Honeywell

Uniformance®

PHD .NET Wrapper

User Guide

R300

Copyright, Notices, and Trademarks

Release Information

Uniformance R300

Document Creation Date: January, 2010

Document ID: pim4201

Support and Other Contacts

United States and Canada

Contact:	Honeywell Solution Support Center
	Phone: 1-800 822-7673.
	Calls are answered by dispatcher between 6:00 A.M. and 4:00 P.M. Mountain Standard Time.
	Emergency calls outside normal working hours are received by an answering service and returned within one hour.
Mail:	Honeywell HPS TAC, MS L17
	1860 W Rose Garden Ln
	Phoenix, Arizona 85027-2708

Europe

Contact:	Honeywell TAC-EMEA
Phone:	+32-2-728-2732
Facsimile:	+32-2-728-2696
Mail:	TAC-BE02
	Hermes Plaza
	Hermeslaan, 1H
	B-1831 Diegem, Belgium

Pacific

Contact:	Honeywell Global TAC – Pacific
Phone:	1300-300-4822 (toll free within Australia)
	+61-8-9362-9559 (outside Australia)
Facsimile:	+61-8-9362-9564
Mail:	Honeywell Limited Australia
	5 Kitchener Way
	Burswood 6100, Western Australia
Email:	GTAC@honeywell.com

India

Contact:	Honeywell Global TAC – India
Phone:	+91-20- 66039400
Facsimile:	+91-20- 66039800
Mail:	Honeywell Automation India Ltd.
	56 and 57, Hadapsar Industrial Estate
	Hadapsar, Pune –411 013, India
Email:	Global-TAC-India@honeywell.com

Korea

Contact:	Honeywell Global TAC – Korea
Phone:	+82-2-799-6317
	+82-11-9227-6324
Facsimile:	+82-2-792-9015
Mail:	Honeywell Co., Ltd
	4F, Sangam IT Tower B4-4 Block
	1590, DMC Sangam-dong, Mapo-gu,
	Seoul, 121-836, Korea
Email:	Global-TAC-Korea@honeywell.com

People's Republic of China

Contact:	Honeywell Global TAC – China
Phone:	+86- 21-52574568
Mail:	Honeywell (China) Co., Ltd
	33/F, Tower A, City Center, 100 Zunyi Rd.
	Shanghai 200051, People's Republic of China
Email:	Global-TAC-China@honeywell.com

Singapore

Contact:	Global TAC – South East Asia
Phone:	+65-6580-3500
Facsimile:	+65-6580-3501
	+65-6445-3033
Mail:	Honeywell Private Limited
	Honeywell Building
	17, Changi Business Park Central 1
	Singapore 486073
Email:	GTAC-SEA@honeywell.com

Taiwan

Contact:	Global TAC – Taiwan
Phone:	+886- 7- 536 2567
Facsimile:	+886-7-536 2039
Mail:	Honeywell Taiwan Ltd.
	17F-1, No. 260, Jhongshan 2nd Road.
	Cianjhen District
	Kaohsiung, Taiwan, ROC
Email:	Global-TAC-Taiwan@honeywell.com

Japan

Contact:	Global TAC – Japan
Phone:	+81-3-6730-7160
Facsimile:	+81-3-6730-7228
Mail:	Honeywell Japan Inc.
	New Pier Takeshiba, South Tower Building,
	20th Floor, 1-16-1 Kaigan, Minato-ku,
	Tokyo 105-0022, Japan
Email:	Global-TAC-JapanJA25@honeywell.com

Elsewhere

Call your nearest Honeywell office.

World Wide Web

Honeywell Solution Support Online: http://www.honeywell.com/ps

Training Classes

Honeywell Automation College: http://www.automationcollege.com

Contents

Contents

1. Introduction

1.1 About this Guide

The PHD .NET Wrapper is available for users developing custom applications using Microsoft .NET technology to provide access to PHD data. This guide contains information and an example on communicating with the PHD .NET Wrapper.

1.2 Overview

Honeywell's Process History Database (PHD) provides real-time data capture, historization, and intelligent process data retrieval required for plant information system support. The result of extensive experience with interfacing real-time systems with plant information systems, PHD forms a comprehensive platform that supports Honeywell's entire family of integrated plant information system modules.

Various methods of accessing information are designed to allow data retrieval from PHD. As the Microsoft.NET technology is popular in recent times, the PHD .NET Wrapper is developed for users who design their client application tools in the .NET Framework environment.

The Wrapper is a set of methods and properties that communicate efficiently with the PHD Server native API calls to fetch data. The next chapter includes a sample program as a quick starting point for developers, a section that elaborates in detail on the methods and attributes available from the Wrapper.

2. Usage

2.1 Example

The following is an example on how to use the PHD .NET Wrapper. This example illustrates a typical three-stepped sequence that is normally found in an application that requires interfacing to a data storage, through a support API.

1. Initializing
2. Data fetching
3. Closing

Developers can use the example as the initial guide, and subsequently expand the application using additional help from the more detailed interface information in the next section, or using the embedded Help system, or take advantage of features like IntelliSense in the Visual Studio development environment. The example is provided in C#, but can be easily adapted to another equivalent language, like VB.NET, and so on.

Sample code:

```
// Locate PHDAPINET.DLL on your system and add a reference to it into this project.

using System;
using Uniformance.PHD;     // This exposes the methods in the .NET Wrapper to this program.

namespace MySpace
{
    class MyClass
    {
        //
        // This program fetches and displays raw data from a tag in the last hour.
        //
        static void Main()
        {
            //
            // Initialize the interfacing to the .NET Wrapper
            //   and setup the type of data to fetch.
            //   PHD Server is running on the localhost.
            //
            PHDHistorian phd       = new PHDHistorian();
            phd.DefaultServer      = new PHDServer("localhost", SERVERVERSION.API200);
            phd.DefaultServer.Port = 3100;
            phd.Sampletype         = SAMPLETYPE.Raw;
            phd.ReductionType      = REDUCTIONTYPE.None;

            //
            // Fetch data.
            //
            Double [] timestamps  = null;
            Double [] values      = null;
            short  [] confidences = null;

            Tag MyTag = new Tag("TEMP.PV");     // Use your tag name here.

            phd.StartTime = "NOW-1H";
            phd.EndTime   = "NOW";

            phd.FetchData(MyTag, ref timestamps, ref values, ref confidences);

            //
            // Display data.
            //
            int count = timestamps.GetUpperBound(0);

            Console.WriteLine("Retrieved {0} data points.",count);
            Console.WriteLine();

            for (int i = 0;i < count; ++i)
            {
                Console.WriteLine("{0}) {1}, {2}, {3}",
                    i+1,
                    DateTime.FromOADate(timestamps[i]),
                    values[i],
                    confidences[i]);
            }

            //
            // Clean up and quit.
            //
            phd.Dispose();
        }
    }
}
```

Output:

```
Retrieved 60 data points.

1) 2009-12-04 09:36:00 am, 115.513603210449, 100
2) 2009-12-04 09:37:00 am, 127.670715332031, 100
3) 2009-12-04 09:38:00 am, 153.79655456543, 100
...
...
58) 2009-12-04 10:33:00 am, 109.614990234375, 100
59) 2009-12-04 10:34:00 am, 126.125640869141, 100
60) 2009-12-04 10:35:00 am, 146.692733764648, 100
```

2.2   The .NET Wrapper Interface – Methods and Properties

Contents

Uniformance.PHD

    LinkListStruct Class

        LinkListStruct Members

        LinkListStruct Constructor

        Properties

            LinkName Property

    ParentTaglistStruct Class

        ParentTaglistStruct Members

        ParentTaglistStruct Constructor

        Properties

            Tagname Property

            Tagno Property

    PHDataStruct Class

        PHDataStruct Members

        PHDataStruct Constructor

        Properties

            Confidence Property

            ErrorMessage Property

            HostName Property

            TagName Property

            TimeStamp Property

            Tolerance Property

            Units Property

            Value Property

    PHDErrorException Class

        PHDErrorException Members

PHDErrorException Constructor

  PHDErrorException Constructor ()

  PHDErrorException Constructor (Int32)

Properties

  Message Property

Methods

  ToString Method

PHDHistorian Class

  PHDHistorian Members

  PHDHistorian Constructor

  Properties

    ConnectionTimeout Property

    DefaultServer Property

    EndTime Property

    MaximumRows Property

    MinimumConfidence Property

    MostRecentRows Property

    Offset Property

    ReductionFrequency Property

    ReductionOffset Property

    ReductionType Property

    RetryPeriod Property

    SampleFrequency Property

    Sampletype Property

    StartTime Property

    UseSampleFrequency Property

    UTCDateTime Property

  Methods

    BrowsingTags Method

BrowsingTags Method (UInt32, TagFilter)

BrowsingTags Method (PHDServer, UInt32, TagFilter)

BrowsingTagsList Method

BrowsingTagsList Method (UInt32, TagFilter)

BrowsingTagsList Method (PHDServer, UInt32, TagFilter)

ClearTagUpdateList Method

ClearTagUpdateList Method ()

ClearTagUpdateList Method (PHDServer)

ConvertToPHDTime Method

ConvertToPHDTime Method (DateTime)

ConvertToPHDTime Method (String, Boolean)

ConvertToPHDTime Method (PHDServer, String, Boolean)

DeleteTagData Method

DeleteTagData Method (Tag, DateTime)

DeleteTagData Method (Tag, String)

Dispose Method

Dispose Method ()

Dispose Method (Boolean)

EnableTagUpdateNotification Method

EnableTagUpdateNotification Method (Boolean)

EnableTagUpdateNotification Method (PHDServer, Boolean)

EngUnitConversion Method

EngUnitConversion Method (String, String, Double, Double, Double, Int32)

EngUnitConversion Method (PHDServer, String, String, Double, Double, Double, Int32)

FetchAllReductionData Method

FetchAllReductionData Method (String, Boolean)

FetchAllReductionData Method (Tags, Boolean)

FetchColumnData Method

GetRDIs Method

  GetRDIs Method ()

  GetRDIs Method (PHDServer)

ModifyTag Method

  ModifyTag Method (Tag, Object)

  ModifyTag Method (Tag, Object, DateTime, SByte, String)

  ModifyTag Method (Tag, Object, String)

  ModifyTag Method (Tag, Object, String, SByte)

  ModifyTag Method (Tag, Object, String, SByte, String)

PutData Method

  PutData Method (Tag, Object)

  PutData Method (Tag, Object, DateTime, SByte, String)

  PutData Method (Tag, Object, String)

  PutData Method (Tag, Object, String, SByte)

  PutData Method (Tag, Object, String, SByte, String)

RelativeToAbsolute Method

  RelativeToAbsolute Method (String, DateTime)

  RelativeToAbsolute Method (String, DateTime, Boolean)

  RelativeToAbsolute Method (String, DateTime, PHDServer)

StringToUniversalTime Method

  StringToUniversalTime Method (String)

  StringToUniversalTime Method (String, PHDServer)

TagDfn Method

  TagDfn Method (String)

  TagDfn Method (Tags)

TagUpdateHandler Method

  TagUpdateHandler Method (PHDServer, TagUpdateNotify)

  TagUpdateHandler Method (TagUpdateNotify)

ValidateCalc Method

[?] CollectorName Property

[?] DataLength Property

[?] DataType Property

[?] ParentTagname Property

[?] ParentTagno Property

[?] SourceTagname Property

[?] SourceUnits Property

[?] Tagname Property

[?] Tagno Property

[?] Units Property

[?] Tags Class

[?] Tags Members

[?] Tags Constructor

[?] Properties

[?] Count Property

[?] Item Property

[?] Item Property (Int32)

[?] Item Property (String)

[?] Methods

[?] Add Method

[?] GetEnumerator Method

[?] Remove Method

[?] RemoveAll Method

[?] TagUpdateNotify Delegate

PHD .NET Wrapper	

Uniformance.PHD Namespace

Namespace hierarchy

Classes

Class	Description
LinkListStruct	PHD link to external systems such as experion etc.
ParentTaglistStruct	Parent Taglist
PHDataStruct	FetchStructData returns an ArrayList of PHDataStruct. The List can return results from multiple tags.
PHDErrorException	PHD Exception Class
PHDHistorian	Main Class
PHDServer	PHDServer class describes a connection to a PHD Server
RDIListStruct	
Tag	Describes a single Tag
TagFilter	A structure is used to specify tag filter conditions
TagListStruct	
Tags	The Tags class is used to hold a collection of Tag items

Delegates

Delegate	Description
TagUpdateNotify	Notification from PHD server when tags are updated

Enumerations

Enumeration	Description
REDUCTIONOFFSET	Identifies the offset applied when retrieving reductions for tags.
REDUCTIONTYPE	Identifies the type of reduction fetched for the tags..
SAMPLETYPE	Sampling types supported.
SERVERVERSION	Enum list if Server API Supported

PHD .NET Wrapper	

LinkListStruct Class

PHD link to external systems such as experion etc.

For a list of all members of this type, see LinkListStruct Members.

System.Object
   Uniformance.PHD.LinkListStruct

```
[Visual Basic]
Public Class LinkListStruct
[C#]
public class LinkListStruct
```

Thread Safety

Public static (**Shared** in Visual Basic) members of this type are safe for multithreaded operations. Instance members are **not** guaranteed to be thread-safe.

Requirements

Namespace: Uniformance.PHD

Assembly: PHDAPINET (in PHDAPINET.dll)

See Also

LinkListStruct Members | Uniformance.PHD Namespace

PHD .NET Wrapper	

LinkListStruct Members

LinkListStruct overview

Public Instance Constructors

LinkListStruct Constructor	Create an instance of the class.

Public Instance Properties

LinkName	PHD Tagname

Public Instance Methods

Equals (inherited from **Object**)	Determines whether the specified Object is equal to the current **Object**.

▤◆GetHashCode (inherited from **Object**)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data structures like a hash table.
▤◆GetType (inherited from **Object**)	Gets the Type of the current instance.
▤◆ToString (inherited from **Object**)	Returns a String that represents the current Object.

Protected Instance Methods

▧◆Finalize (inherited from **Object**)	Allows an Object to attempt to free resources and perform other cleanup operations before the **Object** is reclaimed by garbage collection.
▧◆MemberwiseClone (inherited from **Object**)	Creates a shallow copy of the current Object.

See Also

LinkListStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

LinkListStruct Constructor

Create an instance of the class.

```
[Visual Basic]
Overloads Public Sub New()
[C#]
public LinkListStruct();
```

See Also

LinkListStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

LinkListStruct Properties

The properties of the **LinkListStruct** class are listed below. For a complete list of **LinkListStruct** class members, see the LinkListStruct Members topic.

Public Instance Properties

🖾LinkName	PHD Tagname

See Also

LinkListStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

LinkListStruct.LinkName Property

PHD Tagname

```
[Visual Basic]
Public ReadOnly Property LinkName As String
[C#]
public string LinkName {get;}
```

See Also

LinkListStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

ParentTaglistStruct Class

Parent Taglist

For a list of all members of this type, see ParentTaglistStruct Members.

System.Object
   Uniformance.PHD.ParentTaglistStruct

```
[Visual Basic]
Public Class ParentTaglistStruct
[C#]
public class ParentTaglistStruct
```

Thread Safety

Public static (**Shared** in Visual Basic) members of this type are safe for multithreaded operations. Instance members are **not** guaranteed to be thread-safe.

Requirements

Namespace: Uniformance.PHD

Assembly: PHDAPINET (in PHDAPINET.dll)

See Also

ParentTaglistStruct Members | Uniformance.PHD Namespace

PHD .NET Wrapper	

ParentTaglistStruct Members

[ParentTaglistStruct overview](#)

Public Instance Constructors

◆[ParentTaglistStruct Constructor](#)	Create an instance of the class.

Public Instance Properties

[Tagname](#)	PHD Tagname
[Tagno](#)	PHD Tagno

Public Instance Methods

◆[Equals](#) (inherited from **Object**)	Determines whether the specified [Object](#) is equal to the current **Object**.
◆[GetHashCode](#) (inherited from **Object**)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data structures like a hash table.
◆[GetType](#) (inherited from **Object**)	Gets the [Type](#) of the current instance.
◆[ToString](#) (inherited from **Object**)	Returns a [String](#) that represents the current [Object](#).

Protected Instance Methods

◆[Finalize](#) (inherited from **Object**)	Allows an [Object](#) to attempt to free resources and perform other cleanup operations before the **Object** is reclaimed by garbage collection.
◆[MemberwiseClone](#) (inherited from **Object**)	Creates a shallow copy of the current [Object](#).

See Also

[ParentTaglistStruct Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper	

ParentTaglistStruct Constructor

Create an instance of the class.

```
[Visual Basic]
Overloads Public Sub New()
[C#]
public ParentTaglistStruct();
```

See Also

ParentTaglistStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper

ParentTaglistStruct Properties

The properties of the **ParentTaglistStruct** class are listed below. For a complete list of **ParentTaglistStruct** class members, see the ParentTaglistStruct Members topic.

Public Instance Properties

Tagname	PHD Tagname
Tagno	PHD Tagno

See Also

ParentTaglistStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper

ParentTaglistStruct.Tagname Property

PHD Tagname

```
[Visual Basic]
Public ReadOnly Property Tagname As String
[C#]
public string Tagname {get;}
```

See Also

ParentTaglistStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper

ParentTaglistStruct.Tagno Property

PHD Tagno

```
[Visual Basic]
Public ReadOnly Property Tagno As Integer
[C#]
public int Tagno {get;}
```

See Also

ParentTaglistStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDataStruct Class

FetchStructData returns an ArrayList of PHDataStruct. The List can return results from multiple tags.

For a list of all members of this type, see PHDataStruct Members.

System.Object
  Uniformance.PHD.PHDataStruct

```
[Visual Basic]
Public Class PHDataStruct
[C#]
public class PHDataStruct
```

Thread Safety

Public static (**Shared** in Visual Basic) members of this type are safe for multithreaded operations. Instance members are **not** guaranteed to be thread-safe.

Example

This sample shows how return an ArrayList of PHDataStruct objects

```
PHDataStruct str;

ArrayList lst = oPhd.FetchStructData(oTags)

for(int i=0;i<lst.Count;i++)

{

    str=(PHDataStruct)lst[i];

    if(str.TagName!="kst20000")break;

}
```

Requirements

Namespace: Uniformance.PHD

Assembly: PHDAPINET (in PHDAPINET.dll)

See Also

PHDataStruct Members | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDataStruct Members

[PHDataStruct overview](#)

Public Instance Constructors

[PHDataStruct Constructor](#)	Create an instance of the class.

Public Instance Properties

[Confidence](#)	PHD Confidence
[ErrorMessage](#)	Error Message if there is an error. null indicates there is no error.
[HostName](#)	PHD Hostname
[TagName](#)	PHD Tagname
[TimeStamp](#)	PHD Timestamp
[Tolerance](#)	Tolerance for reductions
[Units](#)	Engineering Units
[Value](#)	PHD Value

Public Instance Methods

[Equals](#) (inherited from **Object**)	Determines whether the specified [Object](#) is equal to the current **Object**.
[GetHashCode](#) (inherited from **Object**)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data structures like a hash table.
[GetType](#) (inherited from **Object**)	Gets the [Type](#) of the current instance.
[ToString](#) (inherited from **Object**)	Returns a [String](#) that represents the current [Object](#).

Protected Instance Methods

[Finalize](#) (inherited from **Object**)	Allows an [Object](#) to attempt to free resources and perform other cleanup operations before the **Object** is reclaimed by garbage collection.
[MemberwiseClone](#) (inherited from **Object**)	Creates a shallow copy of the current [Object](#).

See Also

PHDataStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDataStruct Constructor

Create an instance of the class.

```
[Visual Basic]
Overloads Public Sub New()

[C#]
public PHDataStruct();
```

See Also

PHDataStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDataStruct Properties

The properties of the **PHDataStruct** class are listed below. For a complete list of **PHDataStruct** class members, see the PHDataStruct Members topic.

Public Instance Properties

Confidence	PHD Confidence
ErrorMessage	Error Message if there is an error. null indicates there is no error.
HostName	PHD Hostname
TagName	PHD Tagname
TimeStamp	PHD Timestamp
Tolerance	Tolerance for reductions
Units	Engineering Units
Value	PHD Value

See Also

PHDataStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDDataStruct.Confidence Property

PHD Confidence

```
[Visual Basic]
Public ReadOnly Property Confidence As SByte
[C#]
public sbyte Confidence {get;}
```

See Also

PHDDataStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDDataStruct.ErrorMessage Property

Error Message if there is an error. null indicates there is no error.

```
[Visual Basic]
Public ReadOnly Property ErrorMessage As String
[C#]
public string ErrorMessage {get;}
```

See Also

PHDDataStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDDataStruct.HostName Property

PHD Hostname

```
[Visual Basic]
Public ReadOnly Property HostName As String
[C#]
public string HostName {get;}
```

See Also

PHDDataStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDDataStruct.TagName Property

PHD Tagname

```
[Visual Basic]
Public ReadOnly Property TagName As String
[C#]
public string TagName {get;}
```

See Also

PHDataStruct Class | Uniformance.PHD Namespace

| *PHD .NET Wrapper* | |

PHDataStruct.TimeStamp Property

PHD Timestamp

```
[Visual Basic]
Public ReadOnly Property TimeStamp As Date
[C#]
public System.DateTime TimeStamp {get;}
```

See Also

PHDataStruct Class | Uniformance.PHD Namespace

| *PHD .NET Wrapper* | |

PHDataStruct.Tolerance Property

Tolerance for reductions

```
[Visual Basic]
Public ReadOnly Property Tolerance As Double
[C#]
public double Tolerance {get;}
```

See Also

PHDataStruct Class | Uniformance.PHD Namespace

| *PHD .NET Wrapper* | |

PHDataStruct.Units Property

Engineering Units

```
[Visual Basic]
Public ReadOnly Property Units As String
[C#]
public string Units {get;}
```

See Also

[PHDataStruct Class](#) | [Uniformance.PHD Namespace](#)

| *PHD .NET Wrapper* | |

PHDataStruct.Value Property

PHD Value

```
[Visual Basic]
Public ReadOnly Property Value As Object
[C#]
public object Value {get;}
```

See Also

[PHDataStruct Class](#) | [Uniformance.PHD Namespace](#)

| *PHD .NET Wrapper* | |

PHDErrorException Class

PHD Exception Class

For a list of all members of this type, see [PHDErrorException Members](#).

[System.Object](#)
    [System.Exception](#)
        Uniformance.PHD.PHDErrorException

```
[Visual Basic]
Public Class PHDErrorException
    Inherits Exception
[C#]
public class PHDErrorException : Exception
```

Thread Safety

Public static (**Shared** in Visual Basic) members of this type are safe for multithreaded operations. Instance members are **not** guaranteed to be thread-safe.

Requirements

Namespace: [Uniformance.PHD](#)

Assembly: PHDAPINET (in PHDAPINET.dll)

See Also

[PHDErrorException Members](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper	

PHDErrorException Members

PHDErrorException overview

Public Instance Constructors

PHDErrorException	Overloaded. Initializes a new instance of the PHDErrorException class.

Public Instance Properties

HelpLink (inherited from **Exception**)	Gets or sets a link to the help file associated with this exception.
InnerException (inherited from **Exception**)	Gets the Exception instance that caused the current exception.
Message	Custom Exception Message
Source (inherited from **Exception**)	Gets or sets the name of the application or the object that causes the error.
StackTrace (inherited from **Exception**)	Gets a string representation of the frames on the call stack at the time the current exception was thrown.
TargetSite (inherited from **Exception**)	Gets the method that throws the current exception.

Public Instance Methods

Equals (inherited from **Object**)	Determines whether the specified Object is equal to the current **Object**.
GetBaseException (inherited from **Exception**)	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions.
GetHashCode (inherited from **Object**)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data structures like a hash table.
GetObjectData (inherited from **Exception**)	When overridden in a derived class, sets the SerializationInfo with information about the exception.
GetType (inherited from **Object**)	Gets the Type of the current instance.
ToString	Custom Detail exception Information

Protected Instance Properties

HResult (inherited from **Exception**)	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception.

Protected Instance Methods

Finalize (inherited from **Object**)	Allows an Object to attempt to free resources and perform other cleanup operations before the **Object** is reclaimed by garbage collection.
MemberwiseClone (inherited from **Object**)	Creates a shallow copy of the current Object.

See Also

PHDErrorException Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDErrorException Constructor

Default constructor

Overload List

Default constructor

public PHDErrorException();

PHDAPI Exception handler

public PHDErrorException(int);

See Also

PHDErrorException Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDErrorException Constructor ()

Default constructor

```
[Visual Basic]
Overloads Public Sub New()
[C#]
public PHDErrorException();
```

See Also

PHDErrorException Class | Uniformance.PHD Namespace | PHDErrorException Constructor Overload List

PHD .NET Wrapper	

PHDErrorException Constructor (Int32)

PHDAPI Exception handler

```
[Visual Basic]
Overloads Public Sub New( _
    ByVal ErrorNo As Integer _
)
[C#]
public PHDErrorException(
    int ErrorNo
);
```

Parameters

ErrorNo

PHD Error Number

See Also

PHDErrorException Class | Uniformance.PHD Namespace | PHDErrorException Constructor Overload List

PHD .NET Wrapper	

PHDErrorException Properties

The properties of the **PHDErrorException** class are listed below. For a complete list of **PHDErrorException** class members, see the PHDErrorException Members topic.

Public Instance Properties

HelpLink (inherited from **Exception**)	Gets or sets a link to the help file associated with this exception.
InnerException (inherited from **Exception**)	Gets the Exception instance that caused the current exception.
Message	Custom Exception Message
Source (inherited from **Exception**)	Gets or sets the name of the application or the object that causes the error.
StackTrace (inherited from **Exception**)	Gets a string representation of the frames on the call stack at the time the current exception was thrown.
TargetSite (inherited from **Exception**)	Gets the method that throws the current exception.

Protected Instance Properties

HResult (inherited from **Exception**)	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception.

See Also

PHDErrorException Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDErrorException.Message Property

Custom Exception Message

```
[Visual Basic]
Overrides Public ReadOnly Property Message As String
[C#]
public override string Message {get;}
```

See Also

PHDErrorException Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDErrorException Methods

The methods of the **PHDErrorException** class are listed below. For a complete list of **PHDErrorException** class members, see the PHDErrorException Members topic.

Public Instance Methods

Equals (inherited from **Object**)	Determines whether the specified Object is equal to the current **Object**.
GetBaseException (inherited from **Exception**)	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions.
GetHashCode (inherited from **Object**)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data structures like a hash table.
GetObjectData (inherited from **Exception**)	When overridden in a derived class, sets the SerializationInfo with information about the exception.
GetType (inherited from **Object**)	Gets the Type of the current instance.
ToString	Custom Detail exception Information

Protected Instance Methods

Finalize (inherited from **Object**)	Allows an Object to attempt to free resources and perform other cleanup operations before the **Object** is reclaimed by garbage collection.
MemberwiseClone (inherited from **Object**)	Creates a shallow copy of the current Object.

See Also

PHDErrorException Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDErrorException.ToString Method

Custom Detail exception Information

```
[Visual Basic]
Overrides Public Function ToString() As String
[C#]
public override string ToString();
```

Return Value

Detail Error information

See Also

PHDErrorException Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian Class

Main Class

For a list of all members of this type, see PHDHistorian Members.

System.Object
   Uniformance.PHD.PHDHistorian

```
[Visual Basic]
Public Class PHDHistorian
  Implements IDisposable
[C#]
public class PHDHistorian : IDisposable
```

Thread Safety

Public static (**Shared** in Visual Basic) members of this type are safe for multithreaded operations. Instance members are **not** guaranteed to be thread-safe.

Example

Create a new instance of PHDHistorian.

```
PHDHistorian oPhd = new PHDHistorian(false);
```

Requirements

Namespace: Uniformance.PHD

Assembly: PHDAPINET (in PHDAPINET.dll)

See Also

PHDHistorian Members | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian Members

PHDHistorian overview

Public Instance Constructors

PHDHistorian Constructor	This Class is the entry point for Accessing the PHD Server through the 150, 200 and R200 API's

Public Instance Properties

ConnectionTimeout	The ConnectionTimeout property identifies the network timeout period used when attempting to connect to the PHD server. If the connection to the PHD server is established within the time period, the request is abandoned. It is in milliseconds.
DefaultServer	Attach a PHDServer object to the Historian
EndTime	Assign a default End Time
MaximumRows	Specifies the maximum number of rows to return in a getdata call.
MinimumConfidence	Specifies the confidence threshold for values for validation. Any values that have a confidence less than these are invalid and return a NULL Value. Values are between 0 and 100 with a default value of 0.
MostRecentRows	Set or return the MostRecentRows. The most recent rows will be returned if it is set to true.

![] Offset	Set or return the Offset
![] ReductionFrequency	Set or return the Reduction Frequency
![] ReductionOffset	Set or return the Reduction Offset.
![] ReductionType	Set or return the Reduction Type. Reduction Type is an Enumerate list
![] RetryPeriod	The RetryPeriod property identifies the period after a connection timeout occurs before another connection request to the PHD server will occur. This property is related to the ConnectionTimeout property for the handling of network timeout conditions in communication with the PHD Server. It is in milliseconds.
![] SampleFrequency	Set or return the Smaple Frequency
![] Sampletype	Set or return the Sample Type. Sample Type is an Enumerate list
![] StartTime	Assign a default Start Time
![] UseSampleFrequency	Set or return the UseSampleFrequency. The SampleFrequency will be used as ReductionFrequency if it is set to true.
![] UTCDateTime	Set or Get the UTCDateTime flag. When UTCDateTime is set as true timestamp data will be returned as Universal Time else local time.

Public Instance Methods

![] BrowsingTags	Overloaded. Return a DataSet of Tags
![] BrowsingTagsList	Overloaded. Return an ArrayList of Tags
![] ClearTagUpdateList	Overloaded. Clear Tag Update List
![] ConvertToPHDTime	Overloaded. Using default server to convert the DateTime string (ts) to the format of "yyyy-MM-dd HH:mm:ss".
![] DeleteTagData	Overloaded. Delete Tag Data
![] Dispose	Overloaded. Clean up all managed and unmanaged code
![] EnableTagUpdateNotification	Overloaded. Enable/disable tag update notification callback
![] EngUnitConversion	Overloaded. Convert from one Engineering Unit to another. example ((UnitIn + PreAdd) * Mult) +

	PostAdd
⇌◆Equals (inherited from **Object**)	Determines whether the specified Object is equal to the current **Object**.
⇌◆FetchAllReductionData	Overloaded. Return Reduction Data for a Tagname
⇌◆FetchColumnData	Overloaded. Fetch a single tag of Data and Display in a Column Format
⇌◆FetchData	Fetch a single tag of Data and return results in an Array
⇌◆FetchRowData	Overloaded. Fetch a single tag of Data and return the data in a DataSet
⇌◆FetchStructData	Return all tags from the Tags Collection
⇌◆GetDBInfo	Overloaded. Get Database Information from default PHD server
⇌◆GetHashCode (inherited from **Object**)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data structures like a hash table.
⇌◆GetLinkList	Overloaded. Returns link list from the default server
⇌◆GetLinks	Overloaded. Returns link list from the default server
⇌◆GetParentTagList	Overloaded. Returns Parent Tag list from the default server
⇌◆GetParentTags	Overloaded. Returns Parent Tag list from the default server
⇌◆GetRDIInfo	Overloaded. A 200 API function that will return the RDI info
⇌◆GetRDIList	Overloaded. Returns RDI list from the default server
⇌◆GetRDIs	Overloaded. Returns RDI list from the default server
⇌◆GetType (inherited from **Object**)	Gets the Type of the current instance.
⇌◆ModifyTag	Overloaded. Modify Tag Data
⇌◆PutData	Overloaded. Insert Tag Data into a DCS for a timestamp of Now
⇌◆RelativeToAbsolute	Overloaded. Convert a PHD relative time to the absolute UTC date and time on the client. An exception will be thrown for a incorrect relative

	input.
![icon] **StringToUniversalTime**	Overloaded. Convert a PHD string/relative time to universal DateTime. The conversion will be performed on the default PHD server if the string starts with "NOW"; otherwise the conversion will be performed on the local machine.
![icon] **TagDfn**	Overloaded. Fetch a single tag of Data
![icon] **TagUpdateHandler**	Overloaded. Specify call back function when PHD Tags have been updated
![icon] **ToString** (inherited from **Object**)	Returns a String that represents the current Object.
![icon] **ValidateCalc**	Overloaded. Have the Default PHD Server Validate a function.

Protected Instance Methods

![icon] **Dispose**	Overloaded. Clean up all managed and unmanaged code
![icon] **Finalize**	destructor to clean up all managed and unmanaged code
![icon] **MemberwiseClone** (inherited from **Object**)	Creates a shallow copy of the current Object.

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian Constructor

This Class is the entry point for Accessing the PHD Server through the 150, 200 and R200 API's

```
[Visual Basic]
Public Sub New()

[C#]
public PHDHistorian();
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian Properties

The properties of the **PHDHistorian** class are listed below. For a complete list of **PHDHistorian** class members, see the PHDHistorian Members topic.

Public Instance Properties

ConnectionTimeout	The ConnectionTimeout property identifies the network timeout period used when attempting to connect to the PHD server. If the connection to the PHD server is established within the time period, the request is abandoned. It is in milliseconds.
DefaultServer	Attach a PHDServer object to the Historian
EndTime	Assign a default End Time
MaximumRows	Specifies the maximum number of rows to return in a getdata call.
MinimumConfidence	Specifies the confidence threshold for values for validation. Any values that have a confidence less than these are invalid and return a NULL Value. Values are between 0 and 100 with a default value of 0.
MostRecentRows	Set or return the MostRecentRows. The most recent rows will be returned if it is set to true.
Offset	Set or return the Offset
ReductionFrequency	Set or return the Reduction Frequency
ReductionOffset	Set or return the Reduction Offset.
ReductionType	Set or return the Reduction Type. Reduction Type is an Enumerate list
RetryPeriod	The RetryPeriod property identifies the period after a connection timeout occurs before another connection request to the PHD server will occur. This property is related to the ConnectionTimeout property for the handling of network timeout conditions in communication with the PHD Server. It is in milliseconds.
SampleFrequency	Set or return the Smaple Frequency
Sampletype	Set or return the Sample Type. Sample Type is an Enumerate list
StartTime	Assign a default Start Time
UseSampleFrequency	Set or return the UseSampleFrequency. The SampleFrequency will be used as ReductionFrequency if it is set to true.

📧UTCDateTime	Set or Get the UTCDateTime flag. When UTCDateTime is set as true timestamp data will be returned as Universal Time else local time.

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.ConnectionTimeout Property

The ConnectionTimeout property identifies the network timeout period used when attempting to connect to the PHD server. If the connection to the PHD server is established within the time period, the request is abandoned. It is in milliseconds.

```
[Visual Basic]
Public Property ConnectionTimeout As UInt32
[C#]
public uint ConnectionTimeout {get; set;}
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.DefaultServer Property

Attach a PHDServer object to the Historian

```
[Visual Basic]
Public Property DefaultServer As PHDServer
[C#]
public PHDServer DefaultServer {get; set;}
```

Example

Create a new instance of a PHDHistorian and connect to a PHD Server on the local machine.

```
VB

Dim oPhd as PHDHistorian = new PHDHistorian(false)

Dim DefaultServer as PHDServer = new PHDServer("localhost")



DefaultServer.APIVersion = Uniformance.PHD.SERVERVERSION.API200

oPhd.Server = DefaultServer

```

```c#
c#

PHDHistorian oPhd = new PHDHistorian(false);

PHDServer DefaultServer = new PHDServer("localhost");



DefaultServer.APIVersion= Uniformance.PHD.SERVERVERSION.API200;

oPhd.Server = DefaultServer;

    `
```

See Also

[PHDHistorian Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper

PHDHistorian.EndTime Property

Assign a default End Time

```
[Visual Basic]
Public Property EndTime As String
[C#]
public string EndTime {get; set;}
```

See Also

[PHDHistorian Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper

PHDHistorian.MaximumRows Property

Specifies the maximum number of rows to return in a getdata call.

```
[Visual Basic]
Public Property MaximumRows As UInt32
[C#]
public uint MaximumRows {get; set;}
```

See Also

[PHDHistorian Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper

PHDHistorian.MinimumConfidence Property

Specifies the confidence threshold for values for validation. Any values that have a confidence less than these are invalid and return a NULL Value. Values are between 0 and 100 with a default value of 0.

```
[Visual Basic]
Public Property MinimumConfidence As Integer
[C#]
public int MinimumConfidence {get; set;}
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.MostRecentRows Property

Set or return the MostRecentRows. The most recent rows will be returned if it is set to true.

```
[Visual Basic]
Public Property MostRecentRows As Boolean
[C#]
public bool MostRecentRows {get; set;}
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.Offset Property

Set or return the Offset

```
[Visual Basic]
Public Property Offset As Integer
[C#]
public int Offset {get; set;}
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.ReductionFrequency Property

Set or return the Reduction Frequency

```
[Visual Basic]
```

```
Public Property ReductionFrequency As UInt32
[C#]
public uint ReductionFrequency {get; set;}
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace

| *PHD .NET Wrapper* | |

PHDHistorian.ReductionOffset Property

Set or return the Reduction Offset.

```
[Visual Basic]
Public Property ReductionOffset As REDUCTIONOFFSET
[C#]
public REDUCTIONOFFSET ReductionOffset {get; set;}
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace

| *PHD .NET Wrapper* | |

PHDHistorian.ReductionType Property

Set or return the Reduction Type. Reduction Type is an Enumerate list

```
[Visual Basic]
Public Property ReductionType As REDUCTIONTYPE
[C#]
public REDUCTIONTYPE ReductionType {get; set;}
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace

| *PHD .NET Wrapper* | |

PHDHistorian.RetryPeriod Property

The RetryPeriod property identifies the period after a connection timeout occurs before another connection request to the PHD server will occur. This property is related to the ConnectionTimeout property for the handling of network timeout conditions in communication with the PHD Server. It is in milliseconds.

```
[Visual Basic]
Public Property RetryPeriod As UInt32
[C#]
public uint RetryPeriod {get; set;}
```

See Also

[PHDHistorian Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper	

PHDHistorian.SampleFrequency Property

Set or return the Smaple Frequency

```
[Visual Basic]
Public Property SampleFrequency As UInt32
[C#]
public uint SampleFrequency {get; set;}
```

See Also

[PHDHistorian Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper	

PHDHistorian.Sampletype Property

Set or return the Sample Type. Sample Type is an Enumerate list

```
[Visual Basic]
Public Property Sampletype As SAMPLETYPE
[C#]
public SAMPLETYPE Sampletype {get; set;}
```

See Also

[PHDHistorian Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper	

PHDHistorian.StartTime Property

Assign a default Start Time

```
[Visual Basic]
Public Property StartTime As String
[C#]
public string StartTime {get; set;}
```

See Also

[PHDHistorian Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper	

PHDHistorian.UseSampleFrequency Property

Set or return the UseSampleFrequency. The SampleFrequency will be used as ReductionFrequency if it is set to true.

```
[Visual Basic]
Public Property UseSampleFrequency As Boolean
[C#]
public bool UseSampleFrequency {get; set;}
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.UTCDateTime Property

Set or Get the UTCDateTime flag. When UTCDateTime is set as true timestamp data will be returned as Universal Time else local time.

```
[Visual Basic]
Public Property UTCDateTime As Boolean
[C#]
public bool UTCDateTime {get; set;}
```

Example

Return timestamp data in Universal Time format

```
VB

Dim oPhd as PHDHistorian = new PHDHistorian(false)

Dim DefaultServer as PHDServer = new PHDServer("localhost")



DefaultServer.APIVersion = Uniformance.PHD.SERVERVERSION.API200

oPhd.Server = DefaultServer

oPhd.UTCDateTime = True



c#

PHDHistorian oPhd = new PHDHistorian(false);

PHDServer DefaultServer = new PHDServer("localhost");

```

```
DefaultServer.APIVersion= Uniformance.PHD.SERVERVERSION.API200;

oPhd.Server = DefaultServer;

oPhd.UTCDateTime = true;

    `
```

See Also

[PHDHistorian Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper	

PHDHistorian Methods

The methods of the **PHDHistorian** class are listed below. For a complete list of **PHDHistorian** class members, see the [PHDHistorian Members](#) topic.

Public Instance Methods

[BrowsingTags](#)	Overloaded. Return a DataSet of Tags
[BrowsingTagsList](#)	Overloaded. Return an ArrayList of Tags
[ClearTagUpdateList](#)	Overloaded. Clear Tag Update List
[ConvertToPHDTime](#)	Overloaded. Using default server to convert the DateTime string (ts) to the format of "yyyy-MM-dd HH:mm:ss".
[DeleteTagData](#)	Overloaded. Delete Tag Data
[Dispose](#)	Overloaded. Clean up all managed and unmanaged code
[EnableTagUpdateNotification](#)	Overloaded. Enable/disable tag update notification callback
[EngUnitConversion](#)	Overloaded. Convert from one Engineering Unit to another. example ((UnitIn + PreAdd) * Mult) + PostAdd
[Equals](#) (inherited from **Object**)	Determines whether the specified [Object](#) is equal to the current **Object**.
[FetchAllReductionData](#)	Overloaded. Return Reduction Data for a Tagname
[FetchColumnData](#)	Overloaded. Fetch a single tag of Data and Display in a Column Format

FetchData	Fetch a single tag of Data and return results in an Array
FetchRowData	Overloaded. Fetch a single tag of Data and return the data in a DataSet
FetchStructData	Return all tags from the Tags Collection
GetDBInfo	Overloaded. Get Database Information from default PHD server
GetHashCode (inherited from **Object**)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data structures like a hash table.
GetLinkList	Overloaded. Returns link list from the default server
GetLinks	Overloaded. Returns link list from the default server
GetParentTagList	Overloaded. Returns Parent Tag list from the default server
GetParentTags	Overloaded. Returns Parent Tag list from the default server
GetRDIInfo	Overloaded. A 200 API function that will return the RDI info
GetRDIList	Overloaded. Returns RDI list from the default server
GetRDIs	Overloaded. Returns RDI list from the default server
GetType (inherited from **Object**)	Gets the Type of the current instance.
ModifyTag	Overloaded. Modify Tag Data
PutData	Overloaded. Insert Tag Data into a DCS for a timestamp of Now
RelativeToAbsolute	Overloaded. Convert a PHD relative time to the absolute UTC date and time on the client. An exception will be thrown for a incorrect relative input.
StringToUniversalTime	Overloaded. Convert a PHD string/relative time to universal DateTime. The conversion will be performed on the default PHD server if the string starts with "NOW"; otherwise the conversion will be performed on the local machine.
TagDfn	Overloaded. Fetch a single tag of Data

<img> TagUpdateHandler	Overloaded. Specify call back function when PHD Tags have been updated
<img> ToString (inherited from **Object**)	Returns a String that represents the current Object.
<img> ValidateCalc	Overloaded. Have the Default PHD Server Validate a function.

Protected Instance Methods

<img> Dispose	Overloaded. Clean up all managed and unmanaged code
<img> Finalize	destructor to clean up all managed and unmanaged code
<img> MemberwiseClone (inherited from **Object**)	Creates a shallow copy of the current Object.

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.BrowsingTags Method

Return a DataSet of Tags

Overload List

Return a DataSet of Tags

public DataSet BrowsingTags(uint,TagFilter);

Return a DataSet of Tags

public DataSet BrowsingTags(PHDServer,uint,TagFilter);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.BrowsingTags Method (UInt32, TagFilter)

Return a DataSet of Tags

```
[Visual Basic]
Overloads Public Function BrowsingTags( _
```

```
    ByVal MaxTagCount As UInt32, _
    ByVal MatchCriteria As TagFilter _
) As DataSet
[C#]
public DataSet BrowsingTags(
    uint MaxTagCount,
    TagFilter MatchCriteria
);
```

Parameters

MaxTagCount

Maximum number of tags to be returned

MatchCriteria

Tag filter conditions

Return Value

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.BrowsingTags Overload List

PHD .NET Wrapper	

PHDHistorian.BrowsingTags Method (PHDServer, UInt32, TagFilter)

Return a DataSet of Tags

```
[Visual Basic]
Overloads Public Function BrowsingTags( _
    ByVal Server As PHDServer, _
    ByVal MaxTagCount As UInt32, _
    ByVal MatchCriteria As TagFilter _
) As DataSet
[C#]
public DataSet BrowsingTags(
    PHDServer Server,
    uint MaxTagCount,
    TagFilter MatchCriteria
);
```

Parameters

Server

PHDServer object

MaxTagCount

Maximum number of tags to be returned

MatchCriteria

Tag filter conditions

Return Value

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.BrowsingTags Overload List

PHD .NET Wrapper	

PHDHistorian.BrowsingTagsList Method

Return an ArrayList of Tags

Overload List

Return an ArrayList of Tags

public ArrayList BrowsingTagsList(uint,TagFilter);

Return an ArrayList of Tags

public ArrayList BrowsingTagsList(PHDServer,uint,TagFilter);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.BrowsingTagsList Method (UInt32, TagFilter)

Return an ArrayList of Tags

```
[Visual Basic]
Overloads Public Function BrowsingTagsList( _
   ByVal MaxTagCount As UInt32, _
   ByVal MatchCriteria As TagFilter _
) As ArrayList
[C#]
public ArrayList BrowsingTagsList(
   uint MaxTagCount,
   TagFilter MatchCriteria
);
```

Parameters

MaxTagCount

Maximum number of tags to be returned

MatchCriteria

Tag filter conditions

Return Value

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.BrowsingTagsList Overload List

PHD .NET Wrapper	

PHDHistorian.BrowsingTagsList Method (PHDServer, UInt32, TagFilter)

Return an ArrayList of Tags

```
[Visual Basic]
Overloads Public Function BrowsingTagsList( _
    ByVal Server As PHDServer, _
    ByVal MaxTagCount As UInt32, _
    ByVal MatchCriteria As TagFilter _
) As ArrayList
[C#]
public ArrayList BrowsingTagsList(
    PHDServer Server,
    uint MaxTagCount,
    TagFilter MatchCriteria
);
```

Parameters

Server

PHDServer object

MaxTagCount

Maximum number of tags to be returned

MatchCriteria

Tag filter conditions

Return Value

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.BrowsingTagsList Overload List

PHD .NET Wrapper	

PHDHistorian.ClearTagUpdateList Method

Clear Tag Update List

Overload List

Clear Tag Update List

public void ClearTagUpdateList();

Clear Tag Update List

public void ClearTagUpdateList(PHDServer);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.ClearTagUpdateList Method ()

Clear Tag Update List

```
[Visual Basic]
Overloads Public Sub ClearTagUpdateList()
[C#]
public void ClearTagUpdateList();
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.ClearTagUpdateList Overload List

PHD .NET Wrapper	

PHDHistorian.ClearTagUpdateList Method (PHDServer)

Clear Tag Update List

```
[Visual Basic]
Overloads Public Sub ClearTagUpdateList( _
    ByVal Server As PHDServer _
)
[C#]
public void ClearTagUpdateList(
    PHDServer Server
);
```

Parameters

Server
PHDServer object

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.ClearTagUpdateList Overload List

PHD .NET Wrapper	

PHDHistorian.ConvertToPHDTime Method

Convert DateTime to its string representation in the formant of "yyyy-MM-dd HH:mm:ss".

Overload List

Convert DateTime to its string representation in the formant of "yyyy-MM-dd HH:mm:ss".

public string ConvertToPHDTime(DateTime);

Using default server to convert the DateTime string (ts) to the format of "yyyy-MM-dd HH:mm:ss".

public string ConvertToPHDTime(string,bool);

Using the server specified to convert the DateTime string (ts) to the format of "yyyy-MM-dd HH:mm:ss".

public string ConvertToPHDTime(PHDServer,string,bool);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.ConvertToPHDTime Method (DateTime)

Convert DateTime to its string representation in the formant of "yyyy-MM-dd HH:mm:ss".

```
[Visual Basic]
Overloads Public Function ConvertToPHDTime( _
   ByVal dt As Date _
) As String
[C#]
public string ConvertToPHDTime(
   DateTime dt
);
```

Parameters

dt

DateTime to be converted.

Return Value

The string representation of dt in the form of "yyyy-MM-dd HH:mm:ss" (fromat 2). Or zero length string if ts is invalid.

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.ConvertToPHDTime Overload List

PHD .NET Wrapper	

PHDHistorian.ConvertToPHDTime Method (String, Boolean)

Using default server to convert the DateTime string (ts) to the format of "yyyy-MM-dd HH:mm:ss".

```
[Visual Basic]
Overloads Public Function ConvertToPHDTime( _
   ByVal ts As String, _
   ByVal bConvertRelative As Boolean _
) As String
[C#]
public string ConvertToPHDTime(
   string ts,
   bool bConvertRelative
);
```

Parameters

> *ts*
>
> The DateTime string to be converted.

bConvertRelative

> Indicate whether or not the relative time string should be converted.

Return Value

> PHD DateTime string in the form of "yyyy-MM-dd HH:mm:ss" (fromat 2).

Example

> Convert a DateTime format to a PHD Datetime format

```
PHDHistorian oPhd = new PHDHistorian(false);

string sOutDate="";



sOutDate = oPHD.ConvertToPHDTime("05/17/02 08:30:30", false);
```

See Also

> PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.ConvertToPHDTime Overload List

PHD .NET Wrapper	

PHDHistorian.ConvertToPHDTime Method (PHDServer, String, Boolean)

Using the server specified to convert the DateTime string (ts) to the format of "yyyy-MM-dd HH:mm:ss".

```
[Visual Basic]
Overloads Public Function ConvertToPHDTime( _
   ByVal Server As PHDServer, _
   ByVal ts As String, _
   ByVal bConvertRelative As Boolean _
) As String
[C#]
public string ConvertToPHDTime(
   PHDServer Server,
   string ts,
   bool bConvertRelative
);
```

Parameters

> *Server*
>
> The PHDServer object that specifies the PHD Server

ts

> The DateTime string to be converted.

bConvertRelative

> Indicate whether or not the relative time string should be converted.

Return Value

PHD DateTime string in the form of "yyyy-MM-dd HH:mm:ss" (fromat 2).

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.ConvertToPHDTime Overload List

PHD .NET Wrapper	

PHDHistorian.DeleteTagData Method

Delete Tag Data

Overload List

Delete Tag Data

public void DeleteTagData(Tag,DateTime);

Delete Tag Data

public void DeleteTagData(Tag,string);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.DeleteTagData Method (Tag, DateTime)

Delete Tag Data

```
[Visual Basic]
Overloads Public Sub DeleteTagData( _
    ByVal tag As Tag, _
    ByVal TimeStamp As Date _
)
[C#]
public void DeleteTagData(
    Tag tag,
    DateTime TimeStamp
);
```

Parameters

tag
Tag object

TimeStamp
UTC TimeStamp

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.DeleteTagData Overload List

PHD .NET Wrapper	

PHDHistorian.DeleteTagData Method (Tag, String)

Delete Tag Data

```
[Visual Basic]
Overloads Public Sub DeleteTagData( _
   ByVal tag As Tag, _
   ByVal TimeStamp As String _
)
[C#]
public void DeleteTagData(
   Tag tag,
   string TimeStamp
);
```

Parameters

tag

The Tag object

TimeStamp

The string timestamp

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.DeleteTagData Overload List

PHD .NET Wrapper	

PHDHistorian.Dispose Method

Clean up all managed and unmanaged code

Overload List

Clean up all managed and unmanaged code

public void Dispose();

Clean up all managed and unmanaged code

protected virtual void Dispose(bool);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.Dispose Method ()

Clean up all managed and unmanaged code

```
[Visual Basic]
NotOverridable Overloads Public Sub Dispose() _
   Implements IDisposable.Dispose
```
```
[C#]
public void Dispose();
```

Implements

IDisposable.Dispose

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.Dispose Overload List

PHD .NET Wrapper	

PHDHistorian.Dispose Method (Boolean)

Clean up all managed and unmanaged code

```
[Visual Basic]
Overridable Overloads Protected Sub Dispose( _
   ByVal disposing As Boolean _
)
```
```
[C#]
protected virtual void Dispose(
   bool disposing
);
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.Dispose Overload List

PHD .NET Wrapper	

PHDHistorian.EnableTagUpdateNotification Method

Enable/disable tag update notification callback

Overload List

Enable/disable tag update notification callback

public void EnableTagUpdateNotification(bool);

Enable/disable tag update notification callback

public void EnableTagUpdateNotification(PHDServer,bool);

See Also

[PHDHistorian Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper	

PHDHistorian.EnableTagUpdateNotification Method (Boolean)

Enable/disable tag update notification callback

```
[Visual Basic]
Overloads Public Sub EnableTagUpdateNotification( _
    ByVal Enable As Boolean _
)
[C#]
public void EnableTagUpdateNotification(
    bool Enable
);
```

Parameters

Enable

A boolean flag. Callback will be enabled if it is true; otherwise callback will be disabled.

See Also

[PHDHistorian Class](#) | [Uniformance.PHD Namespace](#) | [PHDHistorian.EnableTagUpdateNotification Overload List](#)

PHD .NET Wrapper	

PHDHistorian.EnableTagUpdateNotification Method (PHDServer, Boolean)

Enable/disable tag update notification callback

```
[Visual Basic]
Overloads Public Sub EnableTagUpdateNotification( _
    ByVal Server As PHDServer, _
    ByVal Enable As Boolean _
)
[C#]
public void EnableTagUpdateNotification(
    PHDServer Server,
    bool Enable
);
```

Parameters

Server

PHDServer object

Enable

A boolean flag. Callback will be enabled if it is true; otherwise callback will be disabled.

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.EnableTagUpdateNotification Overload List

PHD .NET Wrapper	

PHDHistorian.EngUnitConversion Method

Convert from one Engineering Unit to another. example ((UnitIn + PreAdd) * Mult) + PostAdd

Overload List

Convert from one Engineering Unit to another. example ((UnitIn + PreAdd) * Mult) + PostAdd

public void EngUnitConversion(string,string,ref double,ref double,ref double,ref int);

An overloaded EngUnitConversion method

public void EngUnitConversion(PHDServer,string,string,ref double,ref double,ref double,ref int);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.EngUnitConversion Method (String, String, Double, Double, Double, Int32)

Convert from one Engineering Unit to another. example ((UnitIn + PreAdd) * Mult) + PostAdd

```
[Visual Basic]
Overloads Public Sub EngUnitConversion( _
   ByVal UnitIn As String, _
   ByVal UnitOut As String, _
   ByRef PreAdd As Double, _
   ByRef Mult As Double, _
   ByRef PostAdd As Double, _
   ByRef pCode As Integer _
)
[C#]
public void EngUnitConversion(
   string UnitIn,
   string UnitOut,
   ref double PreAdd,
   ref double Mult,
   ref double PostAdd,
   ref int pCode
);
```

Parameters

UnitIn

The Engineering Unit that is to be converted

UnitOut

The new Engineering Unit

PreAdd

The value that is added to the UnitIn

Mult

The multiplation factor

PostAdd

The value that is added after the multiplication

pCode

Return Value

Example

```
string sEUFrom = "DEGC";

string sEUTo = "DEGF";

double dblPreAdd = 0;

double dblMult = 0;

double dblPostAdd = 0;


int iCode = 0;


try

{

        this.PHD.EngUnitConversion(sEUFrom, sEUTo, ref dblPreAdd, ref dblMult,
ref dblPostAdd, ref iCode);

}

catch(PHDErrorException ex)

{

        sMsg = "Conversion Error: " + ex.Message;

}


    txtResult.Text = "( " + sEUFrom + " + (" + dblPreAdd.ToString() + ") ) *
(" + dblMult.ToString() + ") + (" + dblPostAdd.ToString() + ") => " + sEUTo;


Result:


( DEGC + (0) ) * (1.79999999856) + (32) => DEGF
```

See Also

PHD .NET Wrapper	

PHDHistorian.EngUnitConversion Method (PHDServer, String, String, Double, Double, Double, Int32)

An overloaded EngUnitConversion method

```
[Visual Basic]
Overloads Public Sub EngUnitConversion( _
   ByVal Server As PHDServer, _
   ByVal UnitIn As String, _
   ByVal UnitOut As String, _
   ByRef PreAdd As Double, _
   ByRef Mult As Double, _
   ByRef PostAdd As Double, _
   ByRef pCode As Integer _
)
[C#]
public void EngUnitConversion(
   PHDServer Server,
   string UnitIn,
   string UnitOut,
   ref double PreAdd,
   ref double Mult,
   ref double PostAdd,
   ref int pCode
);
```

Parameters

Server

The PHDServer object to be used for the conversion

UnitIn

The Engineering Unit that is to be converted

UnitOut

The new Engineering Unit

PreAdd

The value that is added to the UnitIn

Mult

The multiplation factor

PostAdd

The value that is added after the multiplication

pCode

See Also

PHD .NET Wrapper	

PHDHistorian.FetchAllReductionData Method

Return Reduction Data for a Tagname

Overload List

Return Reduction Data for a Tagname

public DataSet FetchAllReductionData(string,bool);

Return Reduction Data for a Tag or Tags

public DataSet FetchAllReductionData(Tags,bool);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.FetchAllReductionData Method (String, Boolean)

Return Reduction Data for a Tagname

```
[Visual Basic]
Overloads Public Function FetchAllReductionData( _
   ByVal Tagname As String, _
   ByVal bOverallReductionOnly As Boolean _
) As DataSet
[C#]
public DataSet FetchAllReductionData(
   string Tagname,
   bool bOverallReductionOnly
);
```

Parameters

Tagname

Valid PHD Tag Name

bOverallReductionOnly

Return all reduction data for the collection of Tags

Return Value

Datset containing the Reduction Data

Example

Example 1: Return all Reduction data for a collection of Tags Example 2: Return as specified reduction type for a collection of Tags

```
Example 1:
```

```
DataSet ds = new DataSet();

try

{

        ds=this.PHD.FetchAllReductionData("kst20000", true);

}

catch (PHDErrorException ex)

{

        MessageBox.Show(ex.Message);

        return;

}


Example 2:
```

See Also

[PHDHistorian Class](#) | [Uniformance.PHD Namespace](#) | [PHDHistorian.FetchAllReductionData Overload List](#)

PHD .NET Wrapper	

PHDHistorian.FetchAllReductionData Method (Tags, Boolean)

Return Reduction Data for a Tag or Tags

```
[Visual Basic]
Overloads Public Function FetchAllReductionData( _
   ByVal oTags As Tags, _
   ByVal bOverallReductionOnly As Boolean _
) As DataSet

[C#]
public DataSet FetchAllReductionData(
   Tags oTags,
   bool bOverallReductionOnly
);
```

Parameters

oTags

Collection of Tag items

bOverallReductionOnly

Return all reduction data for the collection of Tags

Return Value

Datset containing the Reduction Data

Example

Example 1: Return all Reduction data for a collection of Tags Example 2: Return as specified reduction type for a collection of Tags

```
PHDHistorian PHD = new PHDHistorian();




Tags oTags = new Tags();

oTags.Add(new Tag("kst20000"));

oTags.Add(new Tag("kst20001"));



DataSet ds = new DataSet();




Example 1:



try

{

        ds=this.PHD.FetchAllReductionData(oTags, true);

}

catch (PHDErrorException ex)

{

        MessageBox.Show(ex.Message);

        return;

}



Tagname    HostName        AvgTimeStamp       AvgValue   AvgConf AvgTolerance
DeltaTimeStamp       DeltaValue

kst20001 wi-ab07-xx00000    12/15/2003 0:29:54  121.2412    11        0          12/15/2003
12:29:54    115.6561355

kst20000 wi-ab07-xx00000    12/15/2003 0:29:54  131.1062    11        0          12/15/2003
12:29:54    187.2834625


```

```
Example 2:



PHD.ReductionFrequency=System.Convert.ToUInt32("60");

PHD.ReductionOffset=(REDUCTIONOFFSET)REDUCTIONOFFSET.Parse(typeof(REDUCTIONOFFSET),"After");



try

{

        ds=this.PHD.FetchAllReductionData(oTags, false);

}

catch (PHDErrorException ex)

{

        MessageBox.Show(ex.Message);

        return;

}




Results:

Tagname    HostName       AvgTimeStamp       AvgValue   AvgConf AvgTolerance
DeltaTimeStamp      DeltaValue     ...

kst20000 wi-ab07-xx00000 12/14/2003 12:36:24 137.8850    0         0           12/14/2003
12:36:54      -0.014007568  ...

kst20000   wi-ab07-xx00000    12/14/2003 12:37:24 137.8710    0        0
12/14/2003 12:37:54    -0.014022827  ...

...

kst20001   wi-ab07-xx00000    12/14/2003 12:36:24 117.8424    0        0
12/14/2003 12:36:54    -8.39233E-05  ...

kst20001   wi-ab07-xx00000    12/14/2003 12:37:24 117.8423    0        0
12/14/2003 12:37:54    -9.15527E-05  ...

...
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.FetchAllReductionData Overload List

PHD .NET Wrapper	

PHDHistorian.FetchColumnData Method

Fetch a single tag of Data and Display in a Column Format

Overload List

Fetch a single tag of Data and Display in a Column Format

public DataSet FetchColumnData(string);

Return all tags from the Tags Collection in a Column Format. The Column data format will not display Raw Sample Types.

public DataSet FetchColumnData(Tags);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.FetchColumnData Method (String)

Fetch a single tag of Data and Display in a Column Format

```
[Visual Basic]
Overloads Public Function FetchColumnData( _
   ByVal Tagname As String _
) As DataSet
[C#]
public DataSet FetchColumnData(
   string Tagname
);
```

Parameters

Tagname
Used to supply the Tag that is to be returned

Example

Retrieve and display in a column format a specified tag

```
PHDHistorian oPhd = new PHDHistorian(false);



oPhd.StartTime="now-1h;

oPhd.EndTime="now";



try

{
```

```
      dsGrid=oPhd.FetchColumnData("kst20000");

}

catch (PHDErrorException ex)

{

    MessageBox.Show(ex.Message);

    return;

}




Grid Display Results:



    timestamp          kst20000

12/14/2003 7:14:57    107.3801

12/14/2003 7:19:57    178.6686

12/14/2003 7:24:57    139.201

12/14/2003 7:29:57    142.3064

12/14/2003 7:34:57    182.0065

12/14/2003 7:39:57    107.7729
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.FetchColumnData Overload List

PHD .NET Wrapper	

PHDHistorian.FetchColumnData Method (Tags)

Return all tags from the Tags Collection in a Column Format. The Column data format will not display Raw Sample Types.

```
[Visual Basic]
Overloads Public Function FetchColumnData( _
   ByVal oTags As Tags _
) As DataSet
[C#]
public DataSet FetchColumnData(
    Tags oTags
);
```

Parameters

oTags

Used to supply the collection of Tags

Example

Retrieve and display data in a column format for a collection of tags

```
PHDHistorian oPhd = new PHDHistorian(false);


oPhd.StartTime="now-1h;

oPhd.EndTime="now";


Tags _tags = new Tags();

_tags.Add(new Tag("kst20000");

_tags.Add(new Tag("kst20001");


try

{

   dsGrid=oPhd.FetchColumnData(_tags);

}

catch (PHDErrorException ex)

{

   MessageBox.Show(ex.Message);

   return;

}




Grid Display Results:


   timestamp          kst20000    kst20001

12/14/2003 7:42:23   123.3794    131.9819

12/14/2003 7:47:23   188.7889    147.0151

12/14/2003 7:52:23   105.5942    174.155

12/14/2003 7:57:23   197.3974    114.2238
```

```
12/14/2003 8:02:23    107.3537    195.8699
```

See Also

[PHDHistorian Class](#) | [Uniformance.PHD Namespace](#) | [PHDHistorian.FetchColumnData Overload List](#)

PHD .NET Wrapper	

PHDHistorian.FetchData Method

Fetch a single tag of Data and return results in an Array

```
[Visual Basic]
Public Sub FetchData( _
    ByVal tag As Tag, _
    ByRef datetime As Double(), _
    ByRef Value As Double(), _
    ByRef confidence As Short() _
)
[C#]
public void FetchData(
    Tag tag,
    ref double[] datetime,
    ref double[] Value,
    ref short[] confidence
);
```

Parameters

tag

Used to supply the Tag that is to be returned

datetime

return array of datetime values

Value

return array of values

confidence

return array of confidence values

Example

```
PHDHistorian oPhd = new PHDHistorian();



PHDServer defaultServer = new PHDServer("HostName");

defaultServer.APIVersion = SERVERVERSION.API200;

oPhd.DefaultServer = defaultServer;

oPhd.StartTime="now-1h;

oPhd.EndTime="now";

oPhd.Sampletype = SAMPLETYPE.Raw;
```

```
double[] Value = null;

double[] timeStamp = null;

short[] conf = null;


try

{

        oPhd.FetchData(new Tag("kst20000"), ref timeStamp, ref Value, ref conf);

}

catch

{

        MessageBox.Show(ex.Message);

        return;

}


..


oPhd.Dispose();


Results:


timeStamp[0]=37969.338021446762    Value[0]=101.846    conf[0]=100

timeStamp[1]=37969.341493668981    Value[1]=198.474    conf[1]=100

timeStamp[2]=37969.344965891207    Value[2]=109.622    conf[2]=100
```

See Also

[PHDHistorian Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper	

PHDHistorian.FetchRowData Method

Fetch a single tag of Data and return the data in a DataSet

Overload List

Fetch a single tag of Data and return the data in a DataSet

public DataSet FetchRowData(string);

Return all tags from the Tags Collection

public DataSet FetchRowData(Tags);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.FetchRowData Method (String)

Fetch a single tag of Data and return the data in a DataSet

```
[Visual Basic]
Overloads Public Function FetchRowData( _
   ByVal Tagname As String _
) As DataSet
[C#]
public DataSet FetchRowData(
   string Tagname
);
```

Parameters

Tagname
Tag that is to be returned

Example

Retrieve and display in a row format time series data for tag kst20000

```
PHDHistorian oPhd = new PHDHistorian();


PHDServer defaultServer = new PHDServer("wi-ab07-xx00000");

defaultServer.APIVersion = SERVERVERSION.API200;

oPhd.DefaultServer = defaultServer;

oPhd.StartTime="now-15m;

oPhd.EndTime="now";


oPhd.Sampletype = SAMPLETYPE.Snapshot;

oPhd.SampleFrequency = 300;
```

```
try

{

        dsGrid = oPhd.FetchRowData("kst20000");

}

catch (PHDErrorException ex)

{

        MessageBox.Show(ex.Message);

        return;

}




Grid Display Results:



    Tag      timestamp          Value    Conf    HostName

kst20000 12/15/2003 7:43:41  121.8072   100     wi-ab07-xx00000

kst20000 12/15/2003 7:48:41  121.7371   100     wi-ab07-xx00000

kst20000 12/15/2003 7:53:41  121.667    100     wi-ab07-xx00000

kst20000 12/15/2003 7:58:41  121.597    100     wi-ab07-xx00000
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.FetchRowData Overload List

PHD .NET Wrapper	

PHDHistorian.FetchRowData Method (Tags)

Return all tags from the Tags Collection

```
[Visual Basic]
Overloads Public Function FetchRowData( _
    ByVal oTags As Tags _
) As DataSet
[C#]
public DataSet FetchRowData(
    Tags oTags
);
```

Parameters

oTags

Used to supply the collection of Tags

Example

Retrieve and display in a row format time series data for collection of Tags

```
PHDHistorian oPhd = new PHDHistorian();


PHDServer defaultServer = new PHDServer("wi-ab07-xx00000");

defaultServer.APIVersion = SERVERVERSION.API200;

oPhd.DefaultServer = defaultServer;

oPhd.StartTime="now-15m;

oPhd.EndTime="now";


oPhd.Sampletype = SAMPLETYPE.Snapshot;

oPhd.SampleFrequency = 300;


oPhd.StartTime="now-1h;

oPhd.EndTime="now";


Tags _tags = new Tags();

_tags.Add(new Tag("kst20000"));

_tags.Add(new Tag("kst20001"));


try

{

      dsGrid=oPhd.FetchRowData(_tags);

}

catch (PHDErrorException ex)

{

      MessageBox.Show(ex.Message);

      return;

}
```

```
   ...



Grid Display Results:



   Tag      timestamp          Value    Conf   HostName

kst20000 12/15/2003 7:43:41  121.8072  100   wi-ab07-xx00000

kst20000 12/15/2003 7:48:41  121.7371  100   wi-ab07-xx00000

kst20000 12/15/2003 7:53:41  121.667   100   wi-ab07-xx00000

kst20000 12/15/2003 7:58:41  121.597   100   wi-ab07-xx00000

kst20001 12/15/2003 7:43:41  111.8272  100   wi-ab07-xx00000

kst20001 12/15/2003 7:48:41  110.7371  100   wi-ab07-xx00000

kst20001 12/15/2003 7:53:41  104.667   100   wi-ab07-xx00000

kst20001 12/15/2003 7:58:41  100.597   100   wi-ab07-xx00000
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.FetchRowData Overload List

PHD .NET Wrapper	

PHDHistorian.FetchStructData Method

Return all tags from the Tags Collection

```
[Visual Basic]
Public Function FetchStructData( _
   ByVal oTags As Tags _
) As ArrayList
[C#]
public ArrayList FetchStructData(
   Tags oTags
);
```

Parameters

oTags

Used to supply the collection of Tags

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.Finalize Method

destructor to clean up all managed and unmanaged code

```
[Visual Basic]
Overrides Protected Sub Finalize()
[C#]
protected override void Finalize();
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.GetDBInfo Method

Get Database Information from default PHD server

Overload List

Get Database Information from default PHD server

public void GetDBInfo(ref string,ref string,ref int);

Get Database Information from the PHD server specified

public void GetDBInfo(PHDServer,ref string,ref string,ref int);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.GetDBInfo Method (String, String, Int32)

Get Database Information from default PHD server

```
[Visual Basic]
Overloads Public Sub GetDBInfo( _
   ByRef DBMProvider As String, _
   ByRef DBMDataSource As String, _
   ByRef UsingINTS As Integer _
)
[C#]
public void GetDBInfo(
   ref string DBMProvider,
   ref string DBMDataSource,
   ref int UsingINTS
);
```

Parameters

> *DBMProvider*
>> DBMProvider

> *DBMDataSource*
>> DBMDataSource

> *UsingINTS*
>> UsingINTS

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.GetDBInfo Overload List

---

PHD .NET Wrapper

PHDHistorian.GetDBInfo Method (PHDServer, String, String, Int32)

Get Database Information from the PHD server specified

```
[Visual Basic]
Overloads Public Sub GetDBInfo( _
   ByVal Server As PHDServer, _
   ByRef DBMProvider As String, _
   ByRef DBMDataSource As String, _
   ByRef UsingINTS As Integer _
)
[C#]
public void GetDBInfo(
   PHDServer Server,
   ref string DBMProvider,
   ref string DBMDataSource,
   ref int UsingINTS
);
```

Parameters

> *Server*
>> PHDServer object

> *DBMProvider*
>> DBMProvider

> *DBMDataSource*
>> DBMDataSource

> *UsingINTS*
>> UsingINTS

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.GetDBInfo Overload List

---

PHD .NET Wrapper

2. Usage

2. Usage
2.2. The .NET Wrapper Interface – Methods and Properties

PHDHistorian.GetLinkList Method

Returns link list from the default server

Overload List

Returns link list from the default server

public ArrayList GetLinkList();

Returns link list from the server specified

public ArrayList GetLinkList(PHDServer);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper

PHDHistorian.GetLinkList Method ()

Returns link list from the default server

```
[Visual Basic]
Overloads Public Function GetLinkList() As ArrayList
[C#]
public ArrayList GetLinkList();
```

Return Value

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.GetLinkList Overload List

---

PHD .NET Wrapper

PHDHistorian.GetLinkList Method (PHDServer)

Returns link list from the server specified

```
[Visual Basic]
Overloads Public Function GetLinkList( _
   ByVal Server As PHDServer _
) As ArrayList
[C#]
public ArrayList GetLinkList(
   PHDServer Server
);
```

Parameters

Server

Return Value

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.GetLinkList Overload List

---

PHD .NET Wrapper

PHDHistorian.GetLinks Method

Returns link list from the default server

Overload List

Returns link list from the default server

public DataSet GetLinks();

Returns link list from the server specified

public DataSet GetLinks(PHDServer);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper

PHDHistorian.GetLinks Method ()

Returns link list from the default server

```
[Visual Basic]
Overloads Public Function GetLinks() As DataSet
[C#]
public DataSet GetLinks();
```

Return Value

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.GetLinks Overload List

---

PHD .NET Wrapper

PHDHistorian.GetLinks Method (PHDServer)

Returns link list from the server specified

```
[Visual Basic]
```

```
Overloads Public Function GetLinks( _
   ByVal Server As PHDServer _
) As DataSet
[C#]
public DataSet GetLinks(
   PHDServer Server
);
```

Parameters

Server

Return Value

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.GetLinks Overload List

---

PHD .NET Wrapper

PHDHistorian.GetParentTagList Method

Returns Parent Tag list from the default server

Overload List

Returns Parent Tag list from the default server

public ArrayList GetParentTagList();

Returns Parent Tag list from the server specified

public ArrayList GetParentTagList(PHDServer);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper

PHDHistorian.GetParentTagList Method ()

Returns Parent Tag list from the default server

```
[Visual Basic]
Overloads Public Function GetParentTagList() As ArrayList
[C#]
public ArrayList GetParentTagList();
```

Return Value

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.GetParentTagList Overload List

---

PHDHistorian.GetParentTagList Method (PHDServer)

Returns Parent Tag list from the server specified

```
[Visual Basic]
Overloads Public Function GetParentTagList( _
   ByVal Server As PHDServer _
) As ArrayList
[C#]
public ArrayList GetParentTagList(
   PHDServer Server
);
```

Parameters

Server

Return Value

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.GetParentTagList Overload List

PHDHistorian.GetParentTags Method

Returns Parent Tag list from the default server

Overload List

Returns Parent Tag list from the default server

public DataSet GetParentTags();

Returns Parent Tag list from the server specified

public DataSet GetParentTags(PHDServer);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHDHistorian.GetParentTags Method ()

Returns Parent Tag list from the default server

```
[Visual Basic]
Overloads Public Function GetParentTags() As DataSet
[C#]
public DataSet GetParentTags();
```

Return Value

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.GetParentTags Overload List

PHD .NET Wrapper	

PHDHistorian.GetParentTags Method (PHDServer)

Returns Parent Tag list from the server specified

```
[Visual Basic]
Overloads Public Function GetParentTags( _
   ByVal Server As PHDServer _
) As DataSet
[C#]
public DataSet GetParentTags(
   PHDServer Server
);
```

Parameters

Server

Return Value

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.GetParentTags Overload List

PHD .NET Wrapper	

PHDHistorian.GetRDIInfo Method

A 200 API function that will return the RDI info

Overload List

A 200 API function that will return the RDI info

public DataSet GetRDIInfo(string);

A overloaded GetRDIInfo methods

public DataSet GetRDIInfo(string,PHDServer);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.GetRDIInfo Method (String)

A 200 API function that will return the RDI info

```
[Visual Basic]
Overloads Public Function GetRDIInfo( _
   ByVal RDIName As String _
) As DataSet
[C#]
public DataSet GetRDIInfo(
   string RDIName
);
```

Parameters

RDIName
The RDI Name to return Information for

Return Value

DataSet with the RDI Info

Exceptions

Exception Type	Condition
Exception	Invalid RDI was requested

Example

Return a dataset of RDI information for the specified RDI. This example assumes that there is a datagrid on the form named dgData

```
DataSet ds = new DataSet();


try

{

       ds=this.PHD.GetRDIInfo(txtRDIName.Text);

       dgData.SetDataBinding(ds,"TEST");

}

catch (PHDErrorException ex)

       MessageBox.Show(ex.Message);

```

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.GetRDIInfo Overload List

PHD .NET Wrapper	

PHDHistorian.GetRDIInfo Method (String, PHDServer)

A overloaded GetRDIInfo methods

```
[Visual Basic]
Overloads Public Function GetRDIInfo( _
    ByVal RDIName As String, _
    ByVal Server As PHDServer _
) As DataSet
[C#]
public DataSet GetRDIInfo(
    string RDIName,
    PHDServer Server
);
```

Parameters

RDIName
RDI Name

Server
The PHDServer object

Return Value

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.GetRDIInfo Overload List

PHD .NET Wrapper	

PHDHistorian.GetRDIList Method

Returns RDI list from the default server

Overload List

Returns RDI list from the default server

public ArrayList GetRDIList();

Returns RDI list from the server specified

public ArrayList GetRDIList(PHDServer);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.GetRDIList Method ()

Returns RDI list from the default server

```
[Visual Basic]
Overloads Public Function GetRDIList() As ArrayList
[C#]
public ArrayList GetRDIList();
```

Return Value

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.GetRDIList Overload List

PHD .NET Wrapper	

PHDHistorian.GetRDIList Method (PHDServer)

Returns RDI list from the server specified

```
[Visual Basic]
Overloads Public Function GetRDIList( _
   ByVal Server As PHDServer _
) As ArrayList
[C#]
public ArrayList GetRDIList(
   PHDServer Server
);
```

Parameters

Server
PHDServer object

Return Value

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.GetRDIList Overload List

PHD .NET Wrapper	

PHDHistorian.GetRDIs Method

Returns RDI list from the default server

Overload List

Returns RDI list from the default server

public DataSet GetRDIs();

Returns RDI list from the server specified

public DataSet GetRDIs(PHDServer);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.GetRDIs Method ()

Returns RDI list from the default server

```
[Visual Basic]
Overloads Public Function GetRDIs() As DataSet
[C#]
public DataSet GetRDIs();
```

Return Value

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.GetRDIs Overload List

PHD .NET Wrapper	

PHDHistorian.GetRDIs Method (PHDServer)

Returns RDI list from the server specified

```
[Visual Basic]
Overloads Public Function GetRDIs( _
    ByVal Server As PHDServer _
) As DataSet
[C#]
public DataSet GetRDIs(
    PHDServer Server
);
```

Parameters

Server

Return Value

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.GetRDIs Overload List

PHD .NET Wrapper	

PHDHistorian.ModifyTag Method

Modify Tag Data

Overload List

Modify Tag Data

public void ModifyTag(Tag,object);

Modify Tag Data

public void ModifyTag(Tag,object,DateTime,sbyte,string);

Modify Tag Data

public void ModifyTag(Tag,object,string);

Modify Tag Data

public void ModifyTag(Tag,object,string,sbyte);

Modify Tag Data

public void ModifyTag(Tag,object,string,sbyte,string);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.ModifyTag Method (Tag, Object)

Modify Tag Data

```
[Visual Basic]
Overloads Public Sub ModifyTag( _
   ByVal tag As Tag, _
   ByVal Value As Object _
)
[C#]
public void ModifyTag(
   Tag tag,
   object Value
);
```

Parameters

tag
The Tag object

Value
The value to be inserted

Example

```
Tag oTag = new Tag();

oTag.TagName="KST20000";

float _value = 123.35;



int istat=this.PHD.ModifyTag(oTag, _value);



if(istat!=0)

{

    error.ErrorNo=istat;

    MessageBox.Show(ex.Message);

}
```

See Also

[PHDHistorian Class](#) | [Uniformance.PHD Namespace](#) | [PHDHistorian.ModifyTag Overload List](#)

PHD .NET Wrapper

PHDHistorian.ModifyTag Method (Tag, Object, DateTime, SByte, String)

Modify Tag Data

```
[Visual Basic]
Overloads Public Sub ModifyTag( _
   ByVal tag As Tag, _
   ByVal Value As Object, _
   ByVal TimeStamp As Date, _
   ByVal Confidence As SByte, _
   ByVal Units As String _
)
[C#]
public void ModifyTag(
   Tag tag,
   object Value,
   DateTime TimeStamp,
   sbyte Confidence,
   string Units
);
```

Parameters

tag
The Tag object

Value

The Value to be inserted

TimeStamp

The UTC timestamp

Confidence

The Confidence

Units

The Engineering Units

See Also

[PHDHistorian Class](#) | [Uniformance.PHD Namespace](#) | [PHDHistorian.ModifyTag Overload List](#)

| *PHD .NET Wrapper* | |

PHDHistorian.ModifyTag Method (Tag, Object, String)

Modify Tag Data

```
[Visual Basic]
Overloads Public Sub ModifyTag( _
   ByVal tag As Tag, _
   ByVal Value As Object, _
   ByVal TimeStamp As String _
)
[C#]
public void ModifyTag(
   Tag tag,
   object Value,
   string TimeStamp
);
```

Parameters

tag

The Tag object

Value

The Value of type float, to be inserted

TimeStamp

The timestamp

See Also

[PHDHistorian Class](#) | [Uniformance.PHD Namespace](#) | [PHDHistorian.ModifyTag Overload List](#)

| *PHD .NET Wrapper* | |

PHDHistorian.ModifyTag Method (Tag, Object, String, SByte)

Modify Tag Data

```
[Visual Basic]
```

```
Overloads Public Sub ModifyTag( _
   ByVal tag As Tag, _
   ByVal Value As Object, _
   ByVal TimeStamp As String, _
   ByVal Confidence As SByte _
)
[C#]
public void ModifyTag(
   Tag tag,
   object Value,
   string TimeStamp,
   sbyte Confidence
);
```

Parameters

tag

The Tag object

Value

The value to be inserted

TimeStamp

The timestamp

Confidence

The Confidence

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.ModifyTag Overload List

PHD .NET Wrapper	

PHDHistorian.ModifyTag Method (Tag, Object, String, SByte, String)

Modify Tag Data

```
[Visual Basic]
Overloads Public Sub ModifyTag( _
   ByVal tag As Tag, _
   ByVal Value As Object, _
   ByVal TimeStamp As String, _
   ByVal Confidence As SByte, _
   ByVal Units As String _
)
[C#]
public void ModifyTag(
   Tag tag,
   object Value,
   string TimeStamp,
   sbyte Confidence,
   string Units
);
```

Parameters

tag

The Tag object

Value

The value to be inserted

TimeStamp

The timestamp

Confidence

The Confidence

Units

The Engineering Units

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.ModifyTag Overload List

---

PHD .NET Wrapper	

PHDHistorian.PutData Method

Insert Tag Data into a DCS for a timestamp of Now

Overload List

Insert Tag Data into a DCS for a timestamp of Now

public void PutData(Tag,object);

Insert Tag Data into a DCS for a specified timestamp and Confidence

public void PutData(Tag,object,DateTime,sbyte,string);

Insert Tag Data into a DCS for a specified timestamp

public void PutData(Tag,object,string);

Insert Tag Data into a DCS for a specified timestamp

public void PutData(Tag,object,string,sbyte);

Insert Tag Data into a DCS for a specified timestamp and Confidence

public void PutData(Tag,object,string,sbyte,string);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.PutData Method (Tag, Object)

Insert Tag Data into a DCS for a timestamp of Now

```
[Visual Basic]
Overloads Public Sub PutData( _
   ByVal tag As Tag, _
   ByVal Value As Object _
```

```
)
[C#]
public void PutData(
    Tag tag,
    object Value
);
```

Parameters

tag

Tag object

Value

The value to be inserted

Example

Insert a value into a DCS for tag KST20000. The record will default to a timestamp of Now and Confidence of 100.

```
Tag oTag = new Tag();

oTag.TagName="KST20000";

float floatValue = 123.45


try

{

    PHD.PutData(oTag, floatValue);

}

catch(PHDErrorException ex)

{

    MessageBox.Show(ex.Message);

    return;

}
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.PutData Overload List

PHD .NET Wrapper

PHDHistorian.PutData Method (Tag, Object, DateTime, SByte, String)

Insert Tag Data into a DCS for a specified timestamp and Confidence

```
[Visual Basic]
Overloads Public Sub PutData( _
   ByVal tag As Tag, _
   ByVal Value As Object, _
   ByVal TimeStamp As Date, _
   ByVal Confidence As SByte, _
   ByVal Units As String _
)
[C#]
public void PutData(
   Tag tag,
   object Value,
   DateTime TimeStamp,
   sbyte Confidence,
   string Units
);
```

Parameters

tag

Tag object

Value

The Value to be inserted

TimeStamp

The UTC timestamp

Confidence

The Confidence

Units

The Engineering Units

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.PutData Overload List

PHD .NET Wrapper	

PHDHistorian.PutData Method (Tag, Object, String)

Insert Tag Data into a DCS for a specified timestamp

```
[Visual Basic]
Overloads Public Sub PutData( _
   ByVal tag As Tag, _
   ByVal Value As Object, _
   ByVal TimeStamp As String _
)
[C#]
public void PutData(
   Tag tag,
   object Value,
   string TimeStamp
);
```

Parameters

tag

Tag object

Value

The value to be inserted

TimeStamp

The timestamp

Example

Insert a value into a DCS for tag KST20000. The record will have a timestamp of Now and Confidence of 100.

```
Tag oTag = new Tag();

oTag.TagName="KST20000";

float _value = 123.456;



try

{

    PHD.PutData(oTag, _value, "Now"));

}

catch(PHDErrorException ex)

{

    MessageBox.Show(ex.Message);

    return;

}
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.PutData Overload List

PHD .NET Wrapper	

PHDHistorian.PutData Method (Tag, Object, String, SByte)

Insert Tag Data into a DCS for a specified timestamp

```
[Visual Basic]
Overloads Public Sub PutData( _
   ByVal tag As Tag, _
   ByVal Value As Object, _
   ByVal TimeStamp As String, _
   ByVal Confidence As SByte _
)
[C#]
```

```
public void PutData(
    Tag tag,
    object Value,
    string TimeStamp,
    sbyte Confidence
);
```

Parameters

tag

Tag object

Value

The value to be inserted

TimeStamp

The timestamp

Confidence

The Confidence

Example

Insert a value into PHD for tag KST20000.

```
Tag oTag = new Tag();

oTag.TagName="KST20000";

float _value = 123.23;

sbyte _conf = 100;



try

{

    PHD.PutData(oTag, _value, "Now",_conf));

}

catch(PHDErrorException ex)

{

    MessageBox.Show(ex.Message);

    return;

}
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.PutData Overload List

PHD .NET Wrapper	

PHDHistorian.PutData Method (Tag, Object, String, SByte, String)

Insert Tag Data into a DCS for a specified timestamp and Confidence

```
[Visual Basic]
Overloads Public Sub PutData( _
    ByVal tag As Tag, _
    ByVal Value As Object, _
    ByVal TimeStamp As String, _
    ByVal Confidence As SByte, _
    ByVal Units As String _
)
[C#]
public void PutData(
    Tag tag,
    object Value,
    string TimeStamp,
    sbyte Confidence,
    string Units
);
```

Parameters

tag

Tag object

Value

The Value to be inserted

TimeStamp

The string timestamp

Confidence

The Confidence

Units

The Engineering Units

Example

Insert a value into PHD for tag KST20000.

```
Tag oTag = new Tag();

oTag.TagName="KST20000";

float _value = 123.35;

byte _conf = 100;

string _engUnit = "DEGC";



try

{
```

```
        PHD.PutData(oTag, _value, "Now",_conf,_engUnit));

}

catch(PHDErrorException ex)

{

    MessageBox.Show(ex.Message);

    return;

}
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.PutData Overload List

---

PHD .NET Wrapper

PHDHistorian.RelativeToAbsolute Method

Convert a PHD relative time to the absolute UTC date and time on the client. An exception will be thrown for a incorrect relative input.

Overload List

Convert a PHD relative time to the absolute UTC date and time on the client. An exception will be thrown for a incorrect relative input.

public void RelativeToAbsolute(string,ref DateTime);

Convert a PHD relative time to the absolute UTC date and time. An exception will be thrown for a incorrect relative input.

public void RelativeToAbsolute(string,ref DateTime,bool);

Convert a PHD relative time to the absolute UTC date and time on the PHD server specified. An exception will be thrown for a incorrect relative input.

public void RelativeToAbsolute(string,ref DateTime,PHDServer);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper

PHDHistorian.RelativeToAbsolute Method (String, DateTime)

Convert a PHD relative time to the absolute UTC date and time on the client. An exception will be thrown for a incorrect relative input.

```
[Visual Basic]
Overloads Public Sub RelativeToAbsolute( _
```

```
    ByVal RelativeDate As String, _
    ByRef AbsoluteDateTimeUTC As Date _
)
[C#]
public void RelativeToAbsolute(
    string RelativeDate,
    ref DateTime AbsoluteDateTimeUTC
);
```

Parameters

RelativeDate

The relative value (eq Now, Now-1D etc)

AbsoluteDateTimeUTC

The Absolute Value Returned

Example

Convert Relative dates to Absolute values on the client

```
PHDHistorian oPhd = new PHDHistorian(false);

...

DateTime dtAbsolute = new DateTime();

oPHD.RelativeToAbsolute(txtRelDate.Text,ref dtAbsolute)

...
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.RelativeToAbsolute Overload List

PHD .NET Wrapper	

PHDHistorian.RelativeToAbsolute Method (String, DateTime, Boolean)

Convert a PHD relative time to the absolute UTC date and time. An exception will be thrown for a incorrect relative input.

```
[Visual Basic]
Overloads Public Sub RelativeToAbsolute( _
    ByVal RelativeDate As String, _
    ByRef AbsoluteDateTimeUTC As Date, _
    ByVal UsingServerTime As Boolean _
)
[C#]
public void RelativeToAbsolute(
    string RelativeDate,
    ref DateTime AbsoluteDateTimeUTC,
    bool UsingServerTime
);
```

Parameters

RelativeDate

The relative value (eq Now, Now-1D etc)

AbsoluteDateTimeUTC

The Absolute Value Returned

UsingServerTime

A flag indicating whether or not the absolute time should be gotten from server or client.

Example

Convert Relative dates to Absolute values on the default PHD server.

```
PHDHistorian oPhd = new PHDHistorian(false);

......

DateTime dtAbsolute = new DateTime();

oPHD.RelativeToAbsolute(txtRelDate.Text, ref dtAbsolute, true);

...
```

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.RelativeToAbsolute Overload List

PHD .NET Wrapper	

PHDHistorian.RelativeToAbsolute Method (String, DateTime, PHDServer)

Convert a PHD relative time to the absolute UTC date and time on the PHD server specified. An exception will be thrown for a incorrect relative input.

```
[Visual Basic]
Overloads Public Sub RelativeToAbsolute( _
   ByVal RelativeDate As String, _
   ByRef AbsoluteDateTimeUTC As Date, _
   ByVal Server As PHDServer _
)
[C#]
public void RelativeToAbsolute(
   string RelativeDate,
   ref DateTime AbsoluteDateTimeUTC,
   PHDServer Server
);
```

Parameters

RelativeDate

The relative value (eq Now, Now-1D etc)

AbsoluteDateTimeUTC

The Absolute Value Returned

Server

The PHDServer object that specifies the PHD Server

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.RelativeToAbsolute Overload List

PHD .NET Wrapper	

PHDHistorian.StringToUniversalTime Method

Convert a PHD string/relative time to universal DateTime. The conversion will be performed on the default PHD server if the string starts with "NOW"; otherwise the conversion will be performed on the local machine.

Overload List

Convert a PHD string/relative time to universal DateTime. The conversion will be performed on the default PHD server if the string starts with "NOW"; otherwise the conversion will be performed on the local machine.

public DateTime StringToUniversalTime(string);

Convert a PHD string/relative time to universal DateTime. The conversion will be performed on the PHD server if the string starts with "NOW"; otherwise the conversion will be performed on the local machine.

public DateTime StringToUniversalTime(string,PHDServer);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDHistorian.StringToUniversalTime Method (String)

Convert a PHD string/relative time to universal DateTime. The conversion will be performed on the default PHD server if the string starts with "NOW"; otherwise the conversion will be performed on the local machine.

```
[Visual Basic]
Overloads Public Function StringToUniversalTime( _
   ByVal StringTime As String _
) As Date
[C#]
public DateTime StringToUniversalTime(
   string StringTime
);
```

Parameters

StringTime

Return Value

A exception will be thrown if the conversion fails

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.StringToUniversalTime Overload List

---

PHD .NET Wrapper

PHDHistorian.StringToUniversalTime Method (String, PHDServer)

Convert a PHD string/relative time to universal DateTime. The conversion will be performed on the PHD server if the string starts with "NOW"; otherwise the conversion will be performed on the local machine.

```
[Visual Basic]
Overloads Public Function StringToUniversalTime( _
   ByVal StringTime As String, _
   ByVal Server As PHDServer _
) As Date
[C#]
public DateTime StringToUniversalTime(
   string StringTime,
   PHDServer Server
);
```

Parameters

StringTime
PHD string/relative time

Server
PHDServer object

Return Value

A exception will be thrown if the conversion fails

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.StringToUniversalTime Overload List

PHD .NET Wrapper

PHDHistorian.TagDfn Method

Fetch a single tag of Data

Overload List

Fetch a single tag of Data

public DataSet TagDfn(string);

Return all tags from the Tags Collection

public DataSet TagDfn(Tags);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper

PHDHistorian.TagDfn Method (String)

Fetch a single tag of Data

```
[Visual Basic]
Overloads Public Function TagDfn( _
   ByVal Tagname As String _
) As DataSet
[C#]
public DataSet TagDfn(
   string Tagname
);
```

Parameters

Tagname
Used to supply the Tag that is to be returned

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.TagDfn Overload List

PHD .NET Wrapper

PHDHistorian.TagDfn Method (Tags)

Return all tags from the Tags Collection

```
[Visual Basic]
Overloads Public Function TagDfn( _
   ByVal oTags As Tags _
) As DataSet
[C#]
public DataSet TagDfn(
   Tags oTags
);
```

Parameters

oTags
Used to supply the collection of Tags

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.TagDfn Overload List

PHD .NET Wrapper

PHDHistorian.TagUpdateHandler Method

Specify call back function when PHD Tags have been updated

Overload List

Specify call back function when PHD Tags have been updated

public void TagUpdateHandler(PHDServer,TagUpdateNotify);

Specify call back function when PHD Tags have been updated

public void TagUpdateHandler(TagUpdateNotify);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper

PHDHistorian.TagUpdateHandler Method (PHDServer, TagUpdateNotify)

Specify call back function when PHD Tags have been updated

```
[Visual Basic]
Overloads Public Sub TagUpdateHandler( _
   ByVal Server As PHDServer, _
   ByVal TagUpdateCallbackFunc As TagUpdateNotify _
)
[C#]
public void TagUpdateHandler(
   PHDServer Server,
   TagUpdateNotify TagUpdateCallbackFunc
);
```

Parameters

Server
PHDServer object

TagUpdateCallbackFunc
Tag update callback function

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.TagUpdateHandler Overload List

---

PHD .NET Wrapper

PHDHistorian.TagUpdateHandler Method (TagUpdateNotify)

Specify call back function when PHD Tags have been updated

```
[Visual Basic]
Overloads Public Sub TagUpdateHandler( _
   ByVal TagUpdateCallbackFunc As TagUpdateNotify _
)
```

```
[C#]
public void TagUpdateHandler(
    TagUpdateNotify TagUpdateCallbackFunc
);
```

Parameters

TagUpdateCallbackFunc

Tag update callback function

See Also

PHDHistorian Class | Uniformance.PHD Namespace | PHDHistorian.TagUpdateHandler Overload List

---

PHD .NET Wrapper

PHDHistorian.ValidateCalc Method

Have the Default PHD Server Validate a function.

Overload List

Have the Default PHD Server Validate a function.

public void ValidateCalc(string);

Have the specified PHD Server Validate a function.

public void ValidateCalc(PHDServer,string);

See Also

PHDHistorian Class | Uniformance.PHD Namespace

PHD .NET Wrapper

PHDHistorian.ValidateCalc Method (String)

Have the Default PHD Server Validate a function.

```
[Visual Basic]
Overloads Public Sub ValidateCalc( _
    ByVal sCalc As String _
)
[C#]
public void ValidateCalc(
    string sCalc
);
```

Parameters

sCalc

The function to be validated as a string

Example

```
try

{

    this.PHD.ValidateCalc("Function Test() return 1 ");

    MessageBox.Show("Calculation Validated");

}

catch (PHDErrorException ex)

{

    lblReturn.Text = ex.Message;

    return;

}
```

See Also

[PHDHistorian Class](#) | [Uniformance.PHD Namespace](#) | [PHDHistorian.ValidateCalc Overload List](#)

PHD .NET Wrapper

PHDHistorian.ValidateCalc Method (PHDServer, String)

Have the specified PHD Server Validate a function.

```
[Visual Basic]
Overloads Public Sub ValidateCalc( _
   ByVal Server As PHDServer, _
   ByVal sCalc As String _
)
[C#]
public void ValidateCalc(
   PHDServer Server,
   string sCalc
);
```

Parameters

Server
The PHDServer object

sCalc
The function to be validated as a string

See Also

[PHDHistorian Class](#) | [Uniformance.PHD Namespace](#) | [PHDHistorian.ValidateCalc Overload List](#)

PHD .NET Wrapper	

PHDServer Class

PHDServer class describes a connection to a PHD Server

For a list of all members of this type, see PHDServer Members.

System.Object
   Uniformance.PHD.PHDServer

```
[Visual Basic]
Public Class PHDServer
   Implements IDisposable
[C#]
public class PHDServer : IDisposable
```

Thread Safety

Public static (**Shared** in Visual Basic) members of this type are safe for multithreaded operations. Instance members are **not** guaranteed to be thread-safe.

Example

Create a PHDServer and make it the default Server. The hostname of the server is stored in a text box labeled txtHostname.Text

```
PHDHistorian oPhd = new PHDHistorian();

PHDServer DefaultServer = new PHDServer(txtHostname.Text);



DefaultServer.APIVersion = Uniformance.PHD.SERVERVERSION.API200;

DefaultServer.UserName = "username";

DefaultServer.UserName = "password";



oPhd.DefaultServer = DefaultServer;

.

.

.



DefaultServer.Dispose();
```

Requirements

Namespace: Uniformance.PHD

Assembly: PHDAPINET (in PHDAPINET.dll)

See Also

PHDServer Members | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDServer Members

PHDServer overview

Public Instance Constructors

PHDServer	Overloaded. Initializes a new instance of the PHDServer class.

Public Instance Properties

APIVersion	PHD Server API Version used to communicate with the historian
HostName	PHD Host Name.
Password	For 200 PHD Systems the Username and password are assigned after PHD has a valid NT security connection
Port	PHD Port Number default is 3000.
RequestTimeout	The ConnectionTimeout property identifies the network timeout period used when attempting to connect to the PHD server. If the connection to the PHD server is established within the time period, the request is abandoned. It is in milliseconds.
ServerID	return a unique identifier for the Server GUID
UserName	For 200 PHD Systems the Username and password are assigned after PHD has a valid NT security connection
WindowsPassword	A Windows user account password. It is only used by Remote API to make initial connection.
WindowsUsername	A Windows user account name. It is only used by Remote API to make initial connection.

Public Instance Methods

Dispose	Implement IDisposable. If there is an open connection to PHD then that connection is closed.
Equals (inherited from **Object**)	Determines whether the specified Object is equal

	to the current **Object**.
GetHashCode (inherited from **Object**)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data structures like a hash table.
GetType (inherited from **Object**)	Gets the Type of the current instance.
ToString (inherited from **Object**)	Returns a String that represents the current Object.

Protected Instance Methods

Finalize	Finalizer
MemberwiseClone (inherited from **Object**)	Creates a shallow copy of the current Object.

See Also

PHDServer Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDServer Constructor

Create a PHD Server Object that defaults to the 150 API and uses the PHD_HOST environment variable as a default host name

Overload List

Create a PHD Server Object that defaults to the 150 API and uses the PHD_HOST environment variable as a default host name

public PHDServer();

Create a PHD Server Object that defaults to the 150 API

public PHDServer(string);

Create a PHD Server Object

public PHDServer(string,SERVERVERSION);

See Also

PHDServer Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDServer Constructor ()

Create a PHD Server Object that defaults to the 150 API and uses the PHD_HOST environment variable as a default host name

```
[Visual Basic]
Overloads Public Sub New()
[C#]
public PHDServer();
```

See Also

PHDServer Class | Uniformance.PHD Namespace | PHDServer Constructor Overload List

PHD .NET Wrapper

PHDServer Constructor (String)

Create a PHD Server Object that defaults to the 150 API

```
[Visual Basic]
Overloads Public Sub New( _
    ByVal Hostname As String _
)
[C#]
public PHDServer(
    string Hostname
);
```

Parameters

Hostname
PHD host that the server object will be assigned to

See Also

PHDServer Class | Uniformance.PHD Namespace | PHDServer Constructor Overload List

PHD .NET Wrapper

PHDServer Constructor (String, SERVERVERSION)

Create a PHD Server Object

```
[Visual Basic]
Overloads Public Sub New( _
    ByVal Hostname As String, _
    ByVal APIVersion As SERVERVERSION _
)
[C#]
public PHDServer(
    string Hostname,
    SERVERVERSION APIVersion
);
```

Parameters

Hostname

PHD host that the server object will be assigned to

APIVersion

Enum API selection

See Also

[PHDServer Class](#) | [Uniformance.PHD Namespace](#) | [PHDServer Constructor Overload List](#)

PHD .NET Wrapper	

PHDServer Properties

The properties of the **PHDServer** class are listed below. For a complete list of **PHDServer** class members, see the [PHDServer Members](#) topic.

Public Instance Properties

[APIVersion](#)	PHD Server API Version used to communicate with the historian
[HostName](#)	PHD Host Name.
[Password](#)	For 200 PHD Systems the Username and password are assigned after PHD has a valid NT security connection
[Port](#)	PHD Port Number default is 3000.
[RequestTimeout](#)	The ConnectionTimeout property identifies the network timeout period used when attempting to connect to the PHD server. If the connection to the PHD server is established within the time period, the request is abandoned. It is in milliseconds.
[ServerID](#)	return a unique identifier for the Server GUID
[UserName](#)	For 200 PHD Systems the Username and password are assigned after PHD has a valid NT security connection
[WindowsPassword](#)	A Windows user account password. It is only used by Remote API to make initial connection.
[WindowsUsername](#)	A Windows user account name. It is only used by Remote API to make initial connection.

See Also

[PHDServer Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper

PHDServer.APIVersion Property

PHD Server API Version used to communicate with the historian

```
[Visual Basic]
Public Property APIVersion As SERVERVERSION
[C#]
public SERVERVERSION APIVersion {get; set;}
```

See Also

[PHDServer Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper

PHDServer.HostName Property

PHD Host Name.

```
[Visual Basic]
Public Property HostName As String
[C#]
public string HostName {get; set;}
```

See Also

[PHDServer Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper

PHDServer.Password Property

For 200 PHD Systems the Username and password are assigned after PHD has a valid NT security connection

```
[Visual Basic]
Public Property Password As String
[C#]
public string Password {get; set;}
```

See Also

[PHDServer Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper

PHDServer.Port Property

PHD Port Number default is 3000.

```
[Visual Basic]
Public Property Port As Integer
[C#]
public int Port {get; set;}
```

See Also

PHDServer Class | Uniformance.PHD Namespace

PHD .NET Wrapper

PHDServer.RequestTimeout Property

The ConnectionTimeout property identifies the network timeout period used when attempting to connect to the PHD server. If the connection to the PHD server is established within the time period, the request is abandoned. It is in milliseconds.

```
[Visual Basic]
Public Property RequestTimeout As UInt32
[C#]
public uint RequestTimeout {get; set;}
```

See Also

PHDServer Class | Uniformance.PHD Namespace

PHD .NET Wrapper

PHDServer.ServerID Property

return a unique identifier for the Server GUID

```
[Visual Basic]
Public ReadOnly Property ServerID As String
[C#]
public string ServerID {get;}
```

See Also

PHDServer Class | Uniformance.PHD Namespace

PHD .NET Wrapper

PHDServer.UserName Property

For 200 PHD Systems the Username and password are assigned after PHD has a valid NT security connection

```
[Visual Basic]
Public Property UserName As String
```

```
[C#]
public string UserName {get; set;}
```

See Also

PHDServer Class | Uniformance.PHD Namespace

| *PHD .NET Wrapper* | |

PHDServer.WindowsPassword Property

A Windows user account password. It is only used by Remote API to make initial connection.

```
[Visual Basic]
Public Property WindowsPassword As String
[C#]
public string WindowsPassword {get; set;}
```

See Also

PHDServer Class | Uniformance.PHD Namespace

| *PHD .NET Wrapper* | |

PHDServer.WindowsUsername Property

A Windows user account name. It is only used by Remote API to make initial connection.

```
[Visual Basic]
Public Property WindowsUsername As String
[C#]
public string WindowsUsername {get; set;}
```

See Also

PHDServer Class | Uniformance.PHD Namespace

| *PHD .NET Wrapper* | |

PHDServer Methods

The methods of the **PHDServer** class are listed below. For a complete list of **PHDServer** class members, see the PHDServer Members topic.

Public Instance Methods

| Dispose | Implement IDisposable. If there is an open connection to PHD then that connection is closed. |

≡♦Equals (inherited from **Object**)	Determines whether the specified Object is equal to the current **Object**.
≡♦GetHashCode (inherited from **Object**)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data structures like a hash table.
≡♦GetType (inherited from **Object**)	Gets the Type of the current instance.
≡♦ToString (inherited from **Object**)	Returns a String that represents the current Object.

Protected Instance Methods

♦Finalize	Finalizer
♦MemberwiseClone (inherited from **Object**)	Creates a shallow copy of the current Object.

See Also

PHDServer Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDServer.Dispose Method

Implement IDisposable. If there is an open connection to PHD then that connection is closed.

```
[Visual Basic]
NotOverridable Public Sub Dispose() _
  Implements IDisposable.Dispose
```
```
[C#]
public void Dispose();
```

Implements

IDisposable.Dispose

See Also

PHDServer Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

PHDServer.Finalize Method

Finalizer

```
[Visual Basic]
Overrides Protected Sub Finalize()
```

```
[C#]
protected override void Finalize();
```

See Also

PHDServer Class | Uniformance.PHD Namespace

| *PHD .NET Wrapper* | |

RDIListStruct Class

For a list of all members of this type, see RDIListStruct Members.

System.Object
  Uniformance.PHD.RDIListStruct

```
[Visual Basic]
Public Class RDIListStruct
[C#]
public class RDIListStruct
```

Thread Safety

Public static (**Shared** in Visual Basic) members of this type are safe for multithreaded operations. Instance members are **not** guaranteed to be thread-safe.

Requirements

Namespace: Uniformance.PHD

Assembly: PHDAPINET (in PHDAPINET.dll)

See Also

RDIListStruct Members | Uniformance.PHD Namespace

| *PHD .NET Wrapper* | |

RDIListStruct Members

RDIListStruct overview

Public Instance Constructors

RDIListStruct Constructor	Create an instance of the class.

Public Instance Properties

RDIName	PHD Tagname

Public Instance Methods

≡◆Equals (inherited from **Object**)	Determines whether the specified Object is equal to the current **Object**.
≡◆GetHashCode (inherited from **Object**)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data structures like a hash table.
≡◆GetType (inherited from **Object**)	Gets the Type of the current instance.
≡◆ToString (inherited from **Object**)	Returns a String that represents the current Object.

Protected Instance Methods

◈◆Finalize (inherited from **Object**)	Allows an Object to attempt to free resources and perform other cleanup operations before the **Object** is reclaimed by garbage collection.
◈◆MemberwiseClone (inherited from **Object**)	Creates a shallow copy of the current Object.

See Also

RDIListStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

RDIListStruct Constructor

Create an instance of the class.

```
[Visual Basic]
Overloads Public Sub New()
[C#]
public RDIListStruct();
```

See Also

RDIListStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

RDIListStruct Properties

The properties of the **RDIListStruct** class are listed below. For a complete list of **RDIListStruct** class members, see the RDIListStruct Members topic.

Public Instance Properties

RDIName	PHD Tagname

See Also

RDIListStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

RDIListStruct.RDIName Property

PHD Tagname

```
[Visual Basic]
Public ReadOnly Property RDIName As String
[C#]
public string RDIName {get;}
```

See Also

RDIListStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

REDUCTIONOFFSET Enumeration

Identifies the offset applied when retrieving reductions for tags.

```
[Visual Basic]
Public Enum REDUCTIONOFFSET
[C#]
public enum REDUCTIONOFFSET
```

Members

Member Name	Description
After	The data value returned specifies the reduction for the interval that follows the timestamp.
Around	The data value returned specifies the reduction for the interval that the timestamp is the midpoint.
Before	The data value returned specifies the reduction for the interval that precedes the timestamp.

Requirements

Namespace: Uniformance.PHD

Assembly: PHDAPINET (in PHDAPINET.dll)

See Also

Uniformance.PHD Namespace

PHD .NET Wrapper	

REDUCTIONTYPE Enumeration

Identifies the type of reduction fetched for the tags..

```
[Visual Basic]
Public Enum REDUCTIONTYPE
[C#]
```
```
public enum REDUCTIONTYPE
```

Members

Member Name	Description
None	No reduction
Average	The average data over the reduction frequency
Delta	The difference between the value and the end and start of the reduction frequency.
Minimum	The minimum value over the reduction frequency.
Maximum	The maximum value over the reduction frequency.
StandardDeviation	The standard deviation of the samples over the reduction frequency.
RegressionSlope	The slope of the regression over the reduction frequency.
RegressionConstant	The y-intercept of the regression line.
RegressionDeviation	The standard deviation of the regression line over the reduction frequency.
First	The value at the start of the reduction frequency.
Last	The value at the end of the reduction frequency.

Requirements

Namespace: Uniformance.PHD

Assembly: PHDAPINET (in PHDAPINET.dll)

See Also

Uniformance.PHD Namespace

| *PHD .NET Wrapper* | |

SAMPLETYPE Enumeration

Sampling types supported.

```
[Visual Basic]
Public Enum SAMPLETYPE
[C#]
public enum SAMPLETYPE
```

Members

Member Name	Description
Snapshot	The value is the interpolation between raw data values at the sample time.
Average	The value is the average from the data around the sample time.
Resampled	The value is resampled from the raw data based on the resample method.
Raw	The value is raw data.
InterpolatedRaw	Similar to the retrieval of raw data except that no data of -1 confidence is returned.

Requirements

Namespace: Uniformance.PHD

Assembly: PHDAPINET (in PHDAPINET.dll)

See Also

Uniformance.PHD Namespace

| *PHD .NET Wrapper* | |

SERVERVERSION Enumeration

Enum list if Server API Supported

```
[Visual Basic]
Public Enum SERVERVERSION
[C#]
public enum SERVERVERSION
```

Members

Member Name	Description
DEFAULT	The default uses the 150 API as it can communicate with both 150 and 200 historians
API150	The 150 API can communicate with both 150 and 200 PHD
API200	The 200 API will only communicate with 200 PHD historians
RAPI200	The 200 RAPI will only communicate with 200 PHD historians

Requirements

Namespace: Uniformance.PHD

Assembly: PHDAPINET (in PHDAPINET.dll)

See Also

Uniformance.PHD Namespace

PHD .NET Wrapper	

Tag Class

Describes a single Tag

For a list of all members of this type, see Tag Members.

System.Object
   Uniformance.PHD.Tag

```
[Visual Basic]
Public Class Tag
[C#]
public class Tag
```

Thread Safety

Public static (**Shared** in Visual Basic) members of this type are safe for multithreaded operations. Instance members are **not** guaranteed to be thread-safe.

Remarks

This class describes the properties and methods associated with a Tag

Example

Create a new instance of a Tag. Add the Tag to the Tags collection

```
Tag _tag = new Tag("kst20000");
```

```
Tags oTags = new Tags();



oTags.Add(_tag);
```

Requirements

> **Namespace:** Uniformance.PHD

> **Assembly:** PHDAPINET (in PHDAPINET.dll)

See Also

> Tag Members | Uniformance.PHD Namespace

PHD .NET Wrapper	

Tag Members

> Tag overview

Public Instance Constructors

Tag	Overloaded. Initializes a new instance of the Tag class.

Public Instance Properties

EnumEnable	The Tagname of the Tag.
Server	The Server Object that the tag is associated with. No Server object supplied the base Server is used
TagName	The Tagname of the Tag.
Units	The Engineering Units to be converted to for the Tag.

Public Instance Methods

Equals (inherited from **Object**)	Determines whether the specified Object is equal to the current **Object**.
GetHashCode (inherited from **Object**)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data structures like a hash table.
GetType (inherited from **Object**)	Gets the Type of the current instance.

ToString (inherited from **Object**)	Returns a String that represents the current Object.

Protected Instance Methods

Finalize (inherited from **Object**)	Allows an Object to attempt to free resources and perform other cleanup operations before the **Object** is reclaimed by garbage collection.
MemberwiseClone (inherited from **Object**)	Creates a shallow copy of the current Object.

See Also

Tag Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

Tag Constructor

Create a new instance of a Tag.

Overload List

Create a new instance of a Tag.

public Tag();

Tag constructor with tagname as a parameter

public Tag(string);

See Also

Tag Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

Tag Constructor ()

Create a new instance of a Tag.

```
[Visual Basic]
Overloads Public Sub New()
[C#]
public Tag();
```

Example

Add a Tag to the Tags collection

```
Tag _tag = new Tag();
```

```
_tag.TagName = "kst20000";

Tags oTags = new Tags();



oTags.Add(_tag);
```

See Also

Tag Class | Uniformance.PHD Namespace | Tag Constructor Overload List

PHD .NET Wrapper

Tag Constructor (String)

Tag constructor with tagname as a parameter

```
[Visual Basic]
Overloads Public Sub New( _
    ByVal tagname As String _
)
[C#]
public Tag(
    string tagname
);
```

Parameters

tagname

Example

Create a new instance of a Tag. Add the Tag to the Tags collection

```
Tag _tag = new Tag("kst20000");

Tags oTags = new Tags();



oTags.Add(_tag);
```

See Also

Tag Class | Uniformance.PHD Namespace | Tag Constructor Overload List

PHD .NET Wrapper

Tag Properties

The properties of the **Tag** class are listed below. For a complete list of **Tag** class members, see the [Tag Members](#) topic.

Public Instance Properties

[EnumEnable](#)	The Tagname of the Tag.
[Server](#)	The Server Object that the tag is associated with. No Server object supplied the base Server is used
[TagName](#)	The Tagname of the Tag.
[Units](#)	The Engineering Units to be converted to for the Tag.

See Also

[Tag Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper

Tag.EnumEnable Property

The Tagname of the Tag.

```
[Visual Basic]
Public Property EnumEnable As Boolean
[C#]
public bool EnumEnable {get; set;}
```

See Also

[Tag Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper

Tag.Server Property

The Server Object that the tag is associated with. No Server object supplied the base Server is used

```
[Visual Basic]
Public Property Server As PHDServer
[C#]
public PHDServer Server {get; set;}
```

See Also

[Tag Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper	

Tag.TagName Property

The Tagname of the Tag.

```
[Visual Basic]
Public Property TagName As String
[C#]
public string TagName {get; set;}
```

See Also

Tag Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

Tag.Units Property

The Engineering Units to be converted to for the Tag.

```
[Visual Basic]
Public Property Units As String
[C#]
public string Units {get; set;}
```

See Also

Tag Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

TagFilter Class

A structure is used to specify tag filter conditions

For a list of all members of this type, see TagFilter Members.

System.Object
  Uniformance.PHD.TagFilter

```
[Visual Basic]
Public Class TagFilter
[C#]
public class TagFilter
```

Thread Safety

Public static (**Shared** in Visual Basic) members of this type are safe for multithreaded operations. Instance members are **not** guaranteed to be thread-safe.

Requirements

Namespace: Uniformance.PHD

Assembly: PHDAPINET (in PHDAPINET.dll)

See Also

TagFilter Members | Uniformance.PHD Namespace

PHD .NET Wrapper	

TagFilter Members

TagFilter overview

Public Instance Constructors

TagFilter Constructor	Constructor

Public Instance Fields

Collector	Collector Name Pattern
DataType	Data Type
ParentTagname	Parent Tag Name Pattern
ScanRate	Scan frequency in second
SourceTagname	Source Tag Name Pattern
SourceUnits	Source Tag Engineering Unit Pattern
Tagname	Tag Name Pattern
TagUnits	Engineering Unit Pattern

Public Instance Methods

Clear	Clear all data members
Equals (inherited from **Object**)	Determines whether the specified Object is equal to the current **Object**.
GetHashCode (inherited from **Object**)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data structures like a hash table.

GetType (inherited from **Object**)	Gets the Type of the current instance.
ToString (inherited from **Object**)	Returns a String that represents the current Object.

Protected Instance Methods

Finalize (inherited from **Object**)	Allows an Object to attempt to free resources and perform other cleanup operations before the **Object** is reclaimed by garbage collection.
MemberwiseClone (inherited from **Object**)	Creates a shallow copy of the current Object.

See Also

TagFilter Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

TagFilter Constructor

Constructor

```
[Visual Basic]
Public Sub New()
[C#]
public TagFilter();
```

See Also

TagFilter Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

TagFilter Fields

The fields of the **TagFilter** class are listed below. For a complete list of **TagFilter** class members, see the TagFilter Members topic.

Public Instance Fields

Collector	Collector Name Pattern
DataType	Data Type
ParentTagname	Parent Tag Name Pattern

◈ ScanRate	Scan frequency in second
◈ SourceTagname	Source Tag Name Pattern
◈ SourceUnits	Source Tag Engineering Unit Pattern
◈ Tagname	Tag Name Pattern
◈ TagUnits	Engineering Unit Pattern

See Also

TagFilter Class | Uniformance.PHD Namespace

PHD .NET Wrapper

TagFilter.Collector Field

Collector Name Pattern

```
[Visual Basic]
Public Collector As String
[C#]
public string Collector;
```

See Also

TagFilter Class | Uniformance.PHD Namespace

PHD .NET Wrapper

TagFilter.DataType Field

Data Type

```
[Visual Basic]
Public DataType As Char
[C#]
public char DataType;
```

See Also

TagFilter Class | Uniformance.PHD Namespace

PHD .NET Wrapper

TagFilter.ParentTagname Field

Parent Tag Name Pattern

```
[Visual Basic]
Public ParentTagname As String
[C#]
public string ParentTagname;
```

See Also

TagFilter Class | Uniformance.PHD Namespace

PHD .NET Wrapper

TagFilter.ScanRate Field

Scan frequency in second

```
[Visual Basic]
Public ScanRate As Long
[C#]
public long ScanRate;
```

See Also

TagFilter Class | Uniformance.PHD Namespace

PHD .NET Wrapper

TagFilter.SourceTagname Field

Source Tag Name Pattern

```
[Visual Basic]
Public SourceTagname As String
[C#]
public string SourceTagname;
```

See Also

TagFilter Class | Uniformance.PHD Namespace

PHD .NET Wrapper

TagFilter.SourceUnits Field

Source Tag Engineering Unit Pattern

```
[Visual Basic]
Public SourceUnits As String
```

```
[C#]
public string SourceUnits;
```

See Also

[TagFilter Class](#) | [Uniformance.PHD Namespace](#)

| *PHD .NET Wrapper* | |

TagFilter.Tagname Field

Tag Name Pattern

```
[Visual Basic]
Public Tagname As String
[C#]
public string Tagname;
```

See Also

[TagFilter Class](#) | [Uniformance.PHD Namespace](#)

| *PHD .NET Wrapper* | |

TagFilter.TagUnits Field

Engineering Unit Pattern

```
[Visual Basic]
Public TagUnits As String
[C#]
public string TagUnits;
```

See Also

[TagFilter Class](#) | [Uniformance.PHD Namespace](#)

| *PHD .NET Wrapper* | |

TagFilter Methods

The methods of the **TagFilter** class are listed below. For a complete list of **TagFilter** class members, see the [TagFilter Members](#) topic.

Public Instance Methods

[Clear](#)	Clear all data members

Equals (inherited from **Object**)	Determines whether the specified Object is equal to the current **Object**.
GetHashCode (inherited from **Object**)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data structures like a hash table.
GetType (inherited from **Object**)	Gets the Type of the current instance.
ToString (inherited from **Object**)	Returns a String that represents the current Object.

Protected Instance Methods

Finalize (inherited from **Object**)	Allows an Object to attempt to free resources and perform other cleanup operations before the **Object** is reclaimed by garbage collection.
MemberwiseClone (inherited from **Object**)	Creates a shallow copy of the current Object.

See Also

TagFilter Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

TagFilter.Clear Method

Clear all data members

```
[Visual Basic]
Public Sub Clear()

[C#]
public void Clear();
```

See Also

TagFilter Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

TagListStruct Class

For a list of all members of this type, see TagListStruct Members.

System.Object
  Uniformance.PHD.TagListStruct

```
[Visual Basic]
Public Class TagListStruct

[C#]
```

```
public class TagListStruct
```

Thread Safety

Public static (**Shared** in Visual Basic) members of this type are safe for multithreaded operations. Instance members are **not** guaranteed to be thread-safe.

Requirements

Namespace: Uniformance.PHD

Assembly: PHDAPINET (in PHDAPINET.dll)

See Also

TagListStruct Members | Uniformance.PHD Namespace

PHD .NET Wrapper	

TagListStruct Members

TagListStruct overview

Public Instance Constructors

TagListStruct Constructor	Create an instance of the class.

Public Instance Properties

CollectorName	Collector Name
DataLength	Data Length
DataType	Data Type
ParentTagname	Parent Tag Name
ParentTagno	Parent Tag Number
SourceTagname	Source Tagname
SourceUnits	Source Units
Tagname	PHD Tagname
Tagno	PHD Tagno
Units	Engineering Units

Public Instance Methods

Equals (inherited from **Object**)	Determines whether the specified Object is equal to the current **Object**.
GetHashCode (inherited from **Object**)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data structures like a hash table.
GetType (inherited from **Object**)	Gets the Type of the current instance.
ToString (inherited from **Object**)	Returns a String that represents the current Object.

Protected Instance Methods

Finalize (inherited from **Object**)	Allows an Object to attempt to free resources and perform other cleanup operations before the **Object** is reclaimed by garbage collection.
MemberwiseClone (inherited from **Object**)	Creates a shallow copy of the current Object.

See Also

TagListStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper

TagListStruct Constructor

Create an instance of the class.

```
[Visual Basic]
Overloads Public Sub New()
[C#]
public TagListStruct();
```

See Also

TagListStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper

TagListStruct Properties

The properties of the **TagListStruct** class are listed below. For a complete list of **TagListStruct** class members, see the TagListStruct Members topic.

Public Instance Properties

⊞CollectorName	Collector Name
⊞DataLength	Data Length
⊞DataType	Data Type
⊞ParentTagname	Parent Tag Name
⊞ParentTagno	Parent Tag Number
⊞SourceTagname	Source Tagname
⊞SourceUnits	Source Units
⊞Tagname	PHD Tagname
⊞Tagno	PHD Tagno
⊞Units	Engineering Units

See Also

TagListStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

TagListStruct.CollectorName Property

Collector Name

```
[Visual Basic]
Public ReadOnly Property CollectorName As String
[C#]
public string CollectorName {get;}
```

See Also

TagListStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

TagListStruct.DataLength Property

Data Length

```
[Visual Basic]
Public ReadOnly Property DataLength As Short
```

```
[C#]
public short DataLength {get;}
```

See Also

[TagListStruct Class](#) | [Uniformance.PHD Namespace](#)

| *PHD .NET Wrapper* | |

TagListStruct.DataType Property

Data Type

```
[Visual Basic]
Public ReadOnly Property DataType As Char
[C#]
public char DataType {get;}
```

See Also

[TagListStruct Class](#) | [Uniformance.PHD Namespace](#)

| *PHD .NET Wrapper* | |

TagListStruct.ParentTagname Property

Parent Tag Name

```
[Visual Basic]
Public ReadOnly Property ParentTagname As String
[C#]
public string ParentTagname {get;}
```

See Also

[TagListStruct Class](#) | [Uniformance.PHD Namespace](#)

| *PHD .NET Wrapper* | |

TagListStruct.ParentTagno Property

Parent Tag Number

```
[Visual Basic]
Public ReadOnly Property ParentTagno As Integer
[C#]
public int ParentTagno {get;}
```

See Also

[TagListStruct Class](#) | [Uniformance.PHD Namespace](#)

PHD .NET Wrapper

TagListStruct.SourceTagname Property

Source Tagname

```
[Visual Basic]
Public ReadOnly Property SourceTagname As String
[C#]
public string SourceTagname {get;}
```

See Also

TagListStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper

TagListStruct.SourceUnits Property

Source Units

```
[Visual Basic]
Public ReadOnly Property SourceUnits As String
[C#]
public string SourceUnits {get;}
```

See Also

TagListStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper

TagListStruct.Tagname Property

PHD Tagname

```
[Visual Basic]
Public ReadOnly Property Tagname As String
[C#]
public string Tagname {get;}
```

See Also

TagListStruct Class | Uniformance.PHD Namespace

PHD .NET Wrapper

TagListStruct.Tagno Property

PHD Tagno

```
[Visual Basic]
Public ReadOnly Property Tagno As Integer
[C#]
public int Tagno {get;}
```

See Also

TagListStruct Class | Uniformance.PHD Namespace

| *PHD .NET Wrapper* | |

TagListStruct.Units Property

Engineering Units

```
[Visual Basic]
Public ReadOnly Property Units As String
[C#]
public string Units {get;}
```

See Also

TagListStruct Class | Uniformance.PHD Namespace

| *PHD .NET Wrapper* | |

Tags Class

The Tags class is used to hold a collection of Tag items

For a list of all members of this type, see Tags Members.

System.Object
  Uniformance.PHD.Tags

```
[Visual Basic]
Public Class Tags
[C#]
public class Tags
```

Thread Safety

Public static (**Shared** in Visual Basic) members of this type are safe for multithreaded operations. Instance members are **not** guaranteed to be thread-safe.

Remarks

Example

Create a new instance of the Tags collection. Add 3 tags to the collection

```vb
vb

Dim oTags as Tags = new Tags()



oTags.Add(new Tag("kst20000"))

oTags.Add(new Tag("kst20001"))

oTags.Add(new Tag("kst20002"))
```

```c#
c#

Tags oTags = new Tags();



oTags.Add(new Tag("kst20000"));

oTags.Add(new Tag("kst20001"));

oTags.Add(new Tag("kst20002"));
```

Requirements

Namespace: Uniformance.PHD

Assembly: PHDAPINET (in PHDAPINET.dll)

See Also

Tags Members | Uniformance.PHD Namespace

PHD .NET Wrapper	

Tags Members

Tags overview

Public Instance Constructors

Tags Constructor	Tags class is a container for Tag items

Public Instance Properties

📇Count	return the number of Tag Items in the Tags Collection.
📇Item	Overloaded. Indexer to provide read access to array list for the hashtable add a tagname and return the Tag object reference

Public Instance Methods

Add	Add Tag to HashTable. Use Tagname as the key when no name is supplied. Use Object Hostname when no Hostname is supplied.
Equals (inherited from **Object**)	Determines whether the specified Object is equal to the current **Object**.
GetEnumerator	Enumerates the elements in the Tag Collection
GetHashCode (inherited from **Object**)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data structures like a hash table.
GetType (inherited from **Object**)	Gets the Type of the current instance.
Remove	Remove a Tag from the Tags Collection.
RemoveAll	Remove all Tag items from the Tags Collection.
ToString (inherited from **Object**)	Returns a String that represents the current Object.

Protected Instance Methods

Finalize (inherited from **Object**)	Allows an Object to attempt to free resources and perform other cleanup operations before the **Object** is reclaimed by garbage collection.
MemberwiseClone (inherited from **Object**)	Creates a shallow copy of the current Object.

See Also

Tags Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

Tags Constructor

Tags class is a container for Tag items

```
[Visual Basic]
Public Sub New()
```

```
[C#]
public Tags();
```

Example

Create a new instance of the Tags collection. Add 3 tags to the collection

```
vb

Dim oTags as Tags = new Tags()


oTags.Add(new Tag("kst20000"))

oTags.Add(new Tag("kst20001"))

oTags.Add(new Tag("kst20002"))




c#

Tags oTags = new Tags();


oTags.Add(new Tag("kst20000"));

oTags.Add(new Tag("kst20001"));

oTags.Add(new Tag("kst20002"));
```

See Also

Tags Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

Tags Properties

The properties of the **Tags** class are listed below. For a complete list of **Tags** class members, see the Tags Members topic.

Public Instance Properties

Count	return the number of Tag Items in the Tags Collection.
Item	Overloaded. Indexer to provide read access to array list for the hashtable add a tagname and return the Tag object reference

See Also

Tags Class | Uniformance.PHD Namespace

PHD .NET Wrapper

Tags.Count Property

return the number of Tag Items in the Tags Collection.

```
[Visual Basic]
Public ReadOnly Property Count As Integer
[C#]
public int Count {get;}
```

Example

Create a new instance of the Tags collection. Add a tag name to the collection return the count of the number of items in the collection

```
vb

Dim oTags as Tags = new Tags()

oTags.Add(new Tag("kst20000"))



Dim _count as integer = oTags.Count




c#

Tags oTags = new Tags();

oTags.Add(new Tag("kst20000"));



int _count = oTags.Count;
```

See Also

Tags Class | Uniformance.PHD Namespace

PHD .NET Wrapper

Tags.Item Property

Indexer to provide read access to array list for the hashtable add a tagname and return the Tag object

reference

Overload List

Indexer to provide read access to array list for the hashtable add a tagname and return the Tag object reference

public Tag this[int] {get;}

Indexer to provide read access to array list for the hashtable add a tagname and return the Tag object reference

public Tag[] this[string] {get;}

See Also

Tags Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

Tags.Item Property (Int32)

Indexer to provide read access to array list for the hashtable add a tagname and return the Tag object reference

```
[Visual Basic]
Overloads Public Default ReadOnly Property Item( _
   ByVal index As Integer _
) As Tag
[C#]
public Tag this[
   int index
] {get;}
```

Example

Create a new instance of the Tags collection. Add a tag name to the collection and assign that item in the collection to a Tag

```
vb

Dim oTags as Tags = new Tags();

oTags.Add(new Tag("kst20000"))



Dim _tag as Tag = oTags(0)




c#

Tags oTags = new Tags();

oTags.Add(new Tag("kst20000"));
```

```
Tag _tag = oTags[0];
```

See Also

Tags Class | Uniformance.PHD Namespace | Tags.Item Overload List

| *PHD .NET Wrapper* | |

Tags.Item Property (String)

Indexer to provide read access to array list for the hashtable add a tagname and return the Tag object reference

```
[Visual Basic]
Overloads Public Default ReadOnly Property Item( _
   ByVal TagName As String _
) As Tag()
[C#]
public Tag[] this[
   string TagName
] {get;}
```

Example

Create a new instance of the Tags collection. Add a tag name to the collection and assign that item in the collection to a Tag[]

```
vb

Dim oTags as Tags = new Tags()

oTags.Add(new Tag("kst20000"))



Dim _tags() as Tag  = oTags("kst20000")




c#

Tags oTags = new Tags();

oTags.Add(new Tag("kst20000"));



Tag[] _tags = oTags["kst20000"];
```

See Also

Tags Class | Uniformance.PHD Namespace | Tags.Item Overload List

PHD .NET Wrapper	

Tags Methods

The methods of the **Tags** class are listed below. For a complete list of **Tags** class members, see the Tags Members topic.

Public Instance Methods

Add	Add Tag to HashTable. Use Tagname as the key when no name is supplied. Use Object Hostname when no Hostname is supplied.
Equals (inherited from **Object**)	Determines whether the specified Object is equal to the current **Object**.
GetEnumerator	Enumerates the elements in the Tag Collection
GetHashCode (inherited from **Object**)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data structures like a hash table.
GetType (inherited from **Object**)	Gets the Type of the current instance.
Remove	Remove a Tag from the Tags Collection.
RemoveAll	Remove all Tag items from the Tags Collection.
ToString (inherited from **Object**)	Returns a String that represents the current Object.

Protected Instance Methods

Finalize (inherited from **Object**)	Allows an Object to attempt to free resources and perform other cleanup operations before the **Object** is reclaimed by garbage collection.
MemberwiseClone (inherited from **Object**)	Creates a shallow copy of the current Object.

See Also

Tags Class | Uniformance.PHD Namespace

PHD .NET Wrapper	

Tags.Add Method

Add Tag to HashTable.

Use Tagname as the key when no name is supplied.

Use Object Hostname when no Hostname is supplied.

```
[Visual Basic]
Public Sub Add( _
    ByVal oTag As Tag _
)
[C#]
public void Add(
    Tag oTag
);
```

See Also

Tags Class | Uniformance.PHD Namespace

PHD .NET Wrapper

Tags.GetEnumerator Method

Enumerates the elements in the Tag Collection

```
[Visual Basic]
Public Function GetEnumerator() As IEnumerator
[C#]
public IEnumerator GetEnumerator();
```

See Also

Tags Class | Uniformance.PHD Namespace

PHD .NET Wrapper

Tags.Remove Method

Remove a Tag from the Tags Collection.

```
[Visual Basic]
Public Sub Remove( _
    ByVal oTag As Tag _
)
[C#]
public void Remove(
    Tag oTag
);
```

Example

Create a new instance of the Tags collection. Add a tag name to the collection then remove that tag from the collection the collection

```vb
Dim oTags as Tags = new Tags()
```

```
Dim _tag as Tag = new Tag("kst20000")

oTags.Add(_tag)



oTags.Remove(_tag)




c#

Tags oTags = new Tags();

Tag _tag = new Tag("kst20000");



oTags.Add(_tag);



oTags.Remove(_tag);
```

See Also

Tags Class | Uniformance.PHD Namespace

PHD .NET Wrapper

Tags.RemoveAll Method

Remove all Tag items from the Tags Collection.

```
[Visual Basic]
Public Sub RemoveAll()
[C#]
public void RemoveAll();
```

See Also

Tags Class | Uniformance.PHD Namespace

PHD .NET Wrapper

TagUpdateNotify Delegate

Notification from PHD server when tags are updated

```
[Visual Basic]
Public Delegate Sub TagUpdateNotify( _
    ByVal TagCount As Integer, _
```

```
     ByVal TagnoList As Integer() _
)
[C#]
public delegate void TagUpdateNotify(
     int TagCount,
     int[] TagnoList
);
```

Remarks

Example

Create a new instance of the Tags collection. Add 3 tags to the collection

```
c#


PHDHistorian oPhd;

.

.

.




try

{

      oPhd = new PHDHistorian(false)

      Uniformance.PHD.TagUpdateNotify callback = new
TagUpdateNotify(TagUpdateCallbackHandler);

      oPHD.TagUpdateHandler(callback);

}

catch(Exception ex)

{

      MessageBox.Show( ex.Message);

}



.

.
```

```
.



private void TagUpdateCallbackHandler(int TagCount, int[] TagnoList)

{

     MessageBox.Show("TagCount=" + TagCount.ToString() + "\nFirst Tagno: " +
TagnoList[0].ToString());



     // Clear the TagUpdate list so that you only see new updates next time

     oPHD.ClearTagUpdateList();

}
```

Requirements

Namespace: Uniformance.PHD

Assembly: PHDAPINET (in PHDAPINET.dll)

See Also

Uniformance.PHD Namespace

Index

Honeywell