



Proficiency Historian 9.1

Java APIs Reference Manual



Proprietary Notice

The information contained in this publication is believed to be accurate and reliable. However, General Electric Company assumes no responsibilities for any errors, omissions or inaccuracies. Information contained in the publication is subject to change without notice.

No part of this publication may be reproduced in any form, or stored in a database or retrieval system, or transmitted or distributed in any form by any means, electronic, mechanical photocopying, recording or otherwise, without the prior written permission of General Electric Company. Information contained herein is subject to change without notice.

© 2021, General Electric Company. All rights reserved.

Trademark Notices

GE, the GE Monogram, and Predix are either registered trademarks or trademarks of General Electric Company.

Microsoft® is a registered trademark of Microsoft Corporation, in the United States and/or other countries.

All other trademarks are the property of their respective owners.

We want to hear from you. If you have any comments, questions, or suggestions about our documentation, send them to the following email address:

doc@ge.com

Table Of Content

com.ge.ip.hds.historian.API	3
ArchiveService	6
ArchiveServiceImpl	21
CollectorOperationsService	27
CollectorOperationsServiceImpl	28
CollectorService	29
CollectorServiceImpl	37
CollectorStatisticsService	40
CollectorStatisticsServiceImpl	44
CollectorTagService	45
CollectorTagServiceImpl	55
ConfigurationManager	59
DHSService	65
DHSServiceImpl	76
DHSSStorageService	80
DHSSStorageServiceImpl	88
DataService	91
DataServiceImpl	109
DataStoreService	115
DataStoreServiceImpl	125
EnumeratedSetService	128
EnumeratedSetServiceImpl	138
InterfaceDefService	141
InterfaceDefServiceImpl	149
JobService	152
JobServiceImpl	163
LogManager	168
MessageService	169
MessageServiceImpl	173
PerformanceCounterService	174
PerformanceCounterServiceImpl	176
PublishService	177
PublishServiceImpl	188
QueryModifierService	192
QueryModifierServiceImpl	195
ServerService	196
ServerServiceImpl	208
TagService	213

TagServiceImpl	233
TagsService	240
TagsServiceImpl	241
UserDefinedTypeService	242
UserDefinedTypeServiceImpl	252
UserService	255
UserServiceImpl	256
com.ge.ip.hds.historian.API.NativeFunctions	258
NativeFunctionBridge	258
com.ge.ip.hds.historianserviceapi	297
Activator	297
Index	298

Package com.ge.ip.hds.historian.API

Interface Summary

[ArchiveService](#)

ArchiveService

[CollectorOperationsService](#)

CollectorOperationsService

[CollectorService](#)

CollectorService

[CollectorStatisticsService](#)

CollectorStatisticsService

[CollectorTagService](#)

CollectorTagService

[DHSService](#)

DHSService

[DHSService](#)

[DataService](#)

DataService

[DataStoreService](#)

DataStoreService

[EnumeratedSetService](#)

EnumeratedSetService

[InterfaceDefService](#)

InterfaceDef

[JobService](#)

JobService

[MessageService](#)

MessageService

[PerformanceCounterService](#)

PerformanceCounterService

[PublishService](#)

PublishService

[QueryModifierService](#)

QueryModifierService

[ServerService](#)

ServerService

[TagService](#)

TagService

[TagsService](#)

TagsService

[UserDefinedTypeService](#)

UserDefinedTypeService

[UserService](#)

UserService

Class Summary

[ArchiveServiceImpl](#)

[CollectorOperationsServiceImpl](#)

[CollectorServiceImpl](#)

[CollectorStatisticsServiceImpl](#)

[CollectorTagServiceImpl](#)

[ConfigurationManager](#)

ConfigurationManager

[DHSServiceImpl](#)

[DHSSStorageServiceImpl](#)

[DataServiceImpl](#)

[DataStoreServiceImpl](#)

[EnumeratedSetServiceImpl](#)

[InterfaceDefServiceImpl](#)

[JobServiceImpl](#)

[LogManager](#)

LogManager

[MessageServiceImpl](#)

[PerformanceCounterServiceImpl](#)

[PublishServiceImpl](#)

[QueryModifierServiceImpl](#)

[ServerServiceImpl](#)

[TagServiceImpl](#)

[TagsServiceImpl](#)

[UserDefinedTypeServiceImpl](#)

[UserServiceImpl](#)

com.ge.ip.hds.historian.API

Interface ArchiveService

< [Methods](#) >

public interface **ArchiveService**

ArchiveService

The ArchiveService Class implements interfaces for dealing with Historian Archive files. Historian stores data in form a binary file with file extension (.iha) termed as Historian Archive files.

Methods

AddArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
AddArchive(com.ge.ip.hds.historian.DataContracts.Archive archive)  
                                                    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for creating a Archive file.

Parameters:

archive - Archive structure.

Returns:

Archive structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
AddArchive(java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.Archive archive)  
                                                    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for creating a Archive file.

Parameters:

effectiveUserId - User name against which operation will be performed.
archive - Archive structure.

Returns:

Archive structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive
AddArchive(java.lang.String historianServerName,
java.lang.String effectiveUserId,
com.ge.ip.hds.historian.DataContracts.Archive archive)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for creating a Archive file.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
archive - Archive structure.

Returns:

Archive structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

BackupArchive

```
public boolean BackupArchive(java.lang.String archiveName,
                               java.lang.String backupFileName,
                               java.lang.String dataStoreName,
                               boolean zipArchive)
                               throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for creating Backup file of a Archive file.

Parameters:

archiveName - Name of Archive
backupFileName - Name of backup Archive file
dataStoreName - Name of data store
zipArchive - if true backup file will be zipped.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

BackupArchive

```
public boolean BackupArchive(java.lang.String effectiveUserId,  
                             java.lang.String archiveName,  
                             java.lang.String backupFileName,  
                             java.lang.String dataStoreName,  
                             boolean zipArchive)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for creating Backup file of a Archive file.

Parameters:

effectiveUserId - User name against which operation will be performed.
archiveName - Name of Archive
backupFileName - Name of backup Archive file
dataStoreName - Name of data store
zipArchive - if true backup file will be zipped.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

BackupArchive

```
public boolean BackupArchive(java.lang.String historianServerName,  
                             java.lang.String effectiveUserId,  
                             java.lang.String archiveName,  
                             java.lang.String backupFileName,  
                             java.lang.String dataStoreName,  
                             boolean zipArchive)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for creating Backup file of a Archive file.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
archiveName - Name of Archive
backupFileName - Name of backup Archive file
dataStoreName - Name of data store
zipArchive - if true backup file will be zipped.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

CloseCurrentArchive

```
public boolean CloseCurrentArchive(java.lang.String dataStoreName)
    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

Current Archive file is a file in which Historian Server is currently writing data. This method closes the current Archive file, and new Archive file becomes the current Archive file.

Parameters:

dataStoreName - Name of data store

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

CloseCurrentArchive

```
public boolean CloseCurrentArchive(java.lang.String effectiveUserId,
    java.lang.String dataStoreName)
    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

Current Archive file is a file in which Historian Server is currently writing data. This method closes the current Archive file, and new Archive file becomes the current Archive file.

Parameters:

effectiveUserId - User name against which operation will be performed.
dataStoreName - Name of data store

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

CloseCurrentArchive

```
public boolean CloseCurrentArchive(java.lang.String historianServerName,  
                                     java.lang.String effectiveUserId,  
                                     java.lang.String dataStoreName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

Current Archive file is a file in which Historian Server is currently writing data. This method closes the current Archive file, and new Archive file becomes the current Archive file.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
dataStoreName - Name of data store

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteArchive

```
public boolean DeleteArchive(java.lang.String archiveName,  
                               java.lang.String dataStoreName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting a Archive file.

Parameters:

archiveName - Name of Archive
dataStoreName - Name of data store

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteArchive

```
public boolean DeleteArchive(java.lang.String effectiveUserId,  
                             java.lang.String archiveName,  
                             java.lang.String dataStoreName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting a Archive file.

Parameters:

effectiveUserId - User name against which operation will be performed.
archiveName - Name of Archive
dataStoreName - Name of data store

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteArchive

```
public boolean DeleteArchive(java.lang.String historianServerName,  
                             java.lang.String effectiveUserId,  
                             java.lang.String archiveName,  
                             java.lang.String dataStoreName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting a Archive file.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
archiveName - Name of Archive
dataStoreName - Name of data store

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetAllArchives

```
public java.util.List GetAllArchives()  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of all Archive files in Historian Server.

Returns:

List of Archive.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetAllArchives

```
public java.util.List GetAllArchives(java.lang.String effectiveUserId)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of all Archive files in Historian Server.

Parameters:

effectiveUserId - User name against which operation will be performed.

Returns:

List of Archive.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetAllArchives

```
public java.util.List GetAllArchives(java.lang.String historianServerName,  
    java.lang.String effectiveUserId)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of all Archive files in Historian Server.

Parameters:

historianServerName - Name of target historian server.

effectiveUserId - User name against which operation will be performed.

Returns:

List of Archive.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
GetArchive(java.lang.String archiveName,  
java.lang.String dataStoreName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving Archive structure for a Archive file.

Parameters:

archiveName - Name of Archive
dataStoreName - Name of data store

Returns:

Archive structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
GetArchive(java.lang.String effectiveUserId,  
java.lang.String archiveName,  
java.lang.String dataStoreName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving Archive structure for a Archive file.

Parameters:

effectiveUserId - User name against which operation will be performed.
archiveName - Name of Archive
dataStoreName - Name of data store

Returns:

Archive structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
GetArchive(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String archiveName,  
java.lang.String dataStoreName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving Archive structure for a Archive file.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
archiveName - Name of Archive
dataStoreName - Name of data store

Returns:

Archive structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetArchiveStatistics

```
public com.ge.ip.hds.historian.DataContracts.ArchiveStatistics  
GetArchiveStatistics(java.lang.String dataStoreName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving statistics for a Archive file.

Parameters:

dataStoreName - Name of data store

Returns:

ArchiveStatistics structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetArchiveStatistics

```
public com.ge.ip.hds.historian.DataContracts.ArchiveStatistics  
GetArchiveStatistics(java.lang.String effectiveUserId,  
java.lang.String dataStoreName)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

throws

This method is for retrieving statistics for a Archive file.

Parameters:

effectiveUserId - User name against which operation will be performed.
dataStoreName - Name of data store

Returns:

ArchiveStatistics structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetArchiveStatistics

```
public com.ge.ip.hds.historian.DataContracts.ArchiveStatistics  
GetArchiveStatistics(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String dataStoreName)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

throws

This method is for retrieving statisticse for a Archive file.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
dataStoreName - Name of data store

Returns:

ArchiveStatistics structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryArchives

```
public java.util.List QueryArchives(java.lang.String archiveNameMask,  
                                     java.lang.String dataStoreNameMask,  
                                     int maxNumber)  
                                     throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of Archive files in Historian Server on some search criteria.

Parameters:

archiveNameMask - This is a name mask pattern used to retrieve list of Archives which matches pattern. The mask can include wildcards, such as asterisks (*).

dataStoreNameMask - This is a name mask pattern used to retrieve list of data stores which matches pattern. The mask can include wildcards, such as asterisks (*).

maxNumber - Maximum number of Archives request.

Returns:

List of Archive.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryArchives

```
public java.util.List QueryArchives(java.lang.String effectiveUserId,  
                                     java.lang.String archiveNameMask,  
                                     java.lang.String dataStoreNameMask,  
                                     int maxNumber)  
                                     throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of Archive files in Historian Server on some search criteria.

Parameters:

effectiveUserId - User name against which operation will be performed.

archiveNameMask - This is a name mask pattern used to retrieve list of Archives which matches pattern. The mask can include wildcards, such as asterisks (*).

dataStoreNameMask - This is a name mask pattern used to retrieve list of data stores which matches pattern. The mask can include wildcards, such as asterisks (*).

maxNumber - Maximum number of Archives request.

Returns:

List of Archive.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryArchives

```
public java.util.List QueryArchives(java.lang.String historianServerName,  
                                     java.lang.String effectiveUserId,  
                                     java.lang.String archiveNameMask,  
                                     java.lang.String dataStoreNameMask,  
                                     int maxNumber)  
                                     throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of Archive files in Historian Server on some search criteria.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
archiveNameMask - This is a name mask pattern used to retrieve list of Archives which matches pattern. The mask can include wildcards, such as asterisks (*).
dataStoreNameMask - This is a name mask pattern used to retrieve list of data stores which matches pattern. The mask can include wildcards, such as asterisks (*).
maxNumber - Maximum number of Archives request.

Returns:

List of Archive.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RestoreArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
RestoreArchive(java.lang.String archiveName,  
               java.lang.String fileName,  
               java.lang.String dataStoreName)  
               throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for restoring a Archive file from backup file.

Parameters:

archiveName - Name of Archive
fileName - Name of backup Archive file
dataStoreName - Name of data store

Returns:

Archive structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RestoreArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
RestoreArchive(java.lang.String effectiveUserId,  
java.lang.String archiveName,  
java.lang.String fileName,  
java.lang.String dataStoreName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for restoring a Archive file from backup file.

Parameters:

effectiveUserId - User name against which operation will be performed.
archiveName - Name of Archive
fileName - Name of backup Archive file
dataStoreName - Name of data store

Returns:

Archive structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RestoreArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
RestoreArchive(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String archiveName,  
java.lang.String fileName,  
java.lang.String dataStoreName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for restoring a Archive file from backup file.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
archiveName - Name of Archive
fileName - Name of backup Archive file
dataStoreName - Name of data store

Returns:

Archive structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
UpdateArchive(java.lang.String archiveName,  
com.ge.ip.hds.historian.DataContracts.Archive archive)  
throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for updating Archive property of Archive file.

Parameters:

archiveName - Name of Archive
archive - Archive structure.

Returns:

Archive structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
UpdateArchive(java.lang.String effectiveUserId,  
java.lang.String archiveName,  
com.ge.ip.hds.historian.DataContracts.Archive archive)  
throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for updating Archive property of Archive file.

Parameters:

effectiveUserId - User name against which operation will be performed.
archiveName - Name of Archive
archive - Archive structure.

Returns:

Archive structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive
UpdateArchive(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String archiveName,
com.ge.ip.hds.historian.DataContracts.Archive archive)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for updating Archive property of Archive file.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
archiveName - Name of Archive
archive - Archive structure.

Returns:

Archive structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class ArchiveServiceImpl

```
java.lang.Object
|
+--com.ge.ip.hds.historian.API.ArchiveServiceImpl
```

All Implemented Interfaces:

[ArchiveService](#)

< [Constructors](#) > < [Methods](#) >

```
public class ArchiveServiceImpl
extends java.lang.Object
implements ArchiveService
```

Constructors

ArchiveServiceImpl

```
public ArchiveServiceImpl()
```

Methods

AddArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
AddArchive(com.ge.ip.hds.historian.DataContracts.Archive archive)
```

AddArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
AddArchive(java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.Archive archive)
```

AddArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
AddArchive(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.Archive archive)
```

BackupArchive

```
public boolean BackupArchive(java.lang.String archiveName,  
java.lang.String backupFileName,  
java.lang.String dataStoreName,  
boolean zipArchive)
```

BackupArchive

```
public boolean BackupArchive(java.lang.String effectiveUserId,  
java.lang.String archiveName,  
java.lang.String backupFileName,  
java.lang.String dataStoreName,  
boolean zipArchive)
```

DeleteArchive

```
public boolean DeleteArchive(java.lang.String historianServerName,  
                             java.lang.String effectiveUserId,  
                             java.lang.String archiveName,  
                             java.lang.String dataStoreName)
```

GetAllArchives

```
public java.util.List GetAllArchives()
```

GetAllArchives

```
public java.util.List GetAllArchives(java.lang.String effectiveUserId)
```

GetAllArchives

```
public java.util.List GetAllArchives(java.lang.String historianServerName,  
                                       java.lang.String effectiveUserId)
```

GetArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
GetArchive(java.lang.String archiveName,  
            java.lang.String dataStoreName)
```

GetArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
GetArchive(java.lang.String effectiveUserId,  
            java.lang.String archiveName,  
            java.lang.String dataStoreName)
```

QueryArchives

```
public java.util.List QueryArchives(java.lang.String historianServerName,  
                                       java.lang.String effectiveUserId,  
                                       java.lang.String archiveNameMask,  
                                       java.lang.String dataStoreNameMask,  
                                       int maxNumber)
```

RestoreArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
RestoreArchive(java.lang.String archiveName,  
                java.lang.String fileName,  
                java.lang.String dataStoreName)
```

RestoreArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
RestoreArchive(java.lang.String effectiveUserId,  
                java.lang.String archiveName,  
                java.lang.String fileName,  
                java.lang.String dataStoreName)
```

RestoreArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
RestoreArchive(java.lang.String historianServerName,  
                java.lang.String effectiveUserId,  
                java.lang.String archiveName,  
                java.lang.String fileName,  
                java.lang.String dataStoreName)
```

UpdateArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive  
UpdateArchive(java.lang.String archiveName,  
               com.ge.ip.hds.historian.DataContracts.Archive archive)
```

UpdateArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive
UpdateArchive(java.lang.String effectiveUserId,
java.lang.String archiveName,
com.ge.ip.hds.historian.DataContracts.Archive archive)
```

UpdateArchive

```
public com.ge.ip.hds.historian.DataContracts.Archive
UpdateArchive(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String archiveName,
com.ge.ip.hds.historian.DataContracts.Archive archive)
```

com.ge.ip.hds.historian.API

Interface CollectorOperationsService

< [Methods](#) >

public interface **CollectorOperationsService**

CollectorOperationsService

The CollectorOperationsService Class implements interfaces for controlling Collector operation.

Methods

ExecuteAction

```
public void ExecuteAction(java.lang.String collectorName,
com.ge.ip.hds.historian.DataContracts.CollectorOperationAction actionName)
    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for Starting/Stopping a collector.

Parameters:

collectorName - Name of collector.
actionName - Operation type [start/stop].

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

ExecuteAction

```
public void ExecuteAction(java.lang.String effectiveUserId,  
                           java.lang.String collectorName,  
com.ge.ip.hds.historian.DataContracts.CollectorOperationAction actionName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for Starting/Stopping a collector.

Parameters:

effectiveUserId - User name against which operation will be performed.
collectorName - Name of collector.
actionName - Operation type [start/stop].

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

ExecuteAction

```
public void ExecuteAction(java.lang.String historianServerName,  
                           java.lang.String effectiveUserId,  
                           java.lang.String collectorName,  
com.ge.ip.hds.historian.DataContracts.CollectorOperationAction actionName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for Starting/Stopping a collector.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of collector.
actionName - Operation type [start/stop].

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class CollectorOperationsServiceImpl

```
java.lang.Object  
|  
+--com.ge.ip.hds.historian.API.CollectorOperationsServiceImpl
```

All Implemented Interfaces:

[CollectorOperationsService](#)

< [Constructors](#) > < [Methods](#) >

```
public class CollectorOperationsServiceImpl
```

extends java.lang.Object
implements [CollectorOperationsService](#)

Constructors

CollectorOperationsServiceImpl

```
public CollectorOperationsServiceImpl()
```

Methods

ExecuteAction

```
public void ExecuteAction(java.lang.String collectorName,  
com.ge.ip.hds.historian.DataContracts.CollectorOperationAction actionName)
```

ExecuteAction

```
public void ExecuteAction(java.lang.String effectiveUserId,  
java.lang.String collectorName,  
com.ge.ip.hds.historian.DataContracts.CollectorOperationAction actionName)
```

ExecuteAction

```
public void ExecuteAction(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String collectorName,  
com.ge.ip.hds.historian.DataContracts.CollectorOperationAction actionName)
```

com.ge.ip.hds.historian.API

Interface CollectorService

< [Methods](#) >

```
public interface CollectorService
```

CollectorService

The CollectorService Class implements interfaces for Querying/adding/deleting/updating Collector.

Methods

AddCollector

```
public com.ge.ip.hds.historian.DataContracts.CollectorProperty  
AddCollector(com.ge.ip.hds.historian.DataContracts.CollectorProperty  
collectorProperty)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

throws

This method is used for adding a collector.

Parameters:

collectorProperty - property structure of collector.

Returns:

CollectorProperty property structure of collector.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddCollector

```
public com.ge.ip.hds.historian.DataContracts.CollectorProperty  
AddCollector(java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.CollectorProperty collectorProperty)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

throws

This method is used for adding a collector.

Parameters:

effectiveUserId - User name against which operation will be performed.

collectorProperty - property structure of collector.

Returns:

CollectorProperty property structure of collector.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddCollector

```
public com.ge.ip.hds.historian.DataContracts.CollectorProperty
AddCollector(java.lang.String historianServerName,
java.lang.String effectiveUserId,
com.ge.ip.hds.historian.DataContracts.CollectorProperty collectorProperty)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for adding a collector.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorProperty - property structure of collector.

Returns:

CollectorProperty property structure of collector.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteCollector

```
public boolean DeleteCollector(java.lang.String collectorName,
                                boolean deleteCollectorTags)
                                throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for deleting a collector.

Parameters:

collectorName - Name of collector.
deleteCollectorTags - if it set to true, then all tags added by this collector also gets deleted.

Returns:

boolean true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteCollector

```
public boolean DeleteCollector(java.lang.String effectiveUserId,  
                               java.lang.String collectorName,  
                               boolean deleteCollectorTags)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for deleting a collector.

Parameters:

effectiveUserId - User name against which operation will be performed.
collectorName - Name of collector.
deleteCollectorTags - if it set to true, then all tags added by this collector also gets deleted.

Returns:

boolean true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteCollector

```
public boolean DeleteCollector(java.lang.String historianServerName,  
                               java.lang.String effectiveUserId,  
                               java.lang.String collectorName,  
                               boolean deleteCollectorTags)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for deleting a collector.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of collector.
deleteCollectorTags - if it set to true, then all tags added by this collector also gets deleted.

Returns:

boolean true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCollector

```
public com.ge.ip.hds.historian.DataContracts.CollectorProperty  
GetCollector(java.lang.String collectorName)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException throws
```

This method is used for retrieving CollectorProperty of a collector.

Parameters:

collectorName - Name of collector.

Returns:

CollectorProperty property structure of collector.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCollector

```
public com.ge.ip.hds.historian.DataContracts.CollectorProperty  
GetCollector(java.lang.String effectiveUserId,  
java.lang.String collectorName)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException throws
```

This method is used for retrieving CollectorProperty of a collector.

Parameters:

effectiveUserId - User name against which operation will be performed.
collectorName - Name of collector.

Returns:

CollectorProperty property structure of collector.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCollector

```
public com.ge.ip.hds.historian.DataContracts.CollectorProperty
GetCollector(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String collectorName)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving CollectorProperty of a collector.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of collector.

Returns:

CollectorProperty property structure of collector.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryCollector

```
public java.util.List QueryCollector(java.lang.String collectorNameMask,
                                                                    int maxNumber)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the qualified collector list by a given collectorNameMask.

Parameters:

collectorNameMask - This is a name mask pattern used to retrieve list of collectors which matches pattern. The mask can include wildcards, such as asterisks (*).
maxNumber - Maximum number of Collectors that can be queried.

Returns:

List of CollectorPropertyBase

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryCollector

```
public java.util.List QueryCollector(java.lang.String effectiveUserId,  
                                       java.lang.String collectorNameMask,  
                                       int maxNumber)  
                                       throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the qualified collector list by a given collectorNameMask.

Parameters:

effectiveUserId - User name against which operation will be performed.
collectorNameMask - This is a name mask pattern used to retrieve list of collectors which matches pattern. The mask can include wildcards, such as asterisks (*).
maxNumber - Maximum number of Collectors that can be queried.

Returns:

List of CollectorPropertyBase

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryCollector

```
public java.util.List QueryCollector(java.lang.String historianServerName,  
                                       java.lang.String effectiveUserId,  
                                       java.lang.String collectorNameMask,  
                                       int maxNumber)  
                                       throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the qualified collector list by a given collectorNameMask.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorNameMask - This is a name mask pattern used to retrieve list of collectors which matches pattern. The mask can include wildcards, such as asterisks (*).
maxNumber - Maximum number of Collectors that can be queried.

Returns:

List of CollectorPropertyBase

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateCollector

```
public com.ge.ip.hds.historian.DataContracts.CollectorProperty
UpdateCollector(java.lang.String collectorName,
com.ge.ip.hds.historian.DataContracts.CollectorProperty collectorProperty)
throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for updating CollectorProperty for a collector.

Parameters:

collectorName - Name of collector.
collectorProperty - property structure of collector.

Returns:

CollectorProperty property structure of collector.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateCollector

```
public com.ge.ip.hds.historian.DataContracts.CollectorProperty
UpdateCollector(java.lang.String effectiveUserId,
java.lang.String collectorName,
com.ge.ip.hds.historian.DataContracts.CollectorProperty collectorProperty)
throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for updating CollectorProperty for a collector.

Parameters:

effectiveUserId - User name against which operation will be performed.
collectorName - Name of collector.
collectorProperty - property structure of collector.

Returns:

CollectorProperty property structure of collector.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateCollector

```
public com.ge.ip.hds.historian.DataContracts.CollectorProperty
UpdateCollector( java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String collectorName,
com.ge.ip.hds.historian.DataContracts.CollectorProperty collectorProperty)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for updating CollectorProperty for a collector.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of collector.
collectorProperty - property structure of collector.

Returns:

CollectorProperty property structure of collector.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class CollectorServiceImpl

```
java.lang.Object
|
+--com.ge.ip.hds.historian.API.CollectorServiceImpl
```

All Implemented Interfaces:

[CollectorService](#)

< [Constructors](#) > < [Methods](#) >

```
public class CollectorServiceImpl
extends java.lang.Object
implements CollectorService
```

Constructors

CollectorServiceImpl

```
public CollectorServiceImpl()
```

Methods

AddCollector

```
public com.ge.ip.hds.historian.DataContracts.CollectorProperty  
AddCollector(com.ge.ip.hds.historian.DataContracts.CollectorProperty  
collectorProperty)
```

AddCollector

```
public com.ge.ip.hds.historian.DataContracts.CollectorProperty  
AddCollector(java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.CollectorProperty collectorProperty)
```

AddCollector

```
public com.ge.ip.hds.historian.DataContracts.CollectorProperty  
AddCollector(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.CollectorProperty collectorProperty)
```

DeleteCollector

```
public boolean DeleteCollector(java.lang.String collectorName,  
boolean deleteCollectorTags)
```

DeleteCollector

```
public boolean DeleteCollector(java.lang.String effectiveUserId,  
java.lang.String collectorName,  
boolean deleteCollectorTags)
```

DeleteCollector

```
public boolean DeleteCollector(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String collectorName,  
boolean deleteCollectorTags)
```

GetCollector

```
public com.ge.ip.hds.historian.DataContracts.CollectorProperty  
GetCollector(java.lang.String collectorName)
```

GetCollector

```
public com.ge.ip.hds.historian.DataContracts.CollectorProperty  
GetCollector(java.lang.String effectiveUserId,  
java.lang.String collectorName)
```

GetCollector

```
public com.ge.ip.hds.historian.DataContracts.CollectorProperty  
GetCollector(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String collectorName)
```

QueryCollector

```
public java.util.List QueryCollector(java.lang.String collectorNameMask,  
int maxNumber)
```

QueryCollector

```
public java.util.List QueryCollector(java.lang.String effectiveUserId,  
java.lang.String collectorNameMask,  
int maxNumber)
```

QueryCollector

```
public java.util.List QueryCollector(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String collectorNameMask,  
int maxNumber)
```

UpdateCollector

```
public com.ge.ip.hds.historian.DataContracts.CollectorProperty
UpdateCollector(java.lang.String collectorName,
com.ge.ip.hds.historian.DataContracts.CollectorProperty collectorProperty)
```

UpdateCollector

```
public com.ge.ip.hds.historian.DataContracts.CollectorProperty
UpdateCollector(java.lang.String effectiveUserId,
java.lang.String collectorName,
com.ge.ip.hds.historian.DataContracts.CollectorProperty collectorProperty)
```

UpdateCollector

```
public com.ge.ip.hds.historian.DataContracts.CollectorProperty
UpdateCollector(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String collectorName,
com.ge.ip.hds.historian.DataContracts.CollectorProperty collectorProperty)
```

com.ge.ip.hds.historian.API

Interface CollectorStatisticsService

< [Methods](#) >

public interface **CollectorStatisticsService**

CollectorStatisticsService

The CollectorStatisticsService Class implements interfaces for getting collector statistics.

Methods

GetStatistics

```
public com.ge.ip.hds.historian.DataContracts.CollectorStatistics  
GetStatistics(java.lang.String collectorName) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used getting collector statistics.

Parameters:

collectorName - Name of collector.

Returns:

CollectorStatistics of collector

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetStatistics

```
public com.ge.ip.hds.historian.DataContracts.CollectorStatistics  
GetStatistics(java.lang.String effectiveUserId,  
java.lang.String collectorName) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used getting collector statistics.

Parameters:

effectiveUserId - User name against which operation will be performed.
collectorName - Name of collector.

Returns:

CollectorStatistics of collector

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetStatistics

```
public com.ge.ip.hds.historian.DataContracts.CollectorStatistics  
GetStatistics(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String collectorName)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

throws

This method is used getting collector statistics.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of collector.

Returns:

CollectorStatistics of collector

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

SetStatistics

```
public boolean SetStatistics(java.lang.String collectorName,  
com.ge.ip.hds.historian.DataContracts.CollectorStatistics collectorStatistics)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

throws

This method is used setting collector statistics.

Parameters:

collectorName - Name of collector.
collectorStatistics - CollectorStatistics to be updated.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

SetStatistics

```
public boolean SetStatistics(java.lang.String effectiveUserId,  
                             java.lang.String collectorName,  
com.ge.ip.hds.historian.DataContracts.CollectorStatistics collectorStatistics)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used setting collector statistics.

Parameters:

effectiveUserId - User name against which operation will be performed.
collectorName - Name of collector.
collectorStatistics - CollectorStatistics to be updated.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

SetStatistics

```
public boolean SetStatistics(java.lang.String historianServerName,  
                             java.lang.String effectiveUserId,  
                             java.lang.String collectorName,  
com.ge.ip.hds.historian.DataContracts.CollectorStatistics collectorStatistics)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used setting collector statistics.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of collector.
collectorStatistics - CollectorStatistics to be updated.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class CollectorStatisticsServiceImpl

```
java.lang.Object
|
+--com.ge.ip.hds.historian.API.CollectorStatisticsServiceImpl
```

All Implemented Interfaces:

[CollectorStatisticsService](#)

< [Constructors](#) > < [Methods](#) >

```
public class CollectorStatisticsServiceImpl
extends java.lang.Object
implements CollectorStatisticsService
```

Constructors

CollectorStatisticsServiceImpl

```
public CollectorStatisticsServiceImpl()
```

Methods

GetStatistics

```
public com.ge.ip.hds.historian.DataContracts.CollectorStatistics
GetStatistics(java.lang.String collectorName)
```

GetStatistics

```
public com.ge.ip.hds.historian.DataContracts.CollectorStatistics
GetStatistics(java.lang.String effectiveUserId,
java.lang.String collectorName)
```

GetStatistics

```
public com.ge.ip.hds.historian.DataContracts.CollectorStatistics  
GetStatistics(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String collectorName)
```

SetStatistics

```
public boolean SetStatistics(java.lang.String collectorName,  
com.ge.ip.hds.historian.DataContracts.CollectorStatistics collectorStatistics)
```

SetStatistics

```
public boolean SetStatistics(java.lang.String effectiveUserId,  
                             java.lang.String collectorName,  
com.ge.ip.hds.historian.DataContracts.CollectorStatistics collectorStatistics)
```

SetStatistics

```
public boolean SetStatistics(java.lang.String historianServerName,  
                             java.lang.String effectiveUserId,  
                             java.lang.String collectorName,  
com.ge.ip.hds.historian.DataContracts.CollectorStatistics collectorStatistics)
```

com.ge.ip.hds.historian.API

Interface CollectorTagService

< [Methods](#) >

```
public interface CollectorTagService
```

CollectorTagService

The CollectorTagService Class implements interfaces for tag operation for a collector.

Methods

BrowseCollectorTag

```
public com.ge.ip.hds.historian.DataContracts.CollectorTagBrowseResult  
BrowseCollectorTag(java.lang.String collectorName,  
java.lang.String tagNameMask,  
int maxNumber,  
boolean forceToRefresh) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the qualified tag list by a given tagNameMask for a collector.

Parameters:

collectorName - Name of Collector.
tagNameMask - This is a name mask pattern used to retrieve list of tags which matches pattern. The mask can include wildcards, such as asterisks (*).
maxNumber - Maximum tag number that can be queried and also managed on the server side (1000 by default). If the maxNumber parameter is greater than the max tag query number setting in server side, the query complies with the server side setting.
forceToRefresh - if set to true, than tag list keep refreshing in a tag cache.

Returns:

CollectorTagBrowseResult

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

BrowseCollectorTag

```
public com.ge.ip.hds.historian.DataContracts.CollectorTagBrowseResult  
BrowseCollectorTag(java.lang.String effectiveUserId,  
java.lang.String collectorName,  
java.lang.String tagNameMask,  
int maxNumber,  
boolean forceToRefresh) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the qualified tag list by a given tagNameMask for a collector.

Parameters:

effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector.
tagNameMask - This is a name mask pattern used to retrieve list of tags which matches pattern. The mask can include wildcards, such as asterisks (*).
maxNumber - Maximum tag number that can be queried and also managed on the server side (1000 by default). If the maxNumber parameter is greater than the max tag query number setting in server side, the query complies with the server side setting.
forceToRefresh - if set to true, than tag list keep refreshing in a tag cache.

Returns:

CollectorTagBrowseResult

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

BrowseCollectorTag

```
public com.ge.ip.hds.historian.DataContracts.CollectorTagBrowseResult  
BrowseCollectorTag(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String collectorName,  
java.lang.String tagNameMask,  
int maxNumber,  
boolean forceToRefresh) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the qualified tag list by a given tagNameMask for a collector.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector.
tagNameMask - This is a name mask pattern used to retrieve list of tags which matches pattern. The mask can include wildcards, such as asterisks (*).
maxNumber - Maximum tag number that can be queried and also managed on the server side (1000 by default). If the maxNumber parameter is greater than the max tag query number setting in server side, the query complies with the server side setting.
forceToRefresh - if set to true, than tag list keep refreshing in a tag cache.

Returns:

CollectorTagBrowseResult

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

BrowseCollectorTagByPage

```
public com.ge.ip.hds.historian.DataContracts.CollectorTagBrowseResult  
BrowseCollectorTagByPage(java.lang.String collectorName,  
int pageNumber,  
int numberPerPage,  
boolean forceToRefresh) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the qualified tag list by a given tagNameMask for a collector in chunks or pages.

Parameters:

collectorName - Name of Collector.
pageNumber - page index.
numberPerPage - Number of tags in a page.
forceToRefresh - if set to true, than tag list keep refreshing in a tag cache.

Returns:

CollectorTagBrowseResult

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

BrowseCollectorTagByPage

```
public com.ge.ip.hds.historian.DataContracts.CollectorTagBrowseResult  
BrowseCollectorTagByPage(java.lang.String effectiveUserId,  
java.lang.String collectorName,  
int pageNumber,  
int numberPerPage,  
boolean forceToRefresh)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the qualified tag list by a given tagNameMask for a collector in chunks or pages.

Parameters:

effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector.
pageNumber - page index.
numberPerPage - Number of tags in a page.
forceToRefresh - if set to true, than tag list keep refreshing in a tag cache.

Returns:

CollectorTagBrowseResult

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

BrowseCollectorTagByPage

```
public com.ge.ip.hds.historian.DataContracts.CollectorTagBrowseResult  
BrowseCollectorTagByPage(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String collectorName,  
int pageNumber,  
int numberPerPage,  
boolean forceToRefresh)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the qualified tag list by a given tagNameMask for a collector in chunks or pages.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector.
pageNumber - page index.
numberPerPage - Number of tags in a page.
forceToRefresh - if set to true, than tag list keep refreshing in a tag cache.

Returns:

CollectorTagBrowseResult

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCollectorTagProperty

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
GetCollectorTagProperty(java.lang.String collectorName,  
java.lang.String tagName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the TagProperty of a tag added by a collector.

Parameters:

collectorName - Name of Collector.
tagName - Name of tag.

Returns:

TagProperty of a tag.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCollectorTagProperty

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
GetCollectorTagProperty(java.lang.String effectiveUserId,  
java.lang.String collectorName,  
java.lang.String tagName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the TagProperty of a tag added by a collector..

Parameters:

effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector.
tagName - Name of tag.

Returns:

TagProperty of a tag.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCollectorTagProperty

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
GetCollectorTagProperty(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String collectorName,
java.lang.String tagName)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the TagProperty of a tag added by a collector..

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector.
tagName - Name of tag.

Returns:

TagProperty of a tag.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCurrentValue

```
public java.util.List GetCurrentValue(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String collectorName,
java.util.List sourceAddress)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the list of current data value of list of tags of a collector.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector.
sourceAddress - List of source Address of tags.

Returns:

List of DataProperty.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCurrentValue

```
public java.util.List GetCurrentValue(java.lang.String effectiveUserId,  
                                       java.lang.String collectorName,  
                                       java.util.List sourceAddress)  
                                       throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the list of current data value of list of tags of a collector.

Parameters:

effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector.
sourceAddress - List of source Address of tags.

Returns:

List of DataProperty.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCurrentValue

```
public java.util.List GetCurrentValue(java.lang.String collectorName,  
                                       java.util.List sourceAddress)  
                                       throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the list of current data value of list of tags of a collector.

Parameters:

collectorName - Name of Collector.
sourceAddress - List of source Address of tags.

Returns:

List of DataProperty.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

HierarchicalBrowse

```
public com.ge.ip.hds.historian.DataContracts.HierarchicalBrowseResponse  
HierarchicalBrowse(java.lang.String collectorName,  
int collectorType,  
boolean onlyNonServerCollTags,  
java.lang.String position,  
java.lang.String nodeFilter)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

Hierarchical browsing enables you to browse collectors in a hierarchical manner if your server supports hierarchical organization of tags in a tree structure.

Parameters:

collectorName - Name of Collector
collectorType - Type of the collector ex:OPC,Simulation
onlyNonServerCollTags - if false Server to Server Collector ex:Calculation Collector
position - Position of browsing for specific Tag.
nodeFilter - Node filtering required for tag to get hierarchical information.

Returns:

HierarchicalBrowseResponse

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

HierarchicalBrowse

```
public com.ge.ip.hds.historian.DataContracts.HierarchicalBrowseResponse  
HierarchicalBrowse(java.lang.String effectiveUserId,  
java.lang.String collectorName,  
int collectorType,  
boolean onlyNonServerCollTags,  
java.lang.String position,  
java.lang.String nodeFilter)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

Parameters:

effectiveUserId - User name against which operation will be performed
collectorName - Name of Collector
collectorType - Type of the collector ex:OPC,Simulation
onlyNonServerCollTags - if false Server to Server Collector ex:Calculation Collector
position - Position of browsing for specific Tag.
nodeFilter - Node filtering required for tag to get hierarchical information.

Returns:

HierarchicalBrowseResponse

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

HierarchicalBrowse

```
public com.ge.ip.hds.historian.DataContracts.HierarchicalBrowseResponse  
HierarchicalBrowse(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String collectorName,  
int collectorType,  
boolean onlyNonServerCollTags,  
java.lang.String position,  
java.lang.String nodeFilter)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector
collectorType - Type of the collector ex:OPC,Simulation
onlyNonServerCollTags - if false Server to Server Collector ex:Calculation Collector
position - Position of browsing for specific Tag.
nodeFilter - Node filtering required for tag to get hierarchical information.

Returns:

HierarchicalBrowseResponse

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

Reload

```
public boolean Reload(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String collectorName,  
java.util.List tagNames,  
java.util.Date startTime,  
java.util.Date endTime)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector
tagNames - List of Tags data to be retrieved
startTime - Start Time
endTime - End Time

Returns:

true on success

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

Reload

```
public boolean Reload(java.lang.String effectiveUserId,  
                      java.lang.String collectorName,  
                      java.util.List tagNames,  
                      java.util.Date startTime,  
                      java.util.Date endTime)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

Parameters:

effectiveUserId - User name against which operation will be performed
collectorName - Name of Collector
tagNames - List of Tags data to be retrieved
startTime - start time
endTime - end time

Returns:

true on success

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

Reload

```
public boolean Reload(java.lang.String collectorName,  
                      java.util.List tagNames,  
                      java.util.Date startTime,  
                      java.util.Date endTime)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

Collector gets tags data between given start time and end time

Parameters:

collectorName - Name of Collector
tagNames - List of Tags data to be retrieved
startTime - Start time
endTime - End time

Returns:

true on success

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class CollectorTagServiceImpl

```
java.lang.Object
|
+--com.ge.ip.hds.historian.API.CollectorTagServiceImpl
```

All Implemented Interfaces:

[CollectorTagService](#)

< [Constructors](#) > < [Methods](#) >

```
public class CollectorTagServiceImpl
extends java.lang.Object
implements CollectorTagService
```

Constructors

CollectorTagServiceImpl

```
public CollectorTagServiceImpl()
```

Methods

BrowseCollectorTag

```
public com.ge.ip.hds.historian.DataContracts.CollectorTagBrowseResult
BrowseCollectorTag(java.lang.String collectorName,
java.lang.String tagNameMask,
int maxNumber,
boolean forceToRefresh)
```

BrowseCollectorTag

```
public com.ge.ip.hds.historian.DataContracts.CollectorTagBrowseResult
BrowseCollectorTag(java.lang.String effectiveUserId,
java.lang.String collectorName,
java.lang.String tagNameMask,
int maxNumber,
boolean forceToRefresh)
```

BrowseCollectorTag

```
public com.ge.ip.hds.historian.DataContracts.CollectorTagBrowseResult  
BrowseCollectorTag(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String collectorName,  
java.lang.String tagNameMask,  
int maxNumber,  
boolean forceToRefresh)
```

BrowseCollectorTagByPage

```
public com.ge.ip.hds.historian.DataContracts.CollectorTagBrowseResult  
BrowseCollectorTagByPage(java.lang.String collectorName,  
int pageNumber,  
int numberPerPage,  
boolean forceToRefresh)
```

BrowseCollectorTagByPage

```
public com.ge.ip.hds.historian.DataContracts.CollectorTagBrowseResult  
BrowseCollectorTagByPage(java.lang.String effectiveUserId,  
java.lang.String collectorName,  
int pageNumber,  
int numberPerPage,  
boolean forceToRefresh)
```

BrowseCollectorTagByPage

```
public com.ge.ip.hds.historian.DataContracts.CollectorTagBrowseResult  
BrowseCollectorTagByPage(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String collectorName,  
int pageNumber,  
int numberPerPage,  
boolean forceToRefresh)
```

GetCollectorTagProperty

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
GetCollectorTagProperty(java.lang.String collectorName,  
java.lang.String tagName)
```

GetCollectorTagProperty

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
GetCollectorTagProperty(java.lang.String effectiveUserId,  
java.lang.String collectorName,  
java.lang.String tagName)
```

GetCollectorTagProperty

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
GetCollectorTagProperty(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String collectorName,  
java.lang.String tagName)
```

GetCurrentValue

```
public java.util.List GetCurrentValue(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String collectorName,  
java.util.List sourceAddress)
```

GetCurrentValue

```
public java.util.List GetCurrentValue(java.lang.String effectiveUserId,  
java.lang.String collectorName,  
java.util.List sourceAddress)
```

GetCurrentValue

```
public java.util.List GetCurrentValue(java.lang.String collectorName,  
java.util.List sourceAddress)
```

HierarchicalBrowse

```
public com.ge.ip.hds.historian.DataContracts.HierarchicalBrowseResponse
HierarchicalBrowse(java.lang.String collectorName,
int collectorType,
boolean onlyNonServerCollTags,
java.lang.String position,
java.lang.String nodeFilter)
```

HierarchicalBrowse

```
public com.ge.ip.hds.historian.DataContracts.HierarchicalBrowseResponse
HierarchicalBrowse(java.lang.String effectiveUserId,
java.lang.String collectorName,
int collectorType,
boolean onlyNonServerCollTags,
java.lang.String position,
java.lang.String nodeFilter)
```

HierarchicalBrowse

```
public com.ge.ip.hds.historian.DataContracts.HierarchicalBrowseResponse
HierarchicalBrowse(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String collectorName,
int collectorType,
boolean onlyNonServerCollTags,
java.lang.String position,
java.lang.String nodeFilter)
```

Reload

```
public boolean Reload(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String collectorName,
java.util.List tagNames,
java.util.Date startTime,
java.util.Date endTime)
```

Reload

```
public boolean Reload(java.lang.String effectiveUserId,  
                      java.lang.String collectorName,  
                      java.util.List tagNames,  
                      java.util.Date startTime,  
                      java.util.Date endTime)
```

Reload

```
public boolean Reload(java.lang.String collectorName,  
                      java.util.List tagNames,  
                      java.util.Date startTime,  
                      java.util.Date endTime)
```

com.ge.ip.hds.historian.API

Class ConfigurationManager

```
java.lang.Object  
|  
+--com.ge.ip.hds.historian.API.ConfigurationManager
```

< [Methods](#) >

```
public class ConfigurationManager  
extends java.lang.Object
```

ConfigurationManager

The ConfigurationManager Class implements methods for basic setup required for utilizing JAVA API

Example: ConfigurationManager.getInstance().Initialize(cppBringPath,historianServer, "", "", 10000, 10000, true, false);

Methods

AddHistorianServer

```
public com.ge.ip.hds.historian.DataContracts.ErrorCode  
AddHistorianServer(java.lang.String historianServerName)
```

This method adds a HistorianServer

Parameters:

historianServerName - Name of Historian Server

Returns:

ErrorCode

AddHistorianServer

```
public com.ge.ip.hds.historian.DataContracts.ErrorCode  
AddHistorianServer(java.lang.String historianServerName,  
java.lang.String userName,  
java.lang.String password)
```

This method adds a HistorianServer

Parameters:

historianServerName - Name of target Historian Server

userName - username of any existing user in Historian Server

password - password for authentication

Returns:

ErrorCode

DeleteHistorianServer

```
public com.ge.ip.hds.historian.DataContracts.ErrorCode  
DeleteHistorianServer(java.lang.String historianServerName)
```

This method deleted the Historian Server

Parameters:

historianServerName - Name of the target Historian Serve

Returns:

ErrorCode

GetHistorianServers

```
public java.util.List GetHistorianServers(boolean includePrimaryServer)
```

This method gets list of Historian servers

Parameters:

includePrimaryServer - true, if want to include primary historian server in the list

Returns:

List of Historian Servers

Initialize

```
public boolean Initialize(java.lang.String historianServerName,  
                           java.lang.String userName,  
                           java.lang.String password,  
                           int maxDataQueryResultSize,  
                           int maxTagQueryResultSize,  
                           boolean multipleTagsSampleRead,  
                           boolean authorizationEnable)
```

Parameters:

historianServerName - Name of Target Historian Server

userName - username of any existing user in Historian Server

password - password for authentication

maxDataQueryResultSize - maximum number of data results that can be fetched from historian server

maxTagQueryResultSize - maximum tags that can be fetched from historian server

multipleTagsSampleRead - true if multiple tag results are to be allowed in query result

authorizationEnable - true if authorization is enabled

Returns:

true if initialized

Initialize

```
public boolean Initialize(java.lang.String pluginDllPath,  
                           java.lang.String historianServerName,  
                           java.lang.String userName,  
                           java.lang.String password,  
                           int maxDataQueryResultSize,  
                           int maxTagQueryResultSize,  
                           boolean multipleTagsSampleRead,  
                           boolean authorizationEnable)
```

Parameters:

pluginDllPath - Path where C++ bridge DLL file is located
historianServerName - Name of Target Historian Server
userName - username of any existing user in Historian Server
password - password for authentication
maxDataQueryResultSize - maximum number of data results that can be fetched from historian server
maxTagQueryResultSize - maximum tags that can be fetched from historian server
multipleTagsSampleRead - true if multiple tag results are to be allowed in query result
authorizationEnable - true if authorization is enabled

Returns:

true if initialized

InitializeLogging

```
public boolean InitializeLogging(java.lang.String logFilePath,  
com.ge.ip.hds.historian.DataContracts.LoggingLevel loggingLevel,  
                                int maxFileLen)  
                                throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method initializes logging configuration

Parameters:

logFilePath - Path to log files
loggingLevel - log level Ex:INFO,ERROR,DEBUG
maxFileLen - Maximum file length in MB - allows upto 100MB

Returns:

true after initialization

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

InitializePluginDll

```
public boolean InitializePluginDll(java.lang.String pluginDllPath)
```

This methods loads dependent resources from pluginDllPath.

Parameters:

pluginDllPath - Path where C++ bridge DLL file is located

Returns:

true if plug-ins are initialized

InitializePluginDllInLibrary

```
public boolean InitializePluginDllInLibrary(java.lang.String dllFileName)
```

This methods initializes dependent resources from dllFile.

Parameters:

dllFileName - Name of c++ bridge DLL file

Returns:

true if plug-ins are initialized

IsPluginLoaded

```
public boolean IsPluginLoaded()
```

This methods checks if plug-in is loaded

Returns:

true if dependent plug-ins are loaded

SetLoggingConfigure

```
public boolean
SetLoggingConfigure(com.ge.ip.hds.historian.DataContracts.LoggingLevel
loggingLevel,
                    int maxFileLen)
                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method allows user to set Logging Configuration

Parameters:

loggingLevel - log level Ex:INFO,ERROR,DEBUG
maxFileLen - Maximum file length in MB - allows upto 100MB

Returns:

true after configuration is done

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

SetMaxConnectionPoolSize

```
public boolean SetMaxConnectionPoolSize(int maxConnectionPoolSize)
```

This method sets maximum connections that are allowed to Historian server.

Parameters:

maxConnectionPoolSize - max number of connection,should not be less than 1

Returns:

true when configuration is set successfully

SetQuerySizeConfiguration

```
public boolean SetQuerySizeConfiguration(int maxDataQueryResultSize,
                                         int maxTagQueryResultSize,
                                         boolean multipleTagsSampleRead)
                                         throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method allows user to set query size.

Parameters:

maxDataQueryResultSize - maximum number of data results that can be fetched from historian server
maxTagQueryResultSize - maximum tags that can be fetched from historian server
multipleTagsSampleRead - true if multiple tag results are to be allowed in query result

Returns:

true when set size configuration is done

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateHistorianServer

```
public com.ge.ip.hds.historian.DataContracts.ErrorCode  
UpdateHistorianServer(java.lang.String historianServerName,  
java.lang.String userName,  
java.lang.String password,  
boolean authorizationEnable)
```

Parameters:

historianServerName - Name of Target Historian Server
userName - username of any existing user in Historian Server
password - password for authentication
authorizationEnable - true if authorization is required

Returns:

ErrorCode

getInstance

```
public static ConfigurationManager getInstance()
```

This methods gets an instance of Configuration Manager

Returns:

ConfigurationManager structure

com.ge.ip.hds.historian.API

Interface DHSService

< [Methods](#) >

```
public interface DHSService
```

DHSService

The DHSService Class implements interfaces for dealing with Mirror nodes. Historian provides mirroring of stored data on multiple nodes to provide high levels of data reliability.

Methods

AddDHSService

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo  
AddDHSService(com.ge.ip.hds.historian.DataContracts.DHSServiceInfo  
dHSServiceInfo)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is to add DHS Service

Parameters:

dHSServiceInfo - DHSServiceInfo

Returns:

DHSServiceInfo

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddDHSService

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo  
AddDHSService(java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.DHSServiceInfo dHSServiceInfo)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is to add DHS Service

Parameters:

effectiveUserId - User name against which operation will be performed.

dHSServiceInfo - DHSServiceInfo

Returns:

DHSServiceInfo

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddDHSService

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo
AddDHSService(java.lang.String historianServerName,
java.lang.String effectiveUserId,
com.ge.ip.hds.historian.DataContracts.DHSServiceInfo dHSServiceInfo)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is to add DHS Service

Parameters:

historianServerName - Name of target historian server
effectiveUserId - User name against which operation will be performed.
dHSServiceInfo - DHSServiceInfo

Returns:

DHSServiceInfo

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddDHSServiceNode

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo
AddDHSServiceNode(java.lang.String nodeName,
java.lang.String storageName,
boolean replicateNode)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for adding Mirror Node

Parameters:

nodeName - IP Address or Computer name of Target node
storageName - Name of the storage,if null it sets to default
replicateNode - if true,automatically copies the archive files from Primary machine to Mirror machine. If false,user needs to manually copy the files.

Returns:

DHSServiceInfo

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddDHSServiceNode

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo
AddDHSServiceNode(java.lang.String effectiveUserId,
java.lang.String nodeName,
java.lang.String storageName,
boolean replicateNode)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for adding Mirror Node

Parameters:

effectiveUserId - User name against which operation will be performed.

nodeName - IP Address or Computer name of Target node

storageName - Name of the storage,if null it sets to default

replicateNode - if true,automatically copies the archive files from Primary machine to Mirror machine. If false,user needs to manually copy the files.

Returns:

DHSServiceInfo

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddDHSServiceNode

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo
AddDHSServiceNode(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String nodeName,
java.lang.String storageName,
boolean replicateNode)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for adding Mirror Node

Parameters:

historianServerName - Name of target historian server

effectiveUserId - User name against which operation will be performed.

nodeName - IP Address or Computer name of Target node

storageName - Name of the storage,if null it sets to default

replicateNode - if true,automatically copies the archive files from Primary machine to Mirror machine. If false,user needs to manually copy the files.

Returns:

DHSServiceInfo

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteDHSService

```
public boolean DeleteDHSService(java.lang.String dHSServiceName)
    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting a Service.

Parameters:

dHSServiceName - Name of DHS Service ex:"ConfigManager_[nodename]"

Returns:

true on successful deletion

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteDHSService

```
public boolean DeleteDHSService(java.lang.String effectiveUserId,
    java.lang.String dHSServiceName)
    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is to delete DHS Service

Parameters:

effectiveUserId - User name against which operation will be performed.

dHSServiceName - Name of DHS Service ex:"ConfigManager_[nodename]"

Returns:

true on successful deletion

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteDHSService

```
public boolean DeleteDHSService(java.lang.String historianServerName,  
                                java.lang.String effectiveUserId,  
                                java.lang.String dHSServiceName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is to delete DHS Service

Parameters:

historianServerName - Name of target historian server
effectiveUserId - User name against which operation will be performed.
dHSServiceName - Name of DHS Service ex:"ConfigManager_[nodename]"

Returns:

true on successful deletion

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteDHSServiceNode

```
public boolean DeleteDHSServiceNode(java.lang.String nodeName,  
                                     java.lang.String storageName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting Mirror Node

Parameters:

nodeName - IP Address or Computer name of Target node
storageName - Name of the storage,if null it sets to default

Returns:

true on successful deletion

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteDHSServiceNode

```
public boolean DeleteDHSServiceNode( java.lang.String effectiveUserId,  
                                     java.lang.String nodeName,  
                                     java.lang.String storageName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting Mirror Node

Parameters:

effectiveUserId - User name against which operation will be performed.
nodeName - IP Address or Computer name of Target node
storageName - Name of the storage,if null it sets to default

Returns:

true on successful deletion

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteDHSServiceNode

```
public boolean DeleteDHSServiceNode( java.lang.String historianServerName,  
                                     java.lang.String effectiveUserId,  
                                     java.lang.String nodeName,  
                                     java.lang.String storageName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting Mirror Node

Parameters:

historianServerName - Name of target historian server
effectiveUserId - User name against which operation will be performed.
nodeName - IP Address or Computer name of Target node
storageName - Name of the storage,if null it sets to default

Returns:

true on successful deletion

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDHSService

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo  
GetDHSService(java.lang.String dHSServiceName)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

throws

This method is for retrieving DHS Service Properties structure for a Service.

Parameters:

dHSServiceName - Name of DHS Service ex:"ConfigManager_[nodename]"

Returns:

DHSServiceInfo

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDHSService

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo  
GetDHSService(java.lang.String effectiveUserId,  
java.lang.String dHSServiceName)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

throws

This method is for retrieving DHS Service Properties structure for a Service.

Parameters:

effectiveUserId - User name against which operation will be performed.

dHSServiceName - Name of DHS Service ex:"ConfigManager_[nodename]"

Returns:

DHSServiceInfo

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDHSService

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo
GetDHSService(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String dhSServiceName)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving DHS Service Properties structure for a Service.

Parameters:

historianServerName - Name of target historian server
effectiveUserId - User name against which operation will be performed.
dhSServiceName - Name of DHS Service ex:"ConfigManager_[nodename]"

Returns:

DHSServiceInfo

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDHSServices

```
public java.util.List GetDHSServices(java.lang.String dhSServiceMask,
                                                                    boolean withReason)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of all historian server services in a DHS system

Parameters:

dhSServiceMask - This is a name mask pattern used to retrieve list of services which matches pattern. The mask can include wildcards, such as asterisks (*).
withReason - if true, it returns status of the services

Returns:

List of DHSServiceInfo

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDHSServices

```
public java.util.List GetDHSServices(java.lang.String effectiveUserId,  
                                       java.lang.String dHSServiceMask,  
                                       boolean withReason)  
                                       throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of all historian server services in a DHS system.

Parameters:

effectiveUserId - User name against which operation will be performed.
dHSServiceMask - This is a name mask pattern used to retrieve list of services which matches pattern. The mask can include wildcards, such as asterisks (*).
withReason - if true, it returns status of the services

Returns:

List of DHSServiceInfo

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDHSServices

```
public java.util.List GetDHSServices(java.lang.String historianServerName,  
                                       java.lang.String effectiveUserId,  
                                       java.lang.String dHSServiceMask,  
                                       boolean withReason)  
                                       throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of all historian server services in a DHS system.

Parameters:

historianServerName - Name of target historian server
effectiveUserId - User name against which operation will be performed.
dHSServiceMask - This is a name mask pattern used to retrieve list of services which matches pattern. The mask can include wildcards, such as asterisks (*).
withReason - if true, it returns status of the services

Returns:

List of DHSServiceInfo

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateDHSService

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo
UpdateDHSService(java.lang.String dHSServiceName,
com.ge.ip.hds.historian.DataContracts.DHSServiceInfo dHSServiceInfo)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is to update DHS Service

Parameters:

dHSServiceName - Name of DHS Service ex:"ConfigManager_[nodename]"
dHSServiceInfo - DHSServiceInfo

Returns:

DHS Service Properties Structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateDHSService

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo
UpdateDHSService(java.lang.String effectiveUserId,
java.lang.String dHSServiceName,
com.ge.ip.hds.historian.DataContracts.DHSServiceInfo dHSServiceInfo)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is to update DHS Service

Parameters:

effectiveUserId - User name against which operation will be performed.
dHSServiceName - Name of DHS Service ex:"ConfigManager_[nodename]"
dHSServiceInfo - DHSServiceInfo

Returns:

DHSServiceInfo

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateDHSService

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo
UpdateDHSService(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String dHSServiceName,
com.ge.ip.hds.historian.DataContracts.DHSServiceInfo dHSServiceInfo)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is to update DHS Service

Parameters:

historianServerName - Name of target historian server
effectiveUserId - User name against which operation will be performed.
dHSServiceName - Name of DHS Service ex:"ConfigManager_[nodename]"
dHSServiceInfo - DHSServiceInfo

Returns:

DHSServiceInfo

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class DHSServiceImpl

```
java.lang.Object
|
+--com.ge.ip.hds.historian.API.DHSServiceImpl
```

All Implemented Interfaces:

[DHSService](#)

< [Constructors](#) > < [Methods](#) >

```
public class DHSServiceImpl
extends java.lang.Object
implements DHSService
```

Constructors

DHSServiceImpl

```
public DHSServiceImpl()
```

Methods

AddDHSService

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo
AddDHSService(com.ge.ip.hds.historian.DataContracts.DHSServiceInfo
dHSServiceInfo)
```

AddDHSService

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo
AddDHSService(java.lang.String effectiveUserId,
com.ge.ip.hds.historian.DataContracts.DHSServiceInfo dHSServiceInfo)
```

AddDHSService

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo
AddDHSService(java.lang.String historianServerName,
java.lang.String effectiveUserId,
com.ge.ip.hds.historian.DataContracts.DHSServiceInfo dHSServiceInfo)
```

AddDHSServiceNode

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo
AddDHSServiceNode(java.lang.String nodeName,
java.lang.String storageName,
boolean replicateNode)
```

AddDHSServiceNode

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo
AddDHSServiceNode(java.lang.String effectiveUserId,
java.lang.String nodeName,
java.lang.String storageName,
boolean replicateNode)
```

AddDHSServiceNode

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo
AddDHSServiceNode(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String nodeName,
java.lang.String storageName,
boolean replicateNode)
```

DeleteDHSService

```
public boolean DeleteDHSService(java.lang.String dHSServiceName)
```

DeleteDHSService

```
public boolean DeleteDHSService(java.lang.String effectiveUserId,
java.lang.String dHSServiceName)
```

DeleteDHSService

```
public boolean DeleteDHSService(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String dHSServiceName)
```

DeleteDHSServiceNode

```
public boolean DeleteDHSServiceNode(java.lang.String nodeName,
java.lang.String storageName)
```

DeleteDHSServiceNode

```
public boolean DeleteDHSServiceNode(java.lang.String effectiveUserId,
java.lang.String nodeName,
java.lang.String storageName)
```

GetDHSServices

```
public java.util.List GetDHSServices(java.lang.String historianServerName,  
                                       java.lang.String effectiveUserId,  
                                       java.lang.String dHSServiceMask,  
                                       boolean withReason)
```

UpdateDHSService

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo  
UpdateDHSService(java.lang.String dHSServiceName,  
com.ge.ip.hds.historian.DataContracts.DHSServiceInfo dHSServiceInfo)
```

UpdateDHSService

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo  
UpdateDHSService(java.lang.String effectiveUserId,  
java.lang.String dHSServiceName,  
com.ge.ip.hds.historian.DataContracts.DHSServiceInfo dHSServiceInfo)
```

UpdateDHSService

```
public com.ge.ip.hds.historian.DataContracts.DHSServiceInfo  
UpdateDHSService(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String dHSServiceName,  
com.ge.ip.hds.historian.DataContracts.DHSServiceInfo dHSServiceInfo)
```

com.ge.ip.hds.historian.API

Interface DHSSStorageService

< [Methods](#) >

```
public interface DHSSStorageService
```

Methods

AddDHSStorage

```
public com.ge.ip.hds.historian.DataContracts.DHSStorageProperty
AddDHSStorage(com.ge.ip.hds.historian.DataContracts.DHSStorageProperty
dHSStorageProperty)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for creating a DHSStorage.

Parameters:

dHSStorageProperty - DHSStorageProperty structure

Returns:

DHSStorageProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddDHSStorage

```
public com.ge.ip.hds.historian.DataContracts.DHSStorageProperty
AddDHSStorage(java.lang.String effectiveUserId,
com.ge.ip.hds.historian.DataContracts.DHSStorageProperty dHSStorageProperty)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for creating a DHSStorage.

Parameters:

effectiveUserId - User name against which operation will be performed.
dHSStorageProperty - DHSStorageProperty structure

Returns:

DHSStorageProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddDHSStorage

```
public com.ge.ip.hds.historian.DataContracts.DHSStorageProperty
AddDHSStorage(java.lang.String historianServerName,
java.lang.String effectiveUserId,
com.ge.ip.hds.historian.DataContracts.DHSStorageProperty dHSStorageProperty)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for creating a DHSStorage.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
dHSStorageProperty - DHSStorageProperty structure

Returns:

DHSStorageProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteDHSStorage

```
public boolean DeleteDHSStorage(java.lang.String dHSStorageName)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting a DHSStorage

Parameters:

dHSStorageName - Name of DHS Storage

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteDHSStorage

```
public boolean DeleteDHSStorage(java.lang.String effectiveUserId,  
                               java.lang.String dHSStorageName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting a DHSStorage

Parameters:

effectiveUserId - User name against which operation will be performed.
dHSStorageName - Name of DHS Storage

Returns:

true on success

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteDHSStorage

```
public boolean DeleteDHSStorage(java.lang.String historianServerName,  
                               java.lang.String effectiveUserId,  
                               java.lang.String dHSStorageName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting a DHSStorage

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
dHSStorageName - Name of DHS Storage

Returns:

true on success

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDHSStorage

```
public com.ge.ip.hds.historian.DataContracts.DHSStorageProperty  
GetDHSStorage(java.lang.String dHSStorageName) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving DHSStorageProperty.

Parameters:

dHSStorageName - Name of DHS Storage

Returns:

DHSStorageProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDHSStorage

```
public com.ge.ip.hds.historian.DataContracts.DHSStorageProperty  
GetDHSStorage(java.lang.String effectiveUserId,  
java.lang.String dHSStorageName) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving DHSStorageProperty.

Parameters:

effectiveUserId - User name against which operation will be performed.
dHSStorageName - Name of DHS Storage

Returns:

DHSStorageProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDHSStorage

```
public com.ge.ip.hds.historian.DataContracts.DHSStorageProperty
GetDHSStorage(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String dHSStorageName)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving DHSStorageProperty.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
dHSStorageName - Name of DHS Storage

Returns:

DHSStorageProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDHSStorages

```
public java.util.List GetDHSStorages(java.lang.String dHSStorageNameMask)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method retrieves the List of DHSStorages in Historian Server on some search criteria.

Parameters:

dHSStorageNameMask - This is a name mask pattern used to retrieve list of data stores which matches pattern. The mask can include wildcards, such as asterisks (*).

Returns:

List of DHSStorageProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDHSStorages

```
public java.util.List GetDHSStorages(java.lang.String effectiveUserId,  
                                       java.lang.String dHSStorageNameMask)  
                                       throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method retrieves the List of DHSStorages in Historian Server on some search criteria.

Parameters:

effectiveUserId - User name against which operation will be performed.
dHSStorageNameMask - This is a name mask pattern used to retrieve list of data stores which matches pattern. The mask can include wildcards, such as asterisks (*).

Returns:

List of DHSStorageProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDHSStorages

```
public java.util.List GetDHSStorages(java.lang.String historianServerName,  
                                       java.lang.String effectiveUserId,  
                                       java.lang.String dHSStorageNameMask)  
                                       throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method retrieves the List of DHSStorages in Historian Server on some search criteria.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
dHSStorageNameMask - This is a name mask pattern used to retrieve list of data stores which matches pattern. The mask can include wildcards, such as asterisks (*).

Returns:

List of DHSStorageProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateDHSStorage

```
public com.ge.ip.hds.historian.DataContracts.DHSStorageProperty
UpdateDHSStorage(java.lang.String dHSStorageName,
com.ge.ip.hds.historian.DataContracts.DHSStorageProperty dHSStorageProperty)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for updating DHSStorage

Parameters:

dHSStorageName - Name of DHS Storage
dHSStorageProperty - DHSStorageProperty structure

Returns:

DHSStorageProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateDHSStorage

```
public com.ge.ip.hds.historian.DataContracts.DHSStorageProperty
UpdateDHSStorage(java.lang.String effectiveUserId,
java.lang.String dHSStorageName,
com.ge.ip.hds.historian.DataContracts.DHSStorageProperty dHSStorageProperty)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for updating DHSStorage

Parameters:

effectiveUserId - User name against which operation will be performed.
dHSStorageName - Name of DHS Storage
dHSStorageProperty - DHSStorageProperty structure

Returns:

DHSStorageProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateDHSStorage

```
public com.ge.ip.hds.historian.DataContracts.DHSStorageProperty
UpdateDHSStorage(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String dHSStorageName,
com.ge.ip.hds.historian.DataContracts.DHSStorageProperty dHSStorageProperty)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for updating DHSStorage

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
dHSStorageName - Name of DHS Storage
dHSStorageProperty - DHSStorageProperty structure

Returns:

DHSStorageProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class DHSStorageServiceImpl

```
java.lang.Object
|
+--com.ge.ip.hds.historian.API.DHSStorageServiceImpl
```

All Implemented Interfaces:

[DHSStorageService](#)

< [Constructors](#) > < [Methods](#) >

```
public class DHSStorageServiceImpl
extends java.lang.Object
implements DHSStorageService
```

Constructors

DHSStorageServiceImpl

```
public DHSStorageServiceImpl()
```

Methods

AddDHSStorage

```
public com.ge.ip.hds.historian.DataContracts.DHSStorageProperty  
AddDHSStorage(com.ge.ip.hds.historian.DataContracts.DHSStorageProperty  
dHSStorageProperty)
```

AddDHSStorage

```
public com.ge.ip.hds.historian.DataContracts.DHSStorageProperty  
AddDHSStorage(java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.DHSStorageProperty dHSStorageProperty)
```

AddDHSStorage

```
public com.ge.ip.hds.historian.DataContracts.DHSStorageProperty  
AddDHSStorage(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.DHSStorageProperty dHSStorageProperty)
```

DeleteDHSStorage

```
public boolean DeleteDHSStorage(java.lang.String dHSStorageName)
```

DeleteDHSStorage

```
public boolean DeleteDHSStorage(java.lang.String effectiveUserId,  
java.lang.String dHSStorageName)
```

DeleteDHSStorage

```
public boolean DeleteDHSStorage(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String dHSStorageName)
```

GetDHSStorage

```
public com.ge.ip.hds.historian.DataContracts.DHSStorageProperty  
GetDHSStorage(java.lang.String dHSStorageName)
```

GetDHSStorage

```
public com.ge.ip.hds.historian.DataContracts.DHSStorageProperty  
GetDHSStorage(java.lang.String effectiveUserId,  
java.lang.String dHSStorageName)
```

GetDHSStorage

```
public com.ge.ip.hds.historian.DataContracts.DHSStorageProperty  
GetDHSStorage(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String dHSStorageName)
```

GetDHSStorages

```
public java.util.List GetDHSStorages(java.lang.String dHSStorageNameMask)
```

GetDHSStorages

```
public java.util.List GetDHSStorages(java.lang.String effectiveUserId,  
java.lang.String dHSStorageNameMask)
```

GetDHSStorages

```
public java.util.List GetDHSStorages(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String dHSStorageNameMask)
```

UpdateDHSStorage

```
public com.ge.ip.hds.historian.DataContracts.DHSStorageProperty
UpdateDHSStorage(java.lang.String dHSStorageName,
com.ge.ip.hds.historian.DataContracts.DHSStorageProperty dHSStorageProperty)
```

UpdateDHSStorage

```
public com.ge.ip.hds.historian.DataContracts.DHSStorageProperty
UpdateDHSStorage(java.lang.String effectiveUserId,
java.lang.String dHSStorageName,
com.ge.ip.hds.historian.DataContracts.DHSStorageProperty dHSStorageProperty)
```

UpdateDHSStorage

```
public com.ge.ip.hds.historian.DataContracts.DHSStorageProperty
UpdateDHSStorage(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String dHSStorageName,
com.ge.ip.hds.historian.DataContracts.DHSStorageProperty dHSStorageProperty)
```

com.ge.ip.hds.historian.API

Interface DataService

< [Methods](#) >

public interface **DataService**

DataService

The DataService Class implements interfaces for retrieving data.

Methods

CreateTagSample

```
public void CreateTagSample(java.lang.String historianServerName,  
                             java.lang.String effectiveUserId,  
                             java.lang.String tagName,  
                             java.util.List dataSample)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for adding data sample in Historian Server for a tag.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
tagName - Name of tag.
dataSample - List of DataSample

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

CreateTagSample

```
public void CreateTagSample(java.lang.String effectiveUserId,  
                             java.lang.String tagName,  
                             java.util.List dataSample)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for adding data sample in Historian Server for a tag.

Parameters:

effectiveUserId - User name against which operation will be performed.
tagName - Name of tag.
dataSample - List of DataSample

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

CreateTagSample

```
public void CreateTagSample(java.lang.String tagName,  
                             java.util.List dataSample)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for adding data sample in Historian Server for a tag.

Parameters:

tagName - Name of tag.
dataSample - List of DataSample

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

CreateTagsSample

```
public void CreateTagsSample(java.lang.String historianServerName,  
                               java.lang.String effectiveUserId,  
                               java.util.List tagsDataSample)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for adding data sample in Historian Server for a list of tag.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
tagsDataSample - List of TagWithData.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

CreateTagsSample

```
public void CreateTagsSample(java.lang.String effectiveUserId,  
                               java.util.List tagsDataSample)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for adding data sample in Historian Server for a list of tag.

Parameters:

effectiveUserId - User name against which operation will be performed.
tagsDataSample - List of TagWithData.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

CreateTagsSample

```
public void CreateTagsSample(java.util.List tagsDataSample)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for adding data sample in Historian Server for a list of tag.

Parameters:

tagsDataSample - List of TagWithData.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCalculatedData

```
public java.util.List GetCalculatedData(java.lang.String historianServerName,  
                                         java.lang.String effectiveUserId,  
                                         java.util.List tagNames,  
                                         java.util.Date startTime,  
                                         java.util.Date endTime,  
                                         int count,  
com.ge.ip.hds.historian.DataContracts.CalculationModeType calculationMode,  
                                         long intervalMs)  
                                         throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving Calculated data values for a time range or on basis of count for list of tags. Historian Archive is capable of doing some mathematical calculations on data while querying. For example Average, StandardDeviation, Total etc. If the count is not zero, the service returns the number of raw samples beginning from the start time. If the count is zero, the services uses the interval, start time, and end time to calculate the required sample number.

Parameters:

- historianServerName - Name of target historian server.
- effectiveUserId - User name against which operation will be performed.
- tagNames - List of tag names.
- startTime - start time.
- endTime - end time.
- count - Number of data samples within each calculation interval.
- calculationMode - Type of calculation.
- intervalMs - Interval in milliseconds for which data samples are requested (Frequency).

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCalculatedData

```
public java.util.List GetCalculatedData(java.lang.String effectiveUserId,  
                                         java.util.List tagNames,  
                                         java.util.Date startTime,  
                                         java.util.Date endTime,  
                                         int count,  
com.ge.ip.hds.historian.DataContracts.CalculationModeType calculationMode,  
                                         long intervalMs)  
                                         throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving Calculated data values for a time range or on basis of count for list of tags. Historian Archive is capable of doing some mathematical calculations on data while querying. For example Average, StandardDeviation, Total etc. If the count is not zero, the service returns the number of raw samples beginning from the start time. If the count is zero, the services uses the interval, start time, and end time to calculate the required sample number.

Parameters:

- effectiveUserId - User name against which operation will be performed.
- tagNames - List of tag names.
- startTime - start time.
- endTime - end time.
- count - Number of data samples within each calculation interval.
- calculationMode - Type of calculation.
- intervalMs - Interval in milliseconds for which data samples are requested (Frequency).

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCalculatedData

```
public java.util.List GetCalculatedData(java.util.List tagNames,  
                                         java.util.Date startTime,  
                                         java.util.Date endTime,  
                                         int count,  
com.ge.ip.hds.historian.DataContracts.CalculationModeType calculationMode,  
                                         long intervalMs)  
                                         throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving Calculated data values for a time range or on basis of count for list of tags. Historian Archive is capable of doing some mathematical calculations on data while querying. For example Average, StandardDeviation, Total etc. If the count is not zero, the service returns the number of raw samples beginning from the start time. If the count is zero, the services uses the interval, start time, and end time to calculate the required sample number.

Parameters:

tagNames - List of tag names.
startTime - start time.
endTime - end time.
count - Number of data samples within each calculation interval.
calculationMode - Type of calculation.
intervalMs - Interval in milliseconds for which data samples are requested (Frequency).

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCurrentData

```
public java.util.List GetCurrentData(java.lang.String historianServerName,  
                                       java.lang.String effectiveUserId,  
                                       java.util.List tagNames)  
                                       throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving current values for a list of tags.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
tagNames - List of tag names.

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCurrentData

```
public java.util.List GetCurrentData(java.lang.String effectiveUserId,  
                                       java.util.List tagNames)  
                                       throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving current values for a list of tags.

Parameters:

effectiveUserId - User name against which operation will be performed.
tagNames - List of tag names.

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCurrentData

```
public java.util.List GetCurrentData(java.util.List tagNames)  
                                       throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving current values for a list of tags.

Parameters:

tagNames - List of tag names.

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetInterpolatedData

```
public java.util.List GetInterpolatedData(java.lang.String
historianServerName,
                                           java.lang.String effectiveUserId,
                                           java.util.List tagNames,
                                           java.util.Date startTime,
                                           java.util.Date endTime,
                                           long intervalMs)
                                           throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving interpolated data values for a time range for list of tags. Historian Archive is capable of doing data interpolation while querying data. If the start time equals the end time, the request returns one sample.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
tagNames - List of tag names.
startTime - start time.
endTime - end time.
intervalMs - Interval in milliseconds for which data samples are requested (Frequency).

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetInterpolatedData

```
public java.util.List GetInterpolatedData(java.lang.String effectiveUserId,
                                           java.util.List tagNames,
                                           java.util.Date startTime,
                                           java.util.Date endTime,
                                           long intervalMs)
                                           throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving interpolated data values for a time range for list of tags. Historian Archive is capable of doing data interpolation while querying data. If the start time equals the end time, the request returns one sample.

Parameters:

effectiveUserId - User name against which operation will be performed.
tagNames - List of tag names.
startTime - start time.
endTime - end time.
intervalMs - Interval in milliseconds for which data samples are requested (Frequency).

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetInterpolatedData

```
public java.util.List GetInterpolatedData(java.util.List tagNames,  
                                           java.util.Date startTime,  
                                           java.util.Date endTime,  
                                           long intervalMs)  
                                           throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving interpolated data values for a time range for list of tags. Historian Archive is capable of doing data interpolation while querying data. If the start time equals the end time, the request returns one sample.

Parameters:

tagNames - List of tag names.
startTime - start time.
endTime - end time.
intervalMs - Interval in milliseconds for which data samples are requested (Frequency).

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetRawDataByNumber

```
public java.util.List GetRawDataByNumber(java.lang.String historianServerName,  
                                           java.lang.String effectiveUserId,  
                                           java.util.List tagNames,  
                                           java.util.Date startTime,  
                                           int count,  
                                           boolean backwardTimeOrder)  
                                           throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving specified number of raw data values beginning from a specified time for list of tags.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
tagNames - List of tag names.
startTime - start time.
count - Requested number of raw data values. If the count is not zero, then the API service returns the number of raw samples taken beginning from the start time. If the count is zero, then the service returns the raw samples taken between the start time and the end time.
backwardTimeOrder - Moves in forward direction from start time if its "false" , else in backward direction.

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetRawDataByNumber

```
public java.util.List GetRawDataByNumber(java.lang.String effectiveUserId,  
                                           java.util.List tagNames,  
                                           java.util.Date startTime,  
                                           int count,  
                                           boolean backwardTimeOrder)  
                                           throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving specified number of raw data values beginning from a specified time for list of tags.

Parameters:

effectiveUserId - User name against which operation will be performed.

tagNames - List of tag names.

startTime - start time.

count - Requested number of raw data values. If the count is not zero, then the API service returns the number of raw samples taken beginning from the start time. If the count is zero, then the service returns the raw samples taken between the start time and the end time.

backwardTimeOrder - Moves in forward direction from start time if its "false" , else in backward direction.

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetRawDataByNumber

```
public java.util.List GetRawDataByNumber(java.util.List tagNames,  
                                           java.util.Date startTime,  
                                           int count,  
                                           boolean backwardTimeOrder)  
                                           throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving specified number of raw data values beginning from a specified time for list of tags.

Parameters:

tagNames - List of tag names.

startTime - start time.

count - Requested number of raw data values. If the count is not zero, then the API service returns the number of raw samples taken beginning from the start time. If the count is zero, then the service returns the raw samples taken between the start time and the end time.

backwardTimeOrder - Moves in forward direction from start time if its "false" , else in backward direction.

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetRawDataByTime

```
public java.util.List GetRawDataByTime(java.lang.String historianServerName,  
                                         java.lang.String effectiveUserId,  
                                         java.util.List tagNames,  
                                         java.util.Date startTime,  
                                         java.util.Date endTime)  
                                         throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving raw data values for a time range for list of tags.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
tagNames - List of tag names.
startTime - start time.
endTime - end time.

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetRawDataByTime

```
public java.util.List GetRawDataByTime(java.lang.String effectiveUserId,  
                                         java.util.List tagNames,  
                                         java.util.Date startTime,  
                                         java.util.Date endTime)  
                                         throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving raw data values for a time range for list of tags.

Parameters:

effectiveUserId - User name against which operation will be performed.
tagNames - List of tag names.
startTime - start time.
endTime - end time.

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetRawDataByTime

```
public java.util.List GetRawDataByTime(java.util.List tagNames,  
                                         java.util.Date startTime,  
                                         java.util.Date endTime)  
                                         throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving raw data values for a time range for list of tags.

Parameters:

tagNames - List of tag names.
startTime - start time.
endTime - end time.

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetSampledData

```
public java.util.List GetSampledData(java.lang.String historianServerName,  
                                       java.lang.String effectiveUserId,  
                                       java.util.List tagNames,  
                                       java.util.Date startTime,  
                                       java.util.Date endTime,  
                                       int count,  
                                       com.ge.ip.hds.historian.DataContracts.SamplingModeType samplingMode,  
                                       long intervalMs,  
                                       boolean backwardTimeOrder,  
                                       java.lang.String filterExpression,  
                                       com.ge.ip.hds.historian.DataContracts.CalculationModeType calculationMode,  
                                       long queryModifier,  
                                       com.ge.ip.hds.historian.DataContracts.FilterModeType filterMode)  
                                       throws  
                                       com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving sampled data values for a time range or on basis of count for list of tags. Historian Archive is capable of doing some data samplings while querying. For example Interpolated, Lab, Trend, RawByTime etc. If the count is not zero, the service returns the number of raw samples beginning from the start time. If the count is zero, the services uses the interval, start time, and end time to calculate the required sample number.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
tagNames - List of tag names.
startTime - start time.
endTime - end time.
count - Number of data samples within each calculation interval.
samplingMode - Type of Sampling.
intervalMs - Interval in milliseconds for which data samples are requested (Frequency).
backwardTimeOrder - Moves in forward direction from start time if its "false" , else in backward direction.
filterExpression - An expression which includes one or more filter conditions. AND Condition, OR Condition, Combination of both AND and OR
calculationMode - Type of calculation.
queryModifier - Query Modifiers like "ONLYGOOD", "HONORENDTIME" etc.
filterMode - Filter mode type.

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetSampledData

```
public java.util.List GetSampledData(java.lang.String effectiveUserId,  
                                       java.util.List tagNames,  
                                       java.util.Date startTime,  
                                       java.util.Date endTime,  
                                       int count,  
com.ge.ip.hds.historian.DataContracts.SamplingModeType samplingMode,  
                                       long intervalMs,  
                                       boolean backwardTimeOrder,  
                                       java.lang.String filterExpression,  
com.ge.ip.hds.historian.DataContracts.CalculationModeType calculationMode,  
                                       long queryModifier,  
com.ge.ip.hds.historian.DataContracts.FilterModeType filterMode)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving sampled data values for a time range or on basis of count for list of tags. Historian Archive is capable of doing some data samplings while querying. For example Interpolated, Lab, Trend, RawByTime etc. If the count is not zero, the service returns the number of raw samples beginning from the start time. If the count is zero, the services uses the interval, start time, and end time to calculate the required sample number.

Parameters:

- effectiveUserId - User name against which operation will be performed.
- tagNames - List of tag names.
- startTime - start time.
- endTime - end time.
- count - Number of data samples within each calculation interval.
- samplingMode - Type of Sampling.
- intervalMs - Interval in milliseconds for which data samples are requested (Frequency).
- backwardTimeOrder - Moves in forward direction from start time if its "false" , else in backward direction.
- filterExpression - An expression which includes one or more filter conditions. AND Condition, OR Condition, Combination of both AND and OR
- calculationMode - Type of calculation.
- queryModifier - Query Modifiers like "ONLYGOOD", "HONORENDTIME" etc.
- filterMode - Filter mode type.

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetSampledData

```
public java.util.List GetSampledData(java.util.List tagNames,  
                                       java.util.Date startTime,  
                                       java.util.Date endTime,  
                                       int count,  
com.ge.ip.hds.historian.DataContracts.SamplingModeType samplingMode,  
                                       long intervalMs,  
                                       boolean backwardTimeOrder,  
                                       java.lang.String filterExpression,  
com.ge.ip.hds.historian.DataContracts.CalculationModeType calculationMode,  
                                       long queryModifier,  
com.ge.ip.hds.historian.DataContracts.FilterModeType filterMode)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving sampled data values for a time range or on basis of count for list of tags. Historian Archive is capable of doing some data samplings while querying. For example Interpolated, Lab, Trend, RawByTime etc. If the count is not zero, the service returns the number of raw samples beginning from the start time. If the count is zero, the services uses the interval, start time, and end time to calculate the required sample number.

Parameters:

- tagNames - List of tag names.
- startTime - start time.
- endTime - end time.
- count - Number of data samples within each calculation interval.
- samplingMode - Type of Sampling.
- intervalMs - Interval in milliseconds for which data samples are requested (Frequency).
- backwardTimeOrder - Moves in forward direction from start time if its "false" , else in backward direction.
- filterExpression - An expression which includes one or more filter conditions. AND Condition, OR Condition, Combination of both AND and OR
- calculationMode - Type of calculation.
- queryModifier - Query Modifiers like "ONLYGOOD", "HONORENDTIME" etc.
- filterMode - Filter mode type.

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetTrendData

```
public java.util.List GetTrendData(java.lang.String historianServerName,  
                                     java.lang.String effectiveUserId,  
                                     java.util.List tagNames,  
                                     java.util.Date startTime,  
                                     java.util.Date endTime,  
                                     int count,  
com.ge.ip.hds.historian.DataContracts.SamplingModeType samplingMode,  
                                     long intervalMs,  
                                     boolean backwardTimeOrder,  
                                     java.lang.String filterExpression,  
com.ge.ip.hds.historian.DataContracts.CalculationModeType calculationMode,  
                                     long queryModifier,  
com.ge.ip.hds.historian.DataContracts.FilterModeType filterMode,  
                                     java.lang.String statisticsItemFilter)  
                                     throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving trend data values for a time range or on basis of count for list of tags. Historian Archive is capable of doing some data samplings while querying. For example Interpolated, Lab, Trend, RawByTime etc. If the count is not zero, the service returns the number of raw samples beginning from the start time. If the count is zero, the services uses the interval, start time, and end time to calculate the required sample number.

Parameters:

- historianServerName - Name of target historian server.
- effectiveUserId - User name against which operation will be performed.
- tagNames - List of tag names.
- startTime - start time.
- endTime - end time.
- count - Number of data samples within each calculation interval.
- samplingMode - Type of Sampling.
- intervalMs - Interval in milliseconds for which data samples are requested (Frequency).
- backwardTimeOrder - Moves in forward direction from start time if its "false" , else in backward direction.
- filterExpression - An expression which includes one or more filter conditions. AND Condition, OR Condition, Combination of both AND and OR
- calculationMode - Type of calculation.
- queryModifier - Query Modifiers like "ONLYGOOD", "HONORENDTIME" etc.
- filterMode - Filter mode type.
- statisticsItemFilter -

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetTrendData

```
public java.util.List GetTrendData(java.lang.String effectiveUserId,  
                                     java.util.List tagNames,  
                                     java.util.Date startTime,  
                                     java.util.Date endTime,  
                                     int count,  
com.ge.ip.hds.historian.DataContracts.SamplingModeType samplingMode,  
                                     long intervalMs,  
                                     boolean backwardTimeOrder,  
                                     java.lang.String filterExpression,  
com.ge.ip.hds.historian.DataContracts.CalculationModeType calculationMode,  
                                     long queryModifier,  
com.ge.ip.hds.historian.DataContracts.FilterModeType filterMode,  
                                     java.lang.String statisticsItemFilter)  
                                     throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving trend data values for a time range or on basis of count for list of tags. Historian Archive is capable of doing some data samplings while querying. For example Interpolated, Lab, Trend, RawByTime etc. If the count is not zero, the service returns the number of raw samples beginning from the start time. If the count is zero, the services uses the interval, start time, and end time to calculate the required sample number.

Parameters:

- effectiveUserId - User name against which operation will be performed.
- tagNames - List of tag names.
- startTime - start time.
- endTime - end time.
- count - Number of data samples within each calculation interval.
- samplingMode - Type of Sampling.
- intervalMs - Interval in milliseconds for which data samples are requested (Frequency).
- backwardTimeOrder - Moves in forward direction from start time if its "false" , else in backward direction.
- filterExpression - An expression which includes one or more filter conditions. AND Condition, OR Condition, Combination of both AND and OR
- calculationMode - Type of calculation.
- queryModifier - Query Modifiers like "ONLYGOOD", "HONORENDTIME" etc.
- filterMode - Filter mode type.
- statisticsItemFilter -

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetTrendData

```
public java.util.List GetTrendData(java.util.List tagNames,  
                                     java.util.Date startTime,  
                                     java.util.Date endTime,  
                                     int count,  
com.ge.ip.hds.historian.DataContracts.SamplingModeType samplingMode,  
                                     long intervalMs,  
                                     boolean backwardTimeOrder,  
                                     java.lang.String filterExpression,  
com.ge.ip.hds.historian.DataContracts.CalculationModeType calculationMode,  
                                     long queryModifier,  
com.ge.ip.hds.historian.DataContracts.FilterModeType filterMode,  
                                     java.lang.String statisticsItemFilter)  
                                     throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving trend data values for a time range or on basis of count for list of tags. Historian Archive is capable of doing some data samplings while querying. For example Interpolated, Lab, Trend, RawByTime etc. If the count is not zero, the service returns the number of raw samples beginning from the start time. If the count is zero, the services uses the interval, start time, and end time to calculate the required sample number.

Parameters:

- tagNames - List of tag names.
- startTime - start time.
- endTime - end time.
- count - Number of data samples within each calculation interval.
- samplingMode - Type of Sampling.
- intervalMs - Interval in milliseconds for which data samples are requested (Frequency).
- backwardTimeOrder - Moves in forward direction from start time if its "false" , else in backward direction.
- filterExpression - An expression which includes one or more filter conditions. AND Condition, OR Condition, Combination of both AND and OR
- calculationMode - Type of calculation.
- queryModifier - Query Modifiers like "ONLYGOOD", "HONORENDTIME" etc.
- filterMode - Filter mode type.
- statisticsItemFilter -

Returns:

List of TagSamples.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetRawDataByTime

```
public java.util.List GetRawDataByTime(java.lang.String historianServerName,  
                                         java.lang.String effectiveUserId,  
                                         java.util.List tagNames,  
                                         java.util.Date startTime,  
                                         java.util.Date endTime)
```

GetRawDataByTime

```
public java.util.List GetRawDataByTime(java.lang.String effectiveUserId,  
                                         java.util.List tagNames,  
                                         java.util.Date startTime,  
                                         java.util.Date endTime)
```

GetRawDataByTime

```
public java.util.List GetRawDataByTime(java.util.List tagNames,  
                                         java.util.Date startTime,  
                                         java.util.Date endTime)
```

GetSampledData

```
public java.util.List GetSampledData(java.lang.String historianServerName,  
                                       java.lang.String effectiveUserId,  
                                       java.util.List tagNames,  
                                       java.util.Date startTime,  
                                       java.util.Date endTime,  
                                       int count,  
                                       com.ge.ip.hds.historian.DataContracts.SamplingModeType samplingMode,  
                                       long intervalMs,  
                                       boolean backwardTimeOrder,  
                                       java.lang.String filterExpression,  
                                       com.ge.ip.hds.historian.DataContracts.CalculationModeType calculationMode,  
                                       long queryModifier,  
                                       com.ge.ip.hds.historian.DataContracts.FilterModeType filterMode)
```

GetSampledData

```
public java.util.List GetSampledData(java.lang.String effectiveUserId,  
                                       java.util.List tagNames,  
                                       java.util.Date startTime,  
                                       java.util.Date endTime,  
                                       int count,  
com.ge.ip.hds.historian.DataContracts.SamplingModeType samplingMode,  
                                       long intervalMs,  
                                       boolean backwardTimeOrder,  
                                       java.lang.String filterExpression,  
com.ge.ip.hds.historian.DataContracts.CalculationModeType calculationMode,  
                                       long queryModifier,  
com.ge.ip.hds.historian.DataContracts.FilterModeType filterMode)
```

GetSampledData

```
public java.util.List GetSampledData(java.util.List tagNames,  
                                       java.util.Date startTime,  
                                       java.util.Date endTime,  
                                       int count,  
com.ge.ip.hds.historian.DataContracts.SamplingModeType samplingMode,  
                                       long intervalMs,  
                                       boolean backwardTimeOrder,  
                                       java.lang.String filterExpression,  
com.ge.ip.hds.historian.DataContracts.CalculationModeType calculationMode,  
                                       long queryModifier,  
com.ge.ip.hds.historian.DataContracts.FilterModeType filterMode)
```

GetTrendData

```
public java.util.List GetTrendData(java.lang.String historianServerName,  
                                       java.lang.String effectiveUserId,  
                                       java.util.List tagNames,  
                                       java.util.Date startTime,  
                                       java.util.Date endTime,  
                                       int count,  
com.ge.ip.hds.historian.DataContracts.SamplingModeType samplingMode,  
                                       long intervalMs,  
                                       boolean backwardTimeOrder,  
                                       java.lang.String filterExpression,  
com.ge.ip.hds.historian.DataContracts.CalculationModeType calculationMode,  
                                       long queryModifier,  
com.ge.ip.hds.historian.DataContracts.FilterModeType filterMode,  
                                       java.lang.String statisticsItemFilter)
```

GetTrendData

```
public java.util.List GetTrendData(java.lang.String effectiveUserId,  
                                     java.util.List tagNames,  
                                     java.util.Date startTime,  
                                     java.util.Date endTime,  
                                     int count,  
com.ge.ip.hds.historian.DataContracts.SamplingModeType samplingMode,  
                                     long intervalMs,  
                                     boolean backwardTimeOrder,  
                                     java.lang.String filterExpression,  
com.ge.ip.hds.historian.DataContracts.CalculationModeType calculationMode,  
                                     long queryModifier,  
com.ge.ip.hds.historian.DataContracts.FilterModeType filterMode,  
                                     java.lang.String statisticsItemFilter)
```

GetTrendData

```
public java.util.List GetTrendData(java.util.List tagNames,  
                                     java.util.Date startTime,  
                                     java.util.Date endTime,  
                                     int count,  
com.ge.ip.hds.historian.DataContracts.SamplingModeType samplingMode,  
                                     long intervalMs,  
                                     boolean backwardTimeOrder,  
                                     java.lang.String filterExpression,  
com.ge.ip.hds.historian.DataContracts.CalculationModeType calculationMode,  
                                     long queryModifier,  
com.ge.ip.hds.historian.DataContracts.FilterModeType filterMode,  
                                     java.lang.String statisticsItemFilter)
```

TagNameAggregate

```
public static java.lang.String TagNameAggregate(java.util.List tagNames)
```

com.ge.ip.hds.historian.API

Interface DataStoreService

< [Methods](#) >

public interface **DataStoreService**

DataStoreService

The DataStoreService Class implements interfaces for dealing with Historian DataStore operations.

Methods

AddDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore  
AddDataStore(com.ge.ip.hds.historian.DataContracts.DataStore dataStore)  
throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for adding a data store.

Parameters:

dataStore - DataStore structure.

Returns:

DataStore structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore  
AddDataStore(java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.DataStore dataStore)  
throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for adding a data store.

Parameters:

effectiveUserId - User name against which operation will be performed.

dataStore - DataStore structure.

Returns:

DataStore structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore
AddDataStore(java.lang.String historianServerName,
java.lang.String effectiveUserId,
com.ge.ip.hds.historian.DataContracts.DataStore dataStore)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for adding a data store.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
dataStore - DataStore structure.

Returns:

DataStore structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteDataStore

```
public boolean DeleteDataStore(java.lang.String dataStoreName)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting a data store.

Parameters:

dataStoreName - Name of data store.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteDataStore

```
public boolean DeleteDataStore(java.lang.String effectiveUserId,  
                               java.lang.String dataStoreName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting a data store.

Parameters:

effectiveUserId - User name against which operation will be performed.
dataStoreName - Name of data store.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteDataStore

```
public boolean DeleteDataStore(java.lang.String historianServerName,  
                               java.lang.String effectiveUserId,  
                               java.lang.String dataStoreName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting a data store.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
dataStoreName - Name of data store.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore  
GetDataStore(java.lang.String dataStoreName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving DataStore structure for a data store.

Parameters:

dataStoreName - Name of data store.

Returns:

DataStore structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore  
GetDataStore(java.lang.String effectiveUserId,  
java.lang.String dataStoreName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving DataStore structure for a data store.

Parameters:

effectiveUserId - User name against which operation will be performed.
dataStoreName - Name of data store.

Returns:

DataStore structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore  
GetDataStore(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String dataStoreName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving DataStore structure for a data store.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
dataStoreName - Name of data store.

Returns:

DataStore structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDataStores

```
public java.util.List GetDataStores(java.lang.String dataStoreNameMask)  
throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of data stores in Historian Server.

Parameters:

dataStoreNameMask - This is a name mask pattern used to retrieve list of data stores which matches pattern. The mask can include wildcards, such as asterisks (*).

Returns:

List of DataStore.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDataStores

```
public java.util.List GetDataStores(java.lang.String effectiveUserId,  
                                     java.lang.String dataStoreNameMask)  
                                     throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of data stores in Historian Server.

Parameters:

effectiveUserId - User name against which operation will be performed.
dataStoreNameMask - This is a name mask pattern used to retrieve list of data stores which matches pattern. The mask can include wildcards, such as asterisks (*).

Returns:

List of DataStore.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDataStores

```
public java.util.List GetDataStores(java.lang.String historianServerName,  
                                     java.lang.String effectiveUserId,  
                                     java.lang.String dataStoreNameMask)  
                                     throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of data stores in Historian Server.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
dataStoreNameMask - This is a name mask pattern used to retrieve list of data stores which matches pattern. The mask can include wildcards, such as asterisks (*).

Returns:

List of DataStore.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RenameDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore  
RenameDataStore( java.lang.String oldName,  
java.lang.String newName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for renaming a data store.

Parameters:

oldName - old data store name.
newName - new data store name.

Returns:

DataStore structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RenameDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore  
RenameDataStore( java.lang.String effectiveUserId,  
java.lang.String oldName,  
java.lang.String newName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for renaming a data store.

Parameters:

effectiveUserId - User name against which operation will be performed.
oldName - old data store name.
newName - new data store name.

Returns:

DataStore structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RenameDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore  
RenameDataStore( java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String oldName,  
java.lang.String newName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for renaming a data store.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
oldName - old data store name.
newName - new data store name.

Returns:

DataStore structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore  
UpdateDataStore( java.lang.String dataStoreName,  
com.ge.ip.hds.historian.DataContracts.DataStore dataStore)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for updating a data store property.

Parameters:

dataStoreName - Name of data store.
dataStore - DataStore structure.

Returns:

DataStore structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore
UpdateDataStore( java.lang.String effectiveUserId,
java.lang.String dataStoreName,
com.ge.ip.hds.historian.DataContracts.DataStore dataStore)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for updating a data store property.

Parameters:

effectiveUserId - User name against which operation will be performed.
dataStoreName - Name of data store.
dataStore - DataStore structure.

Returns:

DataStore structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore
UpdateDataStore( java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String dataStoreName,
com.ge.ip.hds.historian.DataContracts.DataStore dataStore)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for updating a data store property.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
dataStoreName - Name of data store.
dataStore - DataStore structure.

Returns:

DataStore structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class DataStoreServiceImpl

```
java.lang.Object
|
+--com.ge.ip.hds.historian.API.DataStoreServiceImpl
```

All Implemented Interfaces:
[DataStoreService](#)

< [Constructors](#) > < [Methods](#) >

```
public class DataStoreServiceImpl
extends java.lang.Object
implements DataStoreService
```

Constructors

DataStoreServiceImpl

```
public DataStoreServiceImpl()
```

Methods

AddDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore
AddDataStore(com.ge.ip.hds.historian.DataContracts.DataStore dataStore)
```

AddDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore
AddDataStore(java.lang.String effectiveUserId,
com.ge.ip.hds.historian.DataContracts.DataStore dataStore)
```

AddDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore
AddDataStore(java.lang.String historianServerName,
java.lang.String effectiveUserId,
com.ge.ip.hds.historian.DataContracts.DataStore dataStore)
```

DeleteDataStore

```
public boolean DeleteDataStore(java.lang.String dataStoreName)
```

DeleteDataStore

```
public boolean DeleteDataStore(java.lang.String effectiveUserId,
java.lang.String dataStoreName)
```

DeleteDataStore

```
public boolean DeleteDataStore(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String dataStoreName)
```

GetDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore
GetDataStore(java.lang.String dataStoreName)
```

GetDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore
GetDataStore(java.lang.String effectiveUserId,
java.lang.String dataStoreName)
```

GetDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore  
GetDataStore(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String dataStoreName)
```

GetDataStores

```
public java.util.List GetDataStores(java.lang.String dataStoreNameMask)
```

GetDataStores

```
public java.util.List GetDataStores(java.lang.String effectiveUserId,  
java.lang.String dataStoreNameMask)
```

GetDataStores

```
public java.util.List GetDataStores(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String dataStoreNameMask)
```

RenameDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore  
RenameDataStore(java.lang.String oldName,  
java.lang.String newName)
```

RenameDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore  
RenameDataStore(java.lang.String effectiveUserId,  
java.lang.String oldName,  
java.lang.String newName)
```

RenameDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore
RenameDataStore( java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String oldName,
java.lang.String newName)
```

UpdateDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore
UpdateDataStore( java.lang.String dataStoreName,
com.ge.ip.hds.historian.DataContracts.DataStore dataStore)
```

UpdateDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore
UpdateDataStore( java.lang.String effectiveUserId,
java.lang.String dataStoreName,
com.ge.ip.hds.historian.DataContracts.DataStore dataStore)
```

UpdateDataStore

```
public com.ge.ip.hds.historian.DataContracts.DataStore
UpdateDataStore( java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String dataStoreName,
com.ge.ip.hds.historian.DataContracts.DataStore dataStore)
```

com.ge.ip.hds.historian.API

Interface EnumeratedSetService

< [Methods](#) >

public interface **EnumeratedSetService**

EnumeratedSetService

The EnumeratedSetService Class implements interfaces for dealing with EnumeratedSet Types.

Methods

AddEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
AddEnumeratedSet(com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
enumeratedSetProperty)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

throws

This method is for adding a EnumeratedSet.

Parameters:

enumeratedSetProperty - EnumeratedSetProperty structure.

Returns:

EnumeratedSetProperty structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
AddEnumeratedSet(java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
enumeratedSetProperty)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

throws

This method is for adding a EnumeratedSet.

Parameters:

effectiveUserId - User name against which operation will be performed.
enumeratedSetProperty - EnumeratedSetProperty structure.

Returns:

EnumeratedSetProperty structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
AddEnumeratedSet(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
enumeratedSetProperty)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for adding a EnumeratedSet.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
enumeratedSetProperty - EnumeratedSetProperty structure.

Returns:

EnumeratedSetProperty structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteEnumeratedSet

```
public boolean DeleteEnumeratedSet(java.lang.String enumeratedSetName)  
throws
```

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting a EnumeratedSet.

Parameters:

enumeratedSetName - Name of EnumeratedSet.

Returns:

true if success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteEnumeratedSet

```
public boolean DeleteEnumeratedSet(java.lang.String effectiveUserId,  
                                   java.lang.String enumeratedSetName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting a EnumeratedSet.

Parameters:

effectiveUserId - User name against which operation will be performed.
enumeratedSetName - Name of EnumeratedSet.

Returns:

true if success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteEnumeratedSet

```
public boolean DeleteEnumeratedSet(java.lang.String historianServerName,  
                                   java.lang.String effectiveUserId,  
                                   java.lang.String enumeratedSetName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting a EnumeratedSet.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
enumeratedSetName - Name of EnumeratedSet.

Returns:

true if success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
GetEnumeratedSet(java.lang.String enumeratedSetName) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the structure of EnumeratedSet.

Parameters:

enumeratedSetName - Name of EnumeratedSet.

Returns:

EnumeratedSetProperty structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
GetEnumeratedSet(java.lang.String effectiveUserId,  
java.lang.String enumeratedSetName) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the structure of EnumeratedSet.

Parameters:

effectiveUserId - User name against which operation will be performed.
enumeratedSetName - Name of EnumeratedSet.

Returns:

EnumeratedSetProperty structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
GetEnumeratedSet(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String enumeratedSetName) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the structure of EnumeratedSet.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
enumeratedSetName - Name of EnumeratedSet.

Returns:

EnumeratedSetProperty structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryEnumeratedSet

```
public java.util.List QueryEnumeratedSet(java.lang.String  
enumeratedSetNameMask,  
int maxNumber) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of EnumeratedSet which matches EnumeratedSet name mask..

Parameters:

enumeratedSetNameMask - This is a name mask pattern used to retrieve list of EnumeratedSet which matches pattern. The mask can include wildcards, such as asterisks (*).
maxNumber - Maximum number of EnumeratedSet request.

Returns:

List of EnumeratedSetPropertyBase.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryEnumeratedSet

```
public java.util.List QueryEnumeratedSet(java.lang.String effectiveUserId,  
                                             java.lang.String  
enumeratedSetNameMask,  
                                             int maxNumber)  
                                             throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of EnumeratedSet which matches EnumeratedSet name mask..

Parameters:

effectiveUserId - User name against which operation will be performed.
enumeratedSetNameMask - This is a name mask pattern used to retrieve list of EnumeratedSet which matches pattern. The mask can include wildcards, such as asterisks (*).
maxNumber - Maximum number of EnumeratedSet request.

Returns:

List of EnumeratedSetPropertyBase.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryEnumeratedSet

```
public java.util.List QueryEnumeratedSet(java.lang.String historianServerName,  
                                             java.lang.String effectiveUserId,  
                                             java.lang.String  
enumeratedSetNameMask,  
                                             int maxNumber)  
                                             throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of EnumeratedSet which matches EnumeratedSet name mask..

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
enumeratedSetNameMask - This is a name mask pattern used to retrieve list of EnumeratedSet which matches pattern. The mask can include wildcards, such as asterisks (*).
maxNumber - Maximum number of EnumeratedSet request.

Returns:

List of EnumeratedSetPropertyBase.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RenameEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
RenameEnumeratedSet(java.lang.String oldEnumeratedSetName,  
java.lang.String newEnumeratedSetName) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for renaming a EnumeratedSet.

Parameters:

oldEnumeratedSetName - Old Name of EnumeratedSet.
newEnumeratedSetName - New Name of EnumeratedSet.

Returns:

EnumeratedSetProperty structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RenameEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
RenameEnumeratedSet(java.lang.String effectiveUserId,  
java.lang.String oldEnumeratedSetName,  
java.lang.String newEnumeratedSetName) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for renaming a EnumeratedSet.

Parameters:

effectiveUserId - User name against which operation will be performed.
oldEnumeratedSetName - Old Name of EnumeratedSet.
newEnumeratedSetName - New Name of EnumeratedSet.

Returns:

EnumeratedSetProperty structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RenameEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
RenameEnumeratedSet( java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String oldEnumeratedSetName,  
java.lang.String newEnumeratedSetName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for renaming a EnumeratedSet.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
oldEnumeratedSetName - Old Name of EnumeratedSet.
newEnumeratedSetName - New Name of EnumeratedSet.

Returns:

EnumeratedSetProperty structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
UpdateEnumeratedSet( java.lang.String enumeratedSetName,  
com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
enumeratedSetProperty)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for updating a EnumeratedSet.

Parameters:

enumeratedSetName - Name of EnumeratedSet.
enumeratedSetProperty - EnumeratedSetProperty structure.

Returns:

EnumeratedSetProperty structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
UpdateEnumeratedSet(java.lang.String effectiveUserId,  
java.lang.String enumeratedSetName,  
com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
enumeratedSetProperty)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for updating a EnumeratedSet.

Parameters:

effectiveUserId - User name against which operation will be performed.
enumeratedSetName - Name of EnumeratedSet.
enumeratedSetProperty - EnumeratedSetProperty structure.

Returns:

EnumeratedSetProperty structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
UpdateEnumeratedSet(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String enumeratedSetName,  
com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
enumeratedSetProperty)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for updating a EnumeratedSet.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
enumeratedSetName - Name of EnumeratedSet.
enumeratedSetProperty - EnumeratedSetProperty structure.

Returns:

EnumeratedSetProperty structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class EnumeratedSetServiceImpl

```
java.lang.Object
|
+--com.ge.ip.hds.historian.API.EnumeratedSetServiceImpl
```

All Implemented Interfaces:

[EnumeratedSetService](#)

< [Constructors](#) > < [Methods](#) >

```
public class EnumeratedSetServiceImpl
extends java.lang.Object
implements EnumeratedSetService
```

Constructors

EnumeratedSetServiceImpl

```
public EnumeratedSetServiceImpl()
```

Methods

AddEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty
AddEnumeratedSet(com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty
enumeratedSetProperty)
```

AddEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty
AddEnumeratedSet(java.lang.String effectiveUserId,
com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty
enumeratedSetProperty)
```

AddEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
AddEnumeratedSet(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
enumeratedSetProperty)
```

DeleteEnumeratedSet

```
public boolean DeleteEnumeratedSet(java.lang.String enumeratedSetName)
```

DeleteEnumeratedSet

```
public boolean DeleteEnumeratedSet(java.lang.String effectiveUserId,  
java.lang.String enumeratedSetName)
```

DeleteEnumeratedSet

```
public boolean DeleteEnumeratedSet(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String enumeratedSetName)
```

GetEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
GetEnumeratedSet(java.lang.String enumeratedSetName)
```

GetEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
GetEnumeratedSet(java.lang.String effectiveUserId,  
java.lang.String enumeratedSetName)
```

GetEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
GetEnumeratedSet(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String enumeratedSetName)
```

QueryEnumeratedSet

```
public java.util.List QueryEnumeratedSet(java.lang.String  
enumeratedSetNameMask,  
int maxNumber)
```

QueryEnumeratedSet

```
public java.util.List QueryEnumeratedSet(java.lang.String effectiveUserId,  
java.lang.String  
enumeratedSetNameMask,  
int maxNumber)
```

QueryEnumeratedSet

```
public java.util.List QueryEnumeratedSet(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String  
enumeratedSetNameMask,  
int maxNumber)
```

RenameEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
RenameEnumeratedSet(java.lang.String oldEnumeratedSetName,  
java.lang.String newEnumeratedSetName)
```

RenameEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty  
RenameEnumeratedSet(java.lang.String effectiveUserId,  
java.lang.String oldEnumeratedSetName,  
java.lang.String newEnumeratedSetName)
```

RenameEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty
RenameEnumeratedSet(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String oldEnumeratedSetName,
java.lang.String newEnumeratedSetName)
```

UpdateEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty
UpdateEnumeratedSet(java.lang.String enumeratedSetName,
com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty
enumeratedSetProperty)
```

UpdateEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty
UpdateEnumeratedSet(java.lang.String effectiveUserId,
java.lang.String enumeratedSetName,
com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty
enumeratedSetProperty)
```

UpdateEnumeratedSet

```
public com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty
UpdateEnumeratedSet(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String enumeratedSetName,
com.ge.ip.hds.historian.DataContracts.EnumeratedSetProperty
enumeratedSetProperty)
```

com.ge.ip.hds.historian.API

Interface InterfaceDefService

< [Methods](#) >

```
public interface InterfaceDefService
```

InterfaceDef

Use this interface to specify what labels you want to appear on the collector configuration page of the administrator.

The administrator will show 5 edit boxes for General 1 to General 5 but with this function you can control what text is show next to each edit box.

Â All custom collectors share the same set of labels.

Methods

AddInterfaceDef

```
public com.ge.ip.hds.historian.DataContracts.ErrorCode  
AddInterfaceDef(com.ge.ip.hds.historian.DataContracts.InterfaceDef intDef)  
throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for creating an InterfaceDef.

Parameters:

intDef - InterfaceDef

Returns:

ErrorCode

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddInterfaceDef

```
public com.ge.ip.hds.historian.DataContracts.ErrorCode  
AddInterfaceDef(java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.InterfaceDef intDef)  
throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for creating an InterfaceDef.

Parameters:

effectiveUserId - User name against which operation will be performed.

intDef - InterfaceDef

Returns:

ErrorCode

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddInterfaceDef

```
public com.ge.ip.hds.historian.DataContracts.ErrorCode  
AddInterfaceDef(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.InterfaceDef intDef) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for creating an InterfaceDef.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
intDef - InterfaceDef

Returns:

ErrorCode

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteInterfaceDef

```
public boolean DeleteInterfaceDef(java.lang.String interfaceName) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting an InterfaceDef.

Parameters:

interfaceName - Name of the interface

Returns:

true on success

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteInterfaceDef

```
public boolean DeleteInterfaceDef(java.lang.String effectiveUserId,  
                                 java.lang.String interfaceName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting an InterfaceDef.

Parameters:

effectiveUserId - User name against which operation will be performed.
interfaceName - Name of the interface

Returns:

true on success

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteInterfaceDef

```
public boolean DeleteInterfaceDef(java.lang.String historianServerName,  
                                 java.lang.String effectiveUserId,  
                                 java.lang.String interfaceName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting an InterfaceDef.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
interfaceName - Name of the interface

Returns:

true on success

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetInterfaceDef

```
public com.ge.ip.hds.historian.DataContracts.InterfaceDef  
GetInterfaceDef(java.lang.String interfaceName)
```

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

throws

This method is for retrieving InterfaceDef structure for an InterfaceDef.

Parameters:

interfaceName - Name of the interface

Returns:

InterfaceDef

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetInterfaceDef

```
public com.ge.ip.hds.historian.DataContracts.InterfaceDef  
GetInterfaceDef(java.lang.String effectiveUserId,  
java.lang.String interfaceName)
```

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

throws

This method is for retrieving InterfaceDef structure for an InterfaceDef.

Parameters:

effectiveUserId - User name against which operation will be performed.

interfaceName - Name of the interface

Returns:

InterfaceDef InterfaceDef

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetInterfaceDef

```
public com.ge.ip.hds.historian.DataContracts.InterfaceDef  
GetInterfaceDef(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String interfaceName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving InterfaceDef structure for an InterfaceDef.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
interfaceName - Name of the interface

Returns:

InterfaceDef

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetInterfaceDefs

```
public java.util.List GetInterfaceDefs(java.lang.String interfaceMask)  
throws
```

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of InterfaceDefs in Historian Server on some search criteria.

Parameters:

interfaceMask - This is a name mask pattern used to retrieve list of InterfaceDefs which matches pattern. The mask can include wildcards, such as asterisks (*)

Returns:

List of InterfaceDef

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetInterfaceDefs

```
public java.util.List GetInterfaceDefs(java.lang.String effectiveUserId,  
                                         java.lang.String interfaceMask)  
                                         throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of InterfaceDefs in Historian Server on some search criteria.

Parameters:

effectiveUserId - User name against which operation will be performed.
interfaceMask - This is a name mask pattern used to retrieve list of InterfaceDefs which matches pattern. The mask can include wildcards, such as asterisks (*)

Returns:

List of InterfaceDef

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetInterfaceDefs

```
public java.util.List GetInterfaceDefs(java.lang.String historianServerName,  
                                         java.lang.String effectiveUserId,  
                                         java.lang.String interfaceMask)  
                                         throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of InterfaceDefs in Historian Server on some search criteria.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
interfaceMask - This is a name mask pattern used to retrieve list of InterfaceDefs which matches pattern. The mask can include wildcards, such as asterisks (*)

Returns:

List of InterfaceDef

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateInterfaceDef

```
public com.ge.ip.hds.historian.DataContracts.ErrorCode  
UpdateInterfaceDef(java.lang.String interfaceName,  
com.ge.ip.hds.historian.DataContracts.InterfaceDef intDef)  
throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for updating InterfaceDef property of an InterfaceDef.

Parameters:

interfaceName - Name of the interface
intDef - InterfaceDef

Returns:

ErrorCode

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateInterfaceDef

```
public com.ge.ip.hds.historian.DataContracts.ErrorCode  
UpdateInterfaceDef(java.lang.String effectiveUserId,  
java.lang.String interfaceName,  
com.ge.ip.hds.historian.DataContracts.InterfaceDef intDef)  
throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for updating InterfaceDef property of an InterfaceDef.

Parameters:

effectiveUserId - User name against which operation will be performed.
interfaceName - Name of the interface
intDef - InterfaceDef

Returns:

ErrorCode

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateInterfaceDef

```
public com.ge.ip.hds.historian.DataContracts.ErrorCode
UpdateInterfaceDef( java.lang.String historianServerName,
 java.lang.String effectiveUserId,
 java.lang.String interfaceName,
 com.ge.ip.hds.historian.DataContracts.InterfaceDef intDef)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for updating InterfaceDef property of an InterfaceDef.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
interfaceName - Name of the interface
intDef - InterfaceDef

Returns:

ErrorCode

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class InterfaceDefServiceImpl

```
java.lang.Object
|
+--com.ge.ip.hds.historian.API.InterfaceDefServiceImpl
```

All Implemented Interfaces:

[InterfaceDefService](#)

< [Constructors](#) > < [Methods](#) >

```
public class InterfaceDefServiceImpl
extends java.lang.Object
implements InterfaceDefService
```

Constructors

InterfaceDefServiceImpl

```
public InterfaceDefServiceImpl()
```

Methods

AddInterfaceDef

```
public com.ge.ip.hds.historian.DataContracts.ErrorCode  
AddInterfaceDef(com.ge.ip.hds.historian.DataContracts.InterfaceDef intDef)
```

AddInterfaceDef

```
public com.ge.ip.hds.historian.DataContracts.ErrorCode  
AddInterfaceDef(java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.InterfaceDef intDef)
```

AddInterfaceDef

```
public com.ge.ip.hds.historian.DataContracts.ErrorCode  
AddInterfaceDef(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.InterfaceDef intDef)
```

DeleteInterfaceDef

```
public boolean DeleteInterfaceDef(java.lang.String interfaceName)
```

DeleteInterfaceDef

```
public boolean DeleteInterfaceDef(java.lang.String effectiveUserId,  
java.lang.String interfaceName)
```

DeleteInterfaceDef

```
public boolean DeleteInterfaceDef(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String interfaceName)
```

GetInterfaceDef

```
public com.ge.ip.hds.historian.DataContracts.InterfaceDef  
GetInterfaceDef(java.lang.String interfaceName)
```

GetInterfaceDef

```
public com.ge.ip.hds.historian.DataContracts.InterfaceDef  
GetInterfaceDef(java.lang.String effectiveUserId,  
java.lang.String interfaceName)
```

GetInterfaceDef

```
public com.ge.ip.hds.historian.DataContracts.InterfaceDef  
GetInterfaceDef(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String interfaceName)
```

GetInterfaceDefs

```
public java.util.List GetInterfaceDefs(java.lang.String interfaceMask)
```

GetInterfaceDefs

```
public java.util.List GetInterfaceDefs(java.lang.String effectiveUserId,  
java.lang.String interfaceMask)
```

GetInterfaceDefs

```
public java.util.List GetInterfaceDefs(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String interfaceMask)
```

UpdateInterfaceDef

```
public com.ge.ip.hds.historian.DataContracts.ErrorCode
UpdateInterfaceDef(java.lang.String interfaceName,
com.ge.ip.hds.historian.DataContracts.InterfaceDef intDef)
```

UpdateInterfaceDef

```
public com.ge.ip.hds.historian.DataContracts.ErrorCode
UpdateInterfaceDef(java.lang.String effectiveUserId,
java.lang.String interfaceName,
com.ge.ip.hds.historian.DataContracts.InterfaceDef intDef)
```

UpdateInterfaceDef

```
public com.ge.ip.hds.historian.DataContracts.ErrorCode
UpdateInterfaceDef(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String interfaceName,
com.ge.ip.hds.historian.DataContracts.InterfaceDef intDef)
```

com.ge.ip.hds.historian.API

Interface JobService

< [Methods](#) >

public interface **JobService**

JobService

The JobService Class implements interfaces for dealing with Historian Jobs. In Historian server their is a concept of Job, Job is a task assigned to Historian Server. These tasks are majorly implemented for syncing, restoring, backing up Historian Archive files. These Jobs are mainly used in Distributed Historian systems.

Methods

BackupArchive

```
public com.ge.ip.hds.historian.DataContracts.JobRecord
BackupArchive(java.lang.String archiveName,
java.lang.String dataStoreName,
java.lang.String fileLocation,
zipFile)
boolean
throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method do the back up of Historian Archive file. Back up file can be zipped also.

Parameters:

archiveName - Name of Archive file
dataStoreName - Name of data store.
fileLocation - Path of Backup file.
zipFile - if set "true", backup file will be zipped else not.

Returns:

JobRecord structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

BackupArchive

```
public com.ge.ip.hds.historian.DataContracts.JobRecord
BackupArchive(java.lang.String effectiveUserId,
java.lang.String archiveName,
java.lang.String dataStoreName,
java.lang.String fileLocation,
zipFile)
boolean
throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method do the back up of Historian Archive file. Back up file can be zipped also.

Parameters:

effectiveUserId - User name against which operation will be performed.
archiveName - Name of Archive file
dataStoreName - Name of data store.
fileLocation - Path of Backup file.
zipFile - if set "true", backup file will be zipped else not.

Returns:

JobRecord structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

BackupArchive

```
public com.ge.ip.hds.historian.DataContracts.JobRecord
BackupArchive(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String archiveName,
java.lang.String dataStoreName,
java.lang.String fileLocation,
boolean
zipFile)
throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method do the back up of Historian Archive file. Back up file can be zipped also.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
archiveName - Name of Archive file
dataStoreName - Name of data store.
fileLocation - Path of Backup file.
zipFile - if set "true", backup file will be zipped else not.

Returns:

JobRecord structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

Cancel

```
public com.ge.ip.hds.historian.DataContracts.JobRecord Cancel(java.lang.String
serviceName,
int jobId)
throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method cancels Job.

Parameters:

serviceName - Name of Job
jobId - Id of Job.

Returns:

JobRecord structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

Cancel

```
public com.ge.ip.hds.historian.DataContracts.JobRecord Cancel( java.lang.String
effectiveUserId,
                                                                    java.lang.String
serviceName,
                                                                    int jobId)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method cancels Job.

Parameters:

effectiveUserId - User name against which operation will be performed.
serviceName - Name of Job
jobId - Id of Job.

Returns:

JobRecord structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

Cancel

```
public com.ge.ip.hds.historian.DataContracts.JobRecord Cancel( java.lang.String
historianServerName,
                                                                    java.lang.String
effectiveUserId,
                                                                    java.lang.String
serviceName,
                                                                    int jobId)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method cancels Job.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
serviceName - Name of Job
jobId - Id of Job.

Returns:

JobRecord structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetJobs

```
public java.util.List GetJobs(boolean includeFinishedJob)
    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method retrieves the List of Jobs from Historian Server.

Parameters:

includeFinishedJob - If set to "true", finished jobs are also retrieved, else only running jobs are retrieved.

Returns:

List of JobRecord

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetJobs

```
public java.util.List GetJobs(java.lang.String effectiveUserId,
    boolean includeFinishedJob)
    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method retrieves the List of Jobs from Historian Server.

Parameters:

effectiveUserId - User name against which operation will be performed.
includeFinishedJob - If set to "true", finished jobs are also retrieved, else only running jobs are retrieved.

Returns:

List of JobRecord

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetJobs

```
public java.util.List GetJobs(java.lang.String historianServerName,  
                               java.lang.String effectiveUserId,  
                               boolean includeFinishedJob)  
                               throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method retrieves the List of Jobs from Historian Server.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
includeFinishedJob - If set to "true", finished jobs are also retrieved, else only running jobs are retrieved.

Returns:

List of JobRecord

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetStatus

```
public com.ge.ip.hds.historian.DataContracts.JobRecord  
GetStatus(java.lang.String serviceName,  
                                                   int jobId)  
                                                   throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method retrieves the current status of Job.

Parameters:

serviceName - Name of Job
jobId - Id of Job.

Returns:

JobRecord structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetStatus

```
public com.ge.ip.hds.historian.DataContracts.JobRecord
GetStatus(java.lang.String effectiveUserId,
java.lang.String serviceName,
                                                    int jobId)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method retrieves the current status of Job.

Parameters:

effectiveUserId - User name against which operation will be performed.
serviceName - Name of Job
jobId - Id of Job.

Returns:

JobRecord structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetStatus

```
public com.ge.ip.hds.historian.DataContracts.JobRecord
GetStatus(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String serviceName,
                                                    int jobId)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method retrieves the current status of Job.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
serviceName - Name of Job
jobId - Id of Job.

Returns:

JobRecord structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

ReplicateNode

```
public com.ge.ip.hds.historian.DataContracts.JobRecord  
ReplicateNode(java.lang.String sourceNode,  
java.lang.String destinationNode)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method replicates the source historian node to a destination historian node.

Parameters:

sourceNode - Name of source historian server.
destinationNode - Name of destination historian server.

Returns:

JobRecord structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

ReplicateNode

```
public com.ge.ip.hds.historian.DataContracts.JobRecord  
ReplicateNode(java.lang.String effectiveUserId,  
java.lang.String sourceNode,  
java.lang.String destinationNode)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method replicates the source historian node to a destination historian node.

Parameters:

effectiveUserId - User name against which operation will be performed.
sourceNode - Name of source historian server.
destinationNode - Name of destination historian server.

Returns:

JobRecord structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

ReplicateNode

```
public com.ge.ip.hds.historian.DataContracts.JobRecord  
ReplicateNode(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String sourceNode,  
java.lang.String destinationNode)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method replicates the source historian node to a destination historian node.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
sourceNode - Name of source historian server.
destinationNode - Name of destination historian server.

Returns:

JobRecord structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RestoreArchive

```
public com.ge.ip.hds.historian.DataContracts.JobRecord  
RestoreArchive(java.lang.String archiveName,  
java.lang.String dataStoreName,  
java.lang.String filePath)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method restore Historian Archive file from backup file.

Parameters:

archiveName - Name of Archive file
dataStoreName - Name of data store.
filePath - Path of Backup file.

Returns:

JobRecord structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RestoreArchive

```
public com.ge.ip.hds.historian.DataContracts.JobRecord  
RestoreArchive(java.lang.String effectiveUserId,  
java.lang.String archiveName,  
java.lang.String dataStoreName,  
java.lang.String filePath)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method restore Historian Archive file from backup file.

Parameters:

effectiveUserId - User name against which operation will be performed.
archiveName - Name of Archive file
dataStoreName - Name of data store.
filePath - Path of Backup file.

Returns:

JobRecord structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RestoreArchive

```
public com.ge.ip.hds.historian.DataContracts.JobRecord  
RestoreArchive(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String archiveName,  
java.lang.String dataStoreName,  
java.lang.String filePath)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method restore Historian Archive file from backup file.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
archiveName - Name of Archive file
dataStoreName - Name of data store.
filePath - Path of Backup file.

Returns:

JobRecord structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

ResyncArchive

```
public com.ge.ip.hds.historian.DataContracts.JobRecord  
ResyncArchive(java.lang.String archiveName,  
java.lang.String dataStoreName,  
java.lang.String dataStorageName,  
java.lang.String sourceNode,  
java.lang.String destinationNode)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method sync a Historian Archive file from one Historian node to another Historian node.

Parameters:

archiveName - Name of Archive file
dataStoreName - Name of data store.
dataStorageName - Name of data storage.
sourceNode - Source Historian Name.
destinationNode - Destination Historian Name.

Returns:

JobRecord structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

ResyncArchive

```
public com.ge.ip.hds.historian.DataContracts.JobRecord  
ResyncArchive(java.lang.String effectiveUserId,  
java.lang.String archiveName,  
java.lang.String dataStoreName,  
java.lang.String dataStorageName,  
java.lang.String sourceNode,  
java.lang.String destinationNode)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method sync a Historian Archive file from one Historian node to another Historian node.

Parameters:

effectiveUserId - User name against which operation will be performed.
archiveName - Name of Archive file
dataStoreName - Name of data store.
dataStorageName - Name of data storage.
sourceNode - Source Historian Name.
destinationNode - Destination Historian Name.

Returns:

JobRecord structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

ResyncArchive

```
public com.ge.ip.hds.historian.DataContracts.JobRecord  
ResyncArchive(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String archiveName,  
java.lang.String dataStoreName,  
java.lang.String dataStorageName,  
java.lang.String sourceNode,  
java.lang.String destinationNode)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method sync a Historian Archive file from one Historian node to another Historian node.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
archiveName - Name of Archive file
dataStoreName - Name of data store.
dataStorageName - Name of data storage.
sourceNode - Source Historian Name.
destinationNode - Destination Historian Name.

Returns:

JobRecord structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class JobServiceImpl

```
java.lang.Object  
|  
+--com.ge.ip.hds.historian.API.JobServiceImpl
```

All Implemented Interfaces:

[JobService](#)

< [Constructors](#) > < [Methods](#) >

```
public class JobServiceImpl  
extends java.lang.Object  
implements JobService
```

Constructors

JobServiceImpl

```
public JobServiceImpl()
```

Methods

BackupArchive

```
public com.ge.ip.hds.historian.DataContracts.JobRecord  
BackupArchive(java.lang.String archiveName,  
java.lang.String dataStoreName,  
java.lang.String fileLocation,  
zipFile) boolean
```

BackupArchive

```
public com.ge.ip.hds.historian.DataContracts.JobRecord  
BackupArchive(java.lang.String effectiveUserId,  
java.lang.String archiveName,  
java.lang.String dataStoreName,  
java.lang.String fileLocation,  
zipFile) boolean
```

BackupArchive

```
public com.ge.ip.hds.historian.DataContracts.JobRecord  
BackupArchive(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String archiveName,  
java.lang.String dataStoreName,  
java.lang.String fileLocation,  
zipFile) boolean
```

Cancel

```
public com.ge.ip.hds.historian.DataContracts.JobRecord Cancel(java.lang.String  
serviceName,  
int jobId)
```

Cancel

```
public com.ge.ip.hds.historian.DataContracts.JobRecord Cancel(java.lang.String  
effectiveUserId,  
serviceName,  
java.lang.String  
int jobId)
```

Cancel

```
public com.ge.ip.hds.historian.DataContracts.JobRecord Cancel(java.lang.String  
historianServerName,  
effectiveUserId,  
serviceName,  
java.lang.String  
java.lang.String  
int jobId)
```

GetJobs

```
public java.util.List GetJobs(boolean includeFinishedJob)
```

GetJobs

```
public java.util.List GetJobs(java.lang.String effectiveUserId,  
boolean includeFinishedJob)
```

GetJobs

```
public java.util.List GetJobs(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
boolean includeFinishedJob)
```

GetStatus

```
public com.ge.ip.hds.historian.DataContracts.JobRecord  
GetStatus(java.lang.String serviceName,  
int jobId)
```

GetStatus

```
public com.ge.ip.hds.historian.DataContracts.JobRecord
GetStatus(java.lang.String effectiveUserId,
java.lang.String serviceName,
int jobId)
```

GetStatus

```
public com.ge.ip.hds.historian.DataContracts.JobRecord
GetStatus(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String serviceName,
int jobId)
```

ReplicateNode

```
public com.ge.ip.hds.historian.DataContracts.JobRecord
ReplicateNode(java.lang.String sourceNode,
java.lang.String destinationNode)
```

ReplicateNode

```
public com.ge.ip.hds.historian.DataContracts.JobRecord
ReplicateNode(java.lang.String effectiveUserId,
java.lang.String sourceNode,
java.lang.String destinationNode)
```

ReplicateNode

```
public com.ge.ip.hds.historian.DataContracts.JobRecord
ReplicateNode(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String sourceNode,
java.lang.String destinationNode)
```

RestoreArchive

```
public com.ge.ip.hds.historian.DataContracts.JobRecord
RestoreArchive(java.lang.String archiveName,
java.lang.String dataStoreName,
java.lang.String filePath)
```

RestoreArchive

```
public com.ge.ip.hds.historian.DataContracts.JobRecord
RestoreArchive(java.lang.String effectiveUserId,
java.lang.String archiveName,
java.lang.String dataStoreName,
java.lang.String filePath)
```

RestoreArchive

```
public com.ge.ip.hds.historian.DataContracts.JobRecord
RestoreArchive(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String archiveName,
java.lang.String dataStoreName,
java.lang.String filePath)
```

ResyncArchive

```
public com.ge.ip.hds.historian.DataContracts.JobRecord
ResyncArchive(java.lang.String archiveName,
java.lang.String dataStoreName,
java.lang.String dataStorageName,
java.lang.String sourceNode,
java.lang.String destinationNode)
```

ResyncArchive

```
public com.ge.ip.hds.historian.DataContracts.JobRecord
ResyncArchive(java.lang.String effectiveUserId,
java.lang.String archiveName,
java.lang.String dataStoreName,
java.lang.String dataStorageName,
java.lang.String sourceNode,
java.lang.String destinationNode)
```

ResyncArchive

```
public com.ge.ip.hds.historian.DataContracts.JobRecord
ResyncArchive(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String archiveName,
java.lang.String dataStoreName,
java.lang.String dataStorageName,
java.lang.String sourceNode,
java.lang.String destinationNode)
```

com.ge.ip.hds.historian.API

Class LogManager

```
java.lang.Object
|
+--com.ge.ip.hds.historian.API.LogManager
```

< [Constructors](#) > < [Methods](#) >

```
public class LogManager
extends java.lang.Object
```

LogManager

Historian uses log4J library for logging events, LogManager class allows users to set basic configurations required for logging.

Constructors

LogManager

```
public LogManager()
```

Methods

debug

```
public static void debug(java.lang.Object message)
```

This method sets the debug message

Parameters:

message - message that needs to be set in log

error

```
public static void error(java.lang.Object message)
```

This method sets the error message

Parameters:

message - message that needs to be set in log

setLogger

```
public static void setLogger(org.apache.log4j.Logger logger)
```

This methods sets the logger

Parameters:

logger - instance of Logger

com.ge.ip.hds.historian.API

Interface MessageService

< [Methods](#) >

```
public interface MessageService
```

MessageService

The MessageService Class implements interfaces for dealing with Message.

Methods

AddMessage

```
public boolean AddMessage(com.ge.ip.hds.historian.DataContracts.Message  
message)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for adding a Message.

Parameters:

message - Message structure.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddMessage

```
public boolean AddMessage(java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.Message  
message)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for adding a Message.

Parameters:

effectiveUserId - User name against which operation will be performed.
message - Message structure.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddMessage

```
public boolean AddMessage(java.lang.String historianServerName,  
                          java.lang.String effectiveUserId,  
                          com.ge.ip.hds.historian.DataContracts.Message  
message)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for adding a Message.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
message - Message structure.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryMessage

```
public java.util.List QueryMessage(java.lang.String historianServerName,  
                                   java.lang.String effectiveUserId,  
                                   java.util.Date start,  
                                   java.util.Date end,  
                                   java.lang.String messageTextMask,  
com.ge.ip.hds.historian.DataContracts.MessageTopic topic,  
                                   int maxNumber)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of Message for a time range..

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
start - Start Time.
end - End Time.
messageTextMask - This is a name mask pattern used to retrieve list of Message which matches pattern. The mask can include wildcards, such as asterisks (*).
topic - Topic of Message.
maxNumber - Maximum number of Message request.

Returns:

List of Message.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryMessage

```
public java.util.List QueryMessage(java.lang.String effectiveUserId,  
                                     java.util.Date start,  
                                     java.util.Date end,  
                                     java.lang.String messageTextMask,  
com.ge.ip.hds.historian.DataContracts.MessageTopic topic,  
                                     int maxNumber)  
                                     throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of Message for a time range..

Parameters:

effectiveUserId - User name against which operation will be performed.
start - Start Time.
end - End Time.
messageTextMask - This is a name mask pattern used to retrieve list of Message which matches pattern. The mask can include wildcards, such as asterisks (*).
topic - Topic of Message.
maxNumber - Maximum number of Message request.

Returns:

List of Message.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryMessage

```
public java.util.List QueryMessage(java.util.Date start,  
                                     java.util.Date end,  
                                     java.lang.String messageTextMask,  
com.ge.ip.hds.historian.DataContracts.MessageTopic topic,  
                                     int maxNumber)  
                                     throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of Message for a time range..

Parameters:

start - Start Time.
end - End Time.
messageTextMask - This is a name mask pattern used to retrieve list of Message which matches pattern. The mask can include wildcards, such as asterisks (*).
topic - Topic of Message.
maxNumber - Maximum number of Message request.

Returns:

List of Message.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class MessageServiceImpl

```
java.lang.Object
|
+--com.ge.ip.hds.historian.API.MessageServiceImpl
```

All Implemented Interfaces:

[MessageService](#)

< [Constructors](#) > < [Methods](#) >

```
public class MessageServiceImpl
extends java.lang.Object
implements MessageService
```

Constructors

MessageServiceImpl

```
public MessageServiceImpl()
```

Methods

AddMessage

```
public boolean AddMessage(com.ge.ip.hds.historian.DataContracts.Message
message)
```

AddMessage

```
public boolean AddMessage(java.lang.String effectiveUserId,
                           com.ge.ip.hds.historian.DataContracts.Message
message)
```

AddMessage

```
public boolean AddMessage(java.lang.String historianServerName,  
                           java.lang.String effectiveUserId,  
                           com.ge.ip.hds.historian.DataContracts.Message  
message)
```

QueryMessage

```
public java.util.List QueryMessage(java.lang.String historianServerName,  
                                    java.lang.String effectiveUserId,  
                                    java.util.Date start,  
                                    java.util.Date end,  
                                    java.lang.String messageTextMask,  
com.ge.ip.hds.historian.DataContracts.MessageTopic topic,  
                                    int maxNumber)
```

QueryMessage

```
public java.util.List QueryMessage(java.lang.String effectiveUserId,  
                                    java.util.Date start,  
                                    java.util.Date end,  
                                    java.lang.String messageTextMask,  
com.ge.ip.hds.historian.DataContracts.MessageTopic topic,  
                                    int maxNumber)
```

QueryMessage

```
public java.util.List QueryMessage(java.util.Date start,  
                                    java.util.Date end,  
                                    java.lang.String messageTextMask,  
com.ge.ip.hds.historian.DataContracts.MessageTopic topic,  
                                    int maxNumber)
```

com.ge.ip.hds.historian.API

Interface PerformanceCounterService

< [Methods](#) >

```
public interface PerformanceCounterService
```


PerformanceCounterService

The PerformanceCounterService Class implements interfaces for getting Performance Counters.

Methods

GetPerformanceCounter

```
public com.ge.ip.hds.historian.DataContracts.PerformanceCounters  
GetPerformanceCounter() throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the Performance counter

Returns:

PerformanceCounters structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetPerformanceCounter

```
public com.ge.ip.hds.historian.DataContracts.PerformanceCounters  
GetPerformanceCounter(java.lang.String effectiveUserId) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the Performance counter

Parameters:

effectiveUserId - User name against which operation will be performed.

Returns:

PerformanceCounters structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetPerformanceCounter

```
public com.ge.ip.hds.historian.DataContracts.PerformanceCounters  
GetPerformanceCounter(java.lang.String historianServerName,  
java.lang.String effectiveUserId)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the Performance counter

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.

Returns:

PerformanceCounters structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class PerformanceCounterServiceImpl

```
java.lang.Object  
|  
+--com.ge.ip.hds.historian.API.PerformanceCounterServiceImpl
```

All Implemented Interfaces:

[PerformanceCounterService](#)

< [Constructors](#) > < [Methods](#) >

```
public class PerformanceCounterServiceImpl  
extends java.lang.Object  
implements PerformanceCounterService
```

Constructors

PerformanceCounterServiceImpl

```
public PerformanceCounterServiceImpl()
```

Methods

GetPerformanceCounter

```
public com.ge.ip.hds.historian.DataContracts.PerformanceCounters  
GetPerformanceCounter()
```

GetPerformanceCounter

```
public com.ge.ip.hds.historian.DataContracts.PerformanceCounters  
GetPerformanceCounter(java.lang.String effectiveUserId)
```

GetPerformanceCounter

```
public com.ge.ip.hds.historian.DataContracts.PerformanceCounters  
GetPerformanceCounter(java.lang.String historianServerName,  
java.lang.String effectiveUserId)
```

com.ge.ip.hds.historian.API

Interface PublishService

< [Methods](#) >

public interface **PublishService**

PublishService

The PublishService Class implements interfaces for Publish functions. The Publish functions present a unified interface for managing data forwarding of local Historian server tags to one or more remote destination servers via the Server-to-Server Distributors. Publishing a tag initiates continuous transmission of its historical data values to a destination server.

Methods

AddTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
AddTag( java.lang.String collectorName,
java.lang.String sourceTagName,
java.lang.String destinationTagName)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method publish a tag to a destination server via the specified collector.

Parameters:

collectorName - Name of Collector
sourceTagName - Name of tag.
destinationTagName - Name of tag in destination server.

Returns:

TagProperty structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
AddTag( java.lang.String effectiveUserId,
java.lang.String collectorName,
java.lang.String sourceTagName,
java.lang.String destinationTagName)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method publish a tag to a destination server via the specified collector.

Parameters:

effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector
sourceTagName - Name of tag.
destinationTagName - Name of tag in destination server.

Returns:

TagProperty structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
AddTag(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String collectorName,
java.lang.String sourceTagName,
java.lang.String destinationTagName)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method publish a tag to a destination server via the specified collector.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector
sourceTagName - Name of tag.
destinationTagName - Name of tag in destination server.

Returns:

TagProperty structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDestinationServer

```
public java.lang.String GetDestinationServer(java.lang.String collectorName)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method retrieve the name of the destination Historian server that is configured for a distributor. .

Parameters:

collectorName - Name of Collector

Returns:

destination Historian server name.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDestinationServer

```
public java.lang.String GetDestinationServer(java.lang.String effectiveUserId,  
                                             java.lang.String collectorName)  
                                             throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method retrieve the name of the destination Historian server that is configured for a distributor. .

Parameters:

effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector

Returns:

destination Historian server name.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetDestinationServer

```
public java.lang.String GetDestinationServer(java.lang.String  
historianServerName,  
                                             java.lang.String effectiveUserId,  
                                             java.lang.String collectorName)  
                                             throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method retrieve the name of the destination Historian server that is configured for a distributor. .

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector

Returns:

destination Historian server name.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetTagProperties

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
GetTagProperties(java.lang.String collectorName,  
java.lang.String tagName)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

throws

This method retrieve TagProperty for a published tag via the specified collector.

Parameters:

collectorName - Name of Collector
tagName - Name of tag.

Returns:

TagProperty structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetTagProperties

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
GetTagProperties(java.lang.String effectiveUserId,  
java.lang.String collectorName,  
java.lang.String tagName)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

throws

This method retrieve TagProperty for a published tag via the specified collector.

Parameters:

effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector
tagName - Name of tag.

Returns:

TagProperty structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetTagProperties

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
GetTagProperties(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String collectorName,
java.lang.String tagName)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method retrieve TagProperty for a published tag via the specified collector.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector
tagName - Name of tag.

Returns:

TagProperty structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryTag

```
public java.util.List QueryTag(java.lang.String collectorName,
                                java.lang.String tagNameMask,
                                int maxNumber)
                                throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method retrieve TagProperty for a list of published tags via the specified collector on basis of tag name mask.

Parameters:

collectorName - Name of Collector
tagNameMask - This is a name mask pattern used to retrieve list of published tags which matches pattern. The mask can include wildcards, such as asterisks (*).
maxNumber - Maximum number of tag request.

Returns:

List of TagPropertyBase structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryTag

```
public java.util.List QueryTag(java.lang.String effectiveUserId,  
                                java.lang.String collectorName,  
                                java.lang.String tagNameMask,  
                                int maxNumber)  
                                throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method retrieve TagProperty for a list of published tags via the specified collector on basis of tag name mask.

Parameters:

effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector
tagNameMask - This is a name mask pattern used to retrieve list of published tags which matches pattern. The mask can include wildcards, such as asterisks (*).
maxNumber - Maximum number of tag request.

Returns:

List of TagPropertyBase structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryTag

```
public java.util.List QueryTag(java.lang.String historianServerName,  
                                java.lang.String effectiveUserId,  
                                java.lang.String collectorName,  
                                java.lang.String tagNameMask,  
                                int maxNumber)  
                                throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method retrieve TagProperty for a list of published tags via the specified collector on basis of tag name mask.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector
tagNameMask - This is a name mask pattern used to retrieve list of published tags which matches pattern. The mask can include wildcards, such as asterisks (*).
maxNumber - Maximum number of tag request.

Returns:

List of TagPropertyBase structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RemoveTag

```
public boolean RemoveTag(java.lang.String collectorName,  
                           java.lang.String tagName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method stops a tag from being published via the specified collector.

Parameters:

collectorName - Name of Collector
tagName - Name of tag.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RemoveTag

```
public boolean RemoveTag(java.lang.String effectiveUserId,  
                           java.lang.String collectorName,  
                           java.lang.String tagName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method stops a tag from being published via the specified collector.

Parameters:

effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector
tagName - Name of tag.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RemoveTag

```
public boolean RemoveTag(java.lang.String historianServerName,  
                           java.lang.String effectiveUserId,  
                           java.lang.String collectorName,  
                           java.lang.String tagName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method stops a tag from being published via the specified collector.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector
tagName - Name of tag.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

SetDestinationServer

```
public boolean SetDestinationServer(java.lang.String collectorName,  
                                     java.lang.String destinationServer)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method associate the given destination Historian server to a collector.

Parameters:

collectorName - Name of Collector
destinationServer - Name of Destination Historian Server.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

SetDestinationServer

```
public boolean SetDestinationServer(java.lang.String effectiveUserId,  
                                     java.lang.String collectorName,  
                                     java.lang.String destinationServer)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method associate the given destination Historian server to a collector.

Parameters:

effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector
destinationServer - Name of Destination Historian Server.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

SetDestinationServer

```
public boolean SetDestinationServer(java.lang.String historianServerName,  
                                     java.lang.String effectiveUserId,  
                                     java.lang.String collectorName,  
                                     java.lang.String destinationServer)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method associate the given destination Historian server to a collector.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector
destinationServer - Name of Destination Historian Server.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

SetTagProperties

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
SetTagProperties(java.lang.String collectorName,
java.lang.String tagName,
com.ge.ip.hds.historian.DataContracts.TagProperty tagProperty)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method sets the tag property of already added tag in destination server. Use this function after tag is added by AddTag function.

Parameters:

collectorName - Name of Collector
tagName - Name of tag.
tagProperty - TagProperty structure

Returns:

TagProperty structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

SetTagProperties

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
SetTagProperties(java.lang.String effectiveUserId,
java.lang.String collectorName,
java.lang.String tagName,
com.ge.ip.hds.historian.DataContracts.TagProperty tagProperty)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method sets the tag property of already added tag in destination server. Use this function after tag is added by AddTag function.

Parameters:

effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector
tagName - Name of tag.
tagProperty - TagProperty structure

Returns:

TagProperty structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

SetTagProperties

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
SetTagProperties(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String collectorName,
java.lang.String tagName,
com.ge.ip.hds.historian.DataContracts.TagProperty tagProperty)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method sets the tag property of already added tag in destination server. Use this function after tag is added by AddTag function.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
collectorName - Name of Collector
tagName - Name of tag.
tagProperty - TagProperty structure

Returns:

TagProperty structure

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class PublishServiceImpl

```
java.lang.Object
|
+--com.ge.ip.hds.historian.API.PublishServiceImpl
```

All Implemented Interfaces:

[PublishService](#)

< [Constructors](#) > < [Methods](#) >

```
public class PublishServiceImpl
extends java.lang.Object
implements PublishService
```

Constructors

PublishServiceImpl

```
public PublishServiceImpl()
```

Methods

AddTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
AddTag(java.lang.String collectorName,
java.lang.String sourceTagName,
java.lang.String destinationTagName)
```

AddTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
AddTag(java.lang.String effectiveUserId,
java.lang.String collectorName,
java.lang.String sourceTagName,
java.lang.String destinationTagName)
```

AddTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
AddTag(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String collectorName,
java.lang.String sourceTagName,
java.lang.String destinationTagName)
```

GetDestinationServer

```
public java.lang.String GetDestinationServer(java.lang.String collectorName)
```

GetDestinationServer

```
public java.lang.String GetDestinationServer(java.lang.String effectiveUserId,
java.lang.String collectorName)
```

GetDestinationServer

```
public java.lang.String GetDestinationServer(java.lang.String
historianServerName,
                                             java.lang.String effectiveUserId,
                                             java.lang.String collectorName)
```

GetTagProperties

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
GetTagProperties(java.lang.String collectorName,
java.lang.String tagName)
```

GetTagProperties

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
GetTagProperties(java.lang.String effectiveUserId,
java.lang.String collectorName,
java.lang.String tagName)
```

GetTagProperties

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
GetTagProperties(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String collectorName,
java.lang.String tagName)
```

QueryTag

```
public java.util.List QueryTag(java.lang.String collectorName,
java.lang.String tagNameMask,
int maxNumber)
```

QueryTag

```
public java.util.List QueryTag(java.lang.String effectiveUserId,
java.lang.String collectorName,
java.lang.String tagNameMask,
int maxNumber)
```

QueryTag

```
public java.util.List QueryTag(java.lang.String historianServerName,  
                                java.lang.String effectiveUserId,  
                                java.lang.String collectorName,  
                                java.lang.String tagNameMask,  
                                int maxNumber)
```

RemoveTag

```
public boolean RemoveTag(java.lang.String collectorName,  
                           java.lang.String tagName)
```

RemoveTag

```
public boolean RemoveTag(java.lang.String effectiveUserId,  
                           java.lang.String collectorName,  
                           java.lang.String tagName)
```

RemoveTag

```
public boolean RemoveTag(java.lang.String historianServerName,  
                           java.lang.String effectiveUserId,  
                           java.lang.String collectorName,  
                           java.lang.String tagName)
```

SetDestinationServer

```
public boolean SetDestinationServer(java.lang.String collectorName,  
                                     java.lang.String destinationServer)
```

SetDestinationServer

```
public boolean SetDestinationServer(java.lang.String effectiveUserId,  
                                     java.lang.String collectorName,  
                                     java.lang.String destinationServer)
```

SetDestinationServer

```
public boolean SetDestinationServer(java.lang.String historianServerName,  
                                     java.lang.String effectiveUserId,  
                                     java.lang.String collectorName,  
                                     java.lang.String destinationServer)
```

SetTagProperties

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
SetTagProperties(java.lang.String collectorName,  
                 java.lang.String tagName,  
                 com.ge.ip.hds.historian.DataContracts.TagProperty tagProperty)
```

SetTagProperties

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
SetTagProperties(java.lang.String effectiveUserId,  
                 java.lang.String collectorName,  
                 java.lang.String tagName,  
                 com.ge.ip.hds.historian.DataContracts.TagProperty tagProperty)
```

SetTagProperties

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
SetTagProperties(java.lang.String historianServerName,  
                 java.lang.String effectiveUserId,  
                 java.lang.String collectorName,  
                 java.lang.String tagName,  
                 com.ge.ip.hds.historian.DataContracts.TagProperty tagProperty)
```

com.ge.ip.hds.historian.API

Interface QueryModifierService

< [Methods](#) >

public interface **QueryModifierService**

QueryModifierService

The QueryModifierService Class implements interfaces for dealing with Query Modifiers.

Methods

GetCriteriaFromString

```
public com.ge.ip.hds.historian.DataContracts.DataCriteria  
GetCriteriaFromString(java.lang.String criteriaString)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the DataCriteria structure on basis of Query Modifier Strings. Their are several Query Modifier Strings like ONLYGOOD, ONLYRAW, HONORENDTIME, .. etc.

Parameters:

criteriaString - Query Modifier String

Returns:

DataCriteria structure .

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCriteriaFromString

```
public com.ge.ip.hds.historian.DataContracts.DataCriteria  
GetCriteriaFromString(java.lang.String effectiveUserId,  
java.lang.String criteriaString)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the DataCriteria structure on basis of Query Modifier Strings. Their are several Query Modifier Strings like ONLYGOOD, ONLYRAW, HONORENDTIME, .. etc.

Parameters:

effectiveUserId - User name against which operation will be performed.

criteriaString - Query Modifier String

Returns:

DataCriteria structure .

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCriteriaFromString

```
public com.ge.ip.hds.historian.DataContracts.DataCriteria  
GetCriteriaFromString(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String criteriaString)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the DataCriteria structure on basis of Query Modifier Strings. Their are several Query Modifier Strings like ONLYGOOD, ONLYRAW, HONORENDTIME, .. etc.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
criteriaString - Query Modifier String

Returns:

DataCriteria structure .

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetQueryModifier

```
public java.util.List GetQueryModifier()
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of query Modifiers.

Returns:

List of QueryModifierProperty.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetQueryModifier

```
public java.util.List GetQueryModifier(java.lang.String effectiveUserId)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of query Modifiers.

Parameters:

effectiveUserId - User name against which operation will be performed.

Returns:

List of QueryModifierProperty.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetQueryModifier

```
public java.util.List GetQueryModifier(java.lang.String historianServerName,  
                                         java.lang.String effectiveUserId)  
                                         throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of query Modifiers.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.

Returns:

List of QueryModifierProperty.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class QueryModifierServiceImpl

```
java.lang.Object  
|  
+--com.ge.ip.hds.historian.API.QueryModifierServiceImpl
```

All Implemented Interfaces:

[QueryModifierService](#)

< [Constructors](#) > < [Methods](#) >

```
public class QueryModifierServiceImpl  
extends java.lang.Object  
implements QueryModifierService
```

Constructors

QueryModifierServiceImpl

```
public QueryModifierServiceImpl()
```

Methods

GetCriteriaFromString

```
public com.ge.ip.hds.historian.DataContracts.DataCriteria  
GetCriteriaFromString(java.lang.String criteriaString)
```

GetCriteriaFromString

```
public com.ge.ip.hds.historian.DataContracts.DataCriteria  
GetCriteriaFromString(java.lang.String effectiveUserId,  
java.lang.String criteriaString)
```

GetCriteriaFromString

```
public com.ge.ip.hds.historian.DataContracts.DataCriteria  
GetCriteriaFromString(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String criteriaString)
```

GetQueryModifier

```
public java.util.List GetQueryModifier()
```

GetQueryModifier

```
public java.util.List GetQueryModifier(java.lang.String effectiveUserId)
```

GetQueryModifier

```
public java.util.List GetQueryModifier(java.lang.String historianServerName,  
java.lang.String effectiveUserId)
```

com.ge.ip.hds.historian.API

Interface ServerService

< [Methods](#) >

public interface **ServerService**

ServerService

The ServerService Class implements interfaces for dealing with Historian Server configuration, Data Store configuration, Options and statistics. Datastore options are some configurable parameters associated with each datastores like "default size", "Read only" etc.

Methods

GetConfiguration

```
public com.ge.ip.hds.historian.DataContracts.HistorianConfiguration  
GetConfiguration() throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving Historian configuration.

Returns:

HistorianConfiguration structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetConfiguration

```
public com.ge.ip.hds.historian.DataContracts.HistorianConfiguration  
GetConfiguration(java.lang.String effectiveUserId) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving Historian configuration.

Parameters:

effectiveUserId - User name against which operation will be performed.

Returns:

HistorianConfiguration structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetConfiguration

```
public com.ge.ip.hds.historian.DataContracts.HistorianConfiguration  
GetConfiguration(java.lang.String historianServerName,  
java.lang.String effectiveUserId)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving Historian configuration.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.

Returns:

HistorianConfiguration structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCurrentTime

```
public java.util.Date GetCurrentTime()  
throws
```

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the current time of Historian Server Machine.

Returns:

Date of Historian Server Machine.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCurrentTime

```
public java.util.Date GetCurrentTime(java.lang.String effectiveUserId)  
throws
```

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the current time of Historian Server Machine.

Parameters:

effectiveUserId - User name against which operation will be performed.

Returns:

Date of Historian Server Machine.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetCurrentTime

```
public java.util.Date GetCurrentTime(java.lang.String historianServerName,  
                                       java.lang.String effectiveUserId)  
                                       throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the current time of Historian Server Machine.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.

Returns:

Date of Historian Server Machine.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetOption

```
public java.lang.String  
GetOption(com.ge.ip.hds.historian.DataContracts.HistorianOption  
historianOption,  
           java.lang.String dataStore)  
           throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving Value of a data store option.

Parameters:

historianOption - data store options which need to be retrieved.
dataStore - Name of data store.

Returns:

value of asked data store option.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetOption

```
public java.lang.String GetOption(java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.HistorianOption historianOption,  
                                     java.lang.String dataStore)  
                                     throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving Value of a data store option.

Parameters:

effectiveUserId - User name against which operation will be performed.
historianOption - data store options which need to be retrieved.
dataStore - Name of data store.

Returns:

value of asked data store option.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetOption

```
public java.lang.String GetOption(java.lang.String historianServerName,  
                                     java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.HistorianOption historianOption,  
                                     java.lang.String dataStore)  
                                     throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving Value of a data store option.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
historianOption - data store options which need to be retrieved.
dataStore - Name of data store.

Returns:

value of asked data store option.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetOptionsInBatch

```
public java.util.Map GetOptionsInBatch(java.lang.String dataStore)
    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving Map of Value and data store options.

Parameters:

dataStore - Name of data store.

Returns:

Map of value and data store options.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetOptionsInBatch

```
public java.util.Map GetOptionsInBatch(java.lang.String effectiveUserId,
    java.lang.String dataStore)
    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving Map of Value and data store options.

Parameters:

effectiveUserId - User name against which operation will be performed.
dataStore - Name of data store.

Returns:

Map of value and data store options.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetOptionsInBatch

```
public java.util.Map GetOptionsInBatch(java.lang.String historianServerName,  
                                         java.lang.String effectiveUserId,  
                                         java.lang.String dataStore)  
                                         throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving Map of Value and data store options.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
dataStore - Name of data store.

Returns:

Map of value and data store options.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetSecurityGroupMembership

```
public java.util.List GetSecurityGroupMembership()  
                    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the List of Security Group names of Historian Server for which this client is added.

Returns:

List of Security Group names of Historian Server.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetSecurityGroupMembership

```
public java.util.List GetSecurityGroupMembership(java.lang.String  
effectiveUserId)  
                    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the List of Security Group names of Historian Server for which this client is added.

Parameters:

effectiveUserId - User name against which operation will be performed.

Returns:

List of Security Group names of Historian Server.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetSecurityGroupMembership

```
public java.util.List GetSecurityGroupMembership(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the List of Security Group names of Historian Server for which this client is added.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.

Returns:

List of Security Group names of Historian Server.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetSecurityGroupNames

```
public java.util.List GetSecurityGroupNames()
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the List of Security Group names of Historian Server.

Returns:

List of Security Group names of Historian Server.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetSecurityGroupNames

```
public java.util.List GetSecurityGroupNames(java.lang.String effectiveUserId)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the List of Security Group names of Historian Server.

Parameters:

effectiveUserId - User name against which operation will be performed.

Returns:

List of Security Group names of Historian Server.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetSecurityGroupNames

```
public java.util.List GetSecurityGroupNames(java.lang.String
historianServerName,
                                             java.lang.String effectiveUserId)
                                             throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the List of Security Group names of Historian Server.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.

Returns:

List of Security Group names of Historian Server.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetStatistics

```
public com.ge.ip.hds.historian.DataContracts.ArchiveStatistics GetStatistics()
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for statistics of Historian Server machine.

Returns:

ArchiveStatistics structure of Historian Server.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetStatistics

```
public com.ge.ip.hds.historian.DataContracts.ArchiveStatistics
GetStatistics(java.lang.String effectiveUserId)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for statistics of Historian Server machine.

Parameters:

effectiveUserId - User name against which operation will be performed.

Returns:

ArchiveStatistics structure of Historian Server.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetStatistics

```
public com.ge.ip.hds.historian.DataContracts.ArchiveStatistics  
GetStatistics(java.lang.String historianServerName,  
java.lang.String effectiveUserId)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException throws
```

This method is for statistics of Historian Server machine.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.

Returns:

ArchiveStatistics structure of Historian Server.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

SetOption

```
public boolean SetOption(com.ge.ip.hds.historian.DataContracts.HistorianOption  
historianOption,  
java.lang.String value,  
java.lang.String dataStore)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException throws
```

This method is for setting value of a data store option.

Parameters:

historianOption - data store option which need to be set.
value - Value of data store option.
dataStore - Name of data store.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

SetOption

```
public boolean SetOption(java.lang.String effectiveUserId,  
                          com.ge.ip.hds.historian.DataContracts.HistorianOption  
historianOption,  
                          java.lang.String value,  
                          java.lang.String dataStore)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for setting value of a data store option.

Parameters:

effectiveUserId - User name against which operation will be performed.
historianOption - data store option which need to be set.
value - Value of data store option.
dataStore - Name of data store.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

SetOption

```
public boolean SetOption(java.lang.String historianServerName,  
                          java.lang.String effectiveUserId,  
                          com.ge.ip.hds.historian.DataContracts.HistorianOption  
historianOption,  
                          java.lang.String value,  
                          java.lang.String dataStore)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for setting value of a data store option.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
historianOption - data store option which need to be set.
value - Value of data store option.
dataStore - Name of data store.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

SetOptionsInBatch

```
public boolean SetOptionsInBatch(java.lang.String historianServerName,  
                                   java.lang.String effectiveUserId,  
                                   java.lang.String dataStore,  
                                   java.util.Map optionsValue)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for setting Map of Value and data store options.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
dataStore - Name of data store.
optionsValue - Map of value and data store options.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

SetOptionsInBatch

```
public boolean SetOptionsInBatch(java.lang.String effectiveUserId,  
                                   java.lang.String dataStore,  
                                   java.util.Map optionsValue)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for setting Map of Value and data store options.

Parameters:

effectiveUserId - User name against which operation will be performed.
dataStore - Name of data store.
optionsValue - Map of value and data store options.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

SetOptionsInBatch

```
public boolean SetOptionsInBatch(java.lang.String dataStore,  
                                   java.util.Map optionsValue)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for setting Map of Value and data store options.

Parameters:

dataStore - Name of data store.
optionsValue - Map of value and data store options.

Returns:

true on success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class ServerServiceImpl

```
java.lang.Object  
|  
+--com.ge.ip.hds.historian.API.ServerServiceImpl
```

All Implemented Interfaces:

[ServerService](#)

< [Constructors](#) > < [Methods](#) >

```
public class ServerServiceImpl  
extends java.lang.Object  
implements ServerService
```

Constructors

ServerServiceImpl

```
public ServerServiceImpl()
```

Methods

GetConfiguration

```
public com.ge.ip.hds.historian.DataContracts.HistorianConfiguration  
GetConfiguration()
```

GetConfiguration

```
public com.ge.ip.hds.historian.DataContracts.HistorianConfiguration  
GetConfiguration(java.lang.String effectiveUserId)
```

GetConfiguration

```
public com.ge.ip.hds.historian.DataContracts.HistorianConfiguration  
GetConfiguration(java.lang.String historianServerName,  
java.lang.String effectiveUserId)
```

GetCurrentTime

```
public java.util.Date GetCurrentTime()
```

GetCurrentTime

```
public java.util.Date GetCurrentTime(java.lang.String effectiveUserId)
```

GetCurrentTime

```
public java.util.Date GetCurrentTime(java.lang.String historianServerName,  
java.lang.String effectiveUserId)
```

GetOption

```
public java.lang.String  
GetOption(com.ge.ip.hds.historian.DataContracts.HistorianOption  
historianOption,  
java.lang.String dataStore)
```

GetOption

```
public java.lang.String GetOption(java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.HistorianOption historianOption,  
java.lang.String dataStore)
```

GetOption

```
public java.lang.String GetOption(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.HistorianOption historianOption,  
java.lang.String dataStore)
```

GetOptionsInBatch

```
public java.util.Map GetOptionsInBatch(java.lang.String dataStore)
```

GetOptionsInBatch

```
public java.util.Map GetOptionsInBatch(java.lang.String dataStore,  
java.lang.String effectiveUserId)
```

GetOptionsInBatch

```
public java.util.Map GetOptionsInBatch(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String dataStore)
```

GetSecurityGroupMembership

```
public java.util.List GetSecurityGroupMembership()
```

GetSecurityGroupMembership

```
public java.util.List GetSecurityGroupMembership(java.lang.String  
effectiveUserId)
```

GetSecurityGroupMembership

```
public java.util.List GetSecurityGroupMembership(java.lang.String  
historianServerName,  
effectiveUserId) java.lang.String
```

GetSecurityGroupNames

```
public java.util.List GetSecurityGroupNames()
```

GetSecurityGroupNames

```
public java.util.List GetSecurityGroupNames(java.lang.String effectiveUserId)
```

GetSecurityGroupNames

```
public java.util.List GetSecurityGroupNames(java.lang.String  
historianServerName,  
effectiveUserId) java.lang.String
```

GetStatistics

```
public com.ge.ip.hds.historian.DataContracts.ArchiveStatistics GetStatistics()
```

GetStatistics

```
public com.ge.ip.hds.historian.DataContracts.ArchiveStatistics  
GetStatistics(java.lang.String effectiveUserId)
```

GetStatistics

```
public com.ge.ip.hds.historian.DataContracts.ArchiveStatistics  
GetStatistics(java.lang.String historianServerName,  
java.lang.String effectiveUserId)
```

SetOption

```
public boolean SetOption(com.ge.ip.hds.historian.DataContracts.HistorianOption  
historianOption,  
java.lang.String value,  
java.lang.String dataStore)
```

SetOption

```
public boolean SetOption(java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.HistorianOption  
historianOption,  
java.lang.String value,  
java.lang.String dataStore)
```

SetOption

```
public boolean SetOption(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.HistorianOption  
historianOption,  
java.lang.String value,  
java.lang.String dataStore)
```

SetOptionsInBatch

```
public boolean SetOptionsInBatch(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String dataStore,  
java.util.Map optionsValue)
```

SetOptionsInBatch

```
public boolean SetOptionsInBatch(java.lang.String dataStore,  
                                   java.lang.String effectiveUserId,  
                                   java.util.Map optionsValue)
```

SetOptionsInBatch

```
public boolean SetOptionsInBatch(java.lang.String dataStore,  
                                   java.util.Map optionsValue)
```

com.ge.ip.hds.historian.API

Interface TagService

< [Methods](#) >

```
public interface TagService
```

TagService

The TagService Class implements interfaces for Tag operations.

Methods

AddTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
AddTag(com.ge.ip.hds.historian.DataContracts.TagProperty tagProperty)  
                                                throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for creating a tag.

Parameters:

tagProperty - TagProperty of a tag.

Returns:

TagProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
AddTag(java.lang.String effectiveUserId,
com.ge.ip.hds.historian.DataContracts.TagProperty tagProperty)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for creating a tag.

Parameters:

effectiveUserId - User name against which operation will be performed.
tagProperty - TagProperty of a tag.

Returns:

TagProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
AddTag(java.lang.String historianServerName,
java.lang.String effectiveUserId,
com.ge.ip.hds.historian.DataContracts.TagProperty tagProperty)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for creating a tag.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
tagProperty - TagProperty of a tag.

Returns:

TagProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

BrowseTag

```
public java.util.List BrowseTag(java.lang.String tagNameMask,  
                                int maxNumber)  
                                throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the qualified tag list by a given tagNameMask.

Parameters:

tagNameMask - This is a name mask pattern used to retrieve list of tags which matches pattern. The mask can include wildcards, such as asterisks (*).
maxNumber - Maximum tag number that can be queried and also managed on the server side (1000 by default). If the maxNumber parameter is greater than the max tag query number setting in server side, the query complies with the server side setting.

Returns:

List of TagPropertyBase

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

BrowseTag

```
public java.util.List BrowseTag(java.lang.String effectiveUserId,  
                                java.lang.String tagNameMask,  
                                int maxNumber)  
                                throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the qualified tag list by a given tagNameMask.

Parameters:

effectiveUserId - User name against which operation will be performed.
tagNameMask - This is a name mask pattern used to retrieve list of tags which matches pattern. The mask can include wildcards, such as asterisks (*).
maxNumber - Maximum tag number that can be queried and also managed on the server side (1000 by default). If the maxNumber parameter is greater than the max tag query number setting in server side, the query complies with the server side setting.

Returns:

List of TagPropertyBase

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

BrowseTag

```
public java.util.List BrowseTag(java.lang.String historianServerName,  
                                java.lang.String effectiveUserId,  
                                java.lang.String tagNameMask,  
                                int maxNumber)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the qualified tag list by a given tagNameMask.

Parameters:

historianServerName - Name of target historian server.

effectiveUserId - User name against which operation will be performed.

tagNameMask - This is a name mask pattern used to retrieve list of tags which matches pattern. The mask can include wildcards, such as asterisks (*).

maxNumber - Maximum tag number that can be queried and also managed on the server side (1000 by default). If the maxNumber parameter is greater than the max tag query number setting in server side, the query complies with the server side setting.

Returns:

List of TagPropertyBase

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

CopyTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
CopyTag(java.lang.String tagName,  
         java.lang.String copyTagName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for copying TagProperty a tag to another tag.

Parameters:

tagName - source tag name.

copyTagName - destination tag name.

Returns:

TagProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

CopyTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
CopyTag(java.lang.String effectiveUserId,  
java.lang.String tagName,  
java.lang.String copyTagName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for copying TagProperty a tag to another tag.

Parameters:

effectiveUserId - User name against which operation will be performed.

tagName - source tag name.

copyTagName - destination tag name.

Returns:

TagProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

CopyTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
CopyTag(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String tagName,  
java.lang.String copyTagName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for copying TagProperty a tag to another tag.

Parameters:

historianServerName - Name of target historian server.

effectiveUserId - User name against which operation will be performed.

tagName - source tag name.

copyTagName - destination tag name.

Returns:

TagProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteTag

```
public void DeleteTag(java.lang.String tagName,
                      boolean permanentDelete)
    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for deleting a tag.

Parameters:

tagName - Name of tag.
permanentDelete - set to true for permanent delete.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteTag

```
public void DeleteTag(java.lang.String effectiveUserId,
                      java.lang.String tagName,
                      boolean permanentDelete)
    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for deleting a tag.

Parameters:

effectiveUserId - User name against which operation will be performed.
tagName - Name of tag.
permanentDelete - set to true for permanent delete.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteTag

```
public void DeleteTag(java.lang.String historianServerName,
                      java.lang.String effectiveUserId,
                      java.lang.String tagName,
                      boolean permanentDelete)
    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for deleting a tag.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
tagName - Name of tag.
permanentDelete - set to true for permanent delete.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetSupportedTagCriterias

```
public java.lang.String GetSupportedTagCriterias()  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving supported tag search criteria.

Returns:

tag search criteria.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetSupportedTagCriterias

```
public java.lang.String GetSupportedTagCriterias(java.lang.String  
effectiveUserId)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving supported tag search criteria.

Parameters:

effectiveUserId - User name against which operation will be performed.

Returns:

tag search criteria.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetSupportedTagCriterias

```
public java.lang.String GetSupportedTagCriterias(java.lang.String  
historianServerName,  
                                                java.lang.String  
effectiveUserId)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving supported tag search criteria.

Parameters:

historianServerName - Name of target historian server.

effectiveUserId - User name against which operation will be performed.

Returns:

tag search criteria.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
GetTag(java.lang.String tagName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving tag property of a tag.

Parameters:

tagName - Name of tag.

Returns:

TagProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
GetTag(java.lang.String effectiveUserId,  
java.lang.String tagName)
```

throws

```
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving tag property of a tag.

Parameters:

effectiveUserId - User name against which operation will be performed.

tagName - Name of tag.

Returns:

TagProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
GetTag(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String tagName)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving tag property of a tag.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
tagName - Name of tag.

Returns:

TagProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetTagAliasName

```
public java.util.List GetTagAliasName(java.lang.String historianServerName,
                                       java.lang.String effectiveUserId,
                                       java.util.List tagNameNames)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving alias name for list of tags.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
tagNameNames - List of tag names.

Returns:

List of alias names.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetTagAliasName

```
public java.util.List GetTagAliasName(java.lang.String effectiveUserId,  
                                       java.util.List tagNames)  
                                       throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving alias name for list of tags.

Parameters:

effectiveUserId - User name against which operation will be performed.
tagNames - List of tag names.

Returns:

List of alias names.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetTagAliasName

```
public java.util.List GetTagAliasName(java.util.List tagNames)  
                                       throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving alias name for list of tags.

Parameters:

tagNames - List of tag names.

Returns:

List of alias names.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryTagByFilter

```
public java.util.List QueryTagByFilter(java.lang.String tagNameMask,  
                                         java.lang.String descriptionMask,  
                                         java.lang.String dataStoreName,  
                                         java.lang.String collectorName,  
                                         java.lang.String customTypeNameMask,  
                                         int maxNumber)  
                                         throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the qualified tag list by multiple search criteria.

Parameters:

tagNameMask - This is a name mask pattern used to retrieve list of tags which matches pattern. The mask can include wildcards, such as asterisks (*).

descriptionMask - This is a description mask pattern used to retrieve list of tags which matches pattern. The mask can include wildcards, such as asterisks (*).

dataStoreName - Data store name where tag data is getting stored.

collectorName - Collector Name collecting tag.

customTypeNameMask - This is a Custom type mask pattern used to retrieve list of tags which matches pattern. The mask can include wildcards, such as asterisks (*).

maxNumber - Maximum tag number that can be queried and also managed on the server side (1000 by default). If the maxNumber parameter is greater than the max tag query number setting in server side, the query complies with the server side setting.

Returns:

List of TagPropertyBase

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryTagByFilter

```
public java.util.List QueryTagByFilter(java.lang.String effectiveUserId,  
                                         java.lang.String tagNameMask,  
                                         java.lang.String descriptionMask,  
                                         java.lang.String dataStoreName,  
                                         java.lang.String collectorName,  
                                         java.lang.String customTypeNameMask,  
                                         int maxNumber)  
                                         throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the qualified tag list by multiple search criteria.

Parameters:

effectiveUserId - User name against which operation will be performed.

tagNameMask - This is a name mask pattern used to retrieve list of tags which matches pattern. The mask can include wildcards, such as asterisks (*).

descriptionMask - This is a description mask pattern used to retrieve list of tags which matches pattern. The mask can include wildcards, such as asterisks (*).

dataStoreName - Data store name where tag data is getting stored.

collectorName - Collector Name collecting tag.

customTypeNameMask - This is a Custom type mask pattern used to retrieve list of tags which matches pattern. The mask can include wildcards, such as asterisks (*).

maxNumber - Maximum tag number that can be queried and also managed on the server side (1000 by default). If the maxNumber parameter is greater than the max tag query number setting in server side, the query complies with the server side setting.

Returns:

List of TagPropertyBase

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryTagByFilter

```
public java.util.List QueryTagByFilter(java.lang.String historianServerName,  
                                         java.lang.String effectiveUserId,  
                                         java.lang.String tagNameMask,  
                                         java.lang.String descriptionMask,  
                                         java.lang.String dataStoreName,  
                                         java.lang.String collectorName,  
                                         java.lang.String customTypeNameMask,  
                                         int maxNumber)  
                                         throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving the qualified tag list by multiple search criteria.

Parameters:

historianServerName - Name of target historian server.

effectiveUserId - User name against which operation will be performed.

tagNameMask - This is a name mask pattern used to retrieve list of tags which matches pattern. The mask can include wildcards, such as asterisks (*).

descriptionMask - This is a description mask pattern used to retrieve list of tags which matches pattern. The mask can include wildcards, such as asterisks (*).

dataStoreName - Data store name where tag data is getting stored.

collectorName - Collector Name collecting tag.

customTypeNameMask - This is a Custom type mask pattern used to retrieve list of tags which matches pattern. The mask can include wildcards, such as asterisks (*).

maxNumber - Maximum tag number that can be queried and also managed on the server side (1000 by default). If the maxNumber parameter is greater than the max tag query number setting in server side, the query complies with the server side setting.

Returns:

List of TagPropertyBase

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryTagByPage

```
public java.util.List QueryTagByPage(java.lang.String tagName,  
                                       int Direction,  
                                       int PageSize,  
                                       int PagesBefore,  
                                       int PagesAfter)  
                                       throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving number of tags in chunks or pages.

Parameters:

tagName - Name of tag after that tag list is returned. Is this is empty, than tag list is returned from begining.

Direction - forward seek = 1, backward seek = 2. The mask can include wildcards, such as asterisks (*).

PageSize - Number of tags retrieved in one call, tune this as per RAM availability of System.

PagesBefore - 0, for forward move.

PagesAfter - 1, for forward move.

Returns:

List of TagPropertyBase

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryTagByPage

```
public java.util.List QueryTagByPage(java.lang.String effectiveUserId,  
                                       java.lang.String tagName,  
                                       int Direction,  
                                       int PageSize,  
                                       int PagesBefore,  
                                       int PagesAfter)  
                                       throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving number of tags in chunks or pages.

Parameters:

effectiveUserId - User name against which operation will be performed.

tagName - Name of tag after that tag list is returned. If this is empty, then tag list is returned from beginning.

Direction - forward seek = 1, backward seek = 2. The mask can include wildcards, such as asterisks (*).

PageSize - Number of tags retrieved in one call, tune this as per RAM availability of System.

PagesBefore - 0, for forward move.

PagesAfter - 1, for forward move.

Returns:

List of TagPropertyBase

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryTagByPage

```
public java.util.List QueryTagByPage(java.lang.String historianServerName,  
                                       java.lang.String effectiveUserId,  
                                       java.lang.String tagName,  
                                       int Direction,  
                                       int PageSize,  
                                       int PagesBefore,  
                                       int PagesAfter)  
                                       throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for retrieving number of tags in chunks or pages.

Parameters:

historianServerName - Name of target historian server.

effectiveUserId - User name against which operation will be performed.

tagName - Name of tag after that tag list is returned. If this is empty, then tag list is returned from beginning.

Direction - forward seek = 1, backward seek = 2. The mask can include wildcards, such as asterisks (*).

PageSize - Number of tags retrieved in one call, tune this as per RAM availability of System.

PagesBefore - 0, for forward move.

PagesAfter - 1, for forward move.

Returns:

List of TagPropertyBase

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RenameTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
RenameTag(java.lang.String oldName,  
           java.lang.String newName,  
           boolean  
trueRename)  
           throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for renaming a tag.

Parameters:

oldName - old Name of tag.

newName - new name of tag.

trueRename - set to true for permanent rename.

Returns:

TagProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RenameTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
RenameTag(java.lang.String effectiveUserId,
java.lang.String oldName,
java.lang.String newName,
trueRename)
boolean
throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for renaming a tag.

Parameters:

effectiveUserId - User name against which operation will be performed.
oldName - old Name of tag.
newName - new name of tag.
trueRename - set to true for permanent rename.

Returns:

TagProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RenameTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
RenameTag(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String oldName,
java.lang.String newName,
trueRename)
boolean
throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for renaming a tag.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
oldName - old Name of tag.
newName - new name of tag.
trueRename - set to true for permanent rename.

Returns:

TagProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

TagExists

```
public boolean TagExists(java.lang.String tagName)
    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for checking tag exist or not.

Parameters:

tagName - tag name.

Returns:

boolean

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

TagExists

```
public boolean TagExists(java.lang.String effectiveUserId,
    java.lang.String tagName)
    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for checking tag exist or not.

Parameters:

effectiveUserId - User name against which operation will be performed.
tagName - tag name.

Returns:

boolean

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

TagExists

```
public boolean TagExists(java.lang.String historianServerName,  
                          java.lang.String effectiveUserId,  
                          java.lang.String tagName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for checking tag exist or not.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
tagName - tag name.

Returns:

boolean

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
UpdateTag(java.lang.String tagName,  
           com.ge.ip.hds.historian.DataContracts.TagProperty tagProperty)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for updating tag property of a tag.

Parameters:

tagName - Name of tag.
tagProperty - updated TagProperty.

Returns:

TagProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
UpdateTag(java.lang.String effectiveUserId,
java.lang.String tagName,
com.ge.ip.hds.historian.DataContracts.TagProperty tagProperty)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for updating tag property of a tag.

Parameters:

effectiveUserId - User name against which operation will be performed.
tagName - Name of tag.
tagProperty - updated TagProperty.

Returns:

TagProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
UpdateTag(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String tagName,
com.ge.ip.hds.historian.DataContracts.TagProperty tagProperty)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for updating tag property of a tag.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
tagName - Name of tag.
tagProperty - updated TagProperty.

Returns:

TagProperty

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class TagServiceImpl

```
java.lang.Object
|
+--com.ge.ip.hds.historian.API.TagServiceImpl
```

All Implemented Interfaces:

[TagService](#)

< [Constructors](#) > < [Methods](#) >

```
public class TagServiceImpl
extends java.lang.Object
implements TagService
```

Constructors

TagServiceImpl

```
public TagServiceImpl()
```

Methods

AddTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
AddTag(com.ge.ip.hds.historian.DataContracts.TagProperty tagProperty)
```

AddTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
AddTag(java.lang.String effectiveUserId,
com.ge.ip.hds.historian.DataContracts.TagProperty tagProperty)
```

AddTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
AddTag(java.lang.String historianServerName,
java.lang.String effectiveUserId,
com.ge.ip.hds.historian.DataContracts.TagProperty tagProperty)
```

BrowseTag

```
public java.util.List BrowseTag(java.lang.String tagNameMask,
int maxNumber)
```

BrowseTag

```
public java.util.List BrowseTag(java.lang.String effectiveUserId,
java.lang.String tagNameMask,
int maxNumber)
```

BrowseTag

```
public java.util.List BrowseTag(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String tagNameMask,
int maxNumber)
```

CopyTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
CopyTag(java.lang.String tagName,
java.lang.String copyTagName)
```

CopyTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
CopyTag(java.lang.String effectiveUserId,
java.lang.String tagName,
java.lang.String copyTagName)
```

CopyTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
CopyTag(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String tagName,  
java.lang.String copyTagName)
```

DeleteTag

```
public void DeleteTag(java.lang.String tagName,  
boolean permanentDelete)
```

DeleteTag

```
public void DeleteTag(java.lang.String effectiveUserId,  
java.lang.String tagName,  
boolean permanentDelete)
```

DeleteTag

```
public void DeleteTag(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String tagName,  
boolean permanentDelete)
```

GetSupportedTagCriterias

```
public java.lang.String GetSupportedTagCriterias()
```

GetSupportedTagCriterias

```
public java.lang.String GetSupportedTagCriterias(java.lang.String  
effectiveUserId)
```

QueryTagByPage

```
public java.util.List QueryTagByPage(java.lang.String effectiveUserId,  
                                       java.lang.String tagName,  
                                       int Direction,  
                                       int PageSize,  
                                       int PagesBefore,  
                                       int PagesAfter)
```

QueryTagByPage

```
public java.util.List QueryTagByPage(java.lang.String historianServerName,  
                                       java.lang.String effectiveUserId,  
                                       java.lang.String tagName,  
                                       int Direction,  
                                       int PageSize,  
                                       int PagesBefore,  
                                       int PagesAfter)
```

RenameTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
RenameTag(java.lang.String oldName,  
           java.lang.String newName,  
           boolean  
trueRename)
```

RenameTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty  
RenameTag(java.lang.String effectiveUserId,  
           java.lang.String oldName,  
           java.lang.String newName,  
           boolean  
trueRename)
```

RenameTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
RenameTag(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String oldName,
java.lang.String newName,
trueRename) boolean
```

TagExists

```
public boolean TagExists(java.lang.String tagName)
```

TagExists

```
public boolean TagExists(java.lang.String effectiveUserId,
java.lang.String tagName)
```

TagExists

```
public boolean TagExists(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String tagName)
```

UpdateTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
UpdateTag(java.lang.String tagName,
com.ge.ip.hds.historian.DataContracts.TagProperty tagProperty)
```

UpdateTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
UpdateTag(java.lang.String effectiveUserId,
java.lang.String tagName,
com.ge.ip.hds.historian.DataContracts.TagProperty tagProperty)
```

UpdateTag

```
public com.ge.ip.hds.historian.DataContracts.TagProperty
UpdateTag(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String tagName,
com.ge.ip.hds.historian.DataContracts.TagProperty tagProperty)
```

com.ge.ip.hds.historian.API

Interface TagsService

< [Methods](#) >

```
public interface TagsService
```

TagsService

The TagsService Class implements interfaces for multiple Tags operations.

Methods

AddTags

```
public java.util.List AddTags(java.lang.String historianServerName,
                             java.lang.String effectiveUserId,
                             java.util.List tagsProperty)
                             throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for creating multiple tags.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
tagsProperty - List of TagProperty.

Returns:

List of ItemError

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddTags

```
public java.util.List AddTags(java.lang.String effectiveUserId,  
                                java.util.List tagsProperty)  
                                throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for creating multiple tags.

Parameters:

effectiveUserId - User name against which operation will be performed.
tagsProperty - List of TagProperty.

Returns:

List of ItemError

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddTags

```
public java.util.List AddTags(java.util.List tagsProperty)  
                                throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is used for creating multiple tags.

Parameters:

tagsProperty - List of TagProperty.

Returns:

List of ItemError

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class TagsServiceImpl

```
java.lang.Object  
|  
+--com.ge.ip.hds.historian.API.TagsServiceImpl
```

All Implemented Interfaces:

[TagsService](#)

< [Constructors](#) > < [Methods](#) >

```
public class TagsServiceImpl  
extends java.lang.Object  
implements TagsService
```

Constructors

TagsServiceImpl

```
public TagsServiceImpl()
```

Methods

AddTags

```
public java.util.List AddTags(java.lang.String historianServerName,  
                             java.lang.String effectiveUserId,  
                             java.util.List tagsProperty)
```

AddTags

```
public java.util.List AddTags(java.lang.String effectiveUserId,  
                             java.util.List tagsProperty)
```

AddTags

```
public java.util.List AddTags(java.util.List tagsProperty)
```

com.ge.ip.hds.historian.API

Interface UserDefinedTypeService

< [Methods](#) >

```
public interface UserDefinedTypeService
```

UserDefinedTypeService

The UserDefinedTypeService Class implements interfaces for dealing with UserDefinedType.

Methods

AddUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType  
AddUserDefinedType(com.ge.ip.hds.historian.DataContracts.UserDefinedType  
userDefinedType)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

throws

This method is for adding a UserDefinedType.

Parameters:

userDefinedType - UserDefinedType structure.

Returns:

UserDefinedType structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType  
AddUserDefinedType(java.lang.String effectiveUserId,  
com.ge.ip.hds.historian.DataContracts.UserDefinedType userDefinedType)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

throws

This method is for adding a UserDefinedType.

Parameters:

effectiveUserId - User name against which operation will be performed.
userDefinedType - UserDefinedType structure.

Returns:

UserDefinedType structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

AddUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType
AddUserDefinedType( java.lang.String historianServerName,
 java.lang.String effectiveUserId,
 com.ge.ip.hds.historian.DataContracts.UserDefinedType userDefinedType)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for adding a UserDefinedType.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
userDefinedType - UserDefinedType structure.

Returns:

UserDefinedType structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteUserDefinedType

```
public boolean DeleteUserDefinedType( java.lang.String userDefinedTypeName)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting a UserDefinedType.

Parameters:

userDefinedTypeName - Name of UserDefinedType.

Returns:

true if success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteUserDefinedType

```
public boolean DeleteUserDefinedType( java.lang.String effectiveUserId,  
                                     java.lang.String userDefinedTypeName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting a UserDefinedType.

Parameters:

effectiveUserId - User name against which operation will be performed.
userDefinedTypeName - Name of UserDefinedType.

Returns:

true if success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

DeleteUserDefinedType

```
public boolean DeleteUserDefinedType( java.lang.String historianServerName,  
                                     java.lang.String effectiveUserId,  
                                     java.lang.String userDefinedTypeName)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for deleting a UserDefinedType.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
userDefinedTypeName - Name of UserDefinedType.

Returns:

true if success.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType  
GetUserDefinedType(java.lang.String userDefinedTypeName) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the structure of UserDefinedType.

Parameters:

userDefinedTypeName - Name of UserDefinedType.

Returns:

UserDefinedType structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType  
GetUserDefinedType(java.lang.String effectiveUserId,  
java.lang.String userDefinedTypeName) throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the structure of UserDefinedType.

Parameters:

effectiveUserId - User name against which operation will be performed.
userDefinedTypeName - Name of UserDefinedType.

Returns:

UserDefinedType structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

GetUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType
GetUserDefinedType( java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String userDefinedTypeName)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving the structure of UserDefinedType.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
userDefinedTypeName - Name of UserDefinedType.

Returns:

UserDefinedType structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryUserDefinedTypes

```
public java.util.ArrayList QueryUserDefinedTypes( java.lang.String
userDefinedTypeNameMask)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of User define types which matches User define type name mask..

Parameters:

userDefinedTypeNameMask - This is a name mask pattern used to retrieve list of User define types which matches pattern. The mask can include wildcards, such as asterisks (*).

Returns:

ArrayList of UserDefinedType.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryUserDefinedTypes

```
public java.util.ArrayList QueryUserDefinedTypes( java.lang.String
effectiveUserId,
                                                    java.lang.String
userDefinedTypeNameMask)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of User define types which matches User define type name mask..

Parameters:

effectiveUserId - User name against which operation will be performed.
userDefinedTypeNameMask - This is a name mask pattern used to retrieve list of User define types which matches pattern. The mask can include wildcards, such as asterisks (*).

Returns:

ArrayList of UserDefinedType.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

QueryUserDefinedTypes

```
public java.util.ArrayList QueryUserDefinedTypes( java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String
userDefinedTypeNameMask)
                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for retrieving list of User define types which matches User define type name mask..

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
userDefinedTypeNameMask - This is a name mask pattern used to retrieve list of User define types which matches pattern. The mask can include wildcards, such as asterisks (*).

Returns:

ArrayList of UserDefinedType.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RenameUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType  
RenameUserDefinedType(java.lang.String oldUserDefinedTypeName,  
java.lang.String newUserDefinedTypeName)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

throws

This method is for renaming a UserDefinedType.

Parameters:

oldUserDefinedTypeName - Old Name of UserDefinedType.
newUserDefinedTypeName - New Name of UserDefinedType.

Returns:

UserDefinedType structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RenameUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType  
RenameUserDefinedType(java.lang.String effectiveUserId,  
java.lang.String oldUserDefinedTypeName,  
java.lang.String newUserDefinedTypeName)  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

throws

This method is for renaming a UserDefinedType.

Parameters:

effectiveUserId - User name against which operation will be performed.
oldUserDefinedTypeName - Old Name of UserDefinedType.
newUserDefinedTypeName - New Name of UserDefinedType.

Returns:

UserDefinedType structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

RenameUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType
RenameUserDefinedType(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String oldUserDefinedTypeName,
java.lang.String newUserDefinedTypeName)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for renaming a UserDefinedType.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
oldUserDefinedTypeName - Old Name of UserDefinedType.
newUserDefinedTypeName - New Name of UserDefinedType.

Returns:

UserDefinedType structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType
UpdateUserDefinedType(java.lang.String userDefinedTypeName,
com.ge.ip.hds.historian.DataContracts.UserDefinedType userDefinedType)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for updating a UserDefinedType.

Parameters:

userDefinedTypeName - Name of UserDefinedType.
userDefinedType - UserDefinedType structure.

Returns:

UserDefinedType structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType
UpdateUserDefinedType(java.lang.String effectiveUserId,
java.lang.String userDefinedTypeName,
com.ge.ip.hds.historian.DataContracts.UserDefinedType userDefinedType)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for updating a UserDefinedType.

Parameters:

effectiveUserId - User name against which operation will be performed.
userDefinedTypeName - Name of UserDefinedType.
userDefinedType - UserDefinedType structure.

Returns:

UserDefinedType structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

UpdateUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType
UpdateUserDefinedType(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String userDefinedTypeName,
com.ge.ip.hds.historian.DataContracts.UserDefinedType userDefinedType)
                                                                    throws
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for updating a UserDefinedType.

Parameters:

historianServerName - Name of target historian server.
effectiveUserId - User name against which operation will be performed.
userDefinedTypeName - Name of UserDefinedType.
userDefinedType - UserDefinedType structure.

Returns:

UserDefinedType structure.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class UserDefinedTypeServiceImpl

```
java.lang.Object
|
+--com.ge.ip.hds.historian.API.UserDefinedTypeServiceImpl
```

All Implemented Interfaces:

[UserDefinedTypeService](#)

< [Constructors](#) > < [Methods](#) >

```
public class UserDefinedTypeServiceImpl
    extends java.lang.Object
    implements UserDefinedTypeService
```

Constructors

UserDefinedTypeServiceImpl

```
public UserDefinedTypeServiceImpl()
```

Methods

AddUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType
AddUserDefinedType(com.ge.ip.hds.historian.DataContracts.UserDefinedType
    userDefinedType)
```

AddUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType
AddUserDefinedType(java.lang.String effectiveUserId,
    com.ge.ip.hds.historian.DataContracts.UserDefinedType userDefinedType)
```

AddUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType
AddUserDefinedType(java.lang.String historianServerName,
java.lang.String effectiveUserId,
com.ge.ip.hds.historian.DataContracts.UserDefinedType userDefinedType)
```

DeleteUserDefinedType

```
public boolean DeleteUserDefinedType(java.lang.String userDefinedTypeName)
```

DeleteUserDefinedType

```
public boolean DeleteUserDefinedType(java.lang.String effectiveUserId,
java.lang.String userDefinedTypeName)
```

DeleteUserDefinedType

```
public boolean DeleteUserDefinedType(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String userDefinedTypeName)
```

GetUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType
GetUserDefinedType(java.lang.String userDefinedTypeName)
```

GetUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType
GetUserDefinedType(java.lang.String effectiveUserId,
java.lang.String userDefinedTypeName)
```

GetUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType  
GetUserDefinedType(java.lang.String historianServerName,  
java.lang.String effectiveUserId,  
java.lang.String userDefinedTypeName)
```

QueryUserDefinedTypes

```
public java.util.ArrayList QueryUserDefinedTypes(java.lang.String  
userDefinedTypeNameMask)
```

QueryUserDefinedTypes

```
public java.util.ArrayList QueryUserDefinedTypes(java.lang.String  
effectiveUserId,  
java.lang.String  
userDefinedTypeNameMask)
```

QueryUserDefinedTypes

```
public java.util.ArrayList QueryUserDefinedTypes(java.lang.String  
historianServerName,  
java.lang.String  
effectiveUserId,  
java.lang.String  
userDefinedTypeNameMask)
```

RenameUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType  
RenameUserDefinedType(java.lang.String oldUserDefinedTypeName,  
java.lang.String newUserDefinedTypeName)
```

RenameUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType  
RenameUserDefinedType(java.lang.String effectiveUserId,  
java.lang.String oldUserDefinedTypeName,  
java.lang.String newUserDefinedTypeName)
```

RenameUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType
RenameUserDefinedType(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String oldUserDefinedTypeName,
java.lang.String newUserDefinedTypeName)
```

UpdateUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType
UpdateUserDefinedType(java.lang.String userDefinedTypeName,
com.ge.ip.hds.historian.DataContracts.UserDefinedType userDefinedType)
```

UpdateUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType
UpdateUserDefinedType(java.lang.String effectiveUserId,
java.lang.String userDefinedTypeName,
com.ge.ip.hds.historian.DataContracts.UserDefinedType userDefinedType)
```

UpdateUserDefinedType

```
public com.ge.ip.hds.historian.DataContracts.UserDefinedType
UpdateUserDefinedType(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String userDefinedTypeName,
com.ge.ip.hds.historian.DataContracts.UserDefinedType userDefinedType)
```

com.ge.ip.hds.historian.API

Interface UserService

< [Methods](#) >

public interface **UserService**

UserService

The UserService Class implements interfaces for User Authentication.

Methods

Authenticate

```
public boolean Authenticate(java.lang.String serverName,  
                             java.lang.String userId,  
                             java.lang.String credentials)  
    throws  
com.ge.ip.hds.historian.DataContracts.HistorianOperationException
```

This method is for checking that a User is authenticated for connecting to a Historian server or not.

Parameters:

serverName - Name of Historian Server.
userId - User name.
credentials - User password.

Returns:

true if it is valid user.

Throws:

com.ge.ip.hds.historian.DataContracts.HistorianOperationException - on API error

com.ge.ip.hds.historian.API

Class UserServiceImpl

```
java.lang.Object  
|  
+--com.ge.ip.hds.historian.API.UserServiceImpl
```

All Implemented Interfaces:

[UserService](#)

< [Constructors](#) > < [Methods](#) >

```
public class UserServiceImpl  
extends java.lang.Object  
implements UserService
```

Constructors

UserServiceImpl

```
public UserServiceImpl()
```

Methods

Package

com.ge.ip.hds.historian.API.NativeFunctions

Class Summary

[NativeFunctionBridge](#)

com.ge.ip.hds.historian.API.NativeFunctions

Class NativeFunctionBridge

```
java.lang.Object
|
|--com.ge.ip.hds.historian.API.NativeFunctions.NativeFunctionBridge
```

< [Constructors](#) > < [Methods](#) >

```
public class NativeFunctionBridge
extends java.lang.Object
```

Constructors

NativeFunctionBridge

```
public NativeFunctionBridge()
```

Methods

AddArchive

```
public static native java.lang.String AddArchive(java.lang.String
historianServerName,
effectiveUserId,
uriBaseAddress,
archiveString)
java.lang.String
java.lang.String
java.lang.String
```

AddCollector

```
public static native java.lang.String AddCollector(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String
uriBaseAddress,
                                                    java.lang.String
collectorProperty)
```

AddDHSService

```
public static native java.lang.String AddDHSService(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String
uriBaseAddress,
                                                    java.lang.String
dHSServiceString)
```

AddDHSServiceNode

```
public static native java.lang.String AddDHSServiceNode(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String
uriBaseAddress,
                                                    java.lang.String
nodeName,
                                                    java.lang.String
storageName,
                                                    boolean replicateNode)
```

AddDHSStorage

```
public static native java.lang.String AddDHSStorage(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String
uriBaseAddress,
                                                    java.lang.String
dHSStorageString)
```

AddDataStore

```
public static native java.lang.String AddDataStore(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String
uriBaseAddress,
                                                    java.lang.String
dataStoreString)
```

AddEnumeratedSet

```
public static native java.lang.String AddEnumeratedSet(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String
uriBaseAddress,
                                                    java.lang.String
enumeratedSetString)
```

AddHistorianServer

```
public static native java.lang.String AddHistorianServer(java.lang.String
historianServerName,
                                                    java.lang.String
userName,
                                                    java.lang.String
password,
                                                    boolean
authorizationEnable)
```

AddHistorianServerByDefaultCredential

```
public static native java.lang.String
AddHistorianServerByDefaultCredential(java.lang.String historianServerName,
boolean authorizationEnable)
```

AddInterfaceDef

```
public static native java.lang.String AddInterfaceDef(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String
uriBaseAddress,
                                                    java.lang.String
intDefPropsString)
```

AddMessage

```
public static native java.lang.String AddMessage(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String
messageString)
```

AddPublishTag

```
public static native java.lang.String AddPublishTag(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String
uriBaseAddress,
                                                    java.lang.String
collectorName,
                                                    java.lang.String
sourceTagName,
                                                    java.lang.String
destinationTagName)
```

AddTag

```
public static native java.lang.String AddTag(java.lang.String
historianServerName,
                                                    java.lang.String effectiveUserId,
                                                    java.lang.String uriBaseAddress,
                                                    java.lang.String
tagPropertyString)
```

DeleteTag

```
public static native java.lang.String DeleteTag(java.lang.String
historianServerName,
                                                java.lang.String
effectiveUserId,
                                                java.lang.String tagName,
                                                boolean permanentDelete)
```

DeleteTagData

```
public static native java.lang.String DeleteTagData(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String tagName,
                                                    java.lang.String
timeStamp)
```

DeleteUserDefinedType

```
public static native java.lang.String DeleteUserDefinedType(java.lang.String
historianServerName,
                                                            java.lang.String
effectiveUserId,
                                                            java.lang.String
userDefinedTypeName)
```

GetAlarmInterfaceProperties

```
public static native java.lang.String
GetAlarmInterfaceProperties(java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String uriBaseAddress,
java.lang.String alarmInterfaceName)
```

GetInterfacePerformanceCounter

```
public static native java.lang.String
GetInterfacePerformanceCounter( java.lang.String historianServerName,
java.lang.String effectiveUserId,
java.lang.String uriBaseAddress,
java.lang.String collectorNameMask,
onlyProblematic,
includeTagCount)
```

boolean
boolean

GetInterpolatedValue

```
public static native java.lang.String GetInterpolatedValue( java.lang.String
historianServerName,
effectiveUserId,
tagNames,
start,
end,
multiTagsQuery)
```

java.lang.String
java.lang.String
java.lang.String
java.lang.String
long intervalMs,
boolean

GetJobStatus

```
public static native java.lang.String GetJobStatus( java.lang.String
historianServerName,
effectiveUserId,
uriBaseAddress,
serviceName,
jobId)
```

java.lang.String
java.lang.String
java.lang.String
java.lang.String

GetJobs

```
public static native java.lang.String GetJobs( java.lang.String
historianServerName,
effectiveUserId,
uriBaseAddress,
includeFinishedJob)
```

java.lang.String
boolean

GetTagProperties

```
public static native java.lang.String GetTagProperties(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String
uriBaseAddress,
                                                    java.lang.String
tagName,
                                                    java.lang.String
tagPropertyFieldsString)
```

GetTrendData

```
public static native java.lang.String GetTrendData(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String tagName,
                                                    java.lang.String start,
                                                    java.lang.String end,
                                                    int count,
                                                    int samplingMode,
                                                    long intervalMs,
                                                    int direction,
                                                    java.lang.String
filterExpression,
                                                    int calculationMode,
                                                    long queryModifier,
                                                    int filterMode,
                                                    java.lang.String
statisticsItemFilter,
                                                    boolean multiTagsQuery)
```

GetUserDefinedType

```
public static native java.lang.String GetUserDefinedType(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String
uriBaseAddress,
                                                    java.lang.String
userDefinedTypeName)
```

QueryTagByPage

```
public static native java.lang.String QueryTagByPage(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String
uriBaseAddress,
                                                    java.lang.String tagName,
int Direction,
int PageSize,
int PagesBefore,
int PagesAfter)
```

RebuildCollectorList

```
public static native java.lang.String RebuildCollectorList(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String
interfaceName)
```

ReloadCollectorTag

```
public static native java.lang.String ReloadCollectorTag(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String
collectorName,
                                                    java.lang.String
tagNamesString,
                                                    java.lang.String
startTimeString,
                                                    java.lang.String
endTimeString,
boolean
multiTagsQuery)
```

SetMaxConnectionPoolSize

```
public static native java.lang.String SetMaxConnectionPoolSize(int
maxConnectionPoolSize)
```

SetPublishTagProperties

```
public static native java.lang.String SetPublishTagProperties(java.lang.String
historianServerName,
                                                                java.lang.String
effectiveUserId,
                                                                java.lang.String
uriBaseAddress,
                                                                java.lang.String
collectorName,
                                                                java.lang.String
tagName,
                                                                java.lang.String
tagProperty)
```

SetQuerySizeConfiguration

```
public static native java.lang.String SetQuerySizeConfiguration(int
maxDataQueryResultSize,
                                                                int
maxTagQueryResultSize,
                                                                boolean
multipleTagsSampleRead)
```

SetTag

```
public static native java.lang.String SetTag(java.lang.String
historianServerName,
                                                                java.lang.String effectiveUserId,
                                                                java.lang.String uriBaseAddress,
                                                                java.lang.String tagName,
                                                                java.lang.String
tagPropertyString)
```

UpdateCollector

```
public static native java.lang.String UpdateCollector(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String
uriBaseAddress,
                                                    java.lang.String
collectorName,
                                                    java.lang.String
collectorProperty)
```

UpdateDHSService

```
public static native java.lang.String UpdateDHSService(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String
uriBaseAddress,
                                                    java.lang.String
dHSServiceName,
                                                    java.lang.String
dHSServiceString)
```

UpdateDHSStorage

```
public static native java.lang.String UpdateDHSStorage(java.lang.String
historianServerName,
                                                    java.lang.String
effectiveUserId,
                                                    java.lang.String
uriBaseAddress,
                                                    java.lang.String
dHSStorageName,
                                                    java.lang.String
dHSStorageString)
```

UpdateInterfaceDef

```
public static native java.lang.String UpdateInterfaceDef( java.lang.String
historianServerName,
                                                                    java.lang.String
effectiveUserId,
                                                                    java.lang.String
uriBaseAddress,
                                                                    java.lang.String
interfaceName,
                                                                    java.lang.String
intDefPropsString)
```

UpdateUserDefinedType

```
public static native java.lang.String UpdateUserDefinedType( java.lang.String
historianServerName,
                                                                    java.lang.String
effectiveUserId,
                                                                    java.lang.String
uriBaseAddress,
                                                                    java.lang.String
userDefinedTypeName,
                                                                    java.lang.String
userDefinedTypeString)
```

Package com.ge.ip.hds.historianserviceapi

Class Summary

[Activator](#)

com.ge.ip.hds.historianserviceapi

Class Activator

```
java.lang.Object
|
+--com.ge.ip.hds.historianserviceapi.Activator
```

All Implemented Interfaces:

org.osgi.framework.BundleActivator

< [Constructors](#) > < [Methods](#) >

```
public class Activator
extends java.lang.Object
implements org.osgi.framework.BundleActivator
```

Constructors

Activator

```
public Activator()
```

Methods

start

```
public void start(org.osgi.framework.BundleContext context)
```

stop

```
public void stop(org.osgi.framework.BundleContext context)
```

INDEX

A

[Activator](#) ... 297
[Activator](#) ... 297
[AddArchive](#) ... 7
[AddArchive](#) ... 7
[AddArchive](#) ... 8
[AddArchive](#) ... 22
[AddArchive](#) ... 22
[AddArchive](#) ... 22
[AddArchive](#) ... 258
[AddCollector](#) ... 30
[AddCollector](#) ... 30
[AddCollector](#) ... 31
[AddCollector](#) ... 38
[AddCollector](#) ... 38
[AddCollector](#) ... 38
[AddCollector](#) ... 259
[AddDataStore](#) ... 116
[AddDataStore](#) ... 116
[AddDataStore](#) ... 117
[AddDataStore](#) ... 125
[AddDataStore](#) ... 125
[AddDataStore](#) ... 126
[AddDataStore](#) ... 260
[AddDHSService](#) ... 66
[AddDHSService](#) ... 66
[AddDHSService](#) ... 67
[AddDHSService](#) ... 77
[AddDHSService](#) ... 77
[AddDHSService](#) ... 77
[AddDHSService](#) ... 259
[AddDHSServiceNode](#) ... 67
[AddDHSServiceNode](#) ... 68
[AddDHSServiceNode](#) ... 68
[AddDHSServiceNode](#) ... 77
[AddDHSServiceNode](#) ... 77
[AddDHSServiceNode](#) ... 78
[AddDHSServiceNode](#) ... 259
[AddDHSSStorage](#) ... 81
[AddDHSSStorage](#) ... 81
[AddDHSSStorage](#) ... 82
[AddDHSSStorage](#) ... 89
[AddDHSSStorage](#) ... 89
[AddDHSSStorage](#) ... 89
[AddDHSSStorage](#) ... 259
[AddEnumeratedSet](#) ... 129
[AddEnumeratedSet](#) ... 129
[AddEnumeratedSet](#) ... 130
[AddEnumeratedSet](#) ... 138
[AddEnumeratedSet](#) ... 138
[AddEnumeratedSet](#) ... 139
[AddEnumeratedSet](#) ... 260
[AddHistorianServer](#) ... 60
[AddHistorianServer](#) ... 60
[AddHistorianServer](#) ... 260
[AddHistorianServerByDefaultCredential](#) ... 260
[AddInterfaceDef](#) ... 142
[AddInterfaceDef](#) ... 142
[AddInterfaceDef](#) ... 143
[AddInterfaceDef](#) ... 150
[AddInterfaceDef](#) ... 150
[AddInterfaceDef](#) ... 150

[AddInterfaceDef](#) ... 261
[AddMessage](#) ... 170
[AddMessage](#) ... 170
[AddMessage](#) ... 171
[AddMessage](#) ... 173
[AddMessage](#) ... 173
[AddMessage](#) ... 174
[AddMessage](#) ... 261
[AddPublishTag](#) ... 261
[AddTag](#) ... 178
[AddTag](#) ... 178
[AddTag](#) ... 179
[AddTag](#) ... 189
[AddTag](#) ... 189
[AddTag](#) ... 189
[AddTag](#) ... 213
[AddTag](#) ... 214
[AddTag](#) ... 214
[AddTag](#) ... 233
[AddTag](#) ... 233
[AddTag](#) ... 234
[AddTag](#) ... 261
[AddTagComment](#) ... 262
[AddTagEx](#) ... 262
[AddTags](#) ... 240
[AddTags](#) ... 241
[AddTags](#) ... 241
[AddTags](#) ... 242
[AddTags](#) ... 242
[AddTags](#) ... 242
[AddTags](#) ... 262
[AddTagsEx](#) ... 262
[AddUserDefinedType](#) ... 243
[AddUserDefinedType](#) ... 243
[AddUserDefinedType](#) ... 244
[AddUserDefinedType](#) ... 252
[AddUserDefinedType](#) ... 252
[AddUserDefinedType](#) ... 253
[AddUserDefinedType](#) ... 263
[AdvanceSearchTags](#) ... 263
[ArchiveService](#) ... 6
[ArchiveServiceImpl](#) ... 21
[ArchiveServiceImpl](#) ... 21
[Authenticate](#) ... 256
[Authenticate](#) ... 257
[Authenticate](#) ... 263

B

[BackupArchive](#) ... 8
[BackupArchive](#) ... 9
[BackupArchive](#) ... 9
[BackupArchive](#) ... 22
[BackupArchive](#) ... 22
[BackupArchive](#) ... 23
[BackupArchive](#) ... 153
[BackupArchive](#) ... 153
[BackupArchive](#) ... 154
[BackupArchive](#) ... 164
[BackupArchive](#) ... 164
[BackupArchive](#) ... 164
[BackupArchive](#) ... 263
[BackupArchiveAsync](#) ... 264
[BrowseCollectorTag](#) ... 46
[BrowseCollectorTag](#) ... 46
[BrowseCollectorTag](#) ... 47
[BrowseCollectorTag](#) ... 55
[BrowseCollectorTag](#) ... 55
[BrowseCollectorTag](#) ... 56
[BrowseCollectorTag](#) ... 264
[BrowseCollectorTagByPage](#) ... 47
[BrowseCollectorTagByPage](#) ... 48
[BrowseCollectorTagByPage](#) ... 48
[BrowseCollectorTagByPage](#) ... 56
[BrowseCollectorTagByPage](#) ... 56
[BrowseCollectorTagByPage](#) ... 56
[BrowseCollectorTagByPage](#) ... 264
[BrowseTag](#) ... 215
[BrowseTag](#) ... 215
[BrowseTag](#) ... 216
[BrowseTag](#) ... 234
[BrowseTag](#) ... 234
[BrowseTag](#) ... 234

C

[Cancel](#) ... 154
[Cancel](#) ... 155
[Cancel](#) ... 155
[Cancel](#) ... 164
[Cancel](#) ... 165
[Cancel](#) ... 165
[CancelJob](#) ... 265
[ChangeCollectorHistorianNode](#) ... 265
[CleanUp](#) ... 265
[CloseCurrentArchive](#) ... 10
[CloseCurrentArchive](#) ... 10
[CloseCurrentArchive](#) ... 11
[CloseCurrentArchive](#) ... 23
[CloseCurrentArchive](#) ... 23
[CloseCurrentArchive](#) ... 23
[CloseCurrentArchive](#) ... 23
[CloseCurrentArchive](#) ... 265
[CollectorBufferControl](#) ... 266
[CollectorDebugModeSet](#) ... 266
[CollectorExecuteAction](#) ... 266
[CollectorOperationsService](#) ... 27
[CollectorOperationsServiceImpl](#) ... 28
[CollectorOperationsServiceImpl](#) ... 29
[CollectorRestart](#) ... 267
[CollectorService](#) ... 29
[CollectorServiceImpl](#) ... 37
[CollectorServiceImpl](#) ... 37
[CollectorStart](#) ... 267
[CollectorStatisticsService](#) ... 40
[CollectorStatisticsServiceImpl](#) ... 44
[CollectorStatisticsServiceImpl](#) ... 44
[CollectorStop](#) ... 267
[CollectorTagService](#) ... 45
[CollectorTagServiceImpl](#) ... 55
[CollectorTagServiceImpl](#) ... 55
[ConfigurationManager](#) ... 59
[CopyTag](#) ... 216
[CopyTag](#) ... 217
[CopyTag](#) ... 217
[CopyTag](#) ... 234
[CopyTag](#) ... 234
[CopyTag](#) ... 235
[CopyTag](#) ... 268
[CreateTagComment](#) ... 268
[CreateTagData](#) ... 268
[CreateTagSample](#) ... 92
[CreateTagSample](#) ... 92
[CreateTagSample](#) ... 92
[CreateTagSample](#) ... 109
[CreateTagSample](#) ... 109
[CreateTagSample](#) ... 110
[CreateTagsData](#) ... 268
[CreateTagsSample](#) ... 93
[CreateTagsSample](#) ... 93
[CreateTagsSample](#) ... 93
[CreateTagsSample](#) ... 110
[CreateTagsSample](#) ... 110
[CreateTagsSample](#) ... 110

D

[debug](#) ... 168
[DataService](#) ... 91
[DataServiceImpl](#) ... 109
[DataServiceImpl](#) ... 109
[DataStoreService](#) ... 115
[DataStoreServiceImpl](#) ... 125
[DataStoreServiceImpl](#) ... 125
[DeleteArchive](#) ... 11
[DeleteArchive](#) ... 12
[DeleteArchive](#) ... 12
[DeleteArchive](#) ... 23
[DeleteArchive](#) ... 23
[DeleteArchive](#) ... 24
[DeleteArchive](#) ... 269
[DeleteCollector](#) ... 31
[DeleteCollector](#) ... 32
[DeleteCollector](#) ... 32
[DeleteCollector](#) ... 38
[DeleteCollector](#) ... 38
[DeleteCollector](#) ... 38
[DeleteCollector](#) ... 269
[DeleteDataStore](#) ... 117
[DeleteDataStore](#) ... 118
[DeleteDataStore](#) ... 118
[DeleteDataStore](#) ... 126
[DeleteDataStore](#) ... 126
[DeleteDataStore](#) ... 126
[DeleteDataStore](#) ... 270
[DeleteDHSService](#) ... 69
[DeleteDHSService](#) ... 69
[DeleteDHSService](#) ... 70
[DeleteDHSService](#) ... 78
[DeleteDHSService](#) ... 78
[DeleteDHSService](#) ... 78
[DeleteDHSService](#) ... 269
[DeleteDHSServiceNode](#) ... 70
[DeleteDHSServiceNode](#) ... 71
[DeleteDHSServiceNode](#) ... 71
[DeleteDHSServiceNode](#) ... 78
[DeleteDHSServiceNode](#) ... 78
[DeleteDHSServiceNode](#) ... 79
[DeleteDHSServiceNode](#) ... 269
[DeleteDHSSStorage](#) ... 82
[DeleteDHSSStorage](#) ... 83
[DeleteDHSSStorage](#) ... 83
[DeleteDHSSStorage](#) ... 89
[DeleteDHSSStorage](#) ... 89
[DeleteDHSSStorage](#) ... 89
[DeleteDHSSStorage](#) ... 270
[DeleteEnumeratedSet](#) ... 130
[DeleteEnumeratedSet](#) ... 131
[DeleteEnumeratedSet](#) ... 131
[DeleteEnumeratedSet](#) ... 139
[DeleteEnumeratedSet](#) ... 139
[DeleteEnumeratedSet](#) ... 139
[DeleteEnumeratedSet](#) ... 270
[DeleteHistorianServer](#) ... 60
[DeleteHistorianServer](#) ... 270
[DeleteInterfaceDef](#) ... 143
[DeleteInterfaceDef](#) ... 144
[DeleteInterfaceDef](#) ... 144
[DeleteInterfaceDef](#) ... 150
[DeleteInterfaceDef](#) ... 150
[DeleteInterfaceDef](#) ... 150
[DeleteInterfaceDef](#) ... 270

[DeleteTag](#) ... 218
[DeleteTag](#) ... 218
[DeleteTag](#) ... 218
[DeleteTag](#) ... 235
[DeleteTag](#) ... 235
[DeleteTag](#) ... 235
[DeleteTag](#) ... 271
[DeleteTagData](#) ... 271
[DeleteUserDefinedType](#) ... 244
[DeleteUserDefinedType](#) ... 245
[DeleteUserDefinedType](#) ... 245
[DeleteUserDefinedType](#) ... 253
[DeleteUserDefinedType](#) ... 253
[DeleteUserDefinedType](#) ... 253
[DeleteUserDefinedType](#) ... 271
[DHSService](#) ... 65
[DHSServiceImpl](#) ... 76
[DHSServiceImpl](#) ... 76
[DHSServiceImpl](#) ... 80
[DHSServiceImpl](#) ... 88
[DHSServiceImpl](#) ... 88

E

[error](#) ... 169
[EnumeratedSetService](#) ... 128
[EnumeratedSetServiceImpl](#) ... 138
[EnumeratedSetServiceImpl](#) ... 138
[ExecuteAction](#) ... 27
[ExecuteAction](#) ... 28
[ExecuteAction](#) ... 28
[ExecuteAction](#) ... 29
[ExecuteAction](#) ... 29
[ExecuteAction](#) ... 29

G

[getInstance](#) ... 65
[GetAlarmInterfaceProperties](#) ... 271
[GetAllArchives](#) ... 13
[GetAllArchives](#) ... 13
[GetAllArchives](#) ... 13
[GetAllArchives](#) ... 24
[GetAllArchives](#) ... 24
[GetAllArchives](#) ... 24
[GetArchive](#) ... 14
[GetArchive](#) ... 14
[GetArchive](#) ... 15
[GetArchive](#) ... 24
[GetArchive](#) ... 24
[GetArchive](#) ... 25
[GetArchive](#) ... 272
[GetArchiveStatistics](#) ... 15
[GetArchiveStatistics](#) ... 16
[GetArchiveStatistics](#) ... 16
[GetArchiveStatistics](#) ... 25
[GetArchiveStatistics](#) ... 25
[GetArchiveStatistics](#) ... 25
[GetArchiveStatistics](#) ... 25
[GetArchiveStatistics](#) ... 272
[GetCalculatedData](#) ... 94
[GetCalculatedData](#) ... 95
[GetCalculatedData](#) ... 96
[GetCalculatedData](#) ... 110
[GetCalculatedData](#) ... 110
[GetCalculatedData](#) ... 111
[GetCalculatedData](#) ... 272
[GetClientPerformanceCounter](#) ... 272
[GetCollector](#) ... 33
[GetCollector](#) ... 33
[GetCollector](#) ... 34
[GetCollector](#) ... 39
[GetCollector](#) ... 39
[GetCollector](#) ... 39
[GetCollector](#) ... 39
[GetCollector](#) ... 273
[GetCollectorStatistics](#) ... 273
[GetCollectorStatus](#) ... 273
[GetCollectorTagCurrentValue](#) ... 273
[GetCollectorTagProperty](#) ... 49
[GetCollectorTagProperty](#) ... 49
[GetCollectorTagProperty](#) ... 50
[GetCollectorTagProperty](#) ... 56
[GetCollectorTagProperty](#) ... 57
[GetCollectorTagProperty](#) ... 57
[GetCollectorTagProperty](#) ... 274
[GetCollectorVersion](#) ... 274
[GetComments](#) ... 274
[GetConfiguration](#) ... 197
[GetConfiguration](#) ... 197
[GetConfiguration](#) ... 198
[GetConfiguration](#) ... 209
[GetConfiguration](#) ... 209
[GetConfiguration](#) ... 209
[GetCriteriaFromString](#) ... 193
[GetCriteriaFromString](#) ... 193
[GetCriteriaFromString](#) ... 194
[GetCriteriaFromString](#) ... 196
[GetCriteriaFromString](#) ... 196
[GetCriteriaFromString](#) ... 196
[GetCriteriaFromString](#) ... 274
[GetCurrentData](#) ... 96
[GetCurrentData](#) ... 97
[GetCurrentData](#) ... 97

[GetCurrentData](#) ... 111
[GetCurrentData](#) ... 111
[GetCurrentData](#) ... 111
[GetCurrentTime](#) ... 198
[GetCurrentTime](#) ... 198
[GetCurrentTime](#) ... 199
[GetCurrentTime](#) ... 209
[GetCurrentTime](#) ... 209
[GetCurrentTime](#) ... 209
[GetCurrentValue](#) ... 50
[GetCurrentValue](#) ... 51
[GetCurrentValue](#) ... 51
[GetCurrentValue](#) ... 57
[GetCurrentValue](#) ... 57
[GetCurrentValue](#) ... 57
[GetCurrentValue](#) ... 275
[GetDataStore](#) ... 119
[GetDataStore](#) ... 119
[GetDataStore](#) ... 120
[GetDataStore](#) ... 126
[GetDataStore](#) ... 126
[GetDataStore](#) ... 127
[GetDataStorePerformanceCounter](#) ... 276
[GetDataStores](#) ... 120
[GetDataStores](#) ... 121
[GetDataStores](#) ... 121
[GetDataStores](#) ... 127
[GetDataStores](#) ... 127
[GetDataStores](#) ... 127
[GetDataStores](#) ... 127
[GetDataStores](#) ... 276
[GetDestinationServer](#) ... 179
[GetDestinationServer](#) ... 180
[GetDestinationServer](#) ... 180
[GetDestinationServer](#) ... 189
[GetDestinationServer](#) ... 189
[GetDestinationServer](#) ... 190
[GetDestinationServer](#) ... 276
[GetDHSService](#) ... 72
[GetDHSService](#) ... 72
[GetDHSService](#) ... 73
[GetDHSService](#) ... 79
[GetDHSService](#) ... 79
[GetDHSService](#) ... 79
[GetDHSService](#) ... 79
[GetDHSService](#) ... 275
[GetDHSServices](#) ... 73
[GetDHSServices](#) ... 74
[GetDHSServices](#) ... 74
[GetDHSServices](#) ... 79
[GetDHSServices](#) ... 79
[GetDHSServices](#) ... 80
[GetDHSServices](#) ... 275
[GetDHSSStorage](#) ... 84
[GetDHSSStorage](#) ... 84
[GetDHSSStorage](#) ... 85
[GetDHSSStorage](#) ... 90
[GetDHSSStorage](#) ... 90
[GetDHSSStorage](#) ... 90
[GetDHSSStorage](#) ... 275
[GetDHSSStorages](#) ... 85
[GetDHSSStorages](#) ... 86
[GetDHSSStorages](#) ... 86
[GetDHSSStorages](#) ... 90
[GetDHSSStorages](#) ... 90
[GetDHSSStorages](#) ... 90
[GetDHSSStorages](#) ... 276
[GetEnumeratedSet](#) ... 132
[GetEnumeratedSet](#) ... 132
[GetEnumeratedSet](#) ... 133
[GetEnumeratedSet](#) ... 139
[GetEnumeratedSet](#) ... 139
[GetEnumeratedSet](#) ... 140
[GetEnumeratedSet](#) ... 277
[GetHistorianServers](#) ... 61
[GetHistorianServers](#) ... 277
[GetInterfaceDef](#) ... 145
[GetInterfaceDef](#) ... 145
[GetInterfaceDef](#) ... 146
[GetInterfaceDef](#) ... 151
[GetInterfaceDef](#) ... 151
[GetInterfaceDef](#) ... 151
[GetInterfaceDef](#) ... 277
[GetInterfaceDefs](#) ... 146
[GetInterfaceDefs](#) ... 147
[GetInterfaceDefs](#) ... 147
[GetInterfaceDefs](#) ... 151
[GetInterfaceDefs](#) ... 151
[GetInterfaceDefs](#) ... 151
[GetInterfaceDefs](#) ... 277
[GetInterfacePerformanceCounter](#) ... 278
[GetInterpolatedData](#) ... 98
[GetInterpolatedData](#) ... 98
[GetInterpolatedData](#) ... 99
[GetInterpolatedData](#) ... 111
[GetInterpolatedData](#) ... 112
[GetInterpolatedData](#) ... 112
[GetInterpolatedValue](#) ... 278
[GetJobs](#) ... 156
[GetJobs](#) ... 156
[GetJobs](#) ... 157
[GetJobs](#) ... 165
[GetJobs](#) ... 165
[GetJobs](#) ... 165
[GetJobs](#) ... 165
[GetJobs](#) ... 278
[GetJobStatus](#) ... 278
[GetNodePerformanceCounter](#) ... 279
[GetOption](#) ... 199
[GetOption](#) ... 200
[GetOption](#) ... 200
[GetOption](#) ... 209
[GetOption](#) ... 210
[GetOption](#) ... 210
[GetOptionsInBatch](#) ... 201
[GetOptionsInBatch](#) ... 201
[GetOptionsInBatch](#) ... 202
[GetOptionsInBatch](#) ... 210
[GetOptionsInBatch](#) ... 210
[GetOptionsInBatch](#) ... 210
[GetPerformanceCounter](#) ... 175
[GetPerformanceCounter](#) ... 175
[GetPerformanceCounter](#) ... 176
[GetPerformanceCounter](#) ... 177
[GetPerformanceCounter](#) ... 177
[GetPerformanceCounter](#) ... 177
[GetPerformanceCounter](#) ... 279
[GetPerfTagData](#) ... 279
[GetPublishTagProperties](#) ... 279
[GetQueryModifier](#) ... 194
[GetQueryModifier](#) ... 194
[GetQueryModifier](#) ... 195
[GetQueryModifier](#) ... 196
[GetQueryModifier](#) ... 196
[GetQueryModifier](#) ... 196
[GetQueryModifier](#) ... 280
[GetRawData](#) ... 280
[GetRawDataByNumber](#) ... 99
[GetRawDataByNumber](#) ... 100

[GetRawDataByNumber](#) ... 100
[GetRawDataByNumber](#) ... 112
[GetRawDataByNumber](#) ... 112
[GetRawDataByNumber](#) ... 112
[GetRawDataByTime](#) ... 101
[GetRawDataByTime](#) ... 101
[GetRawDataByTime](#) ... 102
[GetRawDataByTime](#) ... 113
[GetRawDataByTime](#) ... 113
[GetRawDataByTime](#) ... 113
[GetSampledData](#) ... 103
[GetSampledData](#) ... 104
[GetSampledData](#) ... 105
[GetSampledData](#) ... 113
[GetSampledData](#) ... 114
[GetSampledData](#) ... 114
[GetSampledData](#) ... 280
[GetSampledDataEx](#) ... 281
[GetSecurityGroupMembership](#) ... 202
[GetSecurityGroupMembership](#) ... 202
[GetSecurityGroupMembership](#) ... 203
[GetSecurityGroupMembership](#) ... 210
[GetSecurityGroupMembership](#) ... 211
[GetSecurityGroupMembership](#) ... 211
[GetSecurityGroupNames](#) ... 203
[GetSecurityGroupNames](#) ... 203
[GetSecurityGroupNames](#) ... 204
[GetSecurityGroupNames](#) ... 211
[GetSecurityGroupNames](#) ... 211
[GetSecurityGroupNames](#) ... 211
[GetStatistics](#) ... 41
[GetStatistics](#) ... 41
[GetStatistics](#) ... 42
[GetStatistics](#) ... 44
[GetStatistics](#) ... 44
[GetStatistics](#) ... 45
[GetStatistics](#) ... 204
[GetStatistics](#) ... 204
[GetStatistics](#) ... 205
[GetStatistics](#) ... 211
[GetStatistics](#) ... 211
[GetStatistics](#) ... 212
[GetStatus](#) ... 157
[GetStatus](#) ... 158
[GetStatus](#) ... 158
[GetStatus](#) ... 165
[GetStatus](#) ... 166
[GetStatus](#) ... 166
[GetSupportedTagCriterias](#) ... 219
[GetSupportedTagCriterias](#) ... 219
[GetSupportedTagCriterias](#) ... 219
[GetSupportedTagCriterias](#) ... 235
[GetSupportedTagCriterias](#) ... 235
[GetSupportedTagCriterias](#) ... 236
[GetSupportedTagCriterias](#) ... 281
[GetTag](#) ... 220
[GetTag](#) ... 220
[GetTag](#) ... 221
[GetTag](#) ... 236
[GetTag](#) ... 236
[GetTag](#) ... 236
[GetTag](#) ... 281
[GetTagAlias](#) ... 281
[GetTagAliasName](#) ... 221
[GetTagAliasName](#) ... 222
[GetTagAliasName](#) ... 222
[GetTagAliasName](#) ... 236
[GetTagAliasName](#) ... 236

[GetTagAliasName](#) ... 237
[GetTagProperties](#) ... 181
[GetTagProperties](#) ... 181
[GetTagProperties](#) ... 182
[GetTagProperties](#) ... 190
[GetTagProperties](#) ... 190
[GetTagProperties](#) ... 190
[GetTagProperties](#) ... 282
[GetTrendData](#) ... 106
[GetTrendData](#) ... 107
[GetTrendData](#) ... 108
[GetTrendData](#) ... 114
[GetTrendData](#) ... 115
[GetTrendData](#) ... 115
[GetTrendData](#) ... 282
[GetUserDefinedType](#) ... 246
[GetUserDefinedType](#) ... 246
[GetUserDefinedType](#) ... 247
[GetUserDefinedType](#) ... 253
[GetUserDefinedType](#) ... 253
[GetUserDefinedType](#) ... 254
[GetUserDefinedType](#) ... 282
[GetUserDefinedTypes](#) ... 283

H

[HierarchicalBrowse](#) ... 52
[HierarchicalBrowse](#) ... 52
[HierarchicalBrowse](#) ... 53
[HierarchicalBrowse](#) ... 58
[HierarchicalBrowse](#) ... 58
[HierarchicalBrowse](#) ... 58
[HierarchicalBrowse](#) ... 283

I

[Initialize](#) ... 61
[Initialize](#) ... 62
[InitializeLogging](#) ... 62
[InitializePluginDll](#) ... 63
[InitializePluginDllInLibrary](#) ... 63
[InterfaceDefService](#) ... 141
[InterfaceDefServiceImpl](#) ... 149
[InterfaceDefServiceImpl](#) ... 149
[IsPluginLoaded](#) ... 63

J

[JobService](#) ... 152
[JobServiceImpl](#) ... 163
[JobServiceImpl](#) ... 164

L

[LogManager](#) ... 168
[LogManager](#) ... 168

M

[MessageService](#) ... 169
[MessageServiceImpl](#) ... 173
[MessageServiceImpl](#) ... 173

N

[NativeFunctionBridge](#) ... 258
[NativeFunctionBridge](#) ... 258

P

[PerformanceCounterService](#) ... 174
[PerformanceCounterServiceImpl](#) ... 176
[PerformanceCounterServiceImpl](#) ... 176
[PublishService](#) ... 177
[PublishServiceImpl](#) ... 188
[PublishServiceImpl](#) ... 188

Q

[QueryArchives](#) ... 17
[QueryArchives](#) ... 17
[QueryArchives](#) ... 18
[QueryArchives](#) ... 25
[QueryArchives](#) ... 25
[QueryArchives](#) ... 26
[QueryArchives](#) ... 283
[QueryCollector](#) ... 34
[QueryCollector](#) ... 35
[QueryCollector](#) ... 35
[QueryCollector](#) ... 39
[QueryCollector](#) ... 39
[QueryCollector](#) ... 39
[QueryCollector](#) ... 284
[QueryEnumeratedSet](#) ... 133
[QueryEnumeratedSet](#) ... 134
[QueryEnumeratedSet](#) ... 134
[QueryEnumeratedSet](#) ... 140
[QueryEnumeratedSet](#) ... 140
[QueryEnumeratedSet](#) ... 140
[QueryEnumeratedSet](#) ... 284
[QueryMessage](#) ... 171
[QueryMessage](#) ... 172
[QueryMessage](#) ... 172
[QueryMessage](#) ... 174
[QueryMessage](#) ... 174
[QueryMessage](#) ... 174
[QueryMessage](#) ... 284
[QueryModifierService](#) ... 192
[QueryModifierServiceImpl](#) ... 195
[QueryModifierServiceImpl](#) ... 195
[QueryPublishTag](#) ... 285
[QueryTag](#) ... 182
[QueryTag](#) ... 183
[QueryTag](#) ... 183
[QueryTag](#) ... 190
[QueryTag](#) ... 190
[QueryTag](#) ... 191
[QueryTag](#) ... 285
[QueryTagByFilter](#) ... 223
[QueryTagByFilter](#) ... 224
[QueryTagByFilter](#) ... 225
[QueryTagByFilter](#) ... 237
[QueryTagByFilter](#) ... 237
[QueryTagByFilter](#) ... 237
[QueryTagByFilter](#) ... 285
[QueryTagByPage](#) ... 226
[QueryTagByPage](#) ... 227
[QueryTagByPage](#) ... 228
[QueryTagByPage](#) ... 237
[QueryTagByPage](#) ... 238
[QueryTagByPage](#) ... 238
[QueryTagByPage](#) ... 286
[QueryUserDefinedTypes](#) ... 247
[QueryUserDefinedTypes](#) ... 248
[QueryUserDefinedTypes](#) ... 248
[QueryUserDefinedTypes](#) ... 254
[QueryUserDefinedTypes](#) ... 254
[QueryUserDefinedTypes](#) ... 254

R

[RebuildCollectorList](#) ... 286
[Reload](#) ... 53
[Reload](#) ... 54
[Reload](#) ... 54
[Reload](#) ... 58
[Reload](#) ... 59
[Reload](#) ... 59
[ReloadCollectorTag](#) ... 286
[RemovePublishTag](#) ... 287
[RemoveTag](#) ... 184
[RemoveTag](#) ... 184
[RemoveTag](#) ... 185
[RemoveTag](#) ... 191
[RemoveTag](#) ... 191
[RemoveTag](#) ... 191
[RenameDataStore](#) ... 122
[RenameDataStore](#) ... 122
[RenameDataStore](#) ... 123
[RenameDataStore](#) ... 127
[RenameDataStore](#) ... 127
[RenameDataStore](#) ... 128
[RenameDataStore](#) ... 287
[RenameEnumeratedSet](#) ... 135
[RenameEnumeratedSet](#) ... 135
[RenameEnumeratedSet](#) ... 136
[RenameEnumeratedSet](#) ... 140
[RenameEnumeratedSet](#) ... 140
[RenameEnumeratedSet](#) ... 141
[RenameEnumeratedSet](#) ... 287
[RenameTag](#) ... 228
[RenameTag](#) ... 229
[RenameTag](#) ... 229
[RenameTag](#) ... 238
[RenameTag](#) ... 238
[RenameTag](#) ... 239
[RenameTag](#) ... 287
[RenameUserDefinedType](#) ... 249
[RenameUserDefinedType](#) ... 249
[RenameUserDefinedType](#) ... 250
[RenameUserDefinedType](#) ... 254
[RenameUserDefinedType](#) ... 254
[RenameUserDefinedType](#) ... 255
[RenameUserDefinedType](#) ... 288
[ReplicateNode](#) ... 159
[ReplicateNode](#) ... 159
[ReplicateNode](#) ... 160
[ReplicateNode](#) ... 166
[ReplicateNode](#) ... 166
[ReplicateNode](#) ... 166
[ReplicateNode](#) ... 288
[RestoreArchive](#) ... 18
[RestoreArchive](#) ... 19
[RestoreArchive](#) ... 19
[RestoreArchive](#) ... 26
[RestoreArchive](#) ... 26
[RestoreArchive](#) ... 26
[RestoreArchive](#) ... 160
[RestoreArchive](#) ... 161
[RestoreArchive](#) ... 161
[RestoreArchive](#) ... 167
[RestoreArchive](#) ... 167
[RestoreArchive](#) ... 167
[RestoreArchive](#) ... 288
[RestoreArchiveAsync](#) ... 289
[ResyncArchive](#) ... 162

[ResyncArchive](#) ... 162
[ResyncArchive](#) ... 163
[ResyncArchive](#) ... 167
[ResyncArchive](#) ... 167
[ResyncArchive](#) ... 168
[ResyncArchiveFile](#) ... 289

S

[setLogger](#) ... 169
[start](#) ... 297
[stop](#) ... 297
[ServerGetOption](#) ... 289
[ServerGetOptionInBatch](#) ... 290
[ServerGetProperty](#) ... 290
[ServerService](#) ... 196
[ServerServiceImpl](#) ... 208
[ServerServiceImpl](#) ... 208
[ServerSetOption](#) ... 290
[ServerSetOptionInBatch](#) ... 290
[SetCollectorStatistics](#) ... 291
[SetConfigure](#) ... 291
[SetDestinationServer](#) ... 185
[SetDestinationServer](#) ... 186
[SetDestinationServer](#) ... 186
[SetDestinationServer](#) ... 191
[SetDestinationServer](#) ... 191
[SetDestinationServer](#) ... 192
[SetDestinationServer](#) ... 291
[SetLoggingConfigure](#) ... 64
[SetLoggingConfigure](#) ... 291
[SetMaxConnectionPoolSize](#) ... 64
[SetMaxConnectionPoolSize](#) ... 292
[SetOption](#) ... 205
[SetOption](#) ... 206
[SetOption](#) ... 206
[SetOption](#) ... 212
[SetOption](#) ... 212
[SetOption](#) ... 212
[SetOptionsInBatch](#) ... 207
[SetOptionsInBatch](#) ... 207
[SetOptionsInBatch](#) ... 208
[SetOptionsInBatch](#) ... 212
[SetOptionsInBatch](#) ... 213
[SetOptionsInBatch](#) ... 213
[SetPublishTagProperties](#) ... 292
[SetQuerySizeConfiguration](#) ... 64
[SetQuerySizeConfiguration](#) ... 292
[SetStatistics](#) ... 42
[SetStatistics](#) ... 43
[SetStatistics](#) ... 43
[SetStatistics](#) ... 45
[SetStatistics](#) ... 45
[SetStatistics](#) ... 45
[SetTag](#) ... 292
[SetTagProperties](#) ... 187
[SetTagProperties](#) ... 187
[SetTagProperties](#) ... 188
[SetTagProperties](#) ... 192
[SetTagProperties](#) ... 192
[SetTagProperties](#) ... 192
[SetTagProperties](#) ... 293

T

[TagExists](#) ... 230
[TagExists](#) ... 230
[TagExists](#) ... 231
[TagExists](#) ... 239
[TagExists](#) ... 239
[TagExists](#) ... 239
[TagExists](#) ... 293
[TagNameAggregate](#) ... 115
[TagService](#) ... 213
[TagServiceImpl](#) ... 233
[TagServiceImpl](#) ... 233
[TagsList](#) ... 293
[TagsService](#) ... 240
[TagsServiceImpl](#) ... 241
[TagsServiceImpl](#) ... 242

U

[UpdateArchive](#) ... 20
[UpdateArchive](#) ... 20
[UpdateArchive](#) ... 21
[UpdateArchive](#) ... 26
[UpdateArchive](#) ... 27
[UpdateArchive](#) ... 27
[UpdateArchive](#) ... 293
[UpdateCollector](#) ... 36
[UpdateCollector](#) ... 36
[UpdateCollector](#) ... 37
[UpdateCollector](#) ... 40
[UpdateCollector](#) ... 40
[UpdateCollector](#) ... 40
[UpdateCollector](#) ... 294
[UpdateDataStore](#) ... 123
[UpdateDataStore](#) ... 124
[UpdateDataStore](#) ... 124
[UpdateDataStore](#) ... 128
[UpdateDataStore](#) ... 128
[UpdateDataStore](#) ... 128
[UpdateDataStore](#) ... 295
[UpdateDHSService](#) ... 75
[UpdateDHSService](#) ... 75
[UpdateDHSService](#) ... 76
[UpdateDHSService](#) ... 80
[UpdateDHSService](#) ... 80
[UpdateDHSService](#) ... 80
[UpdateDHSService](#) ... 294
[UpdateDHSSStorage](#) ... 87
[UpdateDHSSStorage](#) ... 87
[UpdateDHSSStorage](#) ... 88
[UpdateDHSSStorage](#) ... 91
[UpdateDHSSStorage](#) ... 91
[UpdateDHSSStorage](#) ... 91
[UpdateDHSSStorage](#) ... 294
[UpdateEnumeratedSet](#) ... 136
[UpdateEnumeratedSet](#) ... 137
[UpdateEnumeratedSet](#) ... 137
[UpdateEnumeratedSet](#) ... 141
[UpdateEnumeratedSet](#) ... 141
[UpdateEnumeratedSet](#) ... 141
[UpdateEnumeratedSet](#) ... 295
[UpdateHistorianServer](#) ... 65
[UpdateHistorianServer](#) ... 295
[UpdateInterfaceDef](#) ... 148
[UpdateInterfaceDef](#) ... 148
[UpdateInterfaceDef](#) ... 149
[UpdateInterfaceDef](#) ... 152
[UpdateInterfaceDef](#) ... 152
[UpdateInterfaceDef](#) ... 152
[UpdateInterfaceDef](#) ... 296
[UpdateTag](#) ... 231
[UpdateTag](#) ... 232
[UpdateTag](#) ... 232
[UpdateTag](#) ... 239
[UpdateTag](#) ... 239
[UpdateTag](#) ... 240
[UpdateUserDefinedType](#) ... 250
[UpdateUserDefinedType](#) ... 251
[UpdateUserDefinedType](#) ... 251
[UpdateUserDefinedType](#) ... 255
[UpdateUserDefinedType](#) ... 255
[UpdateUserDefinedType](#) ... 255
[UpdateUserDefinedType](#) ... 296
[UserDefinedTypeService](#) ... 242

[UserDefinedTypeServiceImpl](#) ... 252

[UserDefinedTypeServiceImpl](#) ... 252

[UserService](#) ... 255

[UserServiceImpl](#) ... 256

[UserServiceImpl](#) ... 256