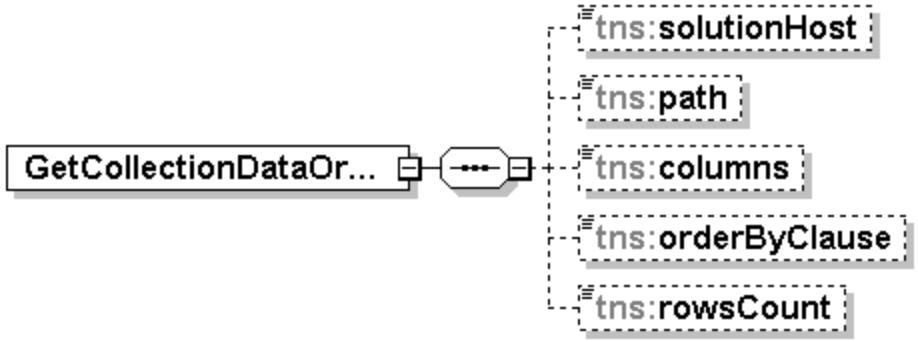
Source:

```
<xsd:element name="GetCollectionDataFilterResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetCollectionDataFilterResult" type="q57:ArrayOfArrayOfstring"
nillable="true" minOccurs="0"/>
    </xsd:sequence>
```

```
  </xsd:complexType>
</xsd:element>
```

element GetCollectionDataOrderBy

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:columns tns:orderByClause tns:rowsCount

Source:

```
<xsd:element name="GetCollectionDataOrderBy">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="columns" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="orderByClause" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="rowsCount" type="xs:int" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetCollectionDataOrderByResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:GetCollectionDataOrderByResult

Source:

```
<xsd:element name="GetCollectionDataOrderByResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetCollectionDataOrderByResult" type="q56:ArrayOfArrayOfstring"
nillable="true" minOccurs="0"/>
    </xsd:sequence>
```

```
  </xsd:complexType>
</xsd:element>
```

element GetCollectionDataResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

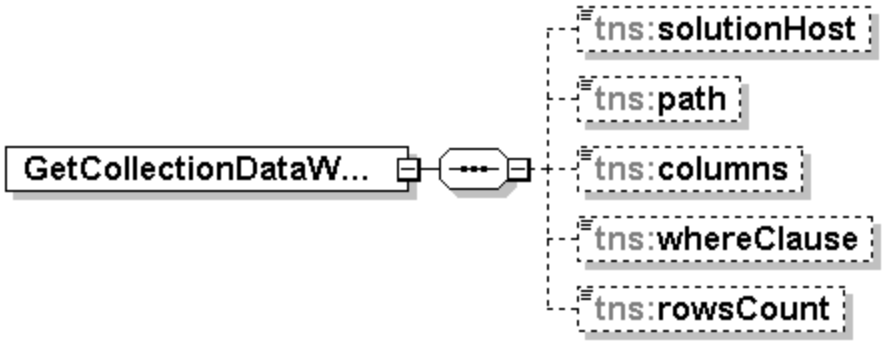
Children: tns:GetCollectionDataResult

Source:

```
<xsd:element name="GetCollectionDataResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetCollectionDataResult" type="q54:ArrayOfArrayOfstring" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetCollectionDataWhere

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:columns tns:whereClause tns:rowsCount

Source:

```
<xsd:element name="GetCollectionDataWhere">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="columns" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="whereClause" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="rowsCount" type="xs:int" minOccurs="0"/>
    </xsd:sequence>
```

```
  </xsd:complexType>
</xsd:element>
```

element GetCollectionDataWhereResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

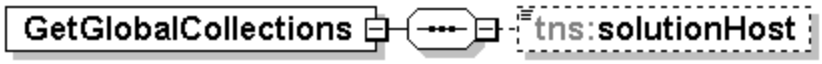
Children: tns:GetCollectionDataWhereResult

Source:

```
<xsd:element name="GetCollectionDataWhereResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetCollectionDataWhereResult" type="q55:ArrayOfArrayOfstring"
nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetGlobalCollections

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost

Source:

```
<xsd:element name="GetGlobalCollections">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetGlobalCollectionsResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

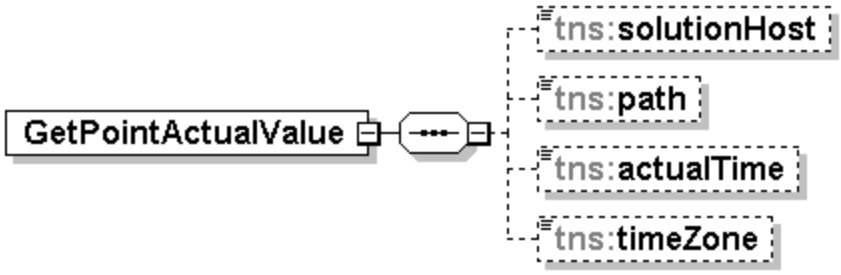
Properties: content complex

Children: tns:GetGlobalCollectionsResult

Source:
```
<xsd:element name="GetGlobalCollectionsResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetGlobalCollectionsResult" type="q4:ArrayOfstring" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointActualValue

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:actualTime tns:timeZone

Source:
```
<xsd:element name="GetPointActualValue">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="actualTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointActualValueResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

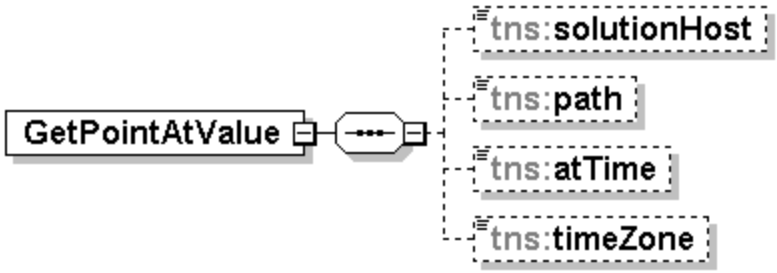
Children: tns:GetPointActualValueResult

Source:
```
<xsd:element name="GetPointActualValueResponse">
  <xsd:complexType>
```

```
    <xsd:sequence>
      <xsd:element name="GetPointActualValueResult" type="q19:VTQ" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointAtValue

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:atTime tns:timeZone

Source:

```
<xsd:element name="GetPointAtValue">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="atTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointAtValueResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:GetPointAtValueResult

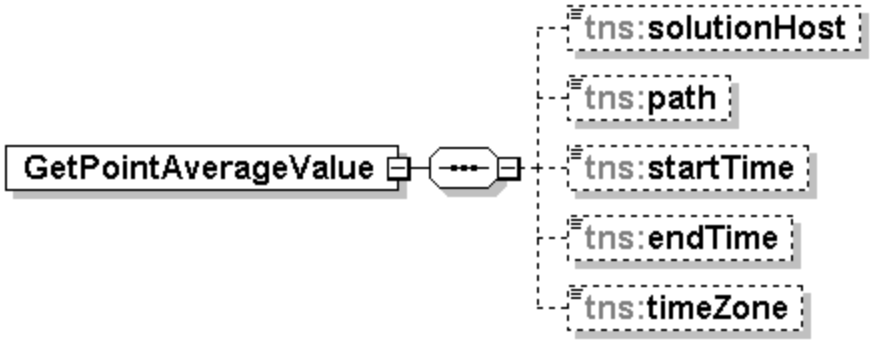
Source:

```
<xsd:element name="GetPointAtValueResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetPointAtValueResult" type="q18:VTQ" nillable="true" minOccurs="0"/>
    </xsd:sequence>
```

```
  </xsd:complexType>
</xsd:element>
```

element GetPointAverageValue

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:startTime tns:endTime tns:timeZone

Source:

```
<xsd:element name="GetPointAverageValue">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointAverageValueResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:GetPointAverageValueResult

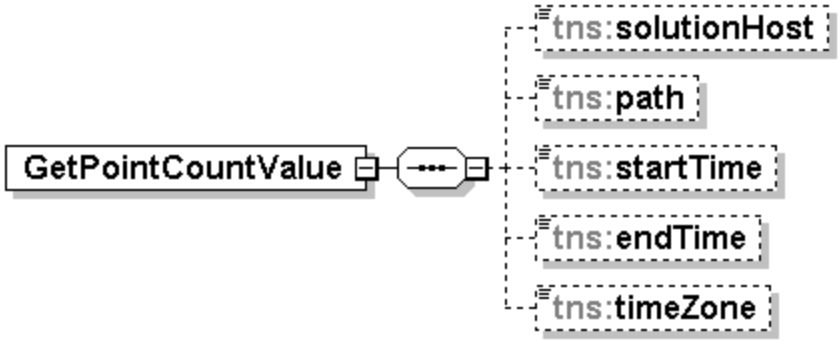
Source:

```
<xsd:element name="GetPointAverageValueResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetPointAverageValueResult" type="q20:VTQ" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
```

```
  </xsd:complexType>
</xsd:element>
```

element GetPointCountValue

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:startTime tns:endTime tns:timeZone

Source:

```
<xsd:element name="GetPointCountValue">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointCountValueResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:GetPointCountValueResult

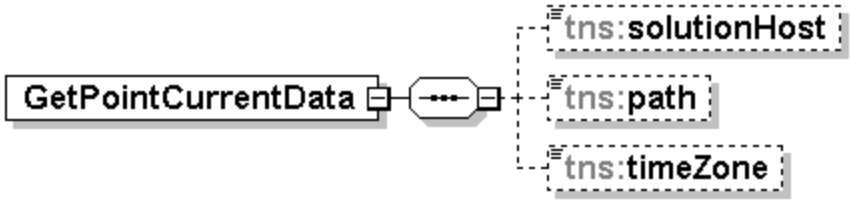
Source:

```
<xsd:element name="GetPointCountValueResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetPointCountValueResult" type="q21:VTQ" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
```

```
</xsd:element>
```

element GetPointCurrentData

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:timeZone

Source:

```
<xsd:element name="GetPointCurrentData">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointCurrentDataResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

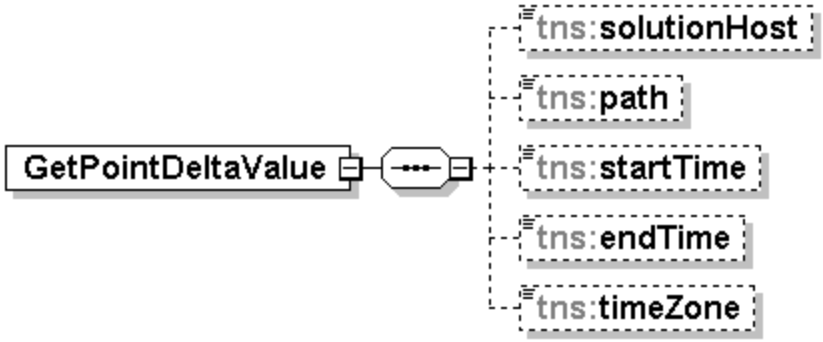
Children: tns:GetPointCurrentDataResult

Source:

```
<xsd:element name="GetPointCurrentDataResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetPointCurrentDataResult" type="q17:VTQ" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointDeltaValue

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:startTime tns:endTime tns:timeZone

Source:

```
<xsd:element name="GetPointDeltaValue">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointDeltaValueResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

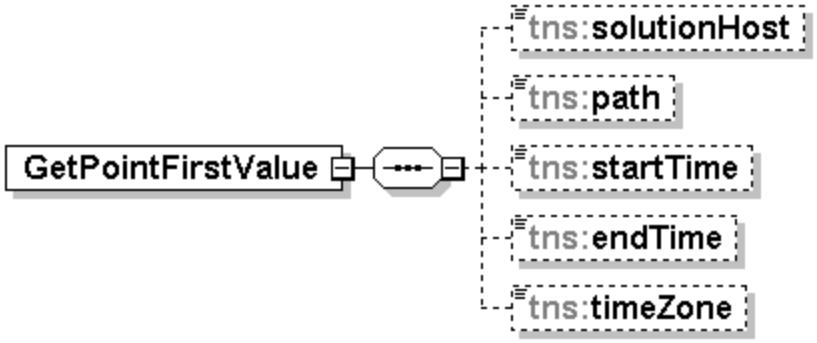
Children: tns:GetPointDeltaValueResult

Source:

```
<xsd:element name="GetPointDeltaValueResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetPointDeltaValueResult" type="q22:VTQ" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointFirstValue

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:startTime tns:endTime tns:timeZone

Source:

```
<xsd:element name="GetPointFirstValue">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointFirstValueResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

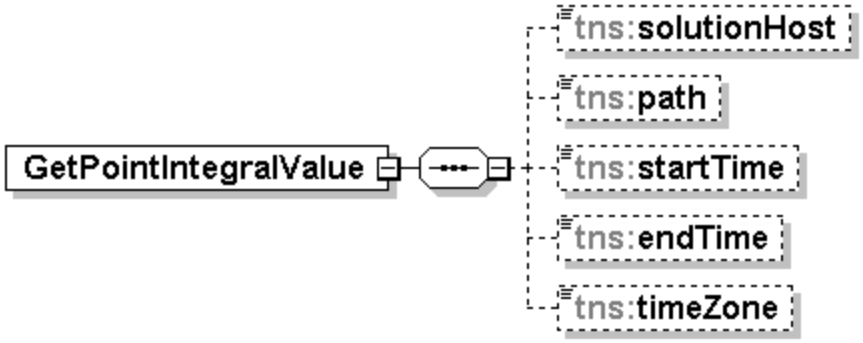
Children: tns:GetPointFirstValueResult

Source:

```
<xsd:element name="GetPointFirstValueResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetPointFirstValueResult" type="q23:VTQ" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointIntegralValue

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:startTime tns:endTime tns:timeZone

Source:

```
<xsd:element name="GetPointIntegralValue">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointIntegralValueResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

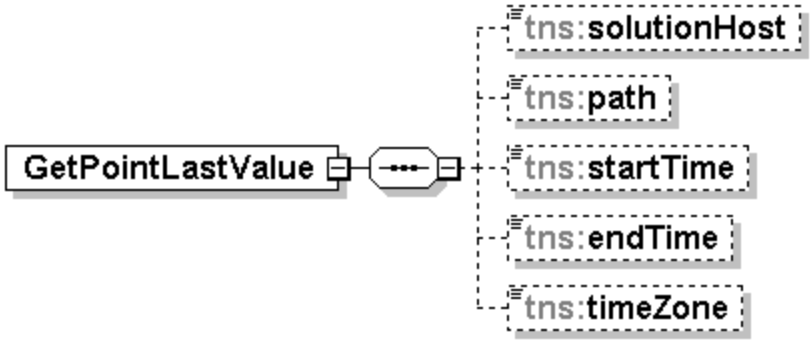
Children: tns:GetPointIntegralValueResult

Source:

```
<xsd:element name="GetPointIntegralValueResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetPointIntegralValueResult" type="q24:VTQ" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointLastValue

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:startTime tns:endTime tns:timeZone

Source:

```
<xsd:element name="GetPointLastValue">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointLastValueResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

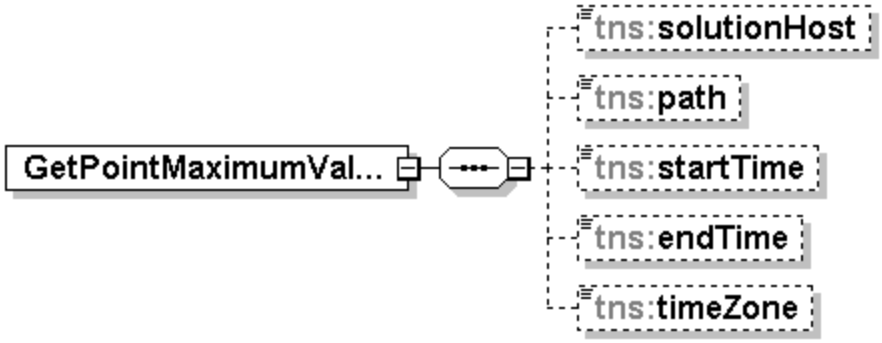
Children: tns:GetPointLastValueResult

Source:

```
<xsd:element name="GetPointLastValueResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetPointLastValueResult" type="q25:VTQ" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointMaximumValue

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:startTime tns:endTime tns:timeZone

Source:

```
<xsd:element name="GetPointMaximumValue">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointMaximumValueResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

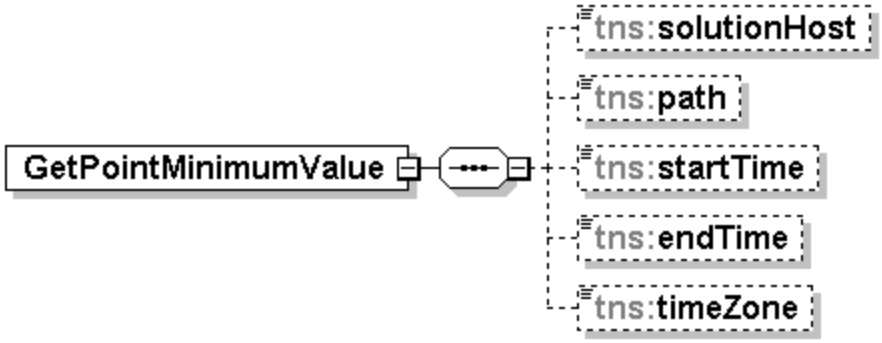
Children: tns:GetPointMaximumValueResult

Source:

```
<xsd:element name="GetPointMaximumValueResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetPointMaximumValueResult" type="q26:VTQ" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointMinimumValue

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:startTime tns:endTime tns:timeZone

Source:

```
<xsd:element name="GetPointMinimumValue">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointMinimumValueResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

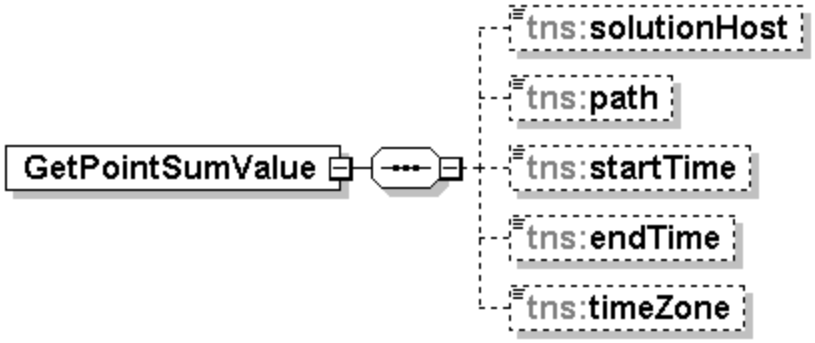
Children: tns:GetPointMinimumValueResult

Source:

```
<xsd:element name="GetPointMinimumValueResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetPointMinimumValueResult" type="q27:VTQ" nillable="true"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointSumValue

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:startTime tns:endTime tns:timeZone

Source:

```
<xsd:element name="GetPointSumValue">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetPointSumValueResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:GetPointSumValueResult

Source:

```
<xsd:element name="GetPointSumValueResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetPointSumValueResult" type="q28:VTQ" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetSolutionRoot

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost

Source:

```
<xsd:element name="GetSolutionRoot">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetSolutionRootResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

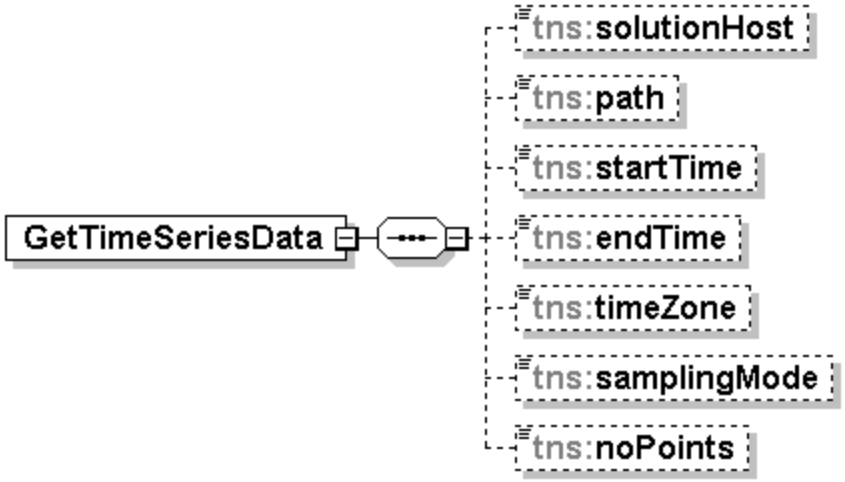
Children: tns:GetSolutionRootResult

Source:

```
<xsd:element name="GetSolutionRootResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GetSolutionRootResult" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetTimeSeriesData

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:startTime tns:endTime tns:timeZone tns:samplingMode tns:noPoints

Source:

```
<xsd:element name="GetTimeSeriesData">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="startTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="endTime" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="timeZone" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="samplingMode" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="noPoints" type="xs:int" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element GetTimeSeriesDataResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:GetTimeSeriesDataResult

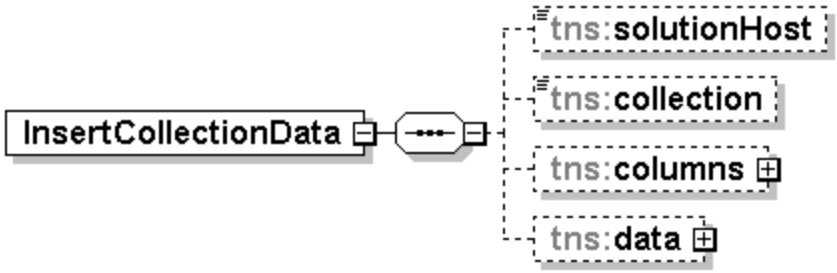
Source:

```
<xsd:element name="GetTimeSeriesDataResponse">
  <xsd:complexType>
    <xsd:sequence>
```

```
    <xsd:element name="GetTimeSeriesDataResult" type="q53:ArrayOfVTQ" nillable="true"
minOccurs="0"/>
   </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element InsertCollectionData

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:collection tns:columns tns:data

Source:

```
<xsd:element name="InsertCollectionData">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="collection" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="columns" type="q9:ArrayOfstring" nillable="true" minOccurs="0"/>
      <xsd:element name="data" type="q10:ArrayOfArrayOfstring" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element InsertCollectionDataResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:InsertCollectionDataResult

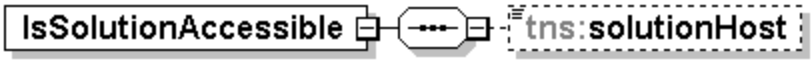
Source:

```
<xsd:element name="InsertCollectionDataResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="InsertCollectionDataResult" type="xs:boolean" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
```

```
</xsd:element>
```

element IsSolutionAccessible

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost

Source:

```
<xsd:element name="IsSolutionAccessible">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element IsSolutionAccessibleResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

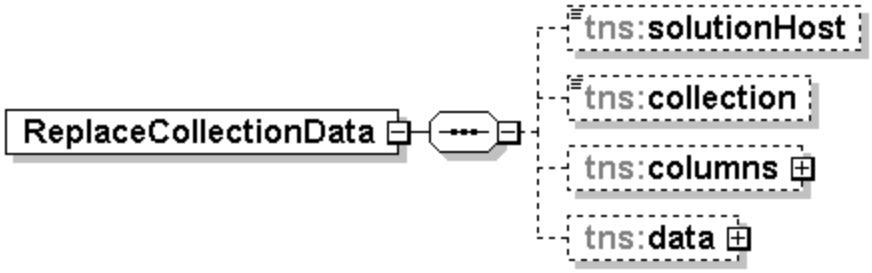
Children: tns:IsSolutionAccessibleResult

Source:

```
<xsd:element name="IsSolutionAccessibleResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="IsSolutionAccessibleResult" type="xs:boolean" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element ReplaceCollectionData

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:collection tns:columns tns:data

Source:

```
<xsd:element name="ReplaceCollectionData">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="collection" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="columns" type="q13:ArrayOfstring" nillable="true" minOccurs="0"/>
      <xsd:element name="data" type="q14:ArrayOfArrayOfstring" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element ReplaceCollectionDataResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

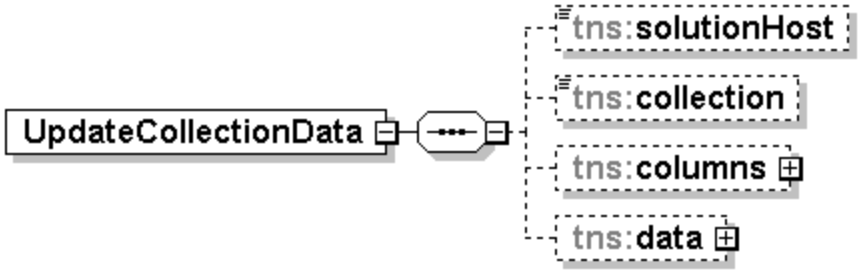
Children: tns:ReplaceCollectionDataResult

Source:

```
<xsd:element name="ReplaceCollectionDataResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="ReplaceCollectionDataResult" type="xs:boolean" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element UpdateCollectionData

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:collection tns:columns tns:data

Source:

```
<xsd:element name="UpdateCollectionData">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="collection" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="columns" type="q11:ArrayOfstring" nillable="true" minOccurs="0"/>
      <xsd:element name="data" type="q12:ArrayOfArrayOfstring" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element UpdateCollectionDataResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

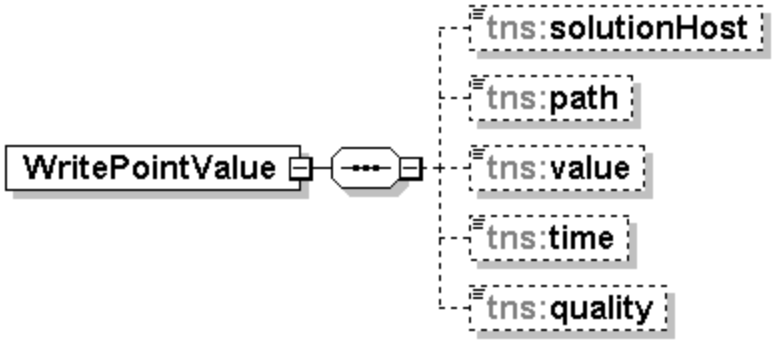
Children: tns:UpdateCollectionDataResult

Source:

```
<xsd:element name="UpdateCollectionDataResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="UpdateCollectionDataResult" type="xs:boolean" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element WritePointValue

Diagram:



Namespace: http://webservices.xhq.siemens.com

Properties: content complex

Children: tns:solutionHost tns:path tns:value tns:time tns:quality

Source:

```
<xsd:element name="WritePointValue">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="solutionHost" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="path" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="value" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="time" type="xs:string" nillable="true" minOccurs="0"/>
      <xsd:element name="quality" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

element WritePointValueResponse

Diagram:

Namespace: http://webservices.xhq.siemens.com

Properties: content complex

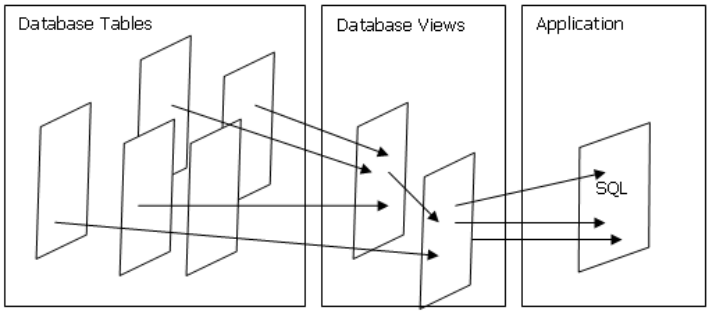
Children: tns:WritePointValueResult

Source:

```
<xsd:element name="WritePointValueResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="WritePointValueResult" type="xs:string" nillable="true" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

C - Database Views

Database views allow read-only access to commonly used information that is extracted from the XHQ High-Performance Database (the embedded database within the XHQ system). In turn, this information can be used by application developers or customer application groups in building customized views and solutions.



Target Management Database Views Model

XHQ provides database views for various features and, for your convenience, are documented in the guide relevant to the given feature:

- Monitoring/XHQStats in the XHQ Connection Guide
- Alert Notification System in the XHQ ANS User's Guide
- Performance Management (Target Management) and View Statistics in the XHQ Performance Management Guide

XHQ database views are for use by the XHQ solution only

You may configure an XHQ internal database connection to have READ access to these database views for the sole purpose of data integration into the XHQ solution (for example, into XHQ views). Any other use would violate the terms of the XHQ license. It is explicitly not permitted to access the XHQ embedded database from outside XHQ in any form or to access any internal tables or views other than reading the database views documented in this section.