



Web Services API Guide

OnPremises and Private Cloud

AgilePoint NX v9.0, Software Update 1

Document Revision r9.1.1

January 2026

Contents

- Legal Statements and Policies..... 11**
- Web Services API..... 13**
- Basic Programming Tasks..... 14**
 - Code Examples..... 14
 - AgilePoint Server Running in Windows Service Mode..... 14
 - Run the Windows Service Client Utility Tool..... 16
 - Authentication..... 17
 - Namespace Reference..... 18
 - Catching Exceptions..... 19
 - Making Calls..... 19
- Methods..... 21**
 - Common Methods..... 21
 - Check Authentication..... 21
 - Surrogate..... 23
 - Surrogate With Application Name and Locale..... 25
 - Set Client Application Name..... 28
 - Set Client Locale..... 29
 - Process Definition Methods..... 31
 - CheckOut Process Definition..... 32
 - Check In Process Definition..... 34
 - Create Process Definition..... 36

Delete Process Definition.....	39
Get Base Process Definition ID.....	40
Get Process Definition By Base Process Definition ID.....	42
Get Process Definition Graphics.....	44
Get Process Definition Name and Version.....	47
Get Process Definitions.....	49
Get Process Definition XML.....	50
Get Released Process Definition ID.....	52
Get Released Process Definitions.....	54
Release Process Definition.....	56
Uncheckout Process Definition.....	57
Update Process Definition.....	59
Methods for Process Instances.....	61
Cancel Process Instance.....	61
Create Process Instance.....	63
Create Process Instance (Extended with Initiator and Work Object Info).....	67
Create Process Instance (Extended with Initiator).....	72
Create Process Instance (Extended Method).....	76
Delete Process Instance.....	81
Get Events By Process Instance ID.....	83
Get Process Instance.....	84
Get Process Instance Attribute.....	86
Get Process Instance Attributes.....	89
Merge Process Instances.....	91
Migrate Process Instances.....	95
Promote Process Instance.....	98

Query Process Instances.....	99
Query Process Instances (Extended Method).....	102
Resume Process Instance.....	104
Rollback Process Instance.....	106
Split Process Instance.....	108
Start Process Instance.....	111
Suspend Process Instance.....	113
Update Process Instance.....	115
Methods for Activity Instances.....	118
Cancel Activity Instance.....	118
Get Activity Instance.....	120
Get Activity Instance Status.....	122
Get Activity Instances By Process Instance ID.....	124
Query Activity Instances.....	126
Rollback Activity Instance.....	128
Rollback Activity Instances.....	130
Methods for Manual Work Items (Tasks).....	133
Assign Work Item.....	133
Assign Work Item (Extended Method).....	135
Cancel Work Item.....	137
Cancel Work Item (Extended Method).....	139
Complete Work Item.....	141
Complete Work Item (Extended Method).....	144
Create Linked Work Item.....	146
Create Linked Work Item (Extended Method).....	149
Create Pseudo Work Item.....	153

Create Work Item.....	157
Get Work Item.....	160
Get Work List By User ID.....	162
Query Work List.....	164
Query Work List (Extended Method).....	166
Reassign Update Work Item.....	168
Reassign Work Item.....	171
Reassign Work Item (Extended Method).....	173
Undo Assign Work Item.....	176
Undo Assign Work Item (Extended Method).....	178
Update Work Item.....	181
Methods for Automatic Work Items (Procedures).....	183
Cancel Procedure.....	183
Complete Procedure.....	185
Get Procedure.....	188
Query Procedure List.....	189
User Delegation.....	191
Activate Delegation.....	191
Add Delegation.....	193
Cancel Delegation.....	195
Get Delegation.....	197
Get Delegations.....	199
Remove Delegation.....	202
Update Delegation.....	204
Methods for Notifications.....	205
Cancel Mail Deliverable.....	206

Get Expecting Send Mail Deliverable.....	208
Get Mail Deliverables.....	209
Resend Mail Deliverable.....	211
Methods for Events.....	213
Get Event.....	213
Send Mail.....	215
Send Mail.....	215
Send Mail (Extended Method).....	218
Send Mail (Extended Method with Priority).....	221
Methods for Custom Attributes.....	225
Get Custom Attribute.....	225
Get Custom Attributes.....	227
Get Custom Attributes (Extended Method).....	229
Remove Custom Attribute.....	231
Remove Custom Attributes.....	233
Set Custom Attribute.....	235
Set Custom Attributes.....	238
Methods for Archiving and Restoring Processes.....	240
Archive Process Instance.....	240
Restore Process Instance.....	242
Query Archived Process Instances.....	244
Group, Role, and Rights.....	246
Add Group.....	246
Add Group Member.....	249
Add Role.....	252
Add Role Member.....	256

Enabled Group Member.....	259
Get Access Right Names.....	262
Get Access Rights.....	263
Get Group.....	265
Get Group Members.....	267
Get Groups.....	269
Get Role.....	271
Get Roles.....	273
Query Role Members.....	275
Remove Group.....	277
Remove Group Member.....	279
Remove Role.....	281
Remove Role Member.....	283
Update Group.....	286
Update Role.....	288
Organization Properties.....	291
Get Organization Properties.....	291
Remove Organization Properties.....	293
Update Organization Properties.....	295
Other Web Services.....	297
Query Audit Trail.....	297
Query Database.....	299
Query Database (Extended Method).....	301
Administrative Service.....	303
Get All EMail Templates.....	303
Get Database Information.....	305

Get Domain Groups.....	306
Get Domain Group Members.....	309
Get Domain Name.....	311
Get Domain Users.....	313
Get EMail Template.....	315
Get Locale.....	317
Get Registered User.....	318
Get Register Users.....	320
Get Register User Icons.....	322
Get Sender Email Address.....	323
Get SMTP Server.....	325
Get System Performance Information.....	326
Get System User.....	328
Query Register Users.....	330
Register User.....	332
Unregister User.....	334
Update Registered User.....	336
Update Registered User Icon.....	338
Report Configuration Methods.....	340
Add Report Configuration.....	340
Get All Report Configurations.....	342
Get Report Configuration.....	344
Read Configuration.....	346
Remove Report Configure.....	347
Update Report Configuration.....	349
Component Administration Methods.....	351

Get Server Component.....	351
Get Server Component Names.....	353
Classes.....	355
KeyValue.....	355
NameValue.....	356
IWFWorkflowService.....	358
IWFTTrackingEventPublisher.....	358
RegisteredUser.....	360
WFAccessRights.....	363
WFAgilePart.....	363
WFAgilePartDescriptor.....	364
WFAgileWork.....	371
WFAgileWorkDescriptor.....	372
WFAny.....	383
WFEvent.....	385
WFIntegratedApplication.....	392
WFIntegratedApplicationDescriptor.....	395
WFPartialRollbackInstruction.....	396
WFProcessMergingInstruction.....	397
WFProcessMigrationInstruction.....	398
WFProcessPluggableAdapter.....	401
WFProcessPluggableAdapterDescriptor.....	403
WFProcessSplittingInstruction.....	406
WFQueryExpr.....	407
WFTimeDuration.....	410
WFTimeUnit.....	412

Legal Statements and Policies

This section provides legal statements and general policies for AgilePoint software and documentation.

Disclaimer of Warranty

AgilePoint, Inc. makes no representations or warranties, either express or implied, by or with respect to anything in this document, and shall not be liable for any implied warranties of merchantability or fitness for a particular purpose or for any indirect, special or consequential damages.

Copyright and Trademarks

© 2026 AgilePoint, Inc. All rights reserved.

AgilePoint is a registered trademark of AgilePoint, Inc. AgilePoint's products, including **NX**, are trademarks of AgilePoint, Inc. References to other companies and their products use trademarks owned by the respective companies and are for reference purpose only.

Government Rights Legend

Use, duplication or disclosure by the U.S. Government is subject to restrictions set forth in the applicable license agreement and as provided in DFARS 227.7202-1(a) and 227.7202-3(a) (1995), DFARS 252.227-7013(c)(1)(ii) (Oct 1988), FAR 12.212(a) (1995), FAR 52.227-19, or FAR 52.227-14, as applicable.

Virus-Free Software Policy

AgilePoint recognizes that viruses are a significant security consideration for our customers. AgilePoint takes the following measures to make sure our software is free of viruses upon delivery:

- AgilePoint is built on top of Microsoft .NET framework. The pre-compiled executable is a .NET Common Language Runtime (CLR) application, not a native machine binary. As far as is known at this time, there are no viruses that infect .NET CLR executables.
- The virtual environment for the product packaging process is fully isolated and protected, and anti-virus software is installed and running during packaging.
- The compiled and packaged software files are scanned by virus scanning software before they are released. An official disclosure document regarding the findings from our virus scanning activities is available upon request.

External Links in Documentation

AgilePoint, Inc. user resources, such as Product Documentation and community forums, sometimes include links or URLs that AgilePoint does not control. Third-party web sites are referenced for convenience only. AgilePoint has no control over these external web sites, so we cannot make any representations about their accuracy or guarantee their security, such as the potential presence of malware. You assume any and all risks associated with visiting non-AgilePoint URLs.

Web Services API

This document describes the AgilePoint remote API, which you can access using a web service or a Windows service (WCF). The AgilePoint remote API is designed and implemented using the Microsoft .NET WCF framework.

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Basic Programming Tasks

This section describes the basic programming tasks that are required for the AgilePoint API:

- Authentication
- Adding a namespace and reference
- Catching exceptions
- Making calls

Code Examples

This section provides general code examples for the remote APIs. The code varies slightly depending upon whether AgilePoint Server is installed in IIS or Windows Service mode.

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

AgilePoint Server Running in Windows Service Mode

GetAdminService

```
public IWFAdminService GetAdminService()
{
    string user = this.Context.User.Identity.Name;

    // Set Credentials - Windows Authentication
    System.Net.ICredentials credentials =
        System.Net.CredentialCache.DefaultCredentials;
    // In case of form authentication
    //System.Net.NetworkCredential credentials = new
        System.Net.NetworkCredential(userName, password, domain);

    string locale = "en-us";

    string adminBinding =
        (String)ConfigurationSettings.AppSettings["AdminBindingUsed"];
    IWFAdminService m_adm = new WCFAdminProxy("MyApplicationName",
        "", locale, user, credentials, adminBinding);

    Return m_adm;
}
```

GetWorkflowService

```
public IWFWorkflowService GetWorkflowService ()
{
    string user = this.Context.User.Identity.Name;

    // Set Credentials - Windows Authentication
    System.Net.ICredentials credentials =
        System.Net.CredentialCache.DefaultCredentials;
    // In case of form authentication
    //System.Net.NetworkCredential credentials = new
        System.Net.NetworkCredential(userName, password, domain);

    string locale = "en-us";

    string workFlowBinding =
        (String)ConfigurationSettings.AppSettings["WorkFlowBindingUsed"];
    IWFWorkflowService m_api = new WCFWorkflowProxy("MyApplicationName",
```

```
        "", locale, user, credentials, workFlowBinding);  
  
    return m_api;  
}
```

Run the Windows Service Client Utility Tool

You must run the [Windows Service Client Utility Tool](#) on your application configuration file to configure the binding with the AgilePoint Server.

- ! AgilePoint recommends creating a backup copy of the configuration file for the application you are configuring before using this utility.

Open the Client Utility Tool on Your Machine

On the AgilePoint Server machine, navigate to **(AgilePoint Server installation folder) C:\Program Files\AgilePoint\AgilePoint Server\SVCUtilityTool\AgilePointWindowsServiceClientUtilityTool.exe**

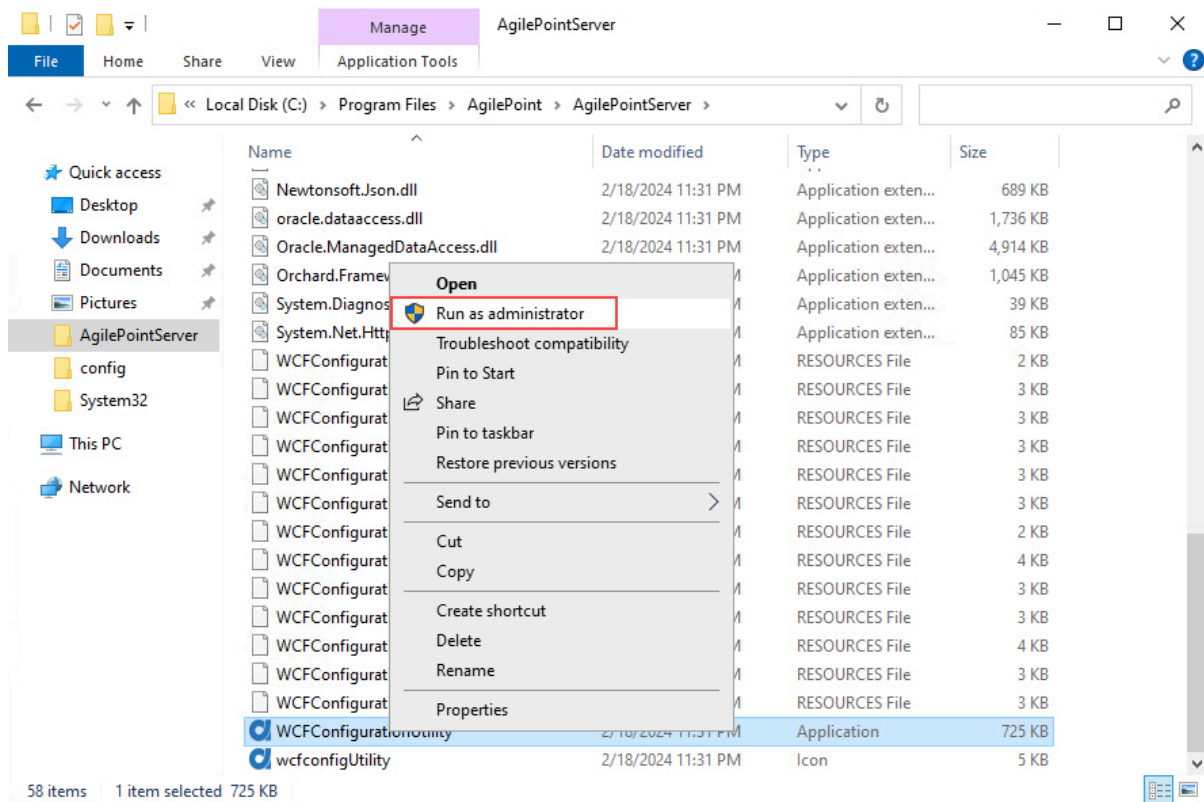
This is only installed on the AgilePoint Server machine.

Open the Client Utility Tool from the AgilePoint Installation Program

Open the AgilePoint installation Setup.exe file, and click **AgilePoint Utilities and Other Support Files > AgilePoint Windows Service Client Utility Tool**.

Open the Client Utility Tool from the AgilePoint Server Manager

1. On the **AgilePoint Server machine**, in Windows Explorer, right-click the file (**AgilePoint Server installation folder**) **C:\Program Files\AgilePoint\AgilePoint Server\WCConfigurationUtility.exe**, and click **Run as Administrator**.



2. Click **SVC Utility** .

More Information

- [Run the Windows Service Client Utility Tool](#)

Authentication

To communicate with the AgilePoint API, you must establish a session, which associates an authenticated user with a set of calls. The client must provide credentials to the AgilePoint Server using Windows authentication.

There are two main ways to provide credentials to the AgilePoint Server: Specify the user's credentials, or use the default credentials.

Specify the Credentials for a User

Specify the user name, password, and domain name for a user. The domain name could be a Windows Domain Name or a Local Host Name. The following example shows the syntax for establishing credentials in this way:

```
System.Net.ICredentials = new
    System.Net.NetworkCredential(userName, password, domainName);
```

Use the Default Credentials

Use the default credentials for a user. The default is the system credentials for the current security context in which the application is running. For a client-side application, these are usually the Windows credentials (user name, password, and domain) of the user running the application. For ASP.NET applications, the default credentials are the user credentials of the authenticated user, or the user being impersonated.

For more information, refer to [Surrogate](#) in the [Documentation Library](#).

```
System.Net.ICredentials =
    System.Net.CredentialCache.DefaultCredentials;
```

Namespace Reference

With an out-of-the-box web service proxy included with AgilePoint, a reference to the namespace must be created. There are two methods to create a namespace reference:

1. Use the AgilePoint pre-compiled web service proxy. This method is recommended because it allows for easier upgrades. If you use AgilePoint proxy, you do not need to change your references to it when you upgrade the AgilePoint server because the new functions are included with the AgilePoint proxy.
 - a. Add the following commands to your assembly file:
 - Ascentn.workflow.shared
 - Ascentn.workflow.WFBase
 - Ascentn.workflow.WFXML
 - Ascentn.AgilePoint.WCFClient

Example:

```
using Ascentn.Workflow.Base;  
using Ascentn.AgilePoint.WCFClient;
```

- b. Add a namespace using `Ascentn.WorkflowBase`
2. Use Microsoft Visual Studio to generate a web service proxy. The disadvantage to this method is that you must regenerate the proxy each time you upgrade the AgilePoint Server.

Catching Exceptions

The exception that a web service throws contains a lot of information, and most of it is not easy to read. AgilePoint Server tags the readable message for end-users.

Call the following function to extract the error message:

```
String error = ShUtil.GetSoapMessage(ex);
```

`ex` is the exception object that contains error message from AgilePoint Server

Making Calls

Calls within the AgilePoint API fall into two categories – synchronous or asynchronous.

Synchronous Call

Synchronous calls are used for short transactions. In a synchronous call, a request is sent to the AgilePoint Server, the server acknowledges the request, and then acts upon it immediately.

Asynchronous Call

Synchronous calls are used for short transactions. In a synchronous call, a request is sent to the AgilePoint Server, the server acknowledges the request, and then acts upon it immediately.

Asynchronous calls are used for longer transactions. In an asynchronous call, a request is sent to the AgilePoint Server, the server acknowledges the request, but it does not immediately act upon it immediately. The server creates a `WFEEvent` object, which contains the call's status, and returns the `WFEEvent` object to the client. The client can call `GetEvent(EventID)` to retrieve the `WFEEvent` object with the updated status. The status can be:

- **Processed** – The transaction was completed successfully.
- **Failed** – The transaction failed, and was not completed.
- **Sent** – The call was received, but it has not been acted upon.

Completing an asynchronous call could take any amount of time, from one second to several days. As a best practice a user interface should handle `GetEvent()` calls to update end users or the application itself regarding the status of asynchronous calls. You might, for example, use Ajax to check status to display on an ASP.NET page in real time.

Methods

This section includes references for all methods within the AgilePoint Web Service API.

Common Methods

This section describes some commonly used methods for the Web Services API.

Check Authentication

API Type

Web Services

Description

This call is used to verify whether the specified user is a registered user on the AgilePoint Server.

Syntax

```
public virtual string CheckAuthenticated()
```

Parameters

Name	Description
None	Not Applicable

Output

If the user is a registered user, the qualified user name in the format of DomainName\UserName is returned. Otherwise, null is returned.

Example

```
//This example is for an ASP.net application.//
public static string Connect(
    System.Web.SessionState.HttpSessionState session,
    System.Net.ICredentials credential, string appName, string locale)
    {
        string url =
            System.Configuration.ConfigurationManager.AppSettings.Get("ServerUrl");

        WorkflowService svc = new WorkflowService(url);
        AdminService adm = new AdminService(url);
        System.Net.CookieContainer cookieContainer = new
            System.Net.CookieContainer();
        svc.Credentials = credential;
        svc.CookieContainer = cookieContainer;
        svc.SetClientAppName(appName);
        svc.SetClientLocale(locale);
        string userName = svc.CheckAuthenticated();
        adm.Credentials = credential;
        adm.CookieContainer = cookieContainer;
        adm.SetClientAppName(appName);
        adm.SetClientLocale(locale);

        //Assume the ASP.net is on session
        WFCommonPage.SetAdm(session, adm);
        WFCommonPage.SetAPI(session, api);

        //return fully qualified Domain username
        return userName;
    }
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Surrogate

API Type

Web Services

Description

IIS does not support users who do not use Windows Active Directory authentication. To address this issue, AgilePoint uses a special type of user called an impersonator. Impersonators enable client applications to authenticate end users who use Active Directory authentication, as well as those who do not.

Impersonators must meet the following requirements:

1. The impersonator must be a Windows Active Directory user to pass IIS authentication.
2. The impersonator must be registered on the AgilePoint Server. The registration can be done through Enterprise Manager.
3. The impersonator must be registered for the application under the Extension of AgilePoint Server Configuration. The application name is case sensitive.
4. The impersonator does not need to be the administrator for AgilePoint Server, or even have workflow execution rights on AgilePoint.

This Surrogate function allows the impersonator to act as a surrogate for the specified user to complete IIS authentication. Once the authentication has passed, the web service API will be called based on the rights granted to the specified user on AgilePoint Server.

This function is called before calling any other AgilePoint Web Service API.

Syntax

```
public virtual void Surrogate(string userName)
```

Parameters

Name	Description
userName	<p><u>Description:</u></p> <p>The user name for the user.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name for a registered AgilePoint user.</p>

Output

None.

Example

```
public IWFWorkflowService GetWorkflowService(
    System.Net.ICredentials credentials, string
    username)
{
    IWFWorkflowService svc = GetAdm(credentials);
    svc.Surrogate(username);
    return svc;
}

//Web service using Impersonator credentials that is registered in the AgilePoint
    Server
//configuration with application name
public IWFWorkflowService GetWorkflowService(System.Net.ICredentials credentials)
{
    string url = ... // AgilePoint Server web service Url
    System.Net.CookieContainer cookieContainer = new
    System.Net.CookieContainer();
    WFWorkflowService svc = new WFWorkflowService(url);
    svc.CookieContainer = cookieContainer;
    svc.Credentials = credentials;
```

```
svc.SetClientAppName([your application name]);  
//Current Locale  
svc.SetClientLocale(Thread.CurrentThread.CurrentUICulture.Name);  
return svc;  
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Surrogate With Application Name and Locale

API Type

Web Services

Description

This function is similar to Surrogate, with the ability to set the application name and locale at the same time. Calling the functions SetClientAppName and SetClientLocale is not needed if this function is called.

Syntax

```
public virtual void Surrogate(string userName, string AppName, string Locale)
```

Parameters

Name	Description
userName	<p><u>Description:</u></p> <p>The user name for the user.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name for a registered AgilePoint user.</p>
AppName	<p><u>Description:</u></p> <p>Specifies the name of the application.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, case-sensitive application name.</p>
Locale	<p><u>Description:</u></p> <p>Specifies the client locale.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid locale in standardized format.</p> <p><u>Example:</u></p> <p>en-US</p>

Output

None.

Example

```
public IWFWorkflowService GetWorkflowService(System.Net.ICredentials
credentials, string userName, string AppName, string Locale)
{
    IWFWorkflowService svc = GetWorkflowService(credentials);
    svc.Surrogate(userName, surrogate AppName, surrogate Locale);
    return svc;
}

public IWFAdminService GetWorkflowService(System.Net.ICredentials credentials)
{
    //The body is the same as Surrogate
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Set Client Application Name

API Type

Web Services

Description

This call is used to set the current application name. The application can be a web/Windows/Windows service application that calls the AgilePoint server for business process actions.

Syntax

```
public virtual void SetClientAppName(string AppName)
```

Parameters

Name	Description
AppName	<p><u>Description:</u></p> <p>Specifies the name of the application.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, case-sensitive application name.</p>

Output

None.

Example

See previous example

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Set Client Locale

API Type

Web Services

Description

This call is used to set the locale for the client application that calls AgilePoint Server. A client application can be a web/Windows/Windows service application.

Syntax

```
public virtual void SetClientLocale(String Locale)
```

Parameters

Name	Description
Locale	<u>Description:</u>

Name	Description
	<p>Specifies the client locale.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid locale in standardized format.</p> <p><u>Example:</u></p> <p>en-US</p>

Output

None.

Example

None.

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

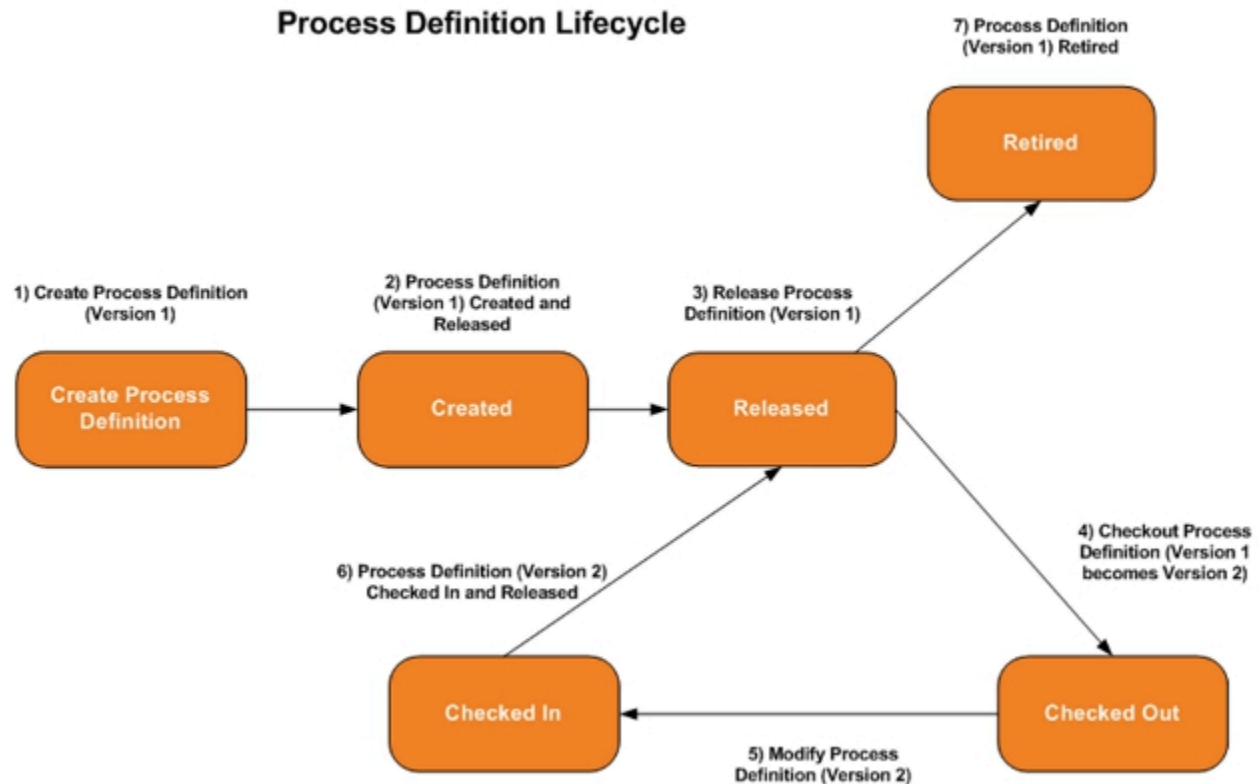
- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Process Definition Methods

This section describes how process definitions are managed. The AgilePoint server supports version control to make sure that process definitions are archived and maintained for later use. A process definition can spawn many process instances. Employing version controls allows the AgilePoint Server to store multiple versions of a single process definition.

The project lifecycle of a process definition can be defined by the following steps:

1. **Create Process Definition** – Create a process definition. The version for the initial process definition is version 1.
2. **Created** – Confirm that the process definition has been created.
3. **Release process definition** – Once a process definition has been created, it must be "released." This allows the process definition to be checked out and edited/modified.
4. **Check out process definition** – If changes or modifications are made, the process definition is "checked out." Changes can then be made to the process definition.
5. **Modify process definition** – After a process definition has been checked out, changes can be made to the process definition.
6. **Check in process definition** – Once all changes/modifications have been made, the process definition is checked in, so the changes are committed to the process definition. The process definition is now version 2.
7. **Retire process definition** – Once version 2 of the process definition has been checked in, the previous version 1 is retired. Retired process definitions can also be deleted.



CheckOut Process Definition

API Type

Web Services

Description

This method is used to manage [process model](#) (process definition) versioning by setting the process model status to [CheckedOut](#) based on a given [process model ID](#).

Good to Know

- Only process models with the [Released status](#) can transition into the [CheckedOut](#) status.
- To call this method, the API account user must have the [access right](#), **Checkin and Checkout a Process Model**.

Syntax

```
public virtual string CheckoutProcDef(string processTemplateID)
```

Parameters

Name	Description
processTemplateID	<p><u>Description:</u></p> <p>The unique identifier for the process definition to be checked out for modification.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process template ID</p>

Output

The process definition, in XML format, that has been checked out.

Example

```
IWFWorkflowService svc = GetWorkflowService();

try
{
    string processTemplateID = ... // process definition to be checked out
    stringprocessDefinitionXML =
    svc.CheckoutProcDef(processTemplateID);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage( ex ) );
}
```

Supported Versions

3.2.0.4 or higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Check In Process Definition

API Type

Web Services

Description

Checks in the [process model](#) (process definition) with an XML deployment file.

Good to Know

- A new version of the process model is created with the status [Created](#).
- To call this method, the API account user must have the [access right](#), **Checkin and Checkout a Process Model**.

Syntax

```
public virtual string CheckinProcDef(string xml)
```

Parameters

Name	Description
xml	<p><u>Description:</u></p> <p>Specifies a process definition in XML format.</p> <p>To generate the process definition file in XML format, in AgilePoint Envision, click File > Export & Import > Save As Deploying File(xml). You can also download the process definition XML from AgilePoint Enterprise Manager.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that contains process definition in XML format.</p>

Output

A new process definition ID.

Example

```

IWFWorkflowService svc = GetWorkflowService();
string xml = ...// Process definition XML as string
WFProcessDefinition pd = new WFProcessDefinition();
GraphicImage g = new GraphicImage();
ProcDefXmlParser parser = new ProcDefXmlParser(new
WFDefaultActivityInstantiator(), pd, g);
parser.Parse(xml);

if (release process definition immediately)
{
    pd.ReleaseDate = DateTime.Now;
    pd.Version = .. // new version
    string procDefID = svc.CheckinProcDef( xml );
    svc.ReleaseProcDef(processDefinitionID);
}

```

```
else if(release process definition at specific date in the future)
{
    pd.ReleaseDate = ...// a specific date in the future
    pd.Version = .. // new version
    string processDefinitionID = svc.CheckinProcDef( xml );
    svc.ReleaseProcDef(processDefinitionID);
}

else // not release process definition
{
    pd.ReleaseDate = Constants.NullDate;
    string processDefinitionID = svc.CheckinProcDef( xml );
}
```

Supported Versions

3.2.0.4 or higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Create Process Definition

API Type

Web Services

Description

Adds a new [process model](#) (process definition) to [AgilePoint Server](#).

Good to Know

- To call this method, the API account user must have the [access right](#), **Add a Process Model**.
- The input parameter is a process model XML that is parsed to create a process model.
- If the process model count exceeds, the [license](#) limit for allowed process models and creation of process model fails.

Syntax

```
public virtual string CreateProcDef(string xml)
```

Parameters

Name	Description
xml	<p><u>Description:</u></p> <p>Specifies a process definition in XML format.</p> <p>To generate the process definition file in XML format, in AgilePoint Envision, click File > Export & Import > Save As Deploying File(xml). You can also download the process definition XML from AgilePoint Enterprise Manager.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that contains process definition in XML format.</p>

Output

Unique ID of the process definition, which the AgilePoint system generates.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string xml = ...// Process definition XML as string

if( set release date )
{
    WFProcessDefinition pd = new WFProcessDefinition();
    GraphicImage g = new GraphicImage();
    ProcDefXmlParser parser = new ProcDefXmlParser(new
    WFDefaultActivityInstantiator(), pd, g);
    parser.Parse(xml);
    pd.ReleaseDate = DateTime.Now; //Set release date
    ProcDefXmlWriter w = new ProcDefXmlWriter(pd, g);
    xml = w.WriteToString();
    string procDefID = svc.CreateProcDef(xml);
}

else // not to release process template
{
    string procDefID = svc.CreateProcDef(xml);
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Delete Process Definition

API Type

Web Services

Description

Permanently deletes the [process model](#) (process definition) and all of the [process instances](#) associated with the process model.

Good to Know

- The process model cannot be deleted if one or more process instances associated with the process model is [running](#) or [Suspended](#).
- The function may take a long time to execute if there are many process instances associated with the process model.
- To call this method, the API account user must have the [access right, Delete and Disable a Process Model](#).

Syntax

```
public virtual void DeleteProcDef(string processTemplateID)
```

Parameters

Name	Description
processTemplateID	<p><u>Description:</u></p> <p>The unique identifier for the process definition to be checked out for modification.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process template ID</p>

Output

None.

Example

```
IWFWorkflowService svc = GetWorkflowService();

string processTemplateID = ..// The unique identifier of the process definition to
    be deleted
svc.DeleteProcDef(processTemplateID);
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Base Process Definition ID

API Type

Web Services

Description

Retrieves the base [process model ID](#) with the specified process model (process definition) name.

Syntax

```
public virtual string GetBaseProcDefID(string procDefName)
```

Parameters

Name	Description
procDefName	<p><u>Description:</u></p> <p>The name of the process definition.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>One line of text (a string).</p> <p>Accepted:</p> <ul style="list-style-type: none">• Letters• Numbers• Spaces

Output

string that contains the base process definition ID.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string procDefName = "CreateRequest";
string baseProcessDefinitionID=
svc.GetBaseProcDefID(procDefName);
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Process Definition By Base Process Definition ID

API Type

Web Services

Description

Retrieves all versions of the [process model](#) (process definition) with the specified base [process model ID](#).

Syntax

```
public virtual WFBBaseProcessDefinition[] GetProcDefByBasePID(string baseprocessTemplateID)
```

Parameters

Name	Description
baseprocessTemplateID	<p><u>Description:</u></p> <p>The ID of the base process definition.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid base process definition ID.</p>

Output

Array of WFBaseProcessDefinition objects.

Example

```
// This is console application
IWfWorkflowService svc = GetWorkflowService();

//Base process definition ID.
string baseprocessTemplateID = ... // for example "1e3d514d43d3465cae6ec3bbbd409168";

try
{
    //Returns Array of WFBaseProcessDefinition for all versions of
    process definition WFBaseProcessDefinition[] processDefinitions =
    svc.GetProcDefByBasePID(baseprocessTemplateID);
    for (int i = 0; i < processDefinitions.Length; i++)
    {
        Console.WriteLine("Definition ID: " + processDefinitions[i].DefID + " ");
        Console.WriteLine("Definition Name: " +
        processDefinitions[i].DefName + " ");
    }
}

catch (Exception ex)
```

```
{  
    Console.WriteLine(ShUtil.GetSoapMessage(ex));  
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Process Definition Graphics

API Type

Web Services

Description

Retrieves graphical data for the [process model](#) (process definition) in XML format.

Good to Know

- The graphical data is used to show the [process](#) visually in the [Process Viewer](#).

Syntax

```
public virtual string GetProcDefGraphics(string processID)
```

Parameters

Name	Description
processID	<p><u>Description:</u></p> <p>The process definition ID for a released process definition.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process definition ID.</p>

Output

Graphics object in XML format.

Example

```
// This is console application sample
IWfWorkflowService svc = GetWorkflowService();
//process definition ID or process instance ID
string processID = ... // for example,
"42544811EC2D4FC18E6BA15CC9FE28DF";

try
{
    //returns an image of a process definition as string.
    string procDefGraphicsXML = svc.GetProcDefGraphics(processID);
    GraphicImage g = new GraphicImage();
    g.FromXml(procDefGraphicsXML);
    byte[] images = g.Image // process image
    NamedRectangle[] shapes = g.Shapes;
}

catch (Exception ex)
{
    Console.WriteLine(ShUtil.GetSoapMessage(ex));
}
```

```

/* This example produces the following results:
process definition Graphics:
<?xml version="1.0" encoding="utf-8"?><Graphics
left="3.333333333333333" right="5.76002857553708"
top="10.3848753378378" bottom="7"><Shapes><Shape
left="3.333333333333333" right="3.833333333333333"
top="10.38541666666667" bottom="9.88541666666667" name="Start "
/><Shape left="5.260416666666667"
right="5.760416666666667" top="7.5" bottom="7" name="Stop"
/><Shape left="4.302083333333333"
right="5.302083333333333" top="9.166666666666667"
bottom="8.666666666666667" name="Text File Writer.3"
/></Shapes><Image>@64R0lGODlh6QBFAXAAACwAAAAA6QB
FAYcAAAAMDawkJCQsLCydaAClAACqAACuAACyAAC2AAC6AAC+\nAADCAADGAADK
AAA8PDzOAADSAA
.....\nTEGgS0+D2pKSHjUEWOzmRMPZljdStSBbQshYsxjRfu7xD
REAAv3vGgai7gAEaA3s3j3bghhl2dbEN\nDAM5bsCFff6yI1uYAAZ8WtowAasI
wudCT76YxwltYQa9/Wlww8ADCmXp+/SHxwvUm4UYBGq02d2S\nWJwAf+brX/9H0ic
BCEbggcPuAAbuvo+YgBV8acwAwNO3PjK+j4EKZzhNwp1PZ4dvfBUAoMEjgHG
b\nBGNA0rYGalw+cpazvMA32cOlMo6cgAAAow==</Image></Graphics>
*/

```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Process Definition Name and Version

API Type

Web Services

Description

Retrieves the name and version of the [process model](#) (process definition) with the specified [process model ID](#).

Syntax

```
public virtual KeyValue GetProcDefNameVersion(string processTemplateID)
```

Parameters

Name	Description
processTemplateID	<p><u>Description:</u></p> <p>The process definition ID for a released process definition.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process definition ID.</p>

Output

[KeyValue](#) object, where Key contains process definition name and Value contains version.

Example

```
// This is console application sample
IWFWorkflowService svc = GetWorkflowService();
```

```
//process definition ID for a process.
string processTemplateID = ... // for example "1e3d514d43d3465cae6ec3bbbd409168";

try
{
    //Returns KeyValuepair, for example "process definition
    Name-process definition Version"
    KeyValue keyValue = _ svc.GetProcDefNameVersion(processTemplateID);
    Console.WriteLine("process definition Name: '" +
    keyValue.Key.ToString() + "' ");
    Console.WriteLine("process definition Version: '" +
    keyValue.Value.ToString() + "' ");
}

catch (Exception ex)
{
    Console.WriteLine( ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Process Definitions

API Type

Web Services

Description

Retrieves all versions of all [process models](#) (process definitions).

Syntax

```
public virtual WFBaseProcessDefinition[] GetProcDefs()
```

Parameters

Name	Description
None	Not Applicable

Output

An array of WFBaseProcessDefinition objects.

Example

```
IWFWorkflowService svc = GetWorkflowService();

try
{
    //Returns Array of WFBaseProcessDefinition type.
    WFBaseProcessDefinition[] processDefinitions = svc.GetProcDefs();
    for (int i = 0; i < processDefinitions.Length; i++)
    {
        Console.WriteLine("Definition ID: '" +
            processDefinitions[i].DefID + "' ");
        Console.WriteLine("Definition Name: '" +
            processDefinitions[i].DefName + "' ");
    }
}
```

```
catch (Exception ex)
{
    Console.WriteLine(ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Process Definition XML

API Type

Web Services

Description

Retrieves a [process model](#) in XML format for the specified [process model ID](#).

Syntax

```
public virtual string GetProcDefXml(string processTemplateID)
```

Parameters

Name	Description
processTemplateID	<p><u>Description:</u></p> <p>The process definition ID for a released process definition.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process definition ID.</p>

Output

string that contains XML format of the process definition.

Example

```
// This is console application sample
IWfWorkflowService svc = GetWorkflowService();
//process definition ID for a process.
string processTemplateID = ..// for example "42544811EC2D4FC18E6BA15CC9FE28DF";

try
{
    //Returns process definition in XML format.
    string procDefXML = svc.GetProcDefXml(processTemplateID);
    Console.WriteLine("process definition XML: \n");
    Console.WriteLine(procDefXML);
}

catch (Exception ex)
{
    Console.WriteLine( ShUtil.GetSoapMessage(ex));
}

/*
This example produces the following results:
```

```

process definition XML:
<?xml version="1.0" encoding="utf-8"
standalone="no"?>\n<?wfmc-xpdl xmlns="http://www.wfmc.org/2002/XPDL1.0"
xmlns:xpdl="http://www.wfmc.org/2002/XPDL1.0"
xsi:schemaLocation="http://www.wfmc.org/2002/XPDL1.0"?>\n<!--Process
Definition, Copyright 2003-2004 AgilePoint Inc., All
Rights Reserved.-->\n<ProcessDefinition
defName="TextFileWriterProcess" preVersion="" version="1.0"
description="" owner="Bipin.Shah" docRef=""
.....\nTEGgS0+D2pKSHjUEWOzmRMPZljdstSBbQshYsxjRfu7
xDREAaV3vGgai7gAEaA3sj3bghh12dbEN\nDAM5bsCFff6
yIluYAAZ8WtowAasIwudCT76YxwltYQa9/W1ww8ADCmXP+/SHxwvUm4UYBGq02d2S\nWJwAf+brX/9H
0icBCEbggcpuAAbuvO+YgBV8acwAwNO3PjK+j4EKZzhNwp
1PZ4dvfBUAoMEjgHGb\nBGNA0rYGalW+cpazvMA32c0lMo6cgAAAOW==</Image>\n
</Graphics>\n</ProcessDefinition>
*/

```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Released Process Definition ID

API Type

Web Services

Description

Gets the [process model IDs](#) of all released version of the [process model](#) (process definition) with the specified process model name.

Syntax

```
public virtual string GetReleasedPID(string procDefName)
```

Parameters

Name	Description
procDefName	<p><u>Description:</u></p> <p>The name of the process definition.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>One line of text (a string).</p> <p><u>Accepted:</u></p> <ul style="list-style-type: none">• Letters• Numbers• Spaces

Output

The ID for the released process definition.

Example

```
//GetReleasedPID
IWfWorkflowService svc= GetWorkflowService();
string procDefName = "Budget Request Approval Process";
string processDefinitionID =
```

```
svc.GetReleasedPID(procDefName);  
Console.WriteLine("Process definition ID=" + processDefinitionID);
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Released Process Definitions

API Type

Web Services

Description

Retrieves the names and IDs of all released [process models](#) (process definitions).

Syntax

```
public virtual KeyValueType[] GetReleasedProcDefs()
```

Parameters

Name	Description
None	Not Applicable

Output

[KeyValue](#) array for pairs of process definition IDs and process definition names.

Example

```
KeyValue[] defs = svc.GetReleasedProcDefs();
for (int i = 0; i < defs.Length; i++)
{
    Console.WriteLine("Key=" + defs[i].Key.ToString() + "*****"+ "Value=" +
        defs[i].Value.ToString());
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Release Process Definition

API Type

Web Services

Description

Releases the [process model](#) (process definition) with the specified [process model ID](#) from [AgilePoint Server](#).

Good to Know

- The [status](#) of the process model changes from [Created](#) to [Released](#).
- To call this method, the API account user must have the [access right](#), **Release a Process Model**.

Syntax

```
public virtual void ReleaseProcDef(string processTemplateID)
```

Parameters

Name	Description
processTemplateID	<p><u>Description:</u></p> <p>The unique identifier for the process definition to be checked out for modification.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process template ID</p>

Output

None.

Example

```
//Sample for using Workflow.ReleaseProcDef using Console Application

try
{
    IWFWorkflowService svc = GetWorkflowService();
    string processTemplateID = ...
    svc.ReleaseProcDef(processTemplateID);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage( ex ) );
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Uncheckout Process Definition

API Type

Web Services

Description

Undoes a check-out for a [process model](#) (process definition).

Good to Know

- This method returns the [status](#) of a process model from [CheckedOut](#) to [Released](#) without making changes to the process model, or changing the version number.
- To call this method, the API account user must have the [access right, Checkin and Checkout a Process Model](#).

Syntax

```
public virtual void UnCheckOutProcDef(string processTemplateID)
```

Parameters

Name	Description
processTemplateID	<p><u>Description:</u></p> <p>The unique identifier for the process definition to be checked out for modification.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process template ID</p>

Output

None.

Example

```
try  
{
```

```
IWFWorkflowService svc = GetWorkflowService();
string processTemplateID = ...
string processDefinitionXML =
svc.UncheckoutProcDef(processTemplateID);
}

catch (Exception ex)
{
    Console.WriteLine("Message:\n" + ShUtil.GetSoapMessage( ex ) );
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Update Process Definition

API Type

Web Services

Description

Updates the [process model](#) (process definition) with the new version of XML based process model file.

Good to Know

- This method is intended for minor changes only, such as typographical errors.
- **Warning:** Changes made with this method circumvent version control, meaning changes are not tracked, and versions can not be managed. Do not use this call for making any major changes to the process model.

Syntax

```
public virtual string UpdateProcDef(string xml)
```

Parameters

Name	Description
xml	<p><u>Description:</u></p> <p>Specifies a process definition in XML format.</p> <p>To generate the process definition file in XML format, in AgilePoint Envision, click File > Export & Import > Save As Deploying File(xml). You can also download the process definition XML from AgilePoint Enterprise Manager.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that contains process definition in XML format.</p>

Output

Returns the unique identifier for the process definition that is updated.

Example

```
// This is console application sample
IWFWorkflowService svc = GetWorkflowService();
string processDefinitionXML = ..// see previous description of how to
get process definition XML
```

```
try
{
    //Update Process definition using updated process xml string
    string processDefinitionID = svc.UpdateProcDef(xml);
}

catch (Exception ex)
{
    Console.WriteLine("Exception:" + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

4.0.1 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Methods for Process Instances

This section describes service calls related to process instances.

Cancel Process Instance

API Type

Web Services

Description

Cancels the [process instance](#) based on a specified [Create Process Instance](#).

Good to Know

- This method cancels all [system activities](#), human [tasks](#), and child process instances.
- To call this method, the API account user must have the [access right](#), **Cancel a Process**.

Syntax

```
public virtual WFEvent CancelProclnst(string processInstanceID)
```

Parameters

Name	Description
processInstanceID	<p><u>Description:</u></p> <p>Specifies the unique ID of a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process instance ID</p>

Output

[WFEvent](#) object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string processInstanceID = ..// the ID of the process instance to be
canceled.

try
{
    WFEvent evt = svc.CancelProcInst(processInstanceID);
}

catch( Exception ex)
{
    base.ShowMessage( base.GetSoapMessage(ex) );
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Create Process Instance

API Type

Web Services

Description

Creates a [process instance](#) for a specified [process instance ID](#) and parameters.

Good to Know

- To call this method, the [process initiator](#) must have the access control, **Initiate a Process**.
- You can not create a process instance for a [process model](#) with the Initiate constraint as child process.
- The [process instance ID](#) and process instance name parameters must be unique.

Syntax

```
public virtual WFEvent CreateProclnst(string ProcessID, string ProcessInstID, string ProclnstName, string WorkObjID, string SuperProclnstID, bool blnStartImmediately)
```

Parameters

Name	Description
ProcessID	<p><u>Description:</u> The process definition ID for a released process definition.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid process definition ID.</p>
ProcessInstID	<p><u>Description:</u> A process instance ID for the process instance you are creating.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A unique, 32 character process instance ID.</p>

Name	Description
	If you set this value to null, the AgilePoint Server generates the ID.
ProclInstName	<p><u>Description:</u></p> <p>A unique process name that is associated with the process definition.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A unique process instance name up to 1024 characters.</p>
WorkObjID	<p><u>Description:</u></p> <p>An ID for an object, such as a document, that is associated with the process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, unique 256-character ID.</p> <p>Even though the field size is 256 characters, in common practice, this will usually return a 32-character GUID.</p>
SuperProclInstID	<p><u>Description:</u></p> <p>A process instance ID that acts as a parent process instance of the process instance that is intended to create. In other words, this is the ID of the process instance on which you want to base your new process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, unique 32-character process instance ID.</p>

Name	Description
blnStartImmediately	<p><u>Description:</u></p> <p>An obsolete, legacy parameter that must be true.</p> <p><u>Type</u></p> <p>bool</p> <p><u>Allowed Values:</u></p> <p>True</p>

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```

IWFWorkflowService svc = GetWorkflowService();
string processDefinitionName = "EmployeeOnboardProcess";

// get UUID of released process definition
string ProcessID = svc.GetReleasedPID(processDefinitionName);

// assign UUID of process instance
string ProcessInstID = UUID.GetID();

// process instance name that has to be unique within process definition ID
string ProcInstName = string.Format("{0}-{1}",
processDefinitionName DateTime.Now.Ticks );

// work object ID
string WorkObjID = UUID.GetID();

```

```
// create process instance
WFEvent event = svc.CreateProcInst(ProcessID, ProcessInstID, ProcInstName,
    WorkObjID, null, true);
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Create Process Instance (Extended with Initiator and Work Object Info)

API Type

Web Services

Description

Creates a process instance with a specified workObjtInfo value. The workObjInfo parameter provides additional information about a work object, such as a URL for a document.

Syntax

```
public virtual WFEvent CreateProcInstEx(string ProcessID, string ProcessInstID, string ProcInstName,
    string WorkObjID, string WorkObjInfo, string SuperProcInstID, string Initiator, string CustomID,
    NameValue[] Attributes, bool blnStartImmediately)
```

Parameters

Name	Description
ProcessID	<p><u>Description:</u></p> <p>The process definition ID or process template name for a released process definition.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process definition ID or process template name.</p>
ProcessInstID	<p><u>Description:</u></p> <p>A process instance ID for the process instance you are creating.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A unique, 32 character process instance ID.</p> <p>If you set this value to null, the AgilePoint Server generates the ID.</p>
ProInstName	<p><u>Description:</u></p> <p>A unique process name that is associated with the process definition.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A unique process instance name up to 1024 characters.</p>
WorkObjID	<p><u>Description:</u></p>

Name	Description
	<p>An ID for an object, such as a document, that is associated with the process instance.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid, unique 256-character ID. Even though the field size is 256 characters, in common practice, this will usually return a 32-character GUID.</p>
WorkObjInfo	<p><u>Description:</u> Usually this parameter is used to hold supplemental information about the work object, such as a URL for a document, within the process instance.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A string up to 1024 characters.</p>
SuperProclnstID	<p><u>Description:</u> A process instance ID that acts as a parent process instance of the process instance that is intended to create. In other words, this is the ID of the process instance on which you want to base your new process instance.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid, unique 32-character process instance ID.</p>
Initiator	<p><u>Description:</u> Specifies the user who initiates a process.</p>

Name	Description
	<p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name.</p>
CustomID	<p><u>Description:</u></p> <p>A work object ID specified within a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>One valid work object ID.</p>
Attributes	<p><u>Description:</u></p> <p>Name-value pairs associated with a custom ID.</p> <p><u>Type</u></p> <p>NameValue</p> <p><u>Allowed Values:</u></p> <p>A valid custom ID with an associated name.</p>
blnStartImmediately	<p><u>Description:</u></p> <p>An obsolete, legacy parameter that must be true.</p> <p><u>Type</u></p> <p>bool</p> <p><u>Allowed Values:</u></p> <p>True</p>

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
... see previous sample
string workObjectInfo = ..// for example, a XML-serialized of an object
WFEvent evt = svc.CreateProcInstEx(
    ProcessID,
    ProcessInstID,
    ProcInstName,
    WorkObjID,
    WorkObjInfo,
    parentProcessInstID,
    Initiator,
    WorkObjID,
    ds.ToArray(),
    true);
```

Supported Versions

4.0.1 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Create Process Instance (Extended with Initiator)

API Type

Web Services

Description

Creates a process instance in which the user name for the user who initiates the process is specified.

Syntax

```
public virtual WFEvent CreateProclnstEx(string ProcessID, string ProcessInstID, string ProclnstName,
string WorkObjID, string SuperProclnstID, string Initiator, string CustomID, NameValue[] Attributes,
bool blnstartImmediately)
```

Parameters

Name	Description
ProcessID	<p><u>Description:</u></p> <p>The process definition ID or process template name for a released process definition.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p>

Name	Description
	A valid process definition ID or process template name.
ProcessInstID	<p><u>Description:</u></p> <p>A process instance ID for the process instance you are creating.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A unique, 32 character process instance ID.</p> <p>If you set this value to null, the AgilePoint Server generates the ID.</p>
ProclInstName	<p><u>Description:</u></p> <p>A unique process name that is associated with the process definition.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A unique process instance name up to 1024 characters.</p>
WorkObjID	<p><u>Description:</u></p> <p>An ID for an object, such as a document, that is associated with the process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, unique 256-character ID.</p> <p>Even though the field size is 256 characters, in common practice, this will usually return a 32-character GUID.</p>
SuperProclInstID	<p><u>Description:</u></p>

Name	Description
	<p>A process instance ID that acts as a parent process instance of the process instance that is intended to create. In other words, this is the ID of the process instance on which you want to base your new process instance.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid, unique 32-character process instance ID.</p>
Initiator	<p><u>Description:</u> Specifies the user who initiates a process.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid user name.</p>
CustomID	<p><u>Description:</u> A work object ID specified within a process instance.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> One valid work object ID.</p>
Attributes	<p><u>Description:</u> Name-value pairs associated with a custom ID.</p> <p><u>Type</u> NameValue</p>

Name	Description
	<u>Allowed Values:</u> A valid custom ID with an associated name.
blnStartImmediately	<u>Description:</u> An obsolete, legacy parameter that must be true. <u>Type</u> bool <u>Allowed Values:</u> True

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```

... see previous sample
string Initiator = ..// System.Environment.UserName
WFEvent evt = svc.CreateProcInstEx(
    ProcessID,
    ProcessInstID,
    ProcInstName,
    WorkObjID,
    parentProcessInstID,
    Initiator,
    WorkObjID,
    ds.ToArray(),
    true);

```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Create Process Instance (Extended Method)

API Type

Web Services

Description

Creates a process instance that can have additional input arguments added to the function.

Syntax

```
public virtual WFEvent CreateProclnstEx(string ProcessID, string ProcessInstID, string ProclnstName, string WorkObjID, string SuperProclnstID, string CustomID, NameValue[] Attributes, bool blstartImmediately)
```

Parameters

Name	Description
ProcessID	<p><u>Description:</u></p> <p>The process definition ID or process template name for a released process definition.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process definition ID or process template name.</p>
ProcessInstID	<p><u>Description:</u></p> <p>A process instance ID for the process instance you are creating.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A unique, 32 character process instance ID.</p> <p>If you set this value to null, the AgilePoint Server generates the ID.</p>
ProInstName	<p><u>Description:</u></p> <p>A unique process name that is associated with the process definition.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A unique process instance name up to 1024 characters.</p>
WorkObjID	<p><u>Description:</u></p>

Name	Description
	<p>An ID for an object, such as a document, that is associated with the process instance.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid, unique 256-character ID. Even though the field size is 256 characters, in common practice, this will usually return a 32-character GUID.</p>
SuperProclnstID	<p><u>Description:</u> A process instance ID that acts as a parent process instance of the process instance that is intended to create. In other words, this is the ID of the process instance on which you want to base your new process instance.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid, unique 32-character process instance ID.</p>
CustomID	<p><u>Description:</u> A work object ID specified within a process instance.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> One valid work object ID.</p>
Attributes	<p><u>Description:</u> Name-value pairs associated with a custom ID.</p> <p><u>Type</u></p>

Name	Description
	<p>NameValue</p> <p><u>Allowed Values:</u></p> <p>A valid custom ID with an associated name.</p>
blnStartImmediately	<p><u>Description:</u></p> <p>An obsolete, legacy parameter that must be true.</p> <p><u>Type</u></p> <p>bool</p> <p><u>Allowed Values:</u></p> <p>True</p>

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string processDefinitionName = "EmployeeOnboardProcess";

// get UUID of released process definition
string ProcessID =
svc.GetReleasedPID(processDefinitionName);

// assign UUID of process instance
string ProcessInstID = UUID.GetID();
```

```
// process instance name that has to be unique within process definition ID
string ProcInstName = string.Format("{0}-{1}",
processDefinitionName DateTime.Now.Ticks );

// work object ID
string WorkObjID = UUID.GetID();
//parent process instance ID is required if this is to create a sub
process. If not, just provide null
string SuperProcInstID = .. // for example,
"09315f0ae769429bbfb243f888bcb09f" or null
List<NameValue> ds = new List<NameValue>();
ds.Add(new NameValue("CustomAttrKey1", "CustomAttrValue1"));
ds.Add(new NameValue("CustomAttrKey2", true));
ds.Add(new NameValue("CustomAttrKey3", 12345));
WFEvent evt = svc.CreateProcInstEx(
    ProcessID,
    ProcessInstID,
    ProcInstName,
    WorkObjID,
    SuperProcInstID,
    WorkObjID,
    ds.ToArray(),
    true);
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Delete Process Instance

API Type

Web Services

Description

Permanently deletes a [process instance](#).

Good to Know

- This method removes the specified process instance and all the associated data from the database, such as [work items](#), e-mails, and [activity instances](#) associated with this process instance.
- This method can only delete the process instance with the [status Completed](#) or [Cancelled](#).
- It may take some time to complete this transaction.
- To call this method, the API account user must have the [access right, Cancel a Process](#).

Syntax

```
public void DeleteProclnst(string processInstanceID)
```

Parameters

Name	Description
processInstanceID	<p><u>Description:</u></p> <p>Specifies the unique ID of a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process instance ID</p>

Output

None.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string processInstanceID = ..// the ID of the process instance to be
canceled.

try
{
    WFEvent evt = svc.DeleteProcInst(processInstanceID);
}

catch( Exception ex)
{
    base.ShowMessage( base.GetSoapMessage(ex) );
}
```

Supported Versions

4.0.1 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Events By Process Instance ID

API Type

Web Services

Description

Retrieves all the events that have occurred for a specified process instance.

Syntax

```
public virtual WFEvent[] GetEventsByProclnstID(string processInstanceID)
```

Parameters

Name	Description
processInstanceID	<p><u>Description:</u></p> <p>Specifies the unique ID of a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process instance ID</p>

Output

An array of [WFEvent](#) objects.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string processInstanceID = ... // process instance ID

try
```

```
{
    WFEvent[] events = svc.GetEventsByProcInstID(processInstanceID);
    for (int i = 0; i < events.Length; i++)
    {
        Console.WriteLine("Event ID: '" + events[i].EventID + "' ");
        Console.WriteLine("Event Name: '" + events[i].EventName + "'");
    }
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Process Instance

API Type

Web Services

Description

Retrieves information about a specified [process instance](#).

Syntax

```
public virtual WFBBaseProcessInstance GetProcInst(string processInstanceID)
```

Parameters

Name	Description
processInstanceID	<p><u>Description:</u></p> <p>Specifies the unique ID of a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process instance ID</p>

Output

WFBBaseProcessInstance object that contains basic information about a process instance. It returns null if the [process instance ID](#) does not exist.

Example

```
// This is sample code for console application
IWFWorkflowService svc = GetWorkflowService();
string processInstanceID = ...// process instance ID

try
{
    //Returns an instance of WFBBaseProcessInstance type.
    WFBBaseProcessInstance processInstance =
    svc.GetProcInst(processInstanceID);
}
```

```
catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Process Instance Attribute

API Type

Web Services

Description

Gets the name-value pair of the specified [process attribute](#) (custom attribute) for the specified [process instance ID](#).

Syntax

```
public virtual KeyValue GetProclnAttr(string processInstanceId, string attributeName)
```

Parameters

Name	Description
processInstanceID	<p><u>Description:</u></p> <p>Specifies the unique ID of a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process instance ID</p>
attributeName	<p><u>Description:</u></p> <p>The name of the process instance attribute you want.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • DefID - The ID of the process definition. • DefName - The name of the process definition. • ProclnstName - The name of the process instance. • Status - The current status of the process instance. • SuperProclnstID - The parent process instance ID. • workObjectID - The ID of the work object. • StartDate - The date and time when the process instance was started. • DueDate - The date that the process instance is expected to be complete. • LastModifiedDate - The date and time that the last modification was made to the process instance.

Output

Returns the [KeyValue](#) for the attribute associated with the process instance as a name-value" pair.

Example

```
//This is console application sample
IWFWorkflowService svc = GetWorkflowService();
string processInstanceID = ...// for example,
"1e3d514d43d3465cae6ec3bbbd409168";
string attributeName = "DefName";

try
{
    //Returns attribute associated with the Process Instance as
    "Name-Value" pair.
    KeyValue processInstanceAttribute =
    svc.GetProcInstAttr(processInstanceID, attributeName);
    Console.WriteLine("{0}={1}", processInstanceAttribute.Name,+
    processInstanceAttribute.Value);
}

catch (Exception ex)
{
    Console.WriteLine(ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Process Instance Attributes

API Type

Web Services

Description

Gets the name-value pairs of [process attributes](#) (custom attributes) with the specified [process instance ID](#).

Syntax

```
public virtual NameValue[] GetProInstAttrs(String processInstanceID)
```

Parameters

Name	Description
processInstanceID	<p><u>Description:</u></p> <p>A process instance ID for the process instance you are creating.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A unique, 32 character process instance ID.</p> <p>If you set this value to null, the AgilePoint Server generates the ID.</p>

Attributes

Name	Description
DefID	The ID of the process definition.
DefName	The name of the process definition.
ProcInstName	The name of the process instance.
Status	The current status of the process instance.
SuperProcInstID	The parent process instance ID .
workObjectID	The ID of the work object.
StartDate	The date and time when the process instance was started.
DueDate	The date that the process instance is expected to be complete
LastModifiedDate	The date and time that the last modification was made to the process instance.

Output

Array of [NameValue](#) objects that holds the values of all the requested attributes.

Example

```
// This is console application sample
IWFWorkflowService svc = GetWorkflowService();
string processInstanceID = ...// process instance ID
NameValue[] attributes = svc.GetProcInstAttrs(processInstanceID);
for (int i = 0; i < attributes.Length; i++)
{
    NameValue nv = attributes[i];
    Console.WriteLine("Process Instance Attribute, {0}={1}", nv.Name, nv.Value);
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Merge Process Instances

API Type

Web Services

Description

Merges 2 or more [process instances](#) into one process instance.

Good to Know

- These process instances must be based on the same [process model](#).
- If more than one process instance is intended to perform the same operation, instead of having more than one process instance, you can merge all the process instances into a single process instance.
- To call this method, the API account user must have the [access right](#), **Cancel a Process**.
- This method is provided for reference for very advanced users – mainly internal AgilePoint teams. If you need help with this method, [contact AgilePoint Professional Services](#).

Syntax

```
public virtual string MergeProInsts(WFProcessMergingInstruction instruction)
```

Parameters

Name	Description
instruction	<p><u>Description:</u></p> <p>Specifies the instructions for merging two process instances.</p> <p><u>Type</u></p> <p>WFProcessMergingInstruction</p> <p><u>Allowed Values:</u></p> <p>A WFProcessMergingInstruction object.</p>

Output

The [process instance ID](#) of the merged process instance.

Example

```
string MergeProcessInstances(IWFWorkflowService svc, string[]
processInstanceIDs)
{
    IWFWorkflowService svc = GetWorkflowService();

    // suspends all of process instances to be merged.
    foreach (string id in processInstanceIDs)
    {
        svc.SuspendProcInst(id);
    }

    // query process instances
    string inExpr = ShUtil.Merge(processInstanceIDs, true);
    WFQueryExpr queryExpr = new WFQueryExpr("PROC_INST_ID", SQLExpr.IN,
WFAny.Create(inExpr), true);
    WFBaseProcessInstance[] pis = svc.QueryProcInsts(queryExpr);

    // merge custom attributes
    NameValue[] mergedCustomAttributes = GetMergeCustomAttributes(api, pis);
    string procInstID = UUID.GetID();
```

```
string procInstName = pis[0].DefName + "_" + DateTime.Now.ToString()
+ " - Merged";
string workObjectID = procInstID + " - Merged";

WFProcessMergingInstruction instruction = new
WFProcessMergingInstruction();
instruction.MergingProcessInstanceIDs = processInstanceIDs;
instruction.MergedProcessInstance = new
WFProcessMergingInstruction.MergedProcessParameter(
    procInstID,
    procInstName,
    workObjectID,
    null,
    mergedCustomAttributes);
instruction.Validate();
return svc.MergeProcInsts(instruction);
}

// sample code for merging custom attributes
private NameValue[] GetMergeCustomAttributes(IWFWorkflowService svc,
WFBaseProcessInstance[] pis)
{
    List<string> workObjectIDs = new List<string>();
    foreach (WFBaseProcessInstance pi in pis)
    {
        workObjectIDs.Add(pi.WorkObjectID);
    }
    KeyValue[] items = svc.GetCustomAttrsEx(workObjectIDs.ToArray());
    Dictionary<string, System.Xml.XmlDocument> dss =
    new Dictionary<string, System.Xml.XmlDocument>();
    foreach (KeyValue item in items)
    {
        WFCustomAttributes ds = new WFCustomAttributes();
        ds.AttrXml = item.Value;
        System.Xml.XmlDocument xmlDoc = new System.Xml.XmlDocument();
        xmlDoc.LoadXml(ds["//"] as string);
        dss[item.Key] = xmlDoc;
    }
    // master processInstanceID and workObjectID
    string masterProcessInstanceID = pis[0].ProcInstID;
    string masterWorkObjectID = pis[0].WorkObjectID;
    System.Xml.XmlDocument masterXmlDoc = dss[masterWorkObjectID];
    System.Xml.XmlNamespaceManager nsm =
    ShUtil.GetNamespaces(masterXmlDoc);
    System.Xml.XmlNode titleNode =
```

```
masterXmlDoc.SelectSingleNode("/pd:issueTracking/pd:issueTitle", nsm);
titleNode.InnerText = "Title - Merged";
System.Xml.XmlNode descriptionNode =

masterXmlDoc.SelectSingleNode("/pd:issueTracking/pd:description", nsm);
descriptionNode.InnerText = "Description - Merged";
System.Xml.XmlNode parent =
masterXmlDoc.SelectSingleNode("/pd:issueTracking/pd:Persons", nsm);

// merge rest of children
foreach (string key in dss.Keys)
{
    if (key == masterWorkObjectID) continue;
    System.Xml.XmlDocument secondaryXmlDoc = dss[key];
    System.Xml.XmlNamespaceManager nsmgr =
    ShUtil.GetNamespaces(secondaryXmlDoc);
    System.Xml.XmlNode node =
    secondaryXmlDoc.SelectSingleNode("/pd:issueTracking/pd:Persons", nsmgr);
    parent.InnerXml += node.InnerXml;
}
return NameValue.Array("//", masterXmlDoc.OuterXml);
}

#endregion
```

Supported Versions

4.5 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Migrate Process Instances

API Type

Web Services

Description

Migrates [process instances](#) from one version to another version (called [process migration](#)).

Good to Know

- It is required to provide Migration instruction with details of a list of [activities](#).
- It is required to suspend a process instance before migrating.
- To call this method, the API account user must have the [access right](#), **Cancel a Process**.
- Once the migration is successful, it is required to call the method **Suspend and Resume a Process**, so that the process instance resumes with the newer version.
- This method is provided for reference for very advanced users – mainly internal AgilePoint teams. If you need help with this method, [contact AgilePoint Professional Services](#).

Syntax

```
public virtual void MigrateProInst(WFProcessMigrationInstruction instruction, string processInstanceID, string reserved)
```

Parameters

Name	Description
instruction	<p><u>Description:</u></p> <p>Specifies the instructions for migrating a process to a new version.</p> <p><u>Type</u></p> <p>WFProcessMigrationInstruction</p> <p><u>Allowed Values:</u></p> <p>A WFProcessMigrationInstruction object.</p>

Name	Description
processInstanceID	<p><u>Description:</u></p> <p>Specifies the unique ID of a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process instance ID</p>
reserved	<p><u>Description:</u></p> <p>Reserved for future use.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>The literal string null.</p> <p>You must pass this as a literal string at this time. This field will be used for other purposes in a later release.</p>

Output

```
IWFWorkflowService svc = GetWorkflowService();
WFProcessMigrationInstruction pmi = new
WFProcessMigrationInstruction();

// some code...
svc.MigrateProcInst(pmi, currentProcessInstanceID, null);

// some more code...
```

Example

```
IWFWorkflowService svc = GetWorkflow();
string processInstanceID = "DB50CFEFDE464A78AAAA9BD7D6E6D9D0";
WFProcessMigrationInstruction pmi = new
WFProcessMigrationInstruction();

//add the correct CurrentActivityUniqueName
string currentActivityUniqueName = "BudgetRequest";

//add the correct TargetActivityUniqueName
string targetActivityUniqueName = "BudgetRequestNew";
bool bCurrentActivated = false;
pmi.AddMatchingActivity(currentActivityUniqueName, targetActivityUniqueName,
bCurrentActivated);
svc.MigrateProcInst(pmi, currentProcessInstanceID, null);
```

Supported Versions

4.0.1 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Promote Process Instance

API Type

Web Services

Description

Promotes a process instance. This method is obsolete.

Syntax

```
public virtual WFEvent PromoteProclnst(String processInstanceId)
```

Parameters

Name	Description
processInstanceId	<p><u>Description:</u></p> <p>A process instance ID for the process instance you are creating.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A unique, 32 character process instance ID.</p> <p>If you set this value to null, the AgilePoint Server generates the ID.</p>

Output

[WFEvent](#) object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.

- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
None .
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Query Process Instances

API Type

Web Services

Description

Retrieves a list of [process instances](#) that match a specified query expression. The WFQueryExpr string is used to generate a query expression, and the client application specifies the query terms.

Syntax

```
public virtual WFBaseProcessInstance[] QueryProcInsts(WFQueryExpr expr)
```

Parameters

Name	Description
expr	<p><u>Description:</u></p> <p>Specifies the where clause of a SQL query expression.</p> <p><u>Type</u></p> <p>WFQueryExpr</p> <p><u>Allowed Values:</u></p> <p>A valid WFQueryExpr object.</p>

Output

An array of WFBaseProcessInstance objects. It returns null if nothing matches to the query expression.

Example

```
IWFWorkflowService svc = GetWorkflowService();

// query all running process instance
string status = WFProcessInstance.RUNNING;
WFAny any = WFAny.Create(status);
WFQueryExpr expr = new WFQueryExpr("STATUS", SQLExpr.EQ, any, true);

try
{
    // Calling the QueryProcInsts WebMethod, passing the expression as the
    // argument.
    WFBaseProcessInstance[] result = svc.QueryProcInsts(expr);

    if (result != null)
    {
        // Iterating through the list of the Process Instance
    }
}
```

```
        foreach (WFBaseProcessInstance processInstance in result
            {
                //Displaying the Process Instance Details on Console.
                Console.WriteLine("ApplName-->" +
                    processInstance.ApplName);
                Console.WriteLine("DefName-->" + processInstance.DefName);
                Console.WriteLine("DefID-->" + processInstance.DefID);
                Console.WriteLine("CompletedDate-->" +
                    processInstance.CompletedDate);
                Console.WriteLine("LastModifiedBy-->" +
                    processInstance.LastModifiedBy);
            }
        }

    catch(Exception ex)
    {
        Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
    }
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Query Process Instances (Extended Method)

API Type

Web Services

Description

Retrieves a list of process instances with a SQL query expression specified by the client application.

Syntax

```
public virtual WFBaseProcessInstance[] QueryProcInstsEx(string sqlWhereClause)
```

Parameters

Name	Description
sqlWhereClause	<p><u>Description:</u></p> <p>The where clause of the SQL statement you want to query.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>The where clause of a SQL statement.</p>

Output

An array of WFBaseProcessInstance objects. It returns null if nothing matches the SQL query expression.

Example

```
// Console application sample code to illustrate QueryProcInstsEx API.  
IWFWorkflowService svc = GetWorkflowService();  
  
// SQL Expression
```

```
string sqlWhereClause = "STATUS in ('Running','Canceled')";

try
{
    //Calling QueryProcInstsEx WebMethod with sql query expression as argument.
    WFBaseProcessInstance[] result = svc.QueryProcInstsEx(where);

    if (result != null)
    {
        // Iterating through the list of the Process Instance
        foreach (WFBaseProcessInstance processInstance in result)
        {
            //Displaying the Process Instance Details on Console.
            Console.WriteLine("ApplName-->" + processInstance.ApplName);
            Console.WriteLine("DefName-->" + processInstance.DefName);
            Console.WriteLine("DefID-->" + processInstance.DefID);
            Console.WriteLine("CompletedDate-->" +
                processInstance.CompletedDate);
            Console.WriteLine("LastModifiedBy-->" +
                processInstance.LastModifiedBy);
        }
    }

    catch (Exception ex)
    {
        Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
    }
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Resume Process Instance

API Type

Web Services

Description

Resumes a suspended process instance. The process instance status is changed to running, and the statuses of all the work items (tasks) become Active.

Syntax

```
public virtual WFEvent ResumeProclnst(String processInstanceId)
```

Parameters

Name	Description
processInstanceId	<p><u>Description:</u></p> <p>Specifies the unique ID of a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process instance ID</p>

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
// This is console application sample
IWFWorkflowService svc = GetWorkflowService();

try
{
    string processInstanceID = ...// process instance to be suspended.
    WFEvent event = svc.ResumeProcInst(processInstanceID);
}

catch (Exception ex)
{
    Console.WriteLine("Failed: " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Rollback Process Instance

API Type

Web Services

Description

Rolls a [process instance](#) back to a previous specified [activity](#), or skips a specified activity that has not yet been started.

Good to Know

- When this method is invoked, the current or skipped activity becomes Cancelled. When skipping, the [process](#) moves forward regardless of the activity's [status](#).
- This API can be called on process instances in [running](#) status only.
- This API call cancels all the active [activity instances](#) and rollbacks or jumps to a specified activity.
- To call this method, the API account user must have the [access right](#), **Rollback a Process**.

Syntax

```
public virtual WFEEvent RollbackProclnst(String activityInstanceID)
```

Parameters

Name	Description
activityInstanceID	<u>Description:</u> The unique ID for an activity instance.

Name	Description
	<p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid activity instance ID.</p>

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
//This is console application sample
IWfWorkflowService svc = GetWorkflowService();
string activityInstanceID = ..// target activity instance to roll back

try
{
    WfEvent evt = workflowService.RollbackProcInst(activityInstanceID);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Split Process Instance

API Type

Web Services

Description

Splits one [process instance](#) into 2 or more process instances. The original process instance is canceled.

Syntax

```
public virtual string[] SplitProclnst(WFProcessSplittingInstruction instruction)
```

Parameters

Name	Description
instruction	<p><u>Description:</u></p> <p>Specifies the instructions for splitting a process instance.</p> <p><u>Type</u></p> <p>WFProcessSplitting</p>

Name	Description
	<p><u>Allowed Values:</u></p> <p>A WFProcessSplitting object.</p>

Output

A collection of strings that contain the [process instance IDs](#) for the process instances that were created from the split.

Example

```
// This is console application sample
public string[] SplitProcessInstance(string processInstanceID)
{
    IWfWorkflowService svc = GetWorkflowService();
    WfBaseProcessInstance processInstance =
    svc.GetProcInst(processInstanceID);
    WfCustomAttributes ds = ds = new
    WfCustomAttributes(processInstance.WorkObjectID);
    ds.AttrXml = svc.GetCustomAttrs(processInstance.WorkObjectID);
    string xml = ds["//"] as string;
    Console.WriteLine("Splitting custom attributes ...");
    List<NameValue[]> splittedCustomAttributes =
    GetSplittedCustomAttributes(xml);
    WfProcessSplittingInstruction instruction;
    instruction = new WfProcessSplittingInstruction();
    instruction.SplittingProcessInstanceID = procInstID;
    List<string> workObjectIDs = new List<string>();
    for (int i = 0; i < splittedCustomAttributes.Count; ++i)
    {
        string SplittingProcessInstanceID = UUID.GetID();
        string splittedProcInstName = string.Format("{0} - Splitted -
{1}", pi.ProcInstName, i + 1);
        string splittedWorkObjectID = string.Format("{0} - Splitted -
{1}", pi.WorkObjectID, i + 1);
        instruction.Add(
        SplittingProcessInstanceID,
        splittedProcInstName,
        splittedWorkObjectID,
        null,
        splittedCustomAttributes[i]);
    }
}
```

```
    }
    instruction.Validate();
    Console.WriteLine("Suspending process instance...");
    svc.SuspendProcInst(procInstID);
    System.Threading.Thread.Sleep(1000);
    string[] ids = svc.SplitProcInst(instruction);
    return ids;
}

// function to split custom attributes
private List<NameValue[]> GetSplittedCustomAttributes(string xml)
{
    System.Xml.XmlDocument xmlDoc = new System.Xml.XmlDocument();
    xmlDoc.PreserveWhitespace = true; xmlDoc.LoadXml(xml);
    System.Xml.XmlNamespaceManager nsm = ShUtil.GetNamespaces(xmlDoc);
    System.Xml.XmlNode parent =
        xmlDoc.SelectSingleNode("/pd:issueTracking/pd:Persons", nsm);

    // split repeating notes - 'pd:issueTracking/pd:Persons/pd:Person'
    System.Xml.XmlNodeList nodes = parent.SelectNodes("pd:Person", nsm);

    if (nodes == null || nodes.Count < 2)
    {
        throw new InvalidOperationException("Failed to get splitted
            customAttributes,
            'pd:issueTracking/pd:Persons/pd:Person' does not have
            multiple node.");
    }

    // remove child nodes
    System.Xml.XmlNode titleNode =
        xmlDoc.SelectSingleNode("/pd:issueTracking/pd:issueTitle", nsm);
    string title = titleNode.InnerText;
    System.Xml.XmlNode descriptionNode =
        xmlDoc.SelectSingleNode("/pd:issueTracking/pd:description", nsm);
    string description = descriptionNode.InnerText;
    List<NameValue[]> splittedCustomAttributes = new
        List<NameValue[]>();
    for( int i = 0; i < nodes.Count; ++i)
    {
        //change title
        titleNode.InnerText = title + "-Splitted-" + i;
        descriptionNode.InnerText = description + "- Splitted-" + i;
        parent.RemoveAll();
        parent.AppendChild(nodes[i]);
    }
}
```

```
splittedCustomAttributes.Add( NameValue.Array("//",
xmlDoc.OuterXml) );
}
return splittedCustomAttributes;
}
```

Supported Versions

4.5 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Start Process Instance

API Type

Web Services

Description

Starts a process instance that is not set to start immediately on creation. This method is obsolete.

Syntax

```
public virtual WFEvent StartProInst(string processInstanceId)
```

Parameters

Name	Description
processInstanceID	<p><u>Description:</u></p> <p>Specifies the unique ID of a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process instance ID</p>

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

None .

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are

outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Suspend Process Instance

API Type

Web Services

Description

Suspends a [process instance](#) with the specified [process instance ID](#).

Good to Know

- The process instance [status](#) is changed to [Suspended](#), and the statuses of all the work items (tasks) become Pending.
- To call this method, the API account user must have the [access right](#), **Suspend and Resume a Process**.
- Only running process instances can be suspended.
- If there is a parent-child hierarchy, and the [parent process](#) instance is suspended, the running child process instances are suspended automatically.

Syntax

```
public virtual WFEEvent SuspendProInst(string processInstanceID)
```

Parameters

Name	Description
processInstanceID	Description:

Name	Description
	<p>Specifies the unique ID of a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process instance ID</p>

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
// This is console application sample
IWFWorkflowService svc = GetWorkflowService();

try
{
    string processInstanceID = ...// process instance to be suspended.
    WFEvent event = svc.SuspendProcInst(processInstanceID);
}

catch (Exception ex)
{
    Console.WriteLine("Failed: " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Update Process Instance

API Type

Web Services

Description

Updates [process attribute](#), such as process instance name, due date, work object info.

Good to Know

- The process attributes that can be updated are listed in the attribute table.
- To call this method, the API account user must have the [access right](#), **Initiate a Process**.

Syntax

```
public void UpdateProclnst(string processInstanceID, NameValue[] attributes)
```

Parameters

Name	Description
processInstanceID	Description :

Name	Description
	<p>Specifies the unique ID of a process instance.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid process instance ID</p>
attributes	<p><u>Description:</u> Name-value pairs associated with a custom ID.</p> <p><u>Type</u> NameValue</p> <p><u>Allowed Values:</u> A valid custom ID with an associated name.</p>

Attributes

Name	Description
ProclnName	The name of the process instance.
DueDate	The date that the process instance is expected to be complete
workObjectID	The ID of the work object.

Output

None.

Example

```
// This is console application sample to update process instance name
IWfWorkflowService svc = GetWorkflowService();
string processInstanceID = ... // process instance ID
string newProcessInstanceName = "[new process instance name]";
```

```
DateTime newDueDate = DateTime.Now.AddDays(7.0);

try
{
    WFBaseProcessInstance inst = svc.GetProcInst(processInstanceID);
    NameValue[] attributes = new NameValue[]
    {
        new NameValue("ProcInstName", newProcessInstanceName),
        new NameValue("DueDate", newDueDate)),
    };

    // update process instance
    svc.UpdateProcInst(processInstanceID, attributes);

    // check if it has been updated.
    string processInstanceName =
    svc.GetProcInst(processInstanceID).ProcInstName;
    Console.WriteLine("New Process Instance Name= '{0}'",
    processInstanceName);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Methods for Activity Instances

This section describes service calls related to [activity instances](#).

Cancel Activity Instance

API Type

Web Services

Description

Cancels an [activity instance](#) for the specified activity instance ID.

Good to Know

- Changes the [status](#) of an [activity](#) to [Cancelled](#).
- An [activity instance](#) can be associated with one or more [work items](#).
- Once the activity instance is cancelled, the [process instance](#) will move forward to the next activity.
- The process instance must be in the [running](#) status, or an exception is thrown.
- This API cancels all the [tasks](#) from multiple [sessions](#) of this activity if the tasks are in [Assigned](#) or [Overdue](#) status.
- To call this method, the API account user must have the [access right](#), **Cancel a Task**.

Syntax

```
public virtual WFEEvent CancelActivityInst(string activityInstanceID)
```

Parameters

Name	Description
activityInstanceID	<p><u>Description:</u></p> <p>The unique ID for an activity instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid activity instance ID.</p>

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string activityInstanceID = ...// activity instance needs to be
canceled.

try
{
    WFEvent evt = svc.CancelActivityInst(activityInstanceID);
}

catch(Exception ex)
{
```

```
Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));  
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Activity Instance

API Type

Web Services

Description

Retrieves an [activity instance](#) related information for the specified activity instance ID.

Syntax

```
public virtual WFBaseActivityInstance GetActivityInst(string activityInstanceID)
```

Parameters

Name	Description
activityInstanceID	<p><u>Description:</u></p> <p>The unique ID for an activity instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid activity instance ID.</p>

Output

WFBaseActivityInstance object.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string activityInstanceID = ..// activity instance

try
{
    WFBaseActivityInstance activityInstance =
    _svc.GetActivityInst(activityInstanceID);
    Console.WriteLine("DisplayName" + activityInstance.DisplayName);
    Console.WriteLine("ID" + activityInstance.ID);
    Console.WriteLine("CompletedDate" + activityInstance.CompletedDate);
    Console.WriteLine("DueDate" + activityInstance.DueDate);
}

catch( Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Activity Instance Status

API Type

Web Services

Description

Retrieves the [status](#) of all [activities](#) for a specified [process instance](#).

Good to Know

- Results are returned as an array of KeyValues where the key contains activity name, and the value contains the status.

Syntax

```
public virtual KeyValue[] GetActivityInstStatus(string proclnID)
```

Parameters

Name	Description
procInstID	<p><u>Description:</u></p> <p>Specifies the unique ID of a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process instance ID</p>

Output

An array of [KeyValue](#) objects that holds pairs of activity definition names and statuses. The statuses can be Passed, Active, Pending, Activated, Canceled, or null.

Example

```
// This is console application sample.
IWFFWorkflowService svc = GetWorkflowService();
string procInstID = ..// for example, "02C3FA88ADE04750A34B5B3168C25793";

try
{
    KeyValue[] resultList = svc.GetActivityInstStatus(procInstID);
    foreach (KeyValue result in resultList)
    {
        System.Console.WriteLine("Activity Definition ID: '{0}'",result.Key);
        System.Console.WriteLine("Status: '{0}'",
            result.Value);
    }
}

catch( Exception ex )
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Activity Instances By Process Instance ID

API Type

Web Services

Description

Retrieves all [activity instances](#) for the specified [process instance ID](#).

Syntax

```
public virtual WFBaseActivityInstance[] GetActivityInstsByPIID(string processInstanceID)
```

Parameters

Name	Description
processInstanceID	<u>Description:</u> Specifies the unique ID of a process instance.

Name	Description
	<p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process instance ID</p>

Output

An array of WFBaseActivityInstance objects.

Example

```
// This is console application sample
IWfWorkflowService svc = GetWorkflowService();
string processInstanceID = ..//"02C3FA88ADE04750A34B5B3168C25793";

try
{
    WFBaseActivityInstance[] activityInstance =
    svc.GetActivityInstsByPIID(processInstanceID);
    foreach (WFBaseActivityInstance activity in activityInstance)
    {
        System.Console.WriteLine("Activity DispalyName: '{0}'",
activity.DisplayName);
        System.Console.WriteLine("CompletedDate: '{0}'", activity.CompletedDate);
    }
}

catch( Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex) );
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Query Activity Instances

API Type

Web Services

Description

Retrieves [activity instances](#) that match a query expression.

Syntax

```
public virtual WFBaseActivityInstance[] QueryActivityInsts(WFQueryExpr expr)
```

Parameters

Name	Description
expr	<p><u>Description:</u></p> <p>Specifies the where clause of a SQL query expression.</p> <p><u>Type</u></p> <p>WFQueryExpr</p>

Name	Description
	<u>Allowed Values:</u> A valid WFQueryExpr object.

Output

An array of WFBaseActivityInstance objects.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string processInstanceID = ...
WFAny any = WFAny.Create(processInstanceID);
WFQueryExpr expr = new WFQueryExpr("PROC_INST_ID", SQLExpr.IN, any,
true);

try
{
    WFBaseActivityInstance[] ais = svc.QueryActivityInsts(expr);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Rollback Activity Instance

API Type

Web Services

Description

Rolls back an [activity instance](#) with the specified activity instance ID.

Good to Know

- Rolls back a [human task activity](#) instance to the token position EN – that is, the state where the [activity](#) is entered.
- All tasks associated with the human task [activity instance](#) with the [status](#) of [New](#), [Overdue](#), or [Assigned](#) change to [Cancelled](#).
- To call this method, the API account user must have the [access right](#), **Rollback a Process**.

Syntax

```
public virtual WFEvent RollbackActivityInst(string activityInstanceID)
```

Parameters

Name	Description
activityInstanceID	<p><u>Description:</u></p> <p>The unique ID for an activity instance.</p> <p><u>Type</u></p>

Name	Description
	<code>string</code> <u>Allowed Values:</u> A valid activity instance ID.

Output

`WFEvent` object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
// This is console application sample
IWfWorkflowService svc = GetWorkflowService();
string activityInstanceID = ...

try
{
    //Rolling back the activity instance
    WFEvent evt = svc.RollbackActivityInst(activityInstanceID);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Rollback Activity Instances

API Type

Web Services

Description

Activates target [activity instances](#) by cancelling the source activity instances.

Good to Know

- All the source [activity instances](#) must be activated and all the destination activity instances must not be activated when this method is called.
- To call this method, the API account user must have the [access right](#), **Rollback a Process**.

Syntax

```
public virtual WFEvent RollbackActivityInsts(WFPartialRollbackInstruction instruction)
```

Parameters

Name	Description
instruction	<p><u>Description:</u></p> <p>Specifies the instructions for the partial rollback.</p> <p><u>Type</u></p> <p>WFPartialRollbackInstruction</p> <p><u>Allowed Values:</u></p> <p>A WFPartialRollbackInstruction object.</p>

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
//Sample for partial rollback
IWfWorkflowService svc = GetWorkflowService();

// PartialRollback unit
WFPartialRollbackInstruction.PartialRollbackUnit unit1 =
    new WFPartialRollbackInstruction.PartialRollbackUnit();
unit1.DestinationActivityInstanceID =
    ... // destination activity instance ID
unit1.SourceActivityInstanceIDs =
    new string[] { ... }; // array of source activity instance ID
WFPartialRollbackInstruction.PartialRollbackUnit unit2 =
    new WFPartialRollbackInstruction.PartialRollbackUnit();
```

```
unit2.DestinationActivityInstanceID =
    ... // destination activity instance ID
unit2.SourceActivityInstanceIDs =
    new string[] { ... }; // array of source activity instance ID
WFPartialRollbackInstruction instruction =
    new WFPartialRollbackInstruction();
instruction.PartialRollbackUnits =
    new WFPartialRollbackInstruction.PartialRollbackUnit[]
    {unit1,unit2};

try
{
    //Rolling back the activity instance
    WFEvent evt = workflowService. RollbackActivityInsts(instruction);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

4.6 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Methods for Manual Work Items (Tasks)

This section describes service calls related to manual work items (tasks).

Assign Work Item

API Type

Web Services

Description

Assigns a human [task](#) (manual work item) to a user, which often means claiming a task for oneself.

Good to Know

- This is often used with task pools where tasks are created, and then multiple users are notified, but the task is not immediately assigned to a user. A user then claims the task, or his manager assigns it to him.
- The user must have [access rights](#) to claim or assign the task.

Syntax

```
public virtual WFEEvent AssignWorkItem(string workItemID)
```

Parameters

Name	Description
workItemID	<p><u>Description:</u></p> <p>An ID that represents a work item (task).</p> <p><u>Type</u></p> <p>string</p>

Name	Description
	<u>Allowed Values:</u> A valid, unique 32-byte work item (task) ID.

Output

`WFEEvent` class that represents the workflow event instance that is created when the task is assigned.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string workItemID = ...// work item ID

try
{
    WFEEvent evt = svc.AssignWorkItem(workItemID);
}

catch( Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Assign Work Item (Extended Method)

API Type

Web Services

Description

Assigns a work item to a user, which often means claiming a work item for oneself. This is often used with task pools where work items are created, and then multiple users are notified, but the work item is not immediately assigned to a user. A user then claims the work item, or his manager assigns it to him. The user must have privileges to claim or assign the work item. This method extends `AssignWorkItem()` by allowing you to specify client data.

Syntax

```
public virtual WFEvent AssignWorkItemEx(string workItemID, string clientData)
```

Parameters

Name	Description
workItemID	<p><u>Description:</u></p> <p>An ID that represents a work item (task).</p> <p><u>Type</u></p> <p><code>string</code></p> <p><u>Allowed Values:</u></p> <p>A valid, unique 32-byte work item (task) ID.</p>

Name	Description
clientData	<p><u>Description:</u></p> <p>Specifies the client data, which identifies a client for AgilePoint Server.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that contains the client data.</p> <p>If this value is null, the system will keep existing client data. Otherwise the relevant data is overwritten.</p>

Output

WFEvent object representing the workflow event instance raised by the invocation of the work assignment.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string workItemID = ...// work item ID
string clientData = null;

try
{
    WFEvent evt = svc.AssignWorkItemEx(workItemID, clientData);
}

catch( Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Cancel Work Item

API Type

Web Services

Description

Cancels a human [task](#) (manual work item) based on a specified [task ID](#).

Good to Know

- Only the following task [status](#) can transition to a [Cancelled](#) status: [Assigned](#), [New](#), [Pseudo](#), and [Overdue](#).
- To call this method, the API account user must have the [access right](#), **Cancel a Task**.
- It is not possible to cancel a task that has child tasks [running](#).

Syntax

```
public virtual WFEEvent CancelWorkItem(string workItemID)
```

Parameters

Name	Description
workItemID	<p><u>Description:</u></p> <p>An ID that represents a work item (task).</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, unique 32-byte work item (task) ID.</p>

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string workItemID = ... //

try
{
    WFEvent evt = svc.CancelWorkItem(workItemID);
}

catch( Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Cancel Work Item (Extended Method)

API Type

Web Services

Description

Cancels a manual work item based on a specified manual work item identifier. This method also contains an added routine to track the method call duration. Only the following manual work item status can transition to a Canceled status: Assigned, New, Pseudo, and Overdue.

Syntax

```
public virtual WFEEvent CancelWorkItemEx(string workItemID, string clientData)
```

Parameters

Name	Description
workItemID	<p><u>Description:</u></p> <p>An ID that represents a work item (task).</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, unique 32-byte work item (task) ID.</p>
clientData	<p><u>Description:</u></p> <p>Specifies the client data, which identifies a client for AgilePoint Server.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that contains the client data.</p> <p>If this value is null, the system will keep existing client data. Otherwise the relevant data is overwritten.</p>

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string workItemID = ...// work item ID
string clientData = null;

try
{
    WFEvent evt = svc.CancelWorkItemEx(workItemID, clientData);
}

catch( Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Complete Work Item

API Type

Web Services

Description

Completes the specified human **task** (manual work item), and changes its **status** to **Completed** in the database.

Good to Know

- If the **process instance** is not in the **running** state, an exception is thrown.
- This API can be called only by the task owner or AgilePoint System User.
- If the specified task is a parent task with more than one linked tasks in the **Assigned** state, the API call fails.
- If the specified task is already completed, and user makes multiple attempts, the API returns null, with no action taken.

Syntax

```
public virtual WFEvent CompleteWorkItem(string workItemID)
```

Parameters

Name	Description
workItemID	<p><u>Description:</u></p> <p>An ID that represents a work item (task).</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, unique 32-byte work item (task) ID.</p>

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.

- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string workItemID = ...// work item ID

try
{
    WFEvent evt = svc.CompleteWorkItem(workItemID);
}

catch( Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Complete Work Item (Extended Method)

API Type

Web Services

Description

Marks a work item as completed with client data.

Syntax

```
public virtual WFEvent CompleteWorkItemEx(string workItemID, string clientData)
```

Parameters

Name	Description
workItemID	<p><u>Description:</u></p> <p>An ID that represents a work item (task).</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, unique 32-byte work item (task) ID.</p>
clientData	<p><u>Description:</u></p> <p>Specifies the client data, which identifies a client for AgilePoint Server.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that contains the client data.</p>

Name	Description
	If this value is null, the system will keep existing client data. Otherwise the relevant data is overwritten.

Output

[WFEvent](#) object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string workItemID = ...// work item ID
string clientData = null;

try
{
    WFEvent evt = svc.CompleteWorkItemEx(workItemID, clientData);
}

catch( Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are

outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Create Linked Work Item

API Type

Web Services

Description

Creates a human [task](#) (manual work item) that is linked to another human task.

Good to Know

- The task you create may or may not depend on the completion of the task to which it is linked. It is based on the input parameter for the API -- for example Dependent with a value of true or false. In other words, if Dependent is set to true, the parent task cannot be completed until the newly created task gets completed. If Dependent is set to false the parent task can be marked as [Completed](#) before new task is completed.
- To call this method, the API account user must have the [access right](#), **Create a Task**.

Syntax

```
public virtual WFEEvent CreateLinkedWorkItem(string SourceWorkItemID, string WorkToPerform, string UserID, WFTimeDuration duration, string ClientData)
```

Parameters

Name	Description
SourceWorkItemID	Description :

Name	Description
	<p>An ID that represents the original, or source, work item.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid, unique 32-character ID.</p>
WorkToPerform	<p><u>Description:</u> Represents the task that performed by the participants of the activity.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid work to perform name.</p>
UserID	<p><u>Description:</u> Specifies the user ID.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid user ID.</p>
duration	<p><u>Description:</u> Specifies the duration settings of a work item.</p> <p><u>Type</u> WFTimeDuration</p> <p><u>Allowed Values:</u> A valid WFTimeDuration object.</p>

Name	Description
ClientData	<p><u>Description:</u></p> <p>Specifies the client data, which identifies a client for AgilePoint Server.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that contains the client data.</p> <p>If this value is null, the system will keep existing client data. Otherwise the relevant data is overwritten.</p>

Output

None.

Example

```
IWFWorkflowService svc = base.GetWorkflowService();
// get existing work item
string workItemID = ..// for example,
"90CF843AC57644058A391FBFA030F607"

try
{
    // Get the source WFManualWorkItem object
    WFManualWorkItem SourceWorkItemID = svc.GetWorkItem(workItemID)
    string WorkToPerform = SourceWorkItemID.Name; //different
    WorkToPerform can be used if desired
    WFTimeDuration duration = new WFTimeDuration();
    duration.Length = "15"; //for example, 15 days
    duration.Unit = WFTimeUnit.DAY;
    string UserID = @"[DOMAIN NAME]\username"; //the participant of the
    linked work item
    WFEvent evt = svc.CreateLinkedWorkItem(
        SourceWorkItemID.WorkItemID,
        WorkToPerform,
        UserID,
```

```
        duration,  
        null);  
    }  
  
    catch( Exception ex)  
    {  
        Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));  
    }  
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Create Linked Work Item (Extended Method)

API Type

Web Services

Description

Creates a manual work item that is linked to another manual work item. The extended parameter `bDependent` is used to specify the dependency between the original work item and the linked work item. If `bDependent` is false, the work items are independent, just like `CreateLinkedWorkItem()`. If `bDependent` is true, the original (source) work item cannot be marked as completed before new work item has been completed.

Syntax

```
public virtual WFEvent CreateLinkedWorkItemEx(string SourceWorkItemID, string WorkToPerform, string UserID, WFTimeDuration duration, string ClientData, bool bDependent)
```

Parameters

Name	Description
SourceWorkItemID	<p><u>Description:</u> An ID that represents the original, or source, work item.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid, unique 32-character ID.</p>
WorkToPerform	<p><u>Description:</u> Represents the task that performed by the participants of the activity.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid work to perform name.</p>
UserID	<p><u>Description:</u> Specifies the user ID.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid user ID.</p>

Name	Description
duration	<p><u>Description:</u></p> <p>Specifies the duration settings of a work item.</p> <p><u>Type</u></p> <p>WFTimeDuration</p> <p><u>Allowed Values:</u></p> <p>A valid WFTimeDuration object.</p>
ClientData	<p><u>Description:</u></p> <p>Specifies the client data, which identifies a client for AgilePoint Server.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that contains the client data.</p> <p>If this value is null, the system will keep existing client data. Otherwise the relevant data is overwritten.</p>
bDependent	<p><u>Description:</u></p> <p>Specifies whether the target work item waits for a linked work item to be completed or canceled before moving forward.</p> <p><u>Type</u></p> <p>bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - The source work item waits until the linked work item is either completed or canceled, before it can be completed or canceled. • False - The source work item can be completed or canceled regardless of whether the linked work item is completed or canceled.

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
IWFWorkflowService svc = base.GetWorkflowService();

// get existing work item
string workItemID = ..// for example, "90CF843AC57644058A391FBFA030F607"

try
{
    // Get the source WFManualWorkItem object
    WFManualWorkItem SourceWorkItem = svc.GetWorkItem(workItemID)
    string WorkToPerform = sourceWorkItem.Name; //different
    WorkToPerform can be used if desired
    WFTimeDuration duration = new WFTimeDuration("15", WFTimeUnit.DAY, false);
    string UserID = @"[DOMAIN NAME]\username"; //the participant of the
    linked work item
    WFEvent evt = svc.CreateLinkedWorkItem(
        SourceWorkItem.WorkItemID,
        WorkToPerform,
        UserID,
        duration,
        null,
        true);
}

catch( Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

4.6 and above

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Create Pseudo Work Item

API Type

Web Services

Description

Creates a [Carbon](#) copy [task](#) for a manager that does not have to be completed in order for a [process instance](#) to advance to the next [activity](#).

Good to Know

- To call this method, the API account user must have the [access right](#), **Create a Task**.

Syntax

```
public virtual WFEEvent CreatePseudoWorkItem(string SourceWorkItemID, string WorkToPerform, string UserID, WFTimeDuration duration, string ClientData, bool bReserved)
```

Parameters

Name	Description
SourceWorkItemID	<p><u>Description:</u></p> <p>An ID that represents the original, or source, work item.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, unique 32-character ID.</p>
WorkToPerform	<p><u>Description:</u></p> <p>Represents the task that performed by the participants of the activity.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid work to perform name.</p>
UserID	<p><u>Description:</u></p> <p>Specifies the user associated with the work item.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user ID.</p>
duration	<p><u>Description:</u></p> <p>Specifies the duration settings of a work item.</p> <p><u>Type</u></p>

Name	Description
	<p>WFTimeDuration</p> <p><u>Allowed Values:</u> A valid WFTimeDuration object.</p>
ClientData	<p><u>Description:</u> Specifies the client data, which identifies a client for AgilePoint Server.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A string that contains the client data. If this value is null, the system will keep existing client data. Otherwise the relevant data is overwritten.</p>

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
IWFWorkflowService svc = base.GetWorkflowService();

// get existing work item
string workItemID = ..// for example, "90CF843AC57644058A391FBFA030F607"

try
{
```

```
// Get the source WFManualWorkItem object
WFManualWorkItem SourceWorkItem = svc.GetWorkItem(workItemID)
string WorkToPerform = sourceWorkItem.Name; //different
WorkToPerform can be used if desired
WFTimeDuration duration = new WFTimeDuration("15", WFTimeUnit.DAY, false);
string UserID = @"[DOMAIN NAME]\username"; //the participant of the linked work
item
WFEvt evt = svc.CreatePseudoWorkItem (
    SourceWorkItem.WorkItemID,
    WorkToPerform,
    UserID,
    duration,
    null,
    false);
}

catch( Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Create Work Item

API Type

Web Services

Description

Creates a human [task](#) (manual work item) for the specified activity instance ID.

Good to Know

- You must send the activity instance ID, WorkToPerform, user ID, task owner, duration, and ClientData (optional) parameters for this API call.
- To call this method, the API account user must have the [access right](#), **Create a Task**.
- The created task is not dependent on any other tasks.

Syntax

```
public virtual WFEEvent CreateWorkItem(string ActivityInstanceID, string WorkToPerform, string UserID, WFTimeDuration Length, string ClientData)
```

Parameters

Name	Description
ActivityInstanceID	<p><u>Description:</u></p> <p>The unique ID for an activity instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid activity instance ID.</p>
WorkToPerform	<p><u>Description:</u></p> <p>Represents the task that performed by the participants of the activity.</p>

Name	Description
	<p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid work to perform name.</p>
UserID	<p><u>Description:</u></p> <p>Specifies the user associated with the work item.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user ID.</p>
Length	<p><u>Description:</u></p> <p>Specifies the length of time duration.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid time duration length.</p>
ClientData	<p><u>Description:</u></p> <p>Specifies the client data, which identifies a client for AgilePoint Server.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that contains the client data.</p>

Name	Description
	If this value is null, the system will keep existing client data. Otherwise the relevant data is overwritten.

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string ActivityInstanceID = ... // for example, "0172460E0AF943C6A6520044452BCAB3";
string WorkToPerform = ... // for example, "SubmitRequest";
//different WorkToPerform can be used if desired

WFTimeDuration Length = new WFTimeDuration("15", WFTimeUnit.DAY, true );// business
time
string UserID = @"[DOMAIN NAME]\username"; //the participant of the linked work
item

try
{
    WFEventevt = svc.CreateWorkItem(ActivityInstanceID,
    WorkToPerform, UserID, Length, null);
}

catch(Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Work Item

API Type

Web Services

Description

Retrieves a human [task](#) (manual work item) object for the specified [task ID](#).

Syntax

```
public virtual WFManualWorkItem GetWorkItem(string workItemID)
```

Parameters

Name	Description
workItemID	<u>Description:</u> An ID that represents a work item (task).

Name	Description
	<p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, unique 32-byte work item (task) ID.</p>

Output

WFMaunalWorkItem object.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string workItemID = ...// for example, "54A648A0A3004A02981E7F0848820FE7";

try
{
    WFAutomaticWorkItem workItem = svc.GetWorkItem(workItemID);
    Console.WriteLine("{0}", workItem.Name);
}

catch( Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Work List By User ID

API Type

Web Services

Description

Retrieves a human [task](#) (manual work item) collection by specifying a user name and task [status](#).

Good to Know

- Multiple status can be specified with a semicolon (;) delimited list.

Syntax

```
public virtual WFManualWorkItem[] GetWorkListByUserID(string UserName, string Status)
```

Parameters

Name	Description
UserName	<p><u>Description:</u></p> <p>Specifies a qualified user name of the instance. A qualified user name formats as [Domain Name]\[Logon Username] or [Local host name]\[Logon Username].</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p>

Name	Description
	A valid user name for a registered AgilePoint user.
Status	<p><u>Description:</u></p> <p>The status of the work item.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • New - The work item is in a pool, more than one participant shares the work item and needs to be assigned. • Assigned - The work item is assigned to a participant and needs to be performed. • Removed - The work item was originally in a pool (status is new), and didn't get assigned to the user. • Completed - The work item is completed. • Reassigned - The work item is re-assigned to the other user. • Canceled - The work item is canceled. • Overdue - The work item is overdue. • Carbon - The work item is a carbon copy that does not affect process instance running.

Output

Array of WFManualWorkItem objects.

Example

```
//Get all WFManualWorkItem assigned to user
IWFWorkflowService svc = GetWorkflowService();
string userID = ...// for example, @"Demo3\Administrator";
string Status= string.format("{0};{1}",
WFManualWorkItem.ASSIGNED, WFManualWorkItem.OVERDUE);

try
```

```
{
    WFManualWorkItem[] workItems = svc.GetWorkListByUserID(userID,
    Status);
    foreach (WFManualWorkItem workItem in workItems)
    {
        Console.WriteLine("{0}", workItem.ApplName);
        Console.WriteLine("{0}", workItem.AssignedDate);
        Console.WriteLine("{0}", workItem.DefName);
        Console.WriteLine("{0}", workItem.DueDate);
    }
}

catch(Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Query Work List

API Type

Web Services

Description

Retrieves a list of human [tasks](#) (manual work items) using a specified [WHERE clause](#) of an SQL query expression.

Good to Know

- Only pass the WHERE clause without WHERE keyword, not the whole query.

Syntax

```
public virtual WFManualWorkItem[] QueryWorkList(WFQueryExpr expr)
```

Parameters

Name	Description
expr	<p><u>Description:</u></p> <p>Specifies the where clause of a SQL query expression.</p> <p><u>Type</u></p> <p>WFQueryExpr</p> <p><u>Allowed Values:</u></p> <p>A valid WFQueryExpr object.</p>

Output

An array of WFManualWorkItem objects that contain the work item data.

Example

```
IWFWorkflowService svc = GetWorkflowService();
WFAny any = WFAny.Create(WFManualWorkItem.ASSIGNED);
WFQueryExpr expr = new WFQueryExpr("WF_MANUAL_WORKITEM.STATUS",
SQLExpr.EQ, any, true);

try
{
```

```
WFManualWorkItem[] workItems = svc.QueryWorkList(expr);

// Iterating through the list of the ManualWorkItem
foreach (WFManualWorkItem workItem in workItems)
{
    Console.WriteLine("{0}", workItem.ApplName);
    Console.WriteLine("{0}", workItem.AssignedDate);
    Console.WriteLine("{0}", workItem.DefName);
    Console.WriteLine("{0}", workItem.DueDate);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Query Work List (Extended Method)

API Type

Web Services

Description

Retrieves a list of manual work items that match a SQL statement.

Syntax

```
public virtual WFManualWorkItem[] QueryWorkListEx(string sqlWhereClause)
```

Parameters

Name	Description
sqlWhereClause	<p><u>Description:</u></p> <p>The where clause of the SQL statement you want to query.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>The where clause of a SQL statement.</p>

Output

An array of WFManualWorkItem objects.

Example

```
IWFWorkflowService svc = GetWorkflowService();

//Query Expression string for Comparison of the WORK_ITEM_ID
string sqlWhereClause = "WORK_ITEM_ID in ('0006EE0244ED431CB93F6253060DD21F', ...)";

try
{
    // Calling the QueryWorkListEx API with argument sql query expression string.
    WFManualWorkItem[] workItems = _svc.QueryWorkListEx(sqlWhereClause);

    // Iterating through the list of the ManualWorkItem
    foreach (WFManualWorkItem workItem in workItems)
```

```
        {
            Console.WriteLine("{0}", workItem.ApplName);
            Console.WriteLine("{0}", workItem.AssignedDate);
            Console.WriteLine("{0}", workItem.DefName);
            Console.WriteLine("{0}", workItem.DueDate);
        }
    }

    catch (Exception ex)
    {
        Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
    }
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Reassign Update Work Item

API Type

Web Services

Description

Reassigns a manual work item to another participant.

Syntax

```
public virtual WFEvent ReassignUpdateWorkItem(string workItemID, string originalUserID, string newAssigneedUserID, string clientData)
```

Parameters

Name	Description
WorkItemID	<p><u>Description:</u></p> <p>An ID that represents a work item (task).</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, unique 32-byte work item (task) ID.</p>
originalUserID	<p><u>Description:</u></p> <p>The user ID for the original user assigned the work item.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user ID.</p>
newAssigneedUserID	<p><u>Description:</u></p> <p>The user name for a user to whom you want to assign a work item.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name.</p>

Name	Description
clientData	<p><u>Description:</u></p> <p>Specifies the client data, which identifies a client for AgilePoint Server.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that contains the client data.</p> <p>If this value is null, the system will keep existing client data. Otherwise the relevant data is overwritten.</p>

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string WorkItemID = ...// "0006EE0244ED431CB93F6253060DD21F"; // Work item ID
string originalUserID = ...// @"[DOMAIN NAME]\[user name]"; // new user ID
string newAssignedUserID = ...// @"[DOMAIN NAME]\[user name]"; // new user ID

try
{
    WFEvent evt = ReassignUpdateWorkItem(WorkItemID, originalUserID,
    newAssignedUserID, null);
}

catch( Exception ex)
```

```
{  
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));  
}
```

Supported Versions

4.6 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Reassign Work Item

API Type

Web Services

Description

Reassigns a human [task](#) (manual work item) to another [participant](#), and updates the user name field for the task.

Good to Know

- To call this method, the API account user must have the [access right](#), **Reassign a Task**.
- This API can be called only by the task owner or [AgilePoint System Account](#).

Syntax

```
public virtual WFEvent ReassignWorkItem(string WorkItemID, string UserName)
```

Parameters

Name	Description
WorkItemID	<p><u>Description:</u></p> <p>An ID that represents a work item (task).</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, unique 32-byte work item (task) ID.</p>
UserName	<p><u>Description:</u></p> <p>The user name for the user.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name for a registered AgilePoint user.</p>

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string WorkItemID = ...// "0006EE0244ED431CB93F6253060DD21F"; // Work item ID
string UserName = ...// @"[DOMAIN NAME]\[user name]"; // new user ID

try
{
    WFEvent evt = svc.ReassignWorkItem(WorkItemID, UserName);
}

catch(Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Reassign Work Item (Extended Method)

API Type

Web Services

Description

Reassigns a work item to another participant, and update the user name. The extended method includes client data.

Syntax

```
public virtual WFEvent ReassignWorkItemEx(string WorkItemID, string UserName, string ClientData)
```

Parameters

Name	Description
WorkItemID	<p><u>Description:</u></p> <p>An ID that represents a work item (task).</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, unique 32-byte work item (task) ID.</p>
UserName	<p><u>Description:</u></p> <p>The user name for the user.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name for a registered AgilePoint user.</p>
ClientData	<p><u>Description:</u></p> <p>Specifies the client data, which identifies a client for AgilePoint Server.</p> <p><u>Type</u></p> <p>string</p>

Name	Description
	<p><u>Allowed Values:</u></p> <p>A string that contains the client data.</p> <p>If this value is null, the system will keep existing client data. Otherwise the relevant data is overwritten.</p>

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
string ClientData= "<?xml version="1.0" ... ..";
string WorkItemID = ...// for example, "0006EE0244ED431CB93F6253060DD21F"; // Work
item ID
string UserName = ...// for example @"[DOMAIN NAME]\[USER NAME]"; // new user ID

try
{
    WFEvent evt = _svc.ReassignWorkItemEx(WorkItemID,
    UserName, ClientData);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Undo Assign Work Item

API Type

Web Services

Description

Unassigns a human [task](#) (manual work item) that was previously assigned to a user.

Good to Know

- This method applies to the tasks that can be assigned to members of [groups](#) or [roles](#).
- If the task [status](#) is not [Assigned](#) or [Overdue](#), the API fails.
- The API call must be made by task owner or [AgilePoint System Account](#).

Syntax

```
public virtual WFEvent UndoAssignWorkItem(string workItemID)
```

Parameters

Name	Description
workItemID	<p><u>Description:</u></p> <p>An ID that represents a work item (task).</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, unique 32-byte work item (task) ID.</p>

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string workItemID = ...// for example, "03ABD59A0EB74D7A8741709478E83877";

try
{
    WFEvent evt = svc.UndoAssignWorkItem(workItemID);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Undo Assign Work Item (Extended Method)

API Type

Web Services

Description

Unassigns a work item that was previously assigned to a user. This method applies to work items that can be assigned to members of task groups, where a work item can be assigned to or claimed by any of a group of users. The extended method includes client data.

Syntax

```
public virtual WFEEvent UndoAssignWorkItemEx(string workItemID, string clientData)
```

Parameters

Name	Description
workItemID	<p><u>Description:</u></p> <p>An ID that represents a work item (task).</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, unique 32-byte work item (task) ID.</p>
clientData	<p><u>Description:</u></p> <p>Specifies the client data, which identifies a client for AgilePoint Server.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that contains the client data.</p> <p>If this value is null, the system will keep existing client data. Otherwise the relevant data is overwritten.</p>

Output

WFEvent object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
string url = "http://[hostname]:[port]/AgilePointServer";
string workItemID = ...// for example,
"03ABD59A0EB74D7A8741709478E83877";
string clientData = ...//

try
{
    WFEvent evt = svc.UndoAssignWorkItemEx(workItemID, clientData);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Update Work Item

API Type

Web Services

Description

Updates the properties for a human [task](#) (manual work item) or a [system activity](#) (automatic work item).

Good to Know

- Lets you update only the properties: Task Name, Original User ID, ClientData, Pool ID, Pool Info, Status, User ID, Resolve Participant, Priority, or Due Date.

If you try to update any other property, it throws an exception.

- To call this method, the API account user must have the [access right](#), **Create a Task**.

Syntax

```
public virtual void UpdateWorkItem(string workItemID, NameValue[] attributes)
```

Parameters

Name	Description
workItemID	<p><u>Description:</u></p> <p>An ID that represents a work item (task).</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, unique 32-byte work item (task) ID.</p>
attributes	<p><u>Description:</u></p> <p>A NameValue array that contains the attributes that needs to be updated in the work item.</p>

Name	Description
	<p><u>Type</u></p> <p>NameValue</p> <p><u>Allowed Values:</u></p> <p>For a manual work item, the following attributes can be updated:</p> <ul style="list-style-type: none"> • NAME • ORIGINAL_USER_ID • CLIENT_DATA, POOL_ID • POOL_INFO • STATUS • USER_ID • PRIORITY • DUE_DATE <p>For an automatic work item, the following attributes can be updated:</p> <ul style="list-style-type: none"> • DUE_DATE • STATUS - if the value is Canceled, Completed, Overdue, Running, or Waiting.

Output

None.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string workItemID = ...// work item ID of manual work item or automatic work item.

try
{
    NameValue[] attributes = NameValue.Array(
        "NAME", "[New Name]",
        "DUE_DATE", [DateTime]); // for example, DateTime.Now.AddDays(3.0)
    svc.UpdateWorkItem(workItemID, attributes);
}
```

```
}  
  
catch (Exception ex)  
{  
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));  
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Methods for Automatic Work Items (Procedures)

This section describes service calls related to automatic work items.

Cancel Procedure

API Type

Web Services

Description

Cancels the specified [system activity](#) (automatic work item).

Good to Know

- This API can be called for [process instances](#) in the [running](#) status only.
- To call this method, the API account user must have the [access right](#), **Cancel a Task**.

Syntax

```
public virtual WFEvent CancelProcedure(string workItemID)
```

Parameters

Name	Description
workItemID	<p><u>Description:</u></p> <p>An ID that represents a work item (task).</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, unique 32-byte work item (task) ID.</p>

Output

[WFEvent](#) object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.
- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string workItemID = ...//

try
{
    WFEEvent evt = svc.CancelProcedure(workItemID);
}

catch( Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Complete Procedure

API Type

Web Services

Description

Completes the specified [system activity](#) (automatic work item).

Good to Know

- This API can be called for [process instances](#) in the [running](#) or [Suspended status](#) only.
- If **trustedAuthorization** is set to **False**, this method will not execute unless a special access control is turned on.

For more information, refer to [Why Do I Get an Access Denied Error for an API Method Call?](#).

Syntax

```
public virtual WFEvent CompleteProcedure(string workItemID)
```

Parameters

Name	Description
workItemID	<p><u>Description:</u></p> <p>An ID that represents a work item (task).</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, unique 32-byte work item (task) ID.</p>

Output

[WFEvent](#) object that provides the status of the transaction. The possible statuses are:

- **Sent** - Indicates event has been sent to engine for processing.
- **Failed** - Indicates event failed to process.
- **Processed** - Indicates event has been processed successfully.

- **Canceled** - Indicates event was canceled.
- **Deferred** - Indicates event does not need to be sent immediately.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string workItemID = ..//

try
{
    svc.CompleteProcedure(workItemID);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Procedure

API Type

Web Services

Description

Retrieves a [system activity](#) (automatic work item) with the specified [task ID](#).

Syntax

```
public virtual WFAutomaticWorkItem GetProcedure(string workItemID)
```

Parameters

Name	Description
workItemID	<p><u>Description:</u></p> <p>An ID that represents a work item (task).</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid, unique 32-byte work item (task) ID.</p>

Output

WFAutomaticWorkItem object.

Example

```
IWFWorkflowService svc = GetWorkflowService()  
string workItemID = ..// for example,  
"54A648A0A3004A02981E7F0848820FE7";
```

```
try
{
    WFAutomaticWorkItem wItem = svc.GetProcedure(workItemID);
}

catch( Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Query Procedure List

API Type

Web Services

Description

Retrieves a list of [system activities](#) (automatic work items) that match a specified query expression.

Syntax

```
public virtual WFAutomaticWorkItem[] QueryProcedureList(WFQueryExpr expr)
```

Parameters

Name	Description
expr	<p><u>Description:</u></p> <p>Specifies the where clause of a SQL query expression.</p> <p><u>Type</u></p> <p>WFQueryExpr</p> <p><u>Allowed Values:</u></p> <p>A valid WFQueryExpr object.</p>

Output

An array of automatic work items.

Example

```
IWFWorkflowService svc = GetWorkflowService();

try
{
    //WebMethod with sql query expression as argument.
    WFAny any = WFAny.Create(WFAutomaticWorkItem.WAITING);
    WFQueryExpr expr = new WFQueryExpr("STATUS", SQLExpr.EQ, any, true);
    WFAutomaticWorkItem[] result = svc.QueryProcedureList(expr);

    if (result != null)
    {
        // Iterating through the list of the automatic work item
        foreach (WFAutomaticWorkItem re in result)
        {
            Console.WriteLine("ActivityInstID-->" +
                re.ActivityInstID);
        }
    }
}
```

```
        Console.WriteLine("ApplName-->" + re.ApplName);
        Console.WriteLine("ProcInstID-->" + re.ProcInstID);
        Console.WriteLine("CreatedDate-->" + re.CreatedDate);
    }
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

User Delegation

This section describes service calls related to user delegation processes.

Activate Delegation

API Type

Web Services

Description

Activates a [delegation](#) instance using a unique [delegation ID](#).

Good to Know

- Only delegations that are in Created status can be activated, and the status changes to Active.
- Once the duration of the delegation is complete, [AgilePoint Server](#) changes the status of the delegation to Complete without further intervention.

Syntax

```
public virtual void ActivateDelegation(string delegationID)
```

Parameters

Name	Description
delegationID	<p><u>Description:</u></p> <p>The unique ID of a delegation object.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid delegation ID.</p>

Output

None.

Example

```
IWFAdminService svc = GetAdminService();
string delegationID = ...;

try
{
```

```
svc.ActivateDelegation(delegationID);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Add Delegation

API Type

Web Services

Description

Creates a rule for delegating one user's tasks to another user.

Good to Know

- After a [delegation](#) is added, its status is set to Created.
- After it is created, you must call the `ActivateDelegation` method for the delegation to take effect.
- This method uses the `WFDelegation` class. For more information, see the [AgilePoint Class Reference](#).

Syntax

```
public virtual WFDelegation AddDelegation(WFDelegation delegation)
```

Parameters

Name	Description
delegation	<p><u>Description:</u></p> <p>An object that specifies the details of the delegation rule, including the user whose tasks will be delegated and the designated user to whom to delegate the tasks.</p> <p><u>Type</u></p> <p>WFDelegation</p> <p><u>Allowed Values:</u></p> <p>A valid <code>WFDelegation</code> object.</p>

Output

`WFDelegation` object.

Example

```
IWFAdminService svc = GetAdminService();
WFDelegation delegation = new WFDelegation();

//Set the object properties.
delegation.FromUser = "Demo3\\Andy";
delegation.ToUser = "Demo3\\Joe";
delegation.StartDate = DateTime.Now;
```

```
delegation.EndDate = DateTime.Parse("27/10/2009");
delegation.Description = "Delegating Andy's task to Joe";

try
{
    WFDelegation delegation = adminService.AddDelegation(delegation);
    Console.WriteLine("Delegation ID: {0}", delegation.DelegationID);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Cancel Delegation

API Type

Web Services

Description

Deactivates a [delegation](#) instance using a [delegation ID](#).

Good to Know

- On successful execution, the delegation status is set to Cancelled.

Syntax

```
public virtual void CancelDelegation(string delegationID)
```

Parameters

Name	Description
delegationID	<p><u>Description:</u></p> <p>The unique ID of a delegation object.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid delegation ID.</p>

Output

None.

Example

```
IWFAdminService svc = GetAdminService();
string delegationID = ...;

try
{
    svc.CancelDelegation(delegationID);
}
```

```
catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Delegation

API Type

Web Services

Description

Retrieves a [delegation](#) instance using the [delegation ID](#).

Syntax

```
public virtual WFDelegation GetDelegation(string delegationID)
```

Parameters

Name	Description
delegationID	<p><u>Description:</u></p> <p>The unique ID of a delegation object.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid delegation ID.</p>

Output

WFDelegation object that specifies the user whose tasks will be delegated and the designated user to whom to delegate tasks.

Example

```
IWFAdminService svc = GetAdminService();
string delegationID = ...; // for example, "C9A40F4BDA26481FB822C398C4387901"

try
{
    WFDelegation delegation = svc.GetDelegation(delegationID);
    Console.WriteLine("Delegation Id:{0}; From User:{1}; To User:{2};
    Status:{3}",
        delegation.DelegationID,
        delegation.FromUser,
        delegation.ToUser,
        delegation.Status);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

```
/* Output
Delegation Id:C9A40F4BDA26481FB822C398C4387901; From
User:VITBDC\yuvaraj; To User:VITBDC\ravis; Status:Canceled
*/
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Delegations

API Type

Web Services

Description

Retrieves a list of [delegation](#) objects that match the specified parameters. You can leave the parameters null to indicate any.

Syntax

```
public virtual WFDelegation[] GetDelegations(string FromUser, string ToUser, string Status)
```

Parameters

Name	Description
FromUser	<p><u>Description:</u></p> <p>Specifies a user from whom to delegate tasks.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name.</p>
ToUser	<p><u>Description:</u></p> <p>Specifies the user who will receive the delegated tasks.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name.</p>
Status	<p><u>Description:</u></p> <p>The status of the associated item.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid status.</p>

Output

An array of WFDelegation objects.

Example

```
IWFAdminService svc = GetAdminService();
string FromUser = ...// for example, @"vitbdc\yuvarajn"
string ToUser = ...// for example, null for any
string Status = WFDelegation.ACTIVE;

try
{
    WFDelegation[] delegations = svc.GetDelegations( FromUser, ToUser,
    Status);
    foreach(WFDelegation delegation in delegations)
    {
        Console.WriteLine("Delegation Id:{0}; From User:{1}; To User:{2};
            Status:{3}, {4}=>{5}",
                delegation.DelegationID,
                delegation.FromUser,
                delegation.ToUser,
                delegation.Status,
                delegation.StartDate,
                delegation.EndDate);
    }
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Remove Delegation

API Type

Web Services

Description

Permanently removes a [delegation](#) instance using a unique [delegation ID](#).

Syntax

```
public virtual void RemoveDelegation(string delegationID)
```

Parameters

Name	Description
delegationID	<p><u>Description:</u></p> <p>The unique ID of a delegation object.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid delegation ID.</p>

Output

None.

Example

```
IWFAdminService svc = GetAdminService();
string delegationID = ...;

try
{
    svc.RemoveDelegation(delegationID);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Update Delegation

API Type

Web Services

Description

Updates a [delegation](#) object that has already been created.

Syntax

```
WFDelegation UpdateDelegation(WFDelegation delegation)
```

Parameters

Name	Description
delegation	<p><u>Description:</u></p> <p>An object that specifies the details of the delegation rule, including the user whose tasks will be delegated and the designated user to whom to delegate the tasks.</p> <p><u>Type</u></p> <p>WFDelegation</p> <p><u>Allowed Values:</u></p> <p>A valid WFDelegation object.</p>

Output

Returns an updated instance of WFDelegation.

Example

```
IWFAdminService svc = GetAdminService();  
WFDelegation delegation = new WFDelegation();
```

```
delegation.DelegationID = ...// unique ID
delegation.FromUser = ...// from user name
delegation.ToUser = ...// to user name
delegation.StartDate = ... // start date
delegation.EndDate = ...// end date

try
{
WFDelegation updatedDelegation = svc.UpdateDelegation( delegation );
}

catch (Exception ex)
{
Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Methods for Notifications

This section describes service calls related to email notifications.

Cancel Mail Deliverable

API Type

Web Services

Description

Cancels the failed mail deliverable record based on a given message identifier. Note that canceling the failed mail deliverable record prevents it from being recycled or present on a given interval by the AgilePoint engine.

Good to Know

- This API call sets an e-mail [status](#) to **Cancelled**.
- To call this method, the API account user must have the [access right](#), **Resend and Cancel an Email Notification**.

Syntax

```
public virtual void CancelMailDeliverable(string mailID)
```

Parameters

Name	Description
mailID	<p><u>Description:</u></p> <p>Specifies the unique ID for an email notification.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid email notification ID.</p>

Output

None.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string mailID = ...

try
{
    svc.CancelMailDeliverable(mailID);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Expecting Send Mail Deliverable

API Type

Web Services

Description

Retrieves all the failed and scheduled e-mail [notifications](#) to resend.

Syntax

```
public virtual WFMailDeliverable[] GetExpectingSendMailDeliverable()
```

Parameters

Name	Description
None	Not Applicable

Output

Array of WFMailDeliverable objects.

Example

```
IWFWorkflowService svc = GetWorkflowService();

try
{
    //Returns Array of WFMailDeliverable type
    WFMailDeliverable[] mailDeliverables =
    svc.GetExpectingSendMailDeliverable();
    foreach(WFMailDeliverable m in mailDeliverables)
    {
        Console.WriteLine("Mail ID: '{0}'", m.ID);
        Console.WriteLine("Process Instance ID: '{0}'",m.ProcInstID);
        Console.WriteLine("E-Mail Subject: '{0}'", m.Mail.Subject);
    }
}
```

```
catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Mail Deliverables

API Type

Web Services

Description

Retrieves all the mail deliverables for a process instance.

Syntax

```
public virtual WFMailDeliverable[] GetMailDeliverables(string processInstanceId)
```

Parameters

Name	Description
processInstanceID	<p><u>Description:</u></p> <p>Specifies the unique ID of a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process instance ID</p>

Output

Array of WFMailDeliverable objects.

Example

```
//Process Instance ID associated with the Process Instance.
IWFFrameworkService svc = GetWorkflowService();
string processInstanceID = ...// for example, "1e3d514d43d3465cae6ec3bbbd409168";

try
{
    WFMailDeliverable[] emailNotifications =
    svc.GetMailDeliverables(processInstanceID);
    foreach(WFMailDeliverable m in emailNotifications)
    {
        Console.WriteLine("Mail ID: '{0}'", m.ID);
        Console.WriteLine("Process Instance ID: '{0}'", m.ProcInstID);
        Console.WriteLine("E-Mail Subject: '{0}'", m.Mail.Subject);
    }
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Resend Mail Deliverable

API Type

Web Services

Description

Resends the mail deliverable with a specified mail ID.

Syntax

```
public virtual void ResendMailDeliverable(string mailID)
```

Good to Know

- If specified mail ID is already in the e-mail queue to be sent, making this API call does not send another e-mail.
- This API call must be made once for a particular mail ID and wait until the e-mail is sent.
- To call this method, the API account user must have the [access right](#), **Resend and Cancel an Email Notification**.

Parameters

Name	Description
mailID	<p><u>Description:</u></p> <p>Specifies the unique ID for an email notification.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid email notification ID.</p>

Output

None.

Example

```
//Sample for using resendMailDeliverable
IWFFrameworkService svc = GetWorkflowService();
string mailID = ... // for example,
"149C3974240F47D3B28EB6D4A3CDCD3F"

try
{
    svc.ResendMailDeliverable(mailID);
}

catch (Exception ex)
```

```
{  
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));  
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Methods for Events

This section describes service calls related to workflow events in the Web Service API.

Get Event

API Type

Web Services

Description

Retrieves an [event](#) object.

Good to Know

- This method is usually used to check if a service call has been completed.

Syntax

```
public virtual WFEvent GetEvent(string eventID)
```

Parameters

Name	Description
eventID	<p><u>Description:</u></p> <p>Specifies a unique ID for an event.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A unique, 32-character ID.</p>

Output

WFEvent object.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string eventID = ...// for example, "049C3974240F47D3BA8EB6D4A3CDCD3F";

try
{
    WFEvent evt = _workflowAPI.GetEvent(eventID);
    Console.WriteLine("Event ID: '{0}'", evt.EventID);
    Console.WriteLine("Event Name: '{0}'", evt.EventName);
    Console.WriteLine("Event Status: '{0}'", evt.Status);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Send Mail

This section describes service calls related to sending email using AgilePoint.

Send Mail

API Type

Web Services

Description

Sends an email through [AgilePoint Server](#).

Good to Know

- If the variables WorkItemID and ProcesInstanceID are not provided, [process attributes](#) (custom attributes) are not resolved.

Syntax

```
public virtual void SendMail(String To, String CC, String Subject, String Body)
```

Parameters

Name	Description
To	<p><u>Description:</u></p> <p>Specifies the To portion of an email.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that includes one or more email addresses in SMTP format.</p>
CC	<p><u>Description:</u></p> <p>Specifies the CC portion of an email.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that includes one or more email addresses in SMTP format.</p>
Subject	<p><u>Description:</u></p> <p>The subject of an email.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>One line of text (a string).</p> <p><u>Accepted:</u></p>

Name	Description
	<ul style="list-style-type: none"> • Letters • Numbers • Spaces
Body	<p><u>Description:</u></p> <p>Specifies the body portion of an email.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>Free text.</p>

Output

None.

Example

```
//Sample for using Workflow.SendMail
IWFFrameworkService svc = GetWorkflowService();

try
{
    // email recipients
    string To = "bill@tusca.com";

    // CC
    string CC = "bob@tusca.com";

    //Subject of the Mail
    string Subject = "This is email Subject";

    //Body of the Mail
    string Body = "This email Body";
    svc.SendMail(To, CC, Subject, Body);
}
```

```
catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Send Mail (Extended Method)

API Type

Web Services

Description

Sends an email through AgilePoint Server. The extended method enables you to send attachments.

Syntax

```
public virtual void SendMailEx(String From, String To, String CC, String Subject, String Body , String Attachments)
```

Parameters

Name	Description
From	<p><u>Description:</u></p> <p>Specifies the From portion of an email.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that includes one or more email addresses in SMTP format.</p>
To	<p><u>Description:</u></p> <p>Specifies the To portion of an email.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that includes one or more email addresses in SMTP format.</p>
CC	<p><u>Description:</u></p> <p>Specifies the CC portion of an email.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that includes one or more email addresses in SMTP format.</p>
Subject	<p><u>Description:</u></p> <p>The subject of an email.</p> <p><u>Type</u></p>

Name	Description
	<p><code>string</code></p> <p><u>Allowed Values:</u></p> <p>One line of text (a string).</p> <p>Accepted:</p> <ul style="list-style-type: none"> • Letters • Numbers • Spaces
Attachments	<p><u>Description:</u></p> <p>File attachments included with the email.</p> <p><u>Type</u></p> <p><code>string</code></p> <p><u>Allowed Values:</u></p> <p>A valid path and file name on the AgilePoint Server.</p> <p>This parameter must use a file path from the file system (for example, C:\file.txt) on the machine where AgilePoint Server is installed.</p> <p>If there is no attachment, you can pass null or String.Empty.</p>

Output

None.

Example

```
//Sample for using Workflow.SendMailEx
IWFWorkflowService svc = GetWorkflowService();

try
{
    string From = "john@tusca.com"; // or <Full Name>"email address"
    string To = "bill@tusca.com";
    string CC = "bob@tusca.com";
}
```

```
string Subject = "Mail Subject";
string Body = "Mail Body";
string Attachments = "c:\\Temp\\Tempdoc.doc";

//Send Mail
svc.SendMailEx(From, To, CC, Subject, Body, Attachments);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

4.6 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Send Mail (Extended Method with Priority)

API Type

Web Services

Description

Sends an email through AgilePoint Server. The extended method enables you to send attachments.

Syntax

```
public virtual void SendMailEx(String From, String To, String CC, String Subject, String Body , String Attachments, Enum priority)
```

Parameters

Name	Description
From	<p><u>Description:</u> Specifies the From portion of an email.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A string that includes one or more email addresses in SMTP format.</p>
To	<p><u>Description:</u> Specifies the To portion of an email.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A string that includes one or more email addresses in SMTP format.</p>
CC	<p><u>Description:</u> Specifies the CC portion of an email.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A string that includes one or more email addresses in SMTP format.</p>

Name	Description
Subject	<p><u>Description:</u></p> <p>The subject of an email.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>One line of text (a string).</p> <p>Accepted:</p> <ul style="list-style-type: none"> • Letters • Numbers • Spaces
Attachments	<p><u>Description:</u></p> <p>File attachments included with the email.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid path and file name on the AgilePoint Server.</p> <p>This parameter must use a file path from the file system (for example, C:\file.txt) on the machine where AgilePoint Server is installed.</p> <p>If there is no attachment, you can pass null or String.Empty.</p>
priority	<p><u>Description:</u></p> <p>The email's priority.</p> <p><u>Type</u></p> <p>enum</p> <p><u>Allowed Values:</u></p>

Name	Description
	<ul style="list-style-type: none">• high• normal• low

Output

None.

Example

```
//Sample for using Workflow.SendMailEx
IWFWorkflowService svc = GetWorkflowService();

try
{
    string From = "john@tusca.com"; // or <Full Name>"email address"
    string To = "bill@tusca.com";
    string CC = "bob@tusca.com";
    string Subject = "Mail Subject";
    string Body = "Mail Body";
    string Attachments = "c:\\Temp\\Tempdoc.doc";
    enum priority = "high";

    //Send Mail
    svc.SendMailEx(From, To, CC, Subject, Body, Attachments);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

5.0 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Methods for Custom Attributes

This section describes service calls related to custom attributes.

Get Custom Attribute

API Type

Web Services

Description

Retrieves a [process attribute](#) (custom attribute) with the specified process attribute ID and process attribute name.

Syntax

```
public virtual object GetCustomAttr(string customID, string attrName);
```

Parameters

Name	Description
customID	<p><u>Description:</u></p> <p>A work object ID specified within a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>One valid work object ID.</p>
attrName	<p><u>Description:</u></p> <p>The name of an item, such as a property or attribute in a name/value pair.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid name.</p>

Output

Custom attribute value (can be string, integer, float, double, bool, and/or DateTime).

Example

```
IWFWorkflowService svc = GetWorkflowService();
string customID = ...// for example, "013933F128C3415F81D6F545594D4CB6";
string attrName = ...// for example, "/pd:myFields/pd:Name" or "Approval"

try
{
    Object obj = svc.GetCustomAttr(customID, attrName);
}
}
```

```
catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Custom Attributes

API Type

Web Services

Description

Retrieves a collection of custom attributes for a specified custom ID in XML format.

Syntax

```
public virtual string GetCustomAttrs(string customID)
```

Parameters

Name	Description
customID	<p><u>Description:</u></p> <p>A work object ID specified within a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>One valid work object ID.</p>

Output

string that contains all the attributes for the custom ID in XML format.

Example

```
// get custom attributes in xml format
IWfWorkflowService svc = GetWorkflowService();
string customID = ...// for example, "013933F128C3415F81D6F545594D4CB6";

try
{
    string xml = svc.GetCustomAttrs(sessionID, string processInstanceID);
    Console.WriteLine("AttributeXMLstring={0}", resultAttrXML);
    WfCustomAttributes attrs = new WfCustomAttributes();
    attrs.AttrXml = xml; // de-serialize xml
    string[] attributeNames = attrs.GetNames();// get attribute names
    Object value = attrs["MyAttributeName"]; // retrieve attribute value
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Custom Attributes (Extended Method)

API Type

Web Services

Description

Retrieves all custom attributes for a specified a set of custom IDs.

Good to Know

- Values of IDs must be separated by a semicolon (;).

Syntax

```
public virtual KeyValue[] GetCustomAttrsEx(string[] customIDs)
```

Parameters

Name	Description
customIDs	<p><u>Description:</u></p> <p>Multiple work object IDs specified within a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>An array of valid work object IDs.</p>

Output

An array of KeyValue values. The key is a custom ID, and value is a string of custom attributes in XML format.

Example

```
IWFWorkflowService svc = GetWorkflowService();

//Array of custom ID
string[] customIDs = ...// for example,
{"InfoPath:011eaf6c46ac4723b25b4db5772d9912", ...};

try
{
    KeyValue[] keyValues = svc.GetCustomAttrsEx(customIDs);
    foreach (KeyValue kv in keyValues)
    {
        Console.WriteLine("ID: '{0}'", kv.Key);
        Console.WriteLine("Custom Attributes XML: '{0}'", kv.Value);
        WFCustomAttributes attrs = new WFCustomAttributes();
        Attrs.AttrXml = kv.Value;
    }
}

catch (Exception ex)
```

```
{  
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));  
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Remove Custom Attribute

API Type

Web Services

Description

Removes a [process attribute](#) (custom attribute) with the specified custom attribute ID and process attribute name.

Syntax

```
public virtual void RemoveCustomAttr(string customID, string attributeName)
```

Parameters

Name	Description
customID	<p><u>Description:</u></p> <p>A work object ID specified within a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>One valid work object ID.</p>
attributeName	<p><u>Description:</u></p> <p>The name of the process instance attribute you want.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process instance attribute.</p>

Output

None.

Example

```
//Sample for using Workflow.RemoveCustomAttr
IWFWorkflowService svc = GetWorkflowService();
string customID = ..// for example, "InfoPath:01leaf6c46ac4723b25b4db5772d9912"
string attributeName = ...// for example, "//pd:purchaseOrder/pd:secondApproval"

try
{
    svc.RemoveCustomAttr(customID, attributeName);
}
```

```
catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Remove Custom Attributes

API Type

Web Services

Description

Removes multiple [process attribute](#) (custom attributes) with the specified custom attribute ID and process variable names.

Good to Know

- The process variable names can be specified as a semicolon (;) delimited list.

Syntax

```
public virtual void RemoveCustomAttrs(string customID, string[] namesArray)
```

Parameters

Name	Description
customID	<p><u>Description:</u></p> <p>A work object ID specified within a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>One valid work object ID.</p>
namesArray	<p><u>Description:</u></p> <p>The the names of one or more items, such as a properties or attributes.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>An array of valid names.</p>

Output

None.

Example

```
//Sample for using Workflow.RemoveCustomAttr
IWfWorkflowService svc = GetWorkflowService();
string customID = ...// for example,
"InfoPath:011eaf6c46ac4723b25b4db5772d9912"
```

```
// array of attribute name, for example,  
// new string[] { // "//pd:purchaseOrder/pd:secondApproval", ... };  
string[] namesArray = ...  
  
try  
{  
    svc.RemoveCustomAttrs(customID, namesArray);  
}  
  
catch (Exception ex)  
{  
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));  
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Set Custom Attribute

API Type

Web Services

Description

Sets the name and value for a custom attribute for a specified custom ID.

Syntax

```
public virtual SetCustomAttr(String customID, String attributeName, object val)
```

Parameters

Name	Description
customID	<p><u>Description:</u></p> <p>A work object ID specified within a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>One valid work object ID.</p>
attributeName	<p><u>Description:</u></p> <p>The name of the process instance attribute you want.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process instance attribute.</p>
val	<p><u>Description:</u></p> <p>The value of the custom attribute.</p> <p><u>Type</u></p> <p>object</p> <p><u>Allowed Values:</u></p>

Name	Description
	A valid Types object.

Output

None.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string customID = ..// for example,
"InfoPath:011leaf6c46ac4723b25b4db5772d9912"
string attributeName = ...// for example,
"//pd:purchaseOrder/pd:secondApproval"
bool val = true;

try
{
    svc.SetCustomAttr(customID, attributeName, val);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Set Custom Attributes

API Type

Web Services

Description

Sets a list of [process attribute](#) (custom attributes) to the specified values for a specified custom attribute ID. The list of process variables must be specified as a Name-Value array.

Syntax

```
public virtual void SetCustomAttrs(string customID, NameValue[] nameValues)
```

Parameters

Name	Description
customID	<p><u>Description:</u></p> <p>A work object ID specified within a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>One valid work object ID.</p>
attributes	<p><u>Description:</u></p> <p>Name-value pairs associated with a custom ID.</p>

Name	Description
	<p><u>Type</u></p> <p>NameValue</p> <p><u>Allowed Values:</u></p> <p>A valid custom ID with an associated name.</p>

Output

None.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string customID = ..// for example, "InfoPath:011eaf6c46ac4723b25b4db5772d9912"

NameValue[] attributes = new NameValue[]
{
    new NameValue("CustomAttributeName1", "CustomAttributevalue1"),
    new NameValue("CustomAttributeName2", false),
    new NameValue("CustomAttributeName3", 10.0)
};

try
{
    svc.SetCustomAttrs(customID, attributes);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Methods for Archiving and Restoring Processes

This section describes service calls related to archiving and restoration of processes.

Archive Process Instance

API Type

Web Services

Description

[Archives](#) a [process instance](#) based on the specified process instance identifier by moving the set of process instance records from the workflow database to the archive database.

Good to Know

- The process instance records and all of the associated data are deleted from the AgilePoint database.
- The process instance to be archived must be in [Completed](#) or [Cancelled status](#).

- If you archive a process instance, and find that the same process instance was archived, it first deletes the old instance record from the archive database before archiving.
- To call this method, the API account user must have the [access right, Archive and restore process\(es\)](#).

Syntax

```
public virtual void ArchiveProcInst(string processInstanceID)
```

Parameters

Name	Description
processInstanceID	<p><u>Description:</u></p> <p>Specifies the unique ID of a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process instance ID</p>

Output

None.

Example

```
// this is console application sample
IWfWorkflowService svc = GetWorkflowService();
string processInstanceID = ..// the ID of process instance to be
    archived

try
{
    svc.ArchiveProcInst(processInstanceID);
}

catch (Exception ex)
{
```

```
Console.WriteLine("Failed!, {0}", ShUtil.GetSoapMessage(ex));  
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Restore Process Instance

API Type

Web Services

Description

Restores a [process instance](#) and associated data from the archive database to [AgilePoint Server](#).

Good to Know

- The process instance records are written to the AgilePoint database, but also retained in the AgilePoint archive database.
- To call this method, the API account user must have the [access right](#), **Archive and restore process(es)**.

Syntax

```
public virtual void RestoreProcInst(string processInstanceId)
```

Parameters

Name	Description
processInstanceId	<p><u>Description:</u></p> <p>Specifies the unique ID of a process instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid process instance ID</p>

Output

None.

Example

```
// this is console application sample code
IWfWorkflowService svc = GetWorkflowService();
string processInstanceId = ..// the ID of process instance to be
restored.

try
{
    svc.RestoreProcInst(processInstanceId);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex) );
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Query Archived Process Instances

API Type

Web Services

Description

Retrieves process instances that match a SQL query.

Syntax

```
public virtual WFBaseProcessInstance[] QueryArchivedProclnsts(string sqlWhereClause)
```

Parameters

Name	Description
sqlWhereClause	<u>Description:</u> The where clause of the SQL statement you want to query.

Name	Description
	<p data-bbox="508 285 570 317"><u>Type</u></p> <p data-bbox="508 348 578 380">string</p> <p data-bbox="508 432 699 464"><u>Allowed Values:</u></p> <p data-bbox="508 495 954 527">The where clause of a SQL statement.</p>

Output

An array of WFBaseProcessInstance objects.

Example

```

// Build SQL Statement
string processDefinitionName = "Budget Request Approval Process";
string sqlWhereClause = string.Format("DEF_NAME = '{0}'",
processDefinitionName);

try
{
    WFBaseProcessInstance[] archivedProcessInstances =
    svc.QueryArchivedProcInsts(sqlWhereClause)
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}

```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Group, Role, and Rights

This section describes service calls related to the groups, roles, and rights.

Add Group

API Type

Web Services

Description

Creates a [group](#) in AgilePoint NX, and sets the ResponsibleUser parameter as the [group lead](#).

Syntax

```
public virtual WFGGroup AddGroup(string GroupName, string Description, string ResponsibleUser, bool Enabled)
```

Parameters

Name	Description
GroupName	<p><u>Description:</u></p> <p>Specifies the name of a group.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p>

Name	Description
	A valid group name.
Description	<p><u>Description:</u></p> <p>A description of an entity.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that can contain spaces and special characters.</p>
ResponsibleUser	<p><u>Description:</u></p> <p>The user name for the responsible user of this group.</p> <p>The responsible user must be a registered AgilePoint user.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name.</p>
Enabled	<p><u>Description:</u></p> <p>Enables or disables an entity.</p> <p><u>Type</u></p> <p>bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - Enables the entity. • False - Disables the entity.

Output

WFGroup object represented the group that is added.

Example

```
IWFAdminService svc = GetAdminService();
string GroupName = ...;
string Description = ...;
string ResponsibleUser = @"[Domain Name]\[Account Name]", // Group
Lead User Name
bool Enabled = true;

try
{
    WFGroup group = svc.AddGroup(GroupName, Description,
    ResponsibleUser, Enabled);
}

catch( Exception ex )
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Add Group Member

API Type

Web Services

Description

Adds a user to the specified AgilePoint [group](#).

Syntax

```
public virtual WFGroupMember AddGroupMember(string GroupName, string UserName, string Description, string ClientData, bool Enabled)
```

Parameters

Name	Description
GroupName	<p><u>Description:</u> Specifies the name of a group.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid group name.</p>
UserName	<p><u>Description:</u> The user name for the user.</p>

Name	Description
	<p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name for a registered AgilePoint user.</p>
Description	<p><u>Description:</u></p> <p>A description of an entity.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that can contain spaces and special characters.</p>
ClientData	<p><u>Description:</u></p> <p>Specifies the client data, which identifies a client for AgilePoint Server.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that contains the client data.</p> <p>If this value is null, the system will keep existing client data. Otherwise the relevant data is overwritten.</p>
Enabled	<p><u>Description:</u></p> <p>Enables or disables an entity.</p> <p><u>Type</u></p> <p>bool</p> <p><u>Allowed Values:</u></p>

Name	Description
	<ul style="list-style-type: none">• True - Enables the entity.• False - Disables the entity.

Output

WFGroupMember object that contains the data for the new group member.

Example

```
IWFAdminService svc = GetAdminService();
string GroupName = ...;
string Description = ...;
string UserName = @"[Domain Name]\[Account Name]", // Group Lead User Name
string ClientData = null;
bool Enabled = true;

try
{
    WFGroup group = svc.AddGroupMember(GroupName, UserName, Description,
    ClientData, Enabled);
}

catch( Exception ex )
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Add Role

API Type

Web Services

Description

Creates a [role](#) in AgilePoint NX, and specifies the [access right](#) for the users assigned to the role.

Syntax

```
public virtual WFRole AddRole(String RoleName, String Description, int[] Rights, bool Enabled)
```

Parameters

Name	Description
RoleName	<p><u>Description:</u></p> <p>The name of a role.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid role name.</p>
Description	<p><u>Description:</u></p> <p>A description of an entity.</p>

Name	Description
	<p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that can contain spaces and special characters.</p>
Rights	<p><u>Description:</u></p> <p>Specifies the rights that are assigned to the Role.</p> <p><u>Type</u></p> <p>WFAccessRights</p> <p><u>Allowed Values:</u></p> <p>An array of index values. See the table for appropriate indexes.</p> <p>WFAccessRights provides the enums for rights. See the sample code for more information.</p>
Enabled	<p><u>Description:</u></p> <p>Enables or disables an entity.</p> <p><u>Type</u></p> <p>bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - Enables the entity. • False - Disables the entity.

Access Rights

Index	Rights
0	Register and modify the user information

Index	Rights
1	Unregister a user
2	Add and modify role information
3	Remove a role
4	Add and modify group information
5	Remove a group
6	Modify and view system information
7	Add a process definition
8	Check in and check out a process definition
9	Delete or disable a process definition
10	Release a process definition
11	Initiate a process
12	Suspend and resume a process
13	Resend and cancel an email notification
14	Cancel a process
15	Rollback a process
16	Reassign a task
17	Cancel a task
18	Create a task
19	Add, remove and modify delegation
20	Add, remove and modify report configuration
21	Achieve and restore processes
22	Add, remove and modify shared custom attributes
23	View process details

Output

WFRole object for the role that is added.

Example

```
IWFAdminService svc = GetAdminService();
string RoleName = ...// for example, "Process Manager";
string Description = ...;
/*integer array specifying the access rights for this role.
* 11 - Initiate a process
* 14 - Cancel a process
* 23 - view process details
*/
int[] Rights =
    {
        WFAccessRights.InitiateProcessInstance,
        WFAccessRights.CancelProcessInstance,
        WFAccessRights.ViewProcessDetails
    };

try
{
    WFRole role = svc.AddRole(RoleName, Description, Rights, True);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Add Role Member

API Type

Web Services

Description

Adds a user or [group](#) to the specified [role](#).

Good to Know

- This user or group inherits the [access right](#) specified at the role level.

Syntax

```
public virtual WFRoleMember AddRoleMember(string RoleName, string Assignee, string AssigneeType, string ClientData, string ObjectID, string ObjectType);
```

Parameters

Name	Description
RoleName	<p><u>Description:</u></p> <p>The name of a role.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p>

Name	Description
	A valid role name.
Assignee	<p><u>Description:</u></p> <p>The name of the assignee.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name or group name, depending upon the assignee type.</p>
AssigneeType	<p><u>Description:</u></p> <p>The type for the assignee for the task.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • User - Assigns the task to a user. • Group - Assigns the task to a group.
ClientData	<p><u>Description:</u></p> <p>Specifies the client data, which identifies a client for AgilePoint Server.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that contains the client data.</p> <p>If this value is null, the system will keep existing client data. Otherwise the relevant data is overwritten.</p>
ObjectID	<p><u>Description:</u></p>

Name	Description
	Reserved for future use. <u>Type</u> string <u>Allowed Values:</u> A null value.
ObjectType	<u>Description:</u> Reserved for future use. <u>Type</u> string <u>Allowed Values:</u> A null value.

Output

WFRoleMember object.

Example

```

IWFAdminService svc = GetAdminService();

string RoleName = "Administrative";
string Assignee = @"Demo3\Administrator";
string AssigneeType = "User";

try
{
    WFRoleMember member = svc.AddRoleMember(RoleName, Assignee,
    AssigneeType, "", null, null);
}

catch (Exception ex)
{

```

```
Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));  
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Enabled Group Member

API Type

Web Services

Description

Enables or disables a user as a member of a group.

Syntax

```
public virtual WFGroupMember EnabledGroupMember(string GroupName, string userName, bool Enabled)
```

Parameters

Name	Description
GroupName	<p><u>Description:</u> Specifies the name of a group.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid group name.</p>
userName	<p><u>Description:</u> The user name for the user.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid user name for a registered AgilePoint user.</p>
Enabled	<p><u>Description:</u> Enables or disables an entity.</p> <p><u>Type</u> bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none">• True - Enables the entity.• False - Disables the entity.

Output

WFGroupMember object.

Example

```
IWFAdminService svc = GetAdminService();

string GroupName = "Administrative";
string userName = @"Demo3\Administrator";
bool Enabled =false;

try
{
    svc.EnabledGroupMember(GroupName, userName, Enabled);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Access Right Names

API Type

Web Services

Description

Retrieves the names of all the access rights in the AgilePoint system.

Syntax

```
public virtual string[] GetAccessRightNames()
```

Parameters

Name	Description
None	Not Applicable

Output

An array of strings that contain the names of all the access rights for the system.

Example

```
IWFAdminService svc = GetAdminService();

try
{
    string[] permissionNames = svc.GetAccessRightNames();
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Access Rights

API Type

Web Services

Description

Retrieves the access rights for a specified user.

Syntax

```
public virtual int[] GetAccessRights(string userName)
```

Parameters

Name	Description
userName	<u>Description:</u> The user name for the user.

Name	Description
	<p data-bbox="509 285 570 317"><u>Type</u></p> <p data-bbox="509 348 578 380">string</p> <p data-bbox="509 432 699 464"><u>Allowed Values:</u></p> <p data-bbox="509 495 1097 527">A valid user name for a registered AgilePoint user.</p>

Output

An array of integers that represent the access rights of the user.

Example

```
IWFAdminService svc = GetAdminService();
string userName = ...// for example, "Demo3\\ap_svc"

try
{
    string[] accessNames = svc.GetAccessRightNames();
    int[] userRights = svc.GetAccessRights(userName);
    Console.WriteLine("The user has the following rights:");
    foreach (int rightCode in userRights)
    {
        Console.WriteLine(accessNames[rightCode]);
    }
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
/*
```

This example produces the following results:

```
The user has the following rights:
Register and modify the user information
Unregister a user
Add and modify the role information
Remove a role
Add and modify the group information
Remove a group
```

```
Modify/View the system configuration
Add a process definition
Checkin and checkout a process definition
Delete and disable a process definition
...
*/
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Group

API Type

Web Services

Description

Retrieves a [group](#) object with the specified AgilePoint group name.

Syntax

```
public virtual WFGGroup GetGroup(string groupName)
```

Parameters

Name	Description
groupName	<p><u>Description:</u></p> <p>Specifies the name of a group.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid group name.</p>

Output

WFGroup object. If the specified group does not exist, the output is null.

Example

```
IWFAdminService svc = GetAdminService();
string groupName = ...// for example, "Administrators";

try
{
    WFGroup grpInfo = svc.GetGroup(groupName);
    Console.WriteLine("Group Name:{0}; Group Lead:{1}", grpInfo.Name,
        grpInfo.ResponsibleUser);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
/*
This example produces the following results:
Group Name:Administrators; Group Lead:\Administrator
*/
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Group Members

API Type

Web Services

Description

Retrieves all the user members associated with the specified AgilePoint [group](#).

Syntax

```
public virtual WFGroupMember[] GetGroupMembers(string groupName)
```

Parameters

Name	Description
groupName	<u>Description:</u> Specifies the name of a group.

Name	Description
	<p data-bbox="508 287 570 319"><u>Type</u></p> <p data-bbox="508 350 578 382">string</p> <p data-bbox="508 434 699 466"><u>Allowed Values:</u></p> <p data-bbox="508 497 743 529">A valid group name.</p>

Output

An array of WFGroupMember objects.

Example

```

IWFAdminService svc = GetAdminService();
string groupName = ...// for example, "Administrators";

try
{
    WFGroupMember[] grpMembers = adminService.GetGroupMembers(groupName);
    Console.WriteLine("AgilePoint Group {0} has {1} members:",
        groupName, grpMembers.Length);
    foreach (WFGroupMember grpMember in grpMembers)
    {
        Console.WriteLine("Member Name:{0}", grpMember.Member);
    }
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}

/*
This example produces the following results:
AgilePoint Group Administrators has 3 members:
Member Name:\Administrator
Member Name:VITBDC\yuvarajn
Member Name:Demo3\manager
*/

```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Groups

API Type

Web Services

Description

Retrieves all the [group](#) objects in the AgilePoint system.

Syntax

```
public virtual WfGroup[] GetGroups()
```

Parameters

Name	Description
None	Not Applicable

Output

An Array of WFGGroup objects.

Example

```
IWFAdminService svc = GetAdminService();

try
{
    WFGGroup[] apGroups = svc.GetGroups();
    Console.WriteLine("AgilePoint Group {0} has {1} members:",
        groupName, grpMembers.Length);
    Console.WriteLine("AgilePoint Groups:");
    foreach(WFGGroup grp in apGroups)
    {
        System.Console.WriteLine("Name:{0};Group Lead:{1} ", grp.Name,
            grp.ResponsibleUser);
    }
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}

/*
This example produces the following results:
AgilePoint Groups:
Name:Administrators;Group Lead:\Administrator
Name:Domain Users;Group Lead:VITBDC\amarnv
*/
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are

outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Role

API Type

Web Services

Description

Retrieves a [role](#) object by the AgilePoint role name.

Syntax

```
public virtual WFRole GetRole(string roleName);
```

Parameters

Name	Description
roleName	<p><u>Description:</u></p> <p>The name of a role.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid role name.</p>

Output

WFRole object with the specified role name.

Example

```
IWFAdminService svc = GetAdminService();
string rolName = ...// for example, "Administrators"

try
{
    WFRole role = adminService.GetRole(rolName);
    Console.WriteLine("Name = '" + role.Name + "' Description = '" +
        role.Description + "'");
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}

/* This example produces the following results:
Name = 'Administrators' Description = 'Administrators have complete access to
maintain system' */
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Roles

API Type

Web Services

Description

Retrieves a list of all [roles](#) in the AgilePoint system.

Syntax

```
public virtual WFRole[] GetRoles()
```

Parameters

Name	Description
None	Not Applicable

Output

An array of WFRole objects that includes all roles.

Example

```
IWFAdminService svc = GetAdminService();  
  
try  
{  
    WFRole[] roles = adminService.GetRoles();  
}
```

```
foreach (WFRole role in roles)
{
    Console.WriteLine("Name = '{0}', Description = '{1}'",role.Name,
        role.Description);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}

/* This example produces the following results:
Name = 'Administrators' Description = 'Administrators have complete
access to maintain system'
Name = 'Process Runtime Managers' Description = 'Process Runtime
Managers have complete access to manage runtime processes'
Name = 'process definition Designers' Description = 'process
definition Designers have complete access to add, modify and remove process
templates' */
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Query Role Members

API Type

Web Services

Description

Retrieves the members assigned to a [role](#) that match a specified [WHERE clause](#) of an SQL query expression.

Good to Know

- Only pass the WHERE clause without WHERE keyword, not the whole query.

Syntax

```
public virtual WFRoleMember[] QueryRoleMembers(string roleName, string sqlWhereClause)
```

Parameters

Name	Description
roleName	<p><u>Description:</u></p> <p>The name of a role.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid role name.</p>
sqlWhereClause	<p><u>Description:</u></p> <p>The where clause of the SQL statement you want to query.</p> <p><u>Type</u></p> <p>string</p>

Name	Description
	<u>Allowed Values:</u> The where clause of a SQL statement.

Output

Returns array of WFRoleMember members of the role that match the specified SQL statement.

Example

```
IWFAdminService svc = GetAdminService();
string roleName = ...// for examples, "Users"
string sqlWhereClause = ...// for example, "ASSIGNEE_TYPE='User'"

try
{
    WFRoleMember[] roleMembers = svc.QueryRoleMembers(roleName, sqlWhereClause);
    foreach (WFRoleMember member in roleMembers)
    {
        Console.WriteLine("Assignee = '{0}', Created Date = '{1}'",
            member.Assignee,
            member.CreatedDate.ToShortDateString());
    }
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}

/*
This example produces the following results:
Assignee = 'DEMO\\CONTROLLER' & Created Date = '9/18/2009'
Assignee = 'DEMO\\CFO' & Created Date = '9/18/2009'
*/
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Remove Group

API Type

Web Services

Description

Permanently removes an AgilePoint [group](#) with the specified group name.

Syntax

```
public virtual void RemoveGroup(string groupName)
```

Parameters

Name	Description
groupName	<p><u>Description:</u></p> <p>Specifies the name of a group.</p> <p><u>Type</u></p> <p>string</p>

Name	Description
	<u>Allowed Values:</u> A valid group name.

Output

None.

Example

```
IWFAdminService svc = GetAdminService();
string groupName = ...// for examples, "Users"

try
{
    svc.RemoveGroup(groupName);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Remove Group Member

API Type

Web Services

Description

Permanently removes a user member from a specified [group](#).

Good to Know

- If the user has a group task assigned, the user continues to see the task until another group member takes assignment for the task.
- The exception is a Dynamic Group task, which is immediately removed from the user's task list.

Syntax

```
public virtual void RemoveGroupMember(string GroupName, string UserName)
```

Parameters

Name	Description
GroupName	<p><u>Description:</u></p> <p>Specifies the name of a group.</p> <p><u>Type</u></p> <p>string</p>

Name	Description
	<p><u>Allowed Values:</u></p> <p>A valid group name.</p>
UserName	<p><u>Description:</u></p> <p>The user name for the user.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name for a registered AgilePoint user.</p>

Output

None.

Example

```

IWFAdminService svc = GetAdminService();
string GroupName = ...// for examples, "Engineers"
string UserName = ...// for example, @"VIT\ct-002"

try
{
    svc.RemoveGroupMember(GroupName, UserName);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}

```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Remove Role

API Type

Web Services

Description

Permanently removes a [role](#) from the AgilePoint system.

Syntax

```
public virtual void RemoveRole(string roleName)
```

Parameters

Name	Description
roleName	<p><u>Description:</u></p> <p>The name of a role.</p> <p><u>Type</u></p> <p>string</p>

Name	Description
	<u>Allowed Values:</u> A valid role name.

Output

None.

Example

```
IWFAdminService svc = GetAdminService();
string roleName = ...// for example, "Engineers"

try
{
    svc.RemoveRole (roleName);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Remove Role Member

API Type

Web Services

Description

Permanently removes a user or [group](#) member from a [role](#).

Good to Know

- If the user has a group task assigned, the user continues to see the task until another group member takes assignment for the task.
- The exception is a Dynamic Group task, which is immediately removed from the user's task list.

Syntax

```
public virtual void RemoveRoleMember(string RoleName, string Assignee, string AssigneeType, string ObjectID)
```

Parameters

Name	Description
RoleName	<p><u>Description:</u></p> <p>The name of a role.</p> <p><u>Type</u></p> <p>string</p>

Name	Description
	<p><u>Allowed Values:</u> A valid role name.</p>
Assignee	<p><u>Description:</u> The name of the assignee.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid user name or group name, depending upon the assignee type.</p>
AssigneeType	<p><u>Description:</u> The type for the assignee for the task.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • User - Assigns the task to a user. • Group - Assigns the task to a group.
ObjectID	<p><u>Description:</u> Reserved for future use.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A null value.</p>

Output

None.

Example

```
IWFAdminService svc = GetAdminService();
string RoleName = ...// for example, "Engineers"
string Assignee = ...// for example, @"VIT\ct-002"

try
{
    svc.RemoveRoleMember(RoleName, Assignee, "User", null);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Update Group

API Type

Web Services

Description

Updates a [group](#) with a new description or [group lead](#).

Syntax

```
WFGroup UpdateGroup(string GroupName, string Description, string ResponsibleUser, bool Enabled)
```

Parameters

Name	Description
GroupName	<p><u>Description:</u> Specifies the name of a group.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid group name.</p>
Description	<p><u>Description:</u> A description of an entity.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A string that can contain spaces and special characters.</p>
ResponsibleUser	<p><u>Description:</u></p>

Name	Description
	<p>The user name for the responsible user of this group.</p> <p>The responsible user must be a registered AgilePoint user.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name.</p>
Enabled	<p><u>Description:</u></p> <p>Enables or disables an entity.</p> <p><u>Type</u></p> <p>bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - Enables the entity. • False - Disables the entity.

Output

An updated WFGroup.

Example

```

IWFAdminService svc = GetAdminService();
string GroupName = ...// for example, "TestGroup1"
string Description = ... // for example, "This is new description of
the user group"
string ResponsibleUser = ...// for example, "DEMO3\\Administrator"

try
{
WFGroup updatedGroup = Svc.UpdateGroup(GroupName, Description,
ResponsibleUser, true);
}

```

```
catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Update Role

API Type

Web Services

Description

Updates the description or [access right](#) selections of a [role](#).

Syntax

```
public virtual WFRole UpdateRole(string RoleName, string Description, int[] Rights, bool Enabled)
```

Parameters

Name	Description
RoleName	<p><u>Description:</u> The name of a role.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid role name.</p>
Description	<p><u>Description:</u> A description of an entity.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A string that can contain spaces and special characters.</p>
Rights	<p><u>Description:</u> Specifies the rights that are assigned to the Role.</p> <p><u>Type</u> WFAccessRights</p> <p><u>Allowed Values:</u> An array of index values. See the table for appropriate indexes. WFAccessRights provides the enums for rights. See the sample code for more information.</p>
Enabled	<p><u>Description:</u> Enables or disables an entity.</p>

Name	Description
	<p><u>Type</u></p> <p>bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none">• True - Enables the entity.• False - Disables the entity.

Output

An updated WFRole object.

Example

```
IWFAdminService svc = GetAdminService();
string RoleName = ...// for example, "TestRole"
string Description = ...// for example, "This is new description for TestRole"
List<int> list = new List<int>();
list.Add(WFAccessRights.AddModifyGroup);
list.Add(WFAccessRights.AddModifyRole);
list.Add(WFAccessRights.AddModifyUser);
list.Add(WFAccessRights.AddProcessTemplate);
list.Add(WFAccessRights.AddRemoveModifyDelegation);
int[] Rights = list.ToArray();

try
{
    WFRole updatedRole = svc.UpdateRole(RoleName, Description, Rights, true);
}

catch( Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Organization Properties

This section describes service calls related to various organization properties.

Get Organization Properties

API Type

Web Services

Description

Retrieves organization properties such as Title, Department and Location.

Syntax

```
public virtual KeyValue[] GetOrganizationProperties(string Name)
```

Parameters

Name	Description
Name	<u>Description:</u>

Name	Description
	<p>The name of an item, such as a property or attribute in a name/value pair.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid name.</p>

Output

An array of KeyValue objects.

Example

```
IWFAdminService svc = GetAdminService();
string name = ...// for example, "Departments"

try
{
    KeyValue[] organizationProperties =
    svc.GetOrganizationProperties(Name);
    foreach (KeyValue property in organizationProperties)
    {
        Console.WriteLine("Property Name = 'Department' Property Value =
        '" + property.Value + "'");
    }
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Remove Organization Properties

API Type

Web Services

Description

Removes an organization property from AgilePoint.

Syntax

```
public virtual void RemoveOrganizationProperties(string Name)
```

Parameters

Name	Description
Name	<p><u>Description:</u></p> <p>The name of an item, such as a property or attribute in a name/value pair.</p> <p><u>Type</u></p>

Name	Description
	<p data-bbox="508 266 578 296">string</p> <p data-bbox="508 348 699 378"><u>Allowed Values:</u></p> <p data-bbox="508 411 670 441">A valid name.</p>

Output

None.

Example

```
IWFAdminService svc = GetAdminService();
string Name = ...// for example, "Departments"

try
{
    svc.RemoveOrganizationProperties (Name);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Update Organization Properties

API Type

Web Services

Description

Updates organization properties in the AgilePoint the system.

Syntax

```
void UpdateOrganizationProperties(string Name, KeyValue[] list)
```

Parameters

Name	Description
Name	<p><u>Description:</u></p> <p>The name of an item, such as a property or attribute in a name/value pair.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid name.</p>
list	<p><u>Description:</u></p> <p>A list of properties.</p>

Name	Description
	<p><u>Type</u></p> <p>KeyValue</p> <p><u>Allowed Values:</u></p> <p>A list of key-value pairs.</p>

Output

None.

Example

```
IWFAdminService svc = GetAdminService();
string Name = ...// for example, "Titles"
List<KeyValue> list = new List<KeyValue>();
list.Add( new KeyValue( "MANAGER", "Manager" ) );
list.Add( new KeyValue( "REG_SALES_DIRECTOR", "Regional Sales
Director" ) );
list.Add( new KeyValue( "GENERAL_MANAGER", "General Manager" ) );
KeyValue[] properties = list.ToArray();

try
{
    adm.UpdateOrganizationProperties("Titles",properties );
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show

specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Other Web Services

This section describes other additional web services.

Query Audit Trail

API Type

Web Services

Description

Retrieves all audit trail items using a a specified [WHERE clause](#) of an SQL query expression.

Good to Know

- Only pass the WHERE clause without WHERE keyword, not the whole query.

Syntax

```
public virtual WFAuditTrailItem[] QueryAuditTrail(string where)
```

Parameters

Name	Description
where	Description:

Name	Description
	<p>The where clause of the SQL statement you want to query.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>The where clause of a SQL statement.</p>

Output

An array list of WFAuditTrailItem objects.

Example

```
// This is console application sample
IWfWorkflowService svc = GetWorkflowService();

// Checking where condition based on the CATEGORY COLUMNand PURPOSE
COLUMN .
string where = "CATEGORY = 0 AND PURPOSE='Check-in process
definition'";

try
{
    // calling the QueryAuditTrail web method it return a array of Dataset.
    WFAuditTrailItem[] result = svc.QueryAuditTrail(where);
    foreach (WFAuditTrailItem item in result)// Iterating through
    WFAuditTrailItem
    {
        //Displaying the result on the console.
        System.Console.WriteLine("ItemCategory: {0} ,ItemDateOccurred: {1}",
            item.Category,item.DateOccurred);
        System.Console.WriteLine("ItemDescription: {0} ,ItemObjectID: {1}",
            item.Description, item.ObjectID);
        System.Console.WriteLine("ItemPurpose: {0} ,ItemPerformer: {1}",
            item.Purpose item.Performer);
    }
}

catch (Exception ex)
```

```
{  
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));  
}
```

Supported Versions

4.0.1 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Query Database

API Type

Web Services

Description

Queries the database with a valid SQL query and returns the dataset as a string in the XML format.

Good to Know

- If **trustedAuthorization** is set to **False**, this method will not execute unless a special access control is turned on.

For more information, refer to [Why Do I Get an Access Denied Error for an API Method Call?](#)

Syntax

```
public virtual string QueryDatabase(string sql)
```

Parameters

Name	Description
sql	<p><u>Description:</u></p> <p>The where clause of the SQL statement you want to query.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>The where clause of a SQL statement.</p>

Output

An XML string that contains the dataset with the results of the database query.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string sql = "SELECT * FROM WF_AUDIT_TRAILS where CATEGORY = 0 AND
PURPOSE='Check-in process definition'";

try
{
    // calling the QueryDatabase web method and passing the sql query as the
    argument.
    string xml = svc.QueryDatabase(sql);
    Console.WriteLine("{0}", xml); // Displaying the XML string on console.
    System.IO.StringReader sr = new System.IO.StringReader(xml);
    System.Data.DataSet ds = new System.Data.DataSet();
    ds.LoadXml(sr);
}

catch (Exception ex)
```

```
{  
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));  
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Query Database (Extended Method)

API Type

Web Services

Description

Queries the database with any valid SQL query and returns an array of 2 elements in XML string format.

Good to Know

- If **trustedAuthorization** is set to **False**, this method will not execute unless a special access control is turned on.

For more information, refer to [Why Do I Get an Access Denied Error for an API Method Call?](#).

Syntax

```
public virtual string[] QueryDatabaseEx(string sql)
```

Parameters

Name	Description
sql	<p><u>Description:</u></p> <p>The where clause of the SQL statement you want to query.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>The where clause of a SQL statement.</p>

Output

2 elements in a string array, where string[0] is an XML string with the dataset of the query results and string[1] is an XML string with the schema.

Example

```
IWFWorkflowService svc = GetWorkflowService();
string sql = "SELECT * FROM WF_AUDIT_TRAILS where CATEGORY = 0 AND
PURPOSE='Check-in process definition'";

try
{
    // calling the QueryDatabase web method and passing the sql query as the
    argument.
    string[] xmls = svc.QueryDatabase(sql);
    Console.WriteLine("{0}", xmls[0]); // Displaying the XML string on console.
    Console.WriteLine("{0}", xmls[1]); // Displaying the schema on console.
}

catch (Exception ex)
{
```

```
Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));  
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Administrative Service

This section describes service calls related to administrative functions. These calls are found under admin.wsdl.

Get All EMail Templates

API Type

Web Services

Description

Retrieves all the [global email templates](#) in XML format.

Syntax

```
private KeyValueType[] GetAllEmailTemplates()
```

Parameters

Name	Description
None	Not Applicable

Output

Returns all the global email templates.

Example

```
private KeyValue[] GetAllEmailTemplates()

{
    IWFAdminService api = GetAdminService();
    return api.GetAllEmailTemplates();
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Database Information

API Type

Web Services

Description

Retrieves the database information of the current server configuration.

Syntax

```
public virtual DatabaseInfo GetDatabaseInfo()
```

Parameters

Name	Description
None	Not Applicable

Output

A DatabaseInfo object that represents the database information of the system.

Example

```
IWFAdminService svc = GetAdminService();

try
{
    DatabaseInfo dbInfo = svc.GetDatabaseInfo();

    if (dbInfo != null)
    {
        Console.WriteLine("AgilePoint System Database Information:");
        Console.WriteLine("Vendor: {0}", dbInfo.Vendor);
        Console.WriteLine("Provider: {0}", dbInfo.Provider);
        Console.WriteLine("Connection string: {0}", dbInfo.Connstr);
    }
}
```

```
catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
/*
This example produces the following results:
AgilePoint System Database Information:
Vendor: MSSQLDatabase
Provider:
Connection string: application name=AgilePoint Server;connection
lifetime=5;min_pool
size=10;server=Demo3;database=AgilePointDB;trusted_Connection=yes
*/
```

Supported Versions

4.6 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Domain Groups

API Type

Web Services

Description

Retrieves all the [groups](#) in the [Active Directory](#) domain specified in the LDAPPPath parameter.

Good to Know

- This method is used only for AgilePoint internal purposes.

Syntax

```
public virtual KeyValue[] GetDomainGroups(string LDAPPPath, string Filter)
```

Parameters

Name	Description
LDAPPPath	<p><u>Description:</u> The LDAP path to the domain.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid LDAP path. If the value is null, the AgilePoint Server machine domain will be used.</p>
Filter	<p><u>Description:</u> A filter term for groups that is wildcard-enabled.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A string in the format (name=[my wildcard filter]).</p>

Output

Array of KeyValue objects. If the specified group is not found, the output is null.

Example

```
IWFAdminService svc = GetAdminService();
string LDAPPath = ...// for example, LDAP://ou=Sales,dc=Frabrikam,dc=com
string Filter = "A*";
Filter = string.Format("(name={0})", Filter);

try
{
    KeyValue[] grps = svc.GetDomainGroups(LDAPPath, Filter);
    foreach (KeyValue grp in grps)
    {
        Console.WriteLine("Group Name:{0}; Group Distinct Name:{1};", grp.Key,
grp.Value);
    }
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Domain Group Members

API Type

Web Services

Description

Retrieves the members of a [group](#) in the [Active Directory](#) domain.

Good to Know

- This method is used only for AgilePoint internal purposes.
- The distinguished name of the group in the LDAP format is required in the `groupDistinguishedName` parameter.

Syntax

```
public virtual DomainUser[] GetDomainGroupMembers(string groupDistinguishedName)
```

Parameters

Name	Description
groupDistinguished-Name	<p><u>Description:</u></p> <p>Specifies the name of a group in LDAP format.</p> <p><u>Type</u></p> <p><code>string</code></p>

Name	Description
	<u>Allowed Values:</u> A valid a group name in LDAP format.

Output

An array of DomainUser objects.

Example

```
IWFAdminService svc = GetAdminService();
string groupName = "Administrators";
string groupDistinguishedName = "LDAP://" + groupName;

try
{
    DomainUser[] grpUsers = svc.
    GetDomainGroupMembers(groupDistinguishedName);
    foreach (DomainUser usr in grpUsers)
    {
        Console.WriteLine("User Name:{0}; Full Name:{1};", usr.UserName,
        usr.FullName);
    }
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
/*
This example produces the following results:
User Name:Administrator; Full Name:Administrator
User Name:vitbdc/yuvaraj; Full Name:Yuvaraj Nagarajan
*/
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Domain Name

API Type

Web Services

Description

Retrieves the [Active Directory](#) domain name to which [AgilePoint Server](#) connects.

Good to Know

- This method is used only for AgilePoint internal purposes.

Syntax

```
public virtual string GetDomainName()
```

Parameters

Name	Description
None	Not Applicable

Output

A string containing the domain name of the AgilePoint Server machine.

Example

```
IWFAdminService svc = GetAdminService();

try
{
    string domainName = svc.GetDomainName();
    Console.WriteLine("AgilePoint System Domain Name={0}", domainName);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}

/* Sample of output
AgilePoint System Domain Name=LDAP://dc=Frabrikam,dc=com
*/
```

Supported Versions

4.6 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Domain Users

API Type

Web Services

Description

Retrieves all the users in the [Active Directory](#) domain specified in the LDAPPath parameter.

Good to Know

- This method is used only for AgilePoint internal purposes.

Syntax

```
public virtual DomainUser[] GetDomainUsers(string LDAPPath, string Filter)
```

Parameters

Name	Description
LDAPPath	<p><u>Description:</u></p> <p>The LDAP path to the domain.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid LDAP path.</p> <p>If the value is null, the AgilePoint Server machine domain will be used.</p>
Filter	<p><u>Description:</u></p> <p>A filter term for groups that is wildcard-enabled.</p> <p><u>Type</u></p> <p>string</p>

Name	Description
	<p><u>Allowed Values:</u></p> <p>A string in the format (name=[my wildcard filter]).</p>

Output

Array of DomainUser objects. If the specified group does not exist, the return value is null.

Example

```

IWFAdminService svc = GetAdminService();
string LDAPPath = ...// for example,
LDAP://ou=Sales,dc=Frabrikam,dc=com
string Filter = ""; // All Users

try
{
    DomainUser[] users = svc.GetDomainUsers(LDAPPath, Filter);
    foreach (DomainUser user in users)
    {
        Console.WriteLine("Full Name:{0}; Login Name:{1};", user.FullName,
user.UserName);
    }
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}

/*
This example produces the following results:
Full Name:Manager; Login Name:Demo3\Manager;
Full Name:Marcomm; Login Name:Demo3\Marcomm;
Full Name:SharePoint Administrator; Login Name:Demo3\sp_adm;
Full Name:Sujeet Kumar; Login Name:Demo3\sujeetk;
*/

```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get EMail Template

API Type

Web Services

Description

Retrieves a [global email template](#) from the server using a specified mail template ID.

Syntax

```
private string GetEMailTemplate(string processTemplateID)
```

Parameters

Name	Description
processTemplateID	<p><u>Description:</u></p> <p>The unique identifier for the process definition to be checked out for modification.</p> <p><u>Type</u></p>

Name	Description
	<code>string</code> <u>Allowed Values:</u> A valid process template ID

Output

Returns an email templates.

Example

```
private string GetEmailTemplate(string templateID)

{
    IWFAdminService api = GetAdminService();
    return api.GetEmailTemplate(templateID);
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Locale

API Type

Web Services

Description

Retrieves the default **locale** for **AgilePoint Server**.

Syntax

```
public virtual string GetLocale()
```

Parameters

Name	Description
None	Not Applicable

Output

Locale abbreviation—for example, en-US.

Example

```
IWFAdminService svc = GetAdminService();
string activeDirectoryLdapPath = ...// for example,
LDAP://ou=Sales,dc=Frabrikam,dc=com
string userFilter = ""; // All Users

try
{
    string locale = svc.GetLocale();
    Console.WriteLine("Locale = '{0}'", locale);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

```
}

/*
This example produces the following results:
Locale = 'en-US
*/
```

Supported Versions

4.6 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Registered User

API Type

Web Services

Description

Retrieves the profile information for the specified registered user.

Syntax

```
public virtual RegisteredUser GetRegisterUser(string userName)
```

Parameters

Name	Description
userName	<p><u>Description:</u></p> <p>The user name for the user.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name for a registered AgilePoint user.</p>

Output

RegisteredUser object.

Example

```
IWFAdminService svc = GetAdminService();

try
{
    RegisteredUser registerUser =
    svc.GetRegisterUser("domain\\wilson.goodman");
    Console.WriteLine("Name = '{0}', Department = '{1}'",
        registerUser.FullName,
        registerUser.Department);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}

//This example produces the following results:
//Name = 'CFO' Department = 'Finance'
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Register Users

API Type

Web Services

Description

Retrieves profile of all the registered users.

Syntax

```
public virtual RegisteredUser[] GetRegisterUsers()
```

Parameters

Name	Description
None	Not Applicable

Output

Returns an array of registered users.

Example

```
IWFAdminService svc = GetAdminService();

try
{
    RegisteredUser[] registerUsers = svc.GetRegisterUsers();
    foreach (RegisteredUser regUser in registerUsers)
    {
        Console.WriteLine("Name = '{0}', Department = '{1}'",
            registerUser.FullName,
            registerUser.Department);
    }
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Register User Icons

API Type

Web Services

Description

Retrieves an icon for a registered user.

Syntax

```
public virtual byte[] GetRegisteredUserIcon(string userName)
```

Parameters

Name	Description
userName	<p><u>Description:</u></p> <p>The user name for the user.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name for a registered AgilePoint user.</p>

Output

Return an array of bytes that contains image data.

Example

```
For future use.
```

Supported Versions

4.6 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Sender Email Address

API Type

Web Services

Description

Retrieves the sender e-mail address of the [AgilePoint Server](#) System User.

Syntax

```
public virtual string GetSenderEmailAddress()
```

Parameters

Name	Description
None	Not Applicable

Output

The email address that is configured as the sender email address on the AgilePoint Server.

Example

```
IWFAdminService svc = GetAdminService();

try
{
    string senderEmailAddress = svc.GetSenderEmailAddress();
    Console.WriteLine("Sender EMail Address = '{0}'",
        senderEmailAddress);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}

//This example produces the following results:
//Sender EMail Address = 'admin@your-domain.com'
```

Supported Versions

4.6 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show

specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get SMTP Server

API Type

Web Services

Description

Retrieves the SMTP server URI configured for [AgilePoint Server](#).

Syntax

```
public virtual string GetSmtpServer();
```

Parameters

Name	Description
None	Not Applicable

Output

A string that contains the name of the AgilePoint system's SMTP server.

Example

```
IWFAdminService svc = GetAdminService();
```

```
try
{
    string smtpServer = svc.GetSmtpServer();
    Console.WriteLine("SMTP Server Name = '" + smtpServer + "'");
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}

//This example produces the following results:
//SMTP Server Name = 'smtp.vitininfotech.com'
```

Supported Versions

4.6 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get System Performance Information

API Type

Web Services

Description

Retrieves system performance information for [AgilePoint Server](#).

Good to Know

- In an NLB environment, it shows information for the server that received the request.

Syntax

```
public virtual WFSysPerfInfo GetSysPerfInfo()
```

Parameters

Name	Description
None	Not Applicable

Output

WFSysPerfInfo object.

Example

```
IWFAdminService svc = GetAdminService();

try
{
    WFSysPerfInfo sysPerfInfo = svc.GetSysPerfInfo();
    Console.WriteLine("1) ServerID = '{0}'", sysPerfInfo.ServerID);
    Console.WriteLine("2) MemoryAllocated = '{0}'",
        sysPerfInfo.MemoryAllocated);
    //...
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}

/*
```

```
This example produces the following results:
```

```
1) ServerID = 'DEMO/4356'  
2) MemoryAllocated = '7329'  
*/
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get System User

API Type

Web Services

Description

Retrieves the user name of the AgilePoint System User.

Syntax

```
public virtual string GetSystemUser()
```

Parameters

Name	Description
None	Not Applicable

Output

Returns the name of the system user as a string value.

Example

```
IWFAdminService svc = GetAdminService();

try
{
    string systemUser = svc.GetSystemUser();
    Console.WriteLine("AgilePoint System User= '{0}'", systemUser);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}

//This example produces the following results:
//AgilePoint System User = 'Administrator'
```

Supported Versions

4.6 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Query Register Users

API Type

Web Services

Description

Retrieves the list of registered users on the AgilePoint Server.

Syntax

```
public virtual RegisteredUser[] QueryRegisterUsers(string sqlWhereClause)
```

Parameters

Name	Description
sqlWhereClause	<p><u>Description:</u></p> <p>The where clause of the SQL statement you want to query.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>The where clause of a SQL statement.</p>

Output

Returns a list of registered users.

Example

```
IWFAdminService svc = GetAdminService();
string sqlWhereClause = "DEPARTMENT in ('PublicUsers', ...)"

try
{
    RegisteredUser[] registeredUsers =
    adminService.QueryRegisterUsers(sqlWhereClause);
    foreach (RegisteredUser user in registeredUsers)
    {
        Console.WriteLine("User Name = '{0}', Email = '{1}'",
            user.UserName, user.EmailAddress);
    }
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}

/*
This example produces the following results:
User Name = 'DEMO\Author' Email = 'author@DEMO.com'
User Name = 'DEMO\Employee' Email = 'employee@DEMO.com'
*/
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Register User

API Type

Web Services

Description

Registers a new user in the AgilePoint system.

Syntax

```
public virtual void RegisterUser(RegisteredUser user)
```

Parameters

Name	Description
user	<p><u>Description:</u> Specifies information about a user.</p> <p><u>Type</u> RegisteredUser</p> <p><u>Allowed Values:</u> A valid RegisteredUser object.</p>

Output

None.

Example

```
IWFAdminService svc = GetAdminService();

try
{
    RegisteredUser user= new RegisteredUser();
    user.FullName = "Accountant";
    user.UserName = "DEMO\\Accountant";
    user.Department = "Accounts";
    user.EmailAddress = "accountant@DEMO.com";
    user.Locale = "en-US";
    svc.RegisterUser(user);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Unregister User

API Type

Web Services

Description

Permanently removes a user from the AgilePoint system.

Good to Know

- This does not remove the user from the local Windows system or the domain controller.
- If the user has a group task assigned, the user continues to see the task until another group member takes assignment for the task.
- The exception is a Dynamic Group task, which is immediately removed from the user's task list.

Syntax

```
void UnregisterUser(string userName)
```

Parameters

Name	Description
userName	<p><u>Description:</u></p> <p>The user name for the user.</p> <p><u>Type</u></p> <p>string</p>

Name	Description
	<u>Allowed Values:</u> A valid user name for a registered AgilePoint user.

Output

None.

Example

```
IWFAdminService svc = GetAdminService();
string userName = ...// for example, "[Domain Name]\\[User Account
Name]"

try
{
    svc.UnregisterUser(userName);
}

catch( Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Update Registered User

API Type

Web Services

Description

Updates the profile information of a registered user.

Syntax

```
public virtual void UpdateRegisterUser(RegisteredUser user)
```

Parameters

Name	Description
user	<p><u>Description:</u> Specifies information about a user.</p> <p><u>Type</u> RegisteredUser</p> <p><u>Allowed Values:</u> A valid RegisteredUser object.</p>

Output

None.

Example

```
RegisteredUser user = new RegisteredUser();
user.UserName = "DEMO3\\cfo";
user.FullName = "Andy";
user.EmailAddress = "cfo@tusca.com";
user.Title = "CFO";
user.Department = "Marketing";
user.Manager = "DEMO3\\Administrator";
user.Locale = Thread.CurrentThread.CurrentUICulture.Name;
IWFAdminService svc = GetAdminService();

try
{
    svc.UpdateRegisterUser( user );
}

catch( Exception ex )
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Update Registered User Icon

API Type

Web Services

Description

Updates the icon for a registered user.

Syntax

```
public virtual void UpdateRegisteredUserIcon(string userName, byte[] UserIcon)
```

Parameters

Name	Description
userName	<p><u>Description:</u></p> <p>The user name for the user.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name for a registered AgilePoint user.</p>
UserIcon	<p><u>Description:</u></p> <p>A byte array representation of icon image.</p>

Name	Description
	<p data-bbox="508 285 570 317"><u>Type</u></p> <p data-bbox="508 348 561 380">byte</p> <p data-bbox="508 432 699 464"><u>Allowed Values:</u></p> <p data-bbox="508 495 719 527">A valid byte array.</p>

Output

None.

Example

```
try
{
    System.Drawing.ImageConverter converter = new System.Drawing.ImageConverter();
    byte[] imgArray = (byte[])converter.ConvertTo(imageData, typeof(byte[]));
    svc.UpdateRegisteredUserIcon("MYDOMAIN\\cfo", imgArray);
}
catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

4.6 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Report Configuration Methods

This section describes service calls related to reports.

Add Report Configuration

API Type

Web Services

Description

Adds a report configuration to the system.

Syntax

```
public virtual WFReportConfigure AddReportConfigure(string reportName, string configure)
```

Parameters

Name	Description
reportName	<p><u>Description:</u></p> <p>The name of a report.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p>

Name	Description
	A valid report name.
configure	<p><u>Description:</u></p> <p>The report configuration in XML format.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid report configuration in XML format.</p>

Output

WFReportConfigure object.

Example

```

IWFAdminService svc = GetAdminService();
string reportName = ...//for example, "weekly task report"
string configure = ...// xml-serialization of WFReportConfiguration

try
{
    WFReportConfigure reportConfig =
    svc.AddReportConfigure(reportName, configure);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}

```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get All Report Configurations

API Type

Web Services

Description

Retrieves all report configurations from the system.

 The Report Center component (sometimes referred to as AgileReports) can only be used by customers who purchased this component before April 1, 2020. This version is no longer available for purchase, including by current and previous purchasers of AgilePoint NX or the Report Center component. In AgilePoint NX v8.0 and higher, Report Center has been replaced with the [Analytics Center](#) component.

Syntax

```
public virtual WFReportConfigure[] GetAllReportConfigure()
```

Parameters

Name	Description
None	Not Applicable

Output

WFReportConfigure object.

Example

```
IWFAdminService svc = GetAdminService();

try
{
    WFReportConfigure[] reportConfigs = svc.GetAllReportConfigure();
    Console.WriteLine("This AgilePoint Server has {0} reports
configured:", reportConfigs.Length);
    foreach (WFReportConfigure config in reportConfigs)
    {
        Console.WriteLine("Report Name: {0}", config.ReportName);
    }
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
/*
This example produces the following results:
This AgilePoint Server has 1 reports configured:
Report Name:Average Process Time By Automatic Activities
*/
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Report Configuration

API Type

Web Services

Description

Retrieves a report configuration from the system.

Syntax

```
public virtual WFReportConfigure GetReportConfigure(string reportName)
```

Parameters

Name	Description
reportName	<p><u>Description:</u></p> <p>The name of a report.</p> <p><u>Type</u></p> <p>string</p>

Name	Description
	<u>Allowed Values:</u> A valid report name.

Output

WFReportConfigure object.

Example

```
IWFAdminService svc = GetAdminService();
string reportName = ...;

try
{
    WFReportConfigure cfg = svc.GetReportConfigure(reportName);
    Console.WriteLine("Report Name: {0}, config:{1}", cfg.ReportName,
        cfg.Configure);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Read Configuration

API Type

Web Services

Description

Read an event service configuration.

Syntax

```
public string ReadConfiguration()
```

Parameters

Name	Description
None	Not Applicable

Output

[WFEvent](#) object.

Example

```
public string ReadConfiguration()  
  
{  
    IWFEventServiceConfiguration eventservice = GetEventService();
```

```
string configuration = eventservice.ReadConfiguration();  
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Remove Report Configure

API Type

Web Services

Description

Removes a report configuration from the system.

Syntax

```
public virtual void RemoveReportConfigure(string reportName)
```

Parameters

Name	Description
reportName	<p><u>Description:</u></p> <p>The name of a report.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid report name.</p>

Output

None.

Example

```
IWFAdminService svc = GetAdminService();
string reportName = ...;

try
{
    svc.RemoveReportConfigure(reportName);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Update Report Configuration

API Type

Web Services

Description

Updates a report configuration in the AgilePoint system.

Syntax

```
public virtual WFReportConfigure UpdateReportConfigure(string reportName, string configure)
```

Parameters

Name	Description
reportName	<p><u>Description:</u></p> <p>The name of a report.</p> <p><u>Type</u></p> <p>string</p>

Name	Description
	<p><u>Allowed Values:</u></p> <p>A valid report name.</p>
configure	<p><u>Description:</u></p> <p>The report configuration in XML format.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid report configuration in XML format.</p>

Output

WFReportConfigure object.

Example

```

IWFAdminService svc = GetAdminService();
string reportName = ...//for example, "weekly task report"
string configure = ...// xml-serialization of WFReportConfiguration

try
{
    WFReportConfigure reportConfig =
        svc.UpdateReportConfigure(reportName, configure);
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}

```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Component Administration Methods

This section describes service calls related to server component administration.

Get Server Component

API Type

Web Services

Description

Retrieves a server component name.

Syntax

```
public virtual WFCComponent GetServerComponent(string Name)
```

Parameters

Name	Description
Name	<u>Description:</u>

Name	Description
	<p>The name of an item, such as a property or attribute in a name/value pair.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid name.</p>

Output

WFComponent object.

Example

None.

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Get Server Component Names

API Type

Web Services

Description

Retrieves the server component names using the Admin Services.

Syntax

```
public virtual string[] GetServerComponentNames()
```

Parameters

Name	Description
None	Not Applicable

Output

An array of strings that contain the server component names.

Example

```
IWFAdminService svc = GetAdminService();

try
{
    string[] names = svc.GetServerComponentNames();
}

catch (Exception ex)
{
    Console.WriteLine("Failed! " + ShUtil.GetSoapMessage(ex));
}
```

Supported Versions

3.2.0.4 and higher

Code Examples in the AgilePoint NX Documentation

The AgilePoint NX Product Documentation is intended as a basic reference to help you understand how to complete basic coding tasks, such as make API or JavaScript method calls. Code examples that show specific use cases, the solutions to specific business problems, or detailed implementation scenarios are outside the scope of the AgilePoint NX Product Documentation. For specific and/or advanced types of examples that may better meet your requirements, AgilePoint provides several resources:

- [AgilePoint Community Forums](#) - A free, AgilePoint-moderated, crowd-sourcing user forum where you can ask questions about specific techniques, the solutions to use cases, workarounds, or other topics that may not be covered in the Product Documentation.
- [Professional Services](#) - If you can not find the information you need for your specific business problem, mentoring is available through AgilePoint Professional Services.
- [Personalized Training](#) - AgilePoint can provide personalized training for your organization. To request personalized training, [contact AgilePoint Sales](#).

Classes

This section includes references for all classes within the AgilePoint Web Service API.

KeyValue

Description

A class that represents an object with name and value properties.

Syntax

```
public class KeyValue
```

Constructors

```
public KeyValue();  
public KeyValue(string key, string val);
```

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

Properties

Property	Description
Key	<p><u>Description:</u></p> <p>Specifies the key of the instance.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid key.</p>
Value	<p><u>Description:</u></p> <p>The value for an item, such as the value for an attribute in a name-value pair.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid string value.</p>

NameValue

Description

A class that represents an object with the name and value properties.

Syntax

```
public class NameValue
```

Constructors

```
public NameValue();
public NameValue(string name,object value);
```

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.Share (in Ascentn.Workflow.Share.dll)

Properties

Property	Description
Name	<p><u>Description:</u></p> <p>The name of an item, such as a property or attribute in a name/value pair.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid name.</p>
Value	<p><u>Description:</u></p> <p>The value for an item, such as the value for an attribute in a key-value pair.</p> <p><u>Type</u></p> <p>object</p> <p><u>Allowed Values:</u></p>

Property	Description
	A valid object value.

IWFWorkflowService

Description

A class that provides interfaces for the AgilePoint workflow API on the client side.

Syntax

```
public interface IWFWorkflowService
```

Constructors

Not Applicable.

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

IWFTrackingEventPublisher

Description

Provides an interface to publish a process event.

Syntax

```
public interface IWFTrackingEventPublisher
```

Constructors

Not Applicable.

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

Properties

Property	Description
AppNameFilter	<p><u>Description:</u> Gets the application name filter.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid application name filter.</p>
EventTypeFilter	<p><u>Description:</u> Gets the event type filter.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid event type filter.</p>

RegisteredUser

Description

A class that represents an AgilePoint registered user.

Syntax

```
public class RegisteredUser
```

Constructors

```
public RegisteredUser();  
public RegisteredUser(string user, string emailAddress, DateTime registeredDate, string fullName);
```

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

Properties

Property	Description
Department	<p><u>Description:</u></p> <p>Gets and sets department of the user.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p>

Property	Description
	A valid department name.
EMailAddress	<p><u>Description:</u></p> <p>Gets and sets the user's e-mail address.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid e-mail address.</p>
FullName	<p><u>Description:</u></p> <p>Specifies the full name of the user.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>One line of text (a string).</p> <p>Accepted:</p> <ul style="list-style-type: none"> • Letters • Numbers • Spaces
Manager	<p><u>Description:</u></p> <p>Specifies the manager of the user.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name.</p>
RegisteredDate	<p><u>Description:</u></p>

Property	Description
	<p>Gets and sets date registered.</p> <p><u>Type</u> DateTime</p> <p><u>Allowed Values:</u> A valid DateTime value.</p>
Title	<p><u>Description:</u> Specifies the job title of the user.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> One line of text (a string).</p> <p>Accepted:</p> <ul style="list-style-type: none"> • Letters • Numbers • Spaces
UserName	<p><u>Description:</u> Specifies a qualified user name of the instance. A qualified user name formats as [Domain Name]\[Logon Username] or [Local host name]\[Logon Username].</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid user name for a registered AgilePoint user.</p>

WFAccessRights

Description

A class that represents different types of access rights.

Syntax

```
public enum WFAccessRights
```

Constructors

Not Applicable.

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

WFAgilePart

Description

An abstract class of AgilePart.

Syntax

```
public class WFAgilePart
```

Constructors

```
protected WFAgilePart();
```

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

WFAgilePartDescriptor

Description

An abstract class of WFAgilePartDescriptor.

Syntax

```
public class WFAgilePartDescriptor : WFAutomaticActivityDefinition, IWFPProcessDefinitionReference
```

Constructors

```
public WFAgilePartDescriptor();
```

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

Properties

Property	Description
AssemblyName	<p><u>Description:</u> Gets and sets the full name of the assembly.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid assembly name.</p>
ClassName	<p><u>Description:</u> Gets and sets the class name, including the namespace.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid class name.</p>
Description	<p><u>Description:</u> A description of the activity.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A string that can contain spaces and special characters.</p>
DisplayName	<p><u>Description:</u> Gets and sets the display name of the activity.</p> <p><u>Type</u></p>

Property	Description
	<p>string</p> <p><u>Allowed Values:</u> A valid activity display name.</p>
ExpectedDuration	<p><u>Description:</u> Specifies an expected time duration of the activity.</p> <p><u>Type</u> WFTimeDuration</p> <p><u>Allowed Values:</u> A valid WFTimeDuration object.</p>
ID	<p><u>Description:</u> Specifies the activity ID.</p> <p><u>Type</u> WFTimeDuration</p> <p><u>Allowed Values:</u> A valid activity ID.</p>
IncomingActivities	<p><u>Description:</u> Gets the interfaces for all the incoming activities.</p> <p><u>Type</u> WFTimeDuration</p> <p><u>Allowed Values:</u> A valid WFTimeDuration object.</p>
IncomingEmail	<p><u>Description:</u> Specifies an email template that used for an email notification after the start of the activity.</p>

Property	Description
	<p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid email template.</p>
IncomingMailTemplate	<p><u>Description:</u> Specifies an email template that used for an email notification after the start of the activity.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid email template.</p>
Index	<p><u>Description:</u> Specifies the ordering index of the activity.</p> <p><u>Type</u> int</p> <p><u>Allowed Values:</u> An integer.</p>
Item	<p><u>Description:</u> Specifies the value associated with the specified parameter name.</p> <p><u>Type</u> object</p> <p><u>Allowed Values:</u> A valid Types object.</p>

Property	Description
Method	<p><u>Description:</u></p> <p>Specifies the name of the method for the AgilePart.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid method name.</p>
Name	<p><u>Description:</u></p> <p>The name of an item, such as a property or attribute in a name/value pair.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid name.</p>
OutgoingActivities	<p><u>Description:</u></p> <p>Gets the interfaces for all the outgoing activities.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid interface.</p>
OutgoingEmail	<p><u>Description:</u></p> <p>Specifies an email template that used for email notification after the completion of an activity.</p> <p><u>Type</u></p> <p>string</p>

Property	Description
	<p><u>Allowed Values:</u> A valid email template.</p>
OutgoingMailTemplate	<p><u>Description:</u> Specifies an email template to use for email notification when leaving the activity.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid email template.</p>
Procedure	<p><u>Description:</u> Specifies the name of the procedure (method).</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid procedure name.</p>
Process	<p><u>Description:</u> Gets the interface class and sets the process definition.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid process definition.</p>
SaveErrorMessageTo	<p><u>Description:</u> Specifies an attribute name for an error message of the AgilePart execution.</p>

Property	Description
	<p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid attribute name.</p>
SaveStatusTo	<p><u>Description:</u> Specifies an attribute name for the status of the AgilePart execution.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid name of an attribute that is boolean type.</p>
Synchronous	<p><u>Description:</u> Specifies the flag to determine if this is a synchronous call.</p> <p><u>Type</u> bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - The call is synchronous. • False - The call is asynchronous.
TimeoutActivities	<p><u>Description:</u> Gets the interfaces for all the timeout activities.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid interface.</p>

Property	Description
TimeoutMailTemplate	<p><u>Description:</u></p> <p>Specifies an email template that used for email notification when the activity is overdue.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid email template.</p>
WaitAllIncoming	<p><u>Description:</u></p> <p>Gets and sets the flag to determine whether the activity must wait until all predecessor activities complete.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - The activity waits until all predecessor activities complete. • False - The activity starts without waiting for predecessor activities complete.

WFAgileWork

Description

An abstract class of WFAgileWork.

Syntax

```
public class WFAgileWork
```

Constructors

protected WFAgileWork(WFProcessInstance currentProcessInstance, WFManualActivityInstance currentActivityInstance);

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

WFAgileWorkDescriptor

Description

An abstract class of WFAgileWorkDescriptor Custom properties format: [Prefix]:[Assembly Information]: [Class Name]:[Parameter].

Syntax

```
public class WFAgileWorkDescriptor : WFManualActivityDefinition, IWFPProcessDefinitionReference
```

Constructors

public WFAgileWorkDescriptor();

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

Properties

Property	Description
AssemblyName	<p><u>Description:</u></p> <p>Gets and sets the full name of the assembly.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid assembly name.</p>
AutoComplete	<p><u>Description:</u></p> <p>Gets and sets the flag that determines whether the work item is automatically marked as completed.</p> <p><u>Type</u></p> <p>bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - The work item is completed automatically. • False - The work item is not completed automatically.
ClassName	<p><u>Description:</u></p> <p>Gets and sets the class name, including the namespace.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid class name.</p>
CustomProperties	<p><u>Description:</u></p>

Property	Description
	<p>Specifies the custom properties. This member supports the Agile-Point Framework infrastructure and does not intend to use directly from your code.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid custom properties.</p>
Description	<p><u>Description:</u> A description of the activity.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A string that can contain spaces and special characters.</p>
DisplayName	<p><u>Description:</u> Gets and sets the display name of the activity.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid activity display name.</p>
ExpectedDuration	<p><u>Description:</u> Specifies an expected time duration of the activity.</p> <p><u>Type</u> WFTimeDuration</p> <p><u>Allowed Values:</u></p>

Property	Description
	A valid WFTimeDuration object.
ID	<p><u>Description:</u> Specifies the activity ID.</p> <p><u>Type</u> WFTimeDuration</p> <p><u>Allowed Values:</u> A valid activity ID.</p>
IncomingActivities	<p><u>Description:</u> Gets the interfaces for all the incoming activities.</p> <p><u>Type</u> WFTimeDuration</p> <p><u>Allowed Values:</u> A valid WFTimeDuration object.</p>
IncomingEmail	<p><u>Description:</u> Specifies an email template that used for an email notification after the start of the activity.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid email template.</p>
IncomingMailTemplate	<p><u>Description:</u> Specifies an email template that used for an email notification after the start of the activity.</p> <p><u>Type</u></p>

Property	Description
	<p><code>string</code></p> <p><u>Allowed Values:</u> A valid email template.</p>
Index	<p><u>Description:</u> Specifies the ordering index of the activity.</p> <p><u>Type</u> <code>int</code></p> <p><u>Allowed Values:</u> An integer.</p>
Item	<p><u>Description:</u> Specifies the value associated with the specified parameter name.</p> <p><u>Type</u> <code>object</code></p> <p><u>Allowed Values:</u> A valid Types object.</p>
MaxParticipant	<p><u>Description:</u> Specifies the maximum participants for the activity.</p> <p><u>Type</u> <code>int</code></p> <p><u>Allowed Values:</u> An integer.</p>
Name	<p><u>Description:</u> The name of an item, such as a property or attribute in a name/value pair.</p>

Property	Description
	<p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid name.</p>
Optional	<p><u>Description:</u> Gets and sets the flag to indicate if the activity is optional.</p> <p><u>Type</u> bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - The activity is optional. • False - The activity is not optional.
OutgoingActivities	<p><u>Description:</u> Gets the interfaces for all the outgoing activities.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid interface.</p>
OutgoingEmail	<p><u>Description:</u> Specifies an email template that used for email notification after the completion of an activity.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u></p>

Property	Description
	A valid email template.
OutgoingMailTemplate	<p><u>Description:</u></p> <p>Specifies an email template to use for email notification when leaving the activity.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid email template.</p>
Participant	<p><u>Description:</u></p> <p>Specifies the participant of the activity.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name.</p>
Participants	<p><u>Description:</u></p> <p>Specifies the participants of the activity.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid multiple user names separated with a semicolon (;).</p>
Process	<p><u>Description:</u></p> <p>Gets the interface class and sets the process definition.</p> <p><u>Type</u></p> <p>string</p>

Property	Description
	<p><u>Allowed Values:</u> A valid process definition.</p>
ReassigningEmail	<p><u>Description:</u> Specifies the email template that used for an email notification when task is reassigned.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid email template.</p>
ReassigningMailTemplate	<p><u>Description:</u> Specifies the email template for task reassignment.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid email template.</p>
RemindingEmail	<p><u>Description:</u> Specifies an email template that used for reminding email notification.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid email template.</p>
RemindingMailAtBusinessTime	<p><u>Description:</u></p>

Property	Description
	<p>Gets and sets the flag that determines whether reminding an email must send in business time.</p> <p><u>Type</u></p> <p>bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - Reminding emails are sent using business time. • False - Reminding email are sent using real time.
RemindingMailFrequency	<p><u>Description:</u></p> <p>Specifies the frequency of reminding an email notification.</p> <p><u>Type</u></p> <p>int</p> <p><u>Allowed Values:</u></p> <p>An integer.</p>
RemindingMailStartTime	<p><u>Description:</u></p> <p>Specifies the start time in minutes of reminding an email notification before the task expires.</p> <p><u>Type</u></p> <p>int</p> <p><u>Allowed Values:</u></p> <p>An integer.</p>
RemindingMailTemplate	<p><u>Description:</u></p> <p>Specifies the email template for task reminder emails.</p> <p><u>Type</u></p> <p>int</p>

Property	Description
	<p><u>Allowed Values:</u></p> <p>A valid email template.</p>
ReuseParticipant	<p><u>Description:</u></p> <p>Gets and sets the flag that determines whether the participants must remain the same when the activity activated more than one time.</p> <p><u>Type</u></p> <p>bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - The participants remains the same when the activity activated more than one time. • False - The participants do not remain the same when the activity activated more than one time.
TimeoutActivities	<p><u>Description:</u></p> <p>Gets the interfaces for all the timeout activities.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid interface.</p>
TimeoutEmail	<p><u>Description:</u></p> <p>Specifies the email template to use for email notification when task is Overdue.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid email template.</p>

Property	Description
TimeoutMailTemplate	<p><u>Description:</u></p> <p>Specifies an email template that used for email notification when the activity is overdue.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid email template.</p>
WaitAllIncoming	<p><u>Description:</u></p> <p>Gets and sets the flag to determine whether the activity must wait until all predecessor activities complete.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - The activity waits until all predecessor activities complete. • False - The activity starts without waiting for predecessor activities complete.
WaitWorkPerformed	<p><u>Description:</u></p> <p>Specifies whether the activity must be marked as completed by the application before proceeding.</p> <p><u>Type</u></p> <p>bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - The activity must complete before proceeding. • False - The activity can proceed without waiting.

Property	Description
WorkPerformer	<p><u>Description:</u></p> <p>Specifies the WorkPerformer/job of the manual activity.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid WorkPerformer/job name.</p>
WorkToPerform	<p><u>Description:</u></p> <p>Represents the task that performed by the participants of the activity.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid work to perform name.</p>

WFAny

Description

A class that represents a primitive data type with type code.

Syntax

```
public class WFAny
```

Constructors

```
public WFAny();
```

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

Properties

Property	Description
Type	<p><u>Description:</u></p> <p>Gets and sets the type of a WFAny object.</p> <p><u>Type</u></p> <p>int</p> <p><u>Allowed Values:</u></p> <p>An integer.</p>
Value	<p><u>Description:</u></p> <p>The value for an item, such as the value for an attribute in a key-value pair.</p> <p><u>Type</u></p> <p>object</p> <p><u>Allowed Values:</u></p> <p>A valid object value.</p>

WFEvent

Description

A class that represents a workflow event.

Syntax

```
public class WFEvent
```

Constructors

```
public WFEvent();  
public WFEvent(string name);  
public WFEvent(string sender, string name);
```

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

Properties

Property	Description
ActivityInstID	<p><u>Description:</u></p> <p>The unique ID for an activity instance.</p> <p><u>Type</u></p> <p>string</p>

Property	Description
	<p><u>Allowed Values:</u></p> <p>A valid activity instance ID.</p>
AutoStart	<p><u>Description:</u></p> <p>Specifies if the process starts immediately after it is created.</p> <p><u>Type</u></p> <p>bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - The process instance starts immediately after it is created. • False - The process instance does not start immediately after it is created.
clientData	<p><u>Description:</u></p> <p>Specifies the client data, which identifies a client for AgilePoint Server.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that contains the client data.</p> <p>If this value is null, the system will keep existing client data. Otherwise the relevant data is overwritten.</p>
CustomAttributes	<p><u>Description:</u></p> <p>Specifies custom attributes in XML format.</p> <p><u>Type</u></p> <p>NameValue</p> <p><u>Allowed Values:</u></p>

Property	Description
	A valid NameValue pair.
Designated	<p><u>Description:</u></p> <p>Gets and sets the flag that determines if the event should be handled by a designated engine.</p> <p><u>Type</u></p> <p>bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - The event is handled by a designated engine. • False - The event is not handled by a designated engine.
Diagnostic	<p><u>Description:</u></p> <p>Gets and sets the flag that determines if the process data will be deleted from the database after the process is completed.</p> <p><u>Type</u></p> <p>bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - The process data is deleted from the database. • False - The process data is not deleted from the database.
EndDate	<p><u>Description:</u></p> <p>Specifies the completion date of delegation.</p> <p><u>Type</u></p> <p>DateTime</p> <p><u>Allowed Values:</u></p> <p>A valid DateTime value.</p>
Entries	<p><u>Description:</u></p>

Property	Description
	<p>Gets the relay time of the event.</p> <p><u>Type</u> int</p> <p><u>Allowed Values:</u> A valid relay time.</p>
Error	<p><u>Description:</u> Gets and sets the error message of the event.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid error message.</p>
EventID	<p><u>Description:</u> Specifies a unique ID for an event.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A unique, 32-character ID.</p>
EventName	<p><u>Description:</u> Specifies the name of the event.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid event name.</p>

Property	Description
ParamsXml	<p><u>Description:</u> Gets and sets the parameters as XML.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A parameter.</p>
ParentProclnstID	<p><u>Description:</u> Specifies parent process instance ID.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid process instance ID.</p>
ProcDefID	<p><u>Description:</u> Specifies process template (process definition) ID.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid process definition ID.</p>
ProclnstID	<p><u>Description:</u> Specifies the unique ID of a process instance.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u></p>

Property	Description
	A valid process instance ID
ProclnstName	<p><u>Description:</u></p> <p>A unique process name that is associated with the process definition.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A unique process instance name up to 1024 characters.</p>
Sender	<p><u>Description:</u></p> <p>Specifies a sender of the event.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid user name.</p>
SentDate	<p><u>Description:</u></p> <p>Specifies a sent date of the event.</p> <p><u>Type</u></p> <p>DateTime</p> <p><u>Allowed Values:</u></p> <p>A valid DateTime value.</p>
SourceWorkItemID	<p><u>Description:</u></p> <p>An ID that represents the original, or source, work item.</p> <p><u>Type</u></p> <p>string</p>

Property	Description
	<p><u>Allowed Values:</u> A valid, unique 32-character ID.</p>
Status	<p><u>Description:</u> The status of the event.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid status.</p>
ThrowAwayInstance	<p><u>Description:</u> Specifies the flag that determines if AgilePoint server should clear the process from cache.</p> <p><u>Type</u> bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - AgilePoint Server clears the process from the cache. • False - AgilePoint Server does not clear the process from cache.
UserID	<p><u>Description:</u> Specifies the user ID.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid user ID.</p>
WorkItemID	<p><u>Description:</u></p>

Property	Description
	<p>An ID that represents a work item (task).</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid, unique 32-byte work item (task) ID.</p>
workObjectID	<p><u>Description:</u> An ID for an object, such as a document, that is associated with the process instance.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid, unique 256-character ID. Even though the field size is 256 characters, in common practice, this will usually return a 32-character GUID.</p>

WFIntegratedApplication

Description

A class that represents an integrated application, such as an AgileConnector.

Syntax

```
public class WFIntegratedApplication : IWFIntegratedApplication
```

Constructors

```
public WFIntegratedApplication();
```

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

Properties

Property	Description
AppEventKind	<p><u>Description:</u> Specifies the type of application event.</p> <p><u>Type</u> IWFWorkflowService</p> <p><u>Allowed Values:</u> A valid application event.</p>
AppName	<p><u>Description:</u> Specifies the name of the application.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid, case-sensitive application name.</p>
Category	<p><u>Description:</u> Gets and sets a category name.</p> <p><u>Type</u></p>

Property	Description
	<p>string</p> <p><u>Allowed Values:</u> A valid category name.</p>
CustomConfiguration	<p><u>Description:</u> Gets and sets custom configuration settings.</p> <p><u>Type</u> System.Xml.NodeList</p> <p><u>Allowed Values:</u> A valid set of custom configuration settings.</p>
Description	<p><u>Description:</u> A description of the application.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A string that can contain spaces and special characters.</p>
Impersonator	<p><u>Description:</u> Specifies a user name of an impersonator user account.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid impersonator user name.</p>

WFIntegratedApplicationDescriptor

Description

An abstract class that describes the global control module.

Syntax

```
public class WFIntegratedApplicationDescriptor : IWFIntegratedApplicationDescriptor
```

Constructors

```
public WFIntegratedApplicationDescriptor(string integratedApplicationsXML);
```

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

Properties

Property	Description
integratedApplications-Xml	<p><u>Description:</u></p> <p>Gets the configuration settings for the integrated application.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p>

Property	Description
	Valid configuration settings.

WFPartialRollbackInstruction

Description

A class that provides instructions for activating activity instances.

Syntax

```
public class WFPartialRollbackInstruction
```

Constructors

```
public WFPartialRollbackInstruction();
```

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

Properties

Property	Description
PartialRollbackUnits	<p><u>Description:</u></p> <p>An array of instances of the PartialRollbackUnit class.</p> <p><u>Type</u></p>

Property	Description
	<p>PartialRollbackUnit</p> <p><u>Allowed Values:</u></p> <p>One or more PartialRollbackUnit objects.</p>

WFProcessMergingInstruction

Description

A class that provides instructions for merging process instances during runtime.

Syntax

```
public class WFProcessMergingInstruction
```

Constructors

Not Applicable.

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

Properties

Property	Description
MergingProcessInstanceIDs	<p><u>Description:</u></p> <p>An array that includes the process instance IDs of the process instances you are merging.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>An array of valid process instance IDs.</p>
MergedProcessInstance	<p><u>Description:</u></p> <p>An instance of the MergedProcess Parameter class that includes the merged process instance.</p> <p><u>Type</u></p> <p>MergedProcessParameter</p> <p><u>Allowed Values:</u></p> <p>A valid An MergedProcess Parameter object.</p>

WFProcessMigrationInstruction

Description

A class that provides instructions for merging process instances during runtime.

Syntax

```
public class WFProcessMigrationInstruction
```

Constructors

Not Applicable.

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

Properties

Property	Description
MatchingActivityDefinition	<p><u>Description:</u> Specifies information for activity migration.</p> <p><u>Type</u> class</p> <p><u>Allowed Values:</u> A valid class.</p>
SourceProcessDefinitionID	<p><u>Description:</u> The ID of the original, or source, process definition.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid process definition ID.</p>

Property	Description
TargetProcessDefinitionID	<p><u>Description:</u> The ID of the target, or destination, process definition.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid process definition ID.</p>
Action	<p><u>Description:</u> Specifies the migration action.</p> <p><u>Type</u> MigrationAction</p> <p><u>Allowed Values:</u> A valid migration action.</p>
IncludeXmlData	<p><u>Description:</u> Specifies whether a migration includes XML data.</p> <p><u>Type</u> bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - The migration includes an XML data. • False - The migration does not include an XML data.

WFProcessPluggableAdapter

Description

A class that represents an AgileExtender.

Syntax

```
public WFProcessPluggableAdapter
```

Constructors

```
public WFProcessPluggableAdapter(WFProcessInstance currentProcessInstance)
```

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

Properties

Property	Description
ComponentTypeID	<p><u>Description:</u></p> <p>Specifies a unique type ID of an Agile Extender.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p>

Property	Description
	A valid type ID for your AgileExtender.
Name	<p><u>Description:</u></p> <p>Specifies the name of an AgileExtender.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid AgileExtender name.</p>
DisplayName	<p><u>Description:</u></p> <p>Gets and sets the display name of an AgileExtender.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid AgileExtender display name.</p>
Description	<p><u>Description:</u></p> <p>A description of an AgileExtender.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A string that can contain spaces and special characters.</p>
AssemblyFullName	<p><u>Description:</u></p> <p>Gets the full name of an AgileExtender assembly.</p> <p><u>Type</u></p> <p>string</p>

Property	Description
	<p><u>Allowed Values:</u></p> <p>A valid AgileExtender assembly name.</p>
ClassName	<p><u>Description:</u></p> <p>Gets the type of the AgileExtender.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid AgileExtender type.</p>
ProcessInstance	<p><u>Description:</u></p> <p>Gets a runtime process instance object.</p> <p><u>Type</u></p> <p>WFProcessInstance</p> <p><u>Allowed Values:</u></p> <p>A WFProcessInstance object.</p>

WFProcessPluggableAdapterDescriptor

Description

A Design Time supporting class for Agile Extender. This Design Time class extends the class WFProcessPluggableAdapterDescriptor.

Syntax

```
public partial class MyAgileExtenderDescriptor : WFProcessPluggableAdaptterDescriptor
```

Constructors

```
public WFProcessPluggableAdaptterDescriptor()
public WFProcessPluggableAdaptterDescriptor(bool designTime)
```

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

Properties

Property	Description
ComponentTypeID	<p><u>Description:</u></p> <p>Specifies a unique type ID of an Agile Extender.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid type ID for your AgileExtender.</p>
Name	<p><u>Description:</u></p> <p>Specifies the name of an AgileExtender.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid AgileExtender name.</p>

Property	Description
DisplayName	<p><u>Description:</u> Gets and sets the display name of an AgileExtender.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid AgileExtender display name.</p>
Description	<p><u>Description:</u> A description of an AgileExtender.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A string that can contain spaces and special characters.</p>
AssemblyFullName	<p><u>Description:</u> Gets the full name of an AgileExtender assembly.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u> A valid AgileExtender assembly name.</p>
ClassName	<p><u>Description:</u> Gets the type of the AgileExtender.</p> <p><u>Type</u> string</p> <p><u>Allowed Values:</u></p>

Property	Description
	A valid AgileExtender type.
Process	<p><u>Description:</u></p> <p>Gets the interface class and sets the process definition.</p> <p><u>Type</u></p> <p>IWFProcessDefinition</p> <p><u>Allowed Values:</u></p> <p>A IWFProcessDefinition class.</p>
DesignTime	<p><u>Description:</u></p> <p>Specifies whether the object is used at design time.</p> <p><u>Type</u></p> <p>bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - The object is used at design time. • False - The object is not used at design time.

WFProcessSplittingInstruction

Description

A class that provides instructions for splitting a process instance during runtime.

Syntax

```
public class WFProcessSplittingInstruction
```

Constructors

Not Applicable.

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

Properties

Property	Description
SplitProcessInstances	<p><u>Description:</u></p> <p>An array of instances of the SplitProcessParameter class.</p> <p><u>Type</u></p> <p>SplitProcessParameter</p> <p><u>Allowed Values:</u></p> <p>A valid SplitProcessParameter object.</p>

WFQueryExpr

Description

A class that represents a query expression object.

Syntax

```
public class WFQueryExpr
```

Constructors

```
public WFQueryExpr();
public WFQueryExpr(string columnName, int op, WFAAny any, bool val);
```

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

Properties

Property	Description
Any	<p><u>Description:</u></p> <p>Gets and sets the WFAAny object to be compared.</p> <p><u>Type</u></p> <p>WFAAny</p> <p><u>Allowed Values:</u></p> <p>A valid WFAAny object.</p>
ColumnName	<p><u>Description:</u></p> <p>Gets and sets the associated database column name.</p> <p><u>Type</u></p> <p>string</p>

Property	Description
	<p><u>Allowed Values:</u></p> <p>A valid database column name.</p>
IsValue	<p><u>Description:</u></p> <p>Gets and sets the flag that indicates if it compares a value or column of the table.</p> <p><u>Type</u></p> <p>bool</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - It compares a value or column of the table. • False - It does not compare a value or column of the table.
Name	<p><u>Description:</u></p> <p>The name of an item, such as a property or attribute in a name/value pair.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid name.</p>
Operator	<p><u>Description:</u></p> <p>Specifies the operator used for comparison.</p> <p><u>Type</u></p> <p>int</p> <p><u>Allowed Values:</u></p> <p>A valid operator.</p>

WFTimeDuration

Description

A class that represents Time Duration with length, time unit, and business time.

Syntax

```
public class WFTimeDuration : Serializable
```

Constructors

```
public WFTimeDuration();  
public WFTimeDuration(string length, WFTimeUnit unit, bool b);  
public WFTimeDuration(int length, WFTimeUnit unit, bool b);
```

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

Properties

Property	Description
BusinessTime	<p><u>Description:</u></p> <p>Determines whether the system calculates the duration using your business time calendar.</p> <p><u>Type</u></p> <p>bool</p>

Property	Description
	<p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • True - The system calculates duration using business time. • False - The calculates the duration based on real time.
Length	<p><u>Description:</u></p> <p>Specifies the length of time duration.</p> <p><u>Type</u></p> <p>string</p> <p><u>Allowed Values:</u></p> <p>A valid time duration length.</p>
Unit	<p><u>Description:</u></p> <p>Specifies the time unit.</p> <p><u>Type</u></p> <p>WFTimeUnit</p> <p><u>Allowed Values:</u></p> <ul style="list-style-type: none"> • hour • day • week • month • year

WFTimeUnit

Description

A class that represents time unit.

Syntax

```
public class WFTimeUnit
```

Constructors

```
public WFTimeUnit();
```

Namespace and Assembly

Requirement	Value
Namespace	Ascentn.Workflow.Base
Assembly	Ascentn.Workflow.WFBase (in Ascentn.Workflow.WFBase.dll)

Properties

Property	Description
Value	<p><u>Description:</u></p> <p>The value for an item expressed as an integer, such as a time unit.</p> <p><u>Type</u></p> <p>int</p> <p><u>Allowed Values:</u></p>

Property	Description
	An integer.

Data Types - API

A list of all data types used in the AgilePoint Web Service API. Common types refer to the documentation on msdn.com

Name	Description
bool	See the documentation on MSDN .
byte	See the documentation on MSDN .
DateTime	See the documentation on MSDN .
enum	See the documentation on MSDN .
Guid	Instantiates the Guid class.
int	See the documentation on MSDN .
IWFProcessDefinition	Instantiates the IWFProcessDefinition class.
IWFWorkflowService	Instantiates the IWFWorkflowService class.
JSON String	See the documentation on MSDN .
KeyValue	Instantiates the KeyValue class.
List	See the documentation on MSDN .
MergedProcessParameter	Instantiates the MergedProcessParameter class.
MigrationAction	Instantiates the MigrationAction class.
NameValue	Instantiates the NameValue class.
object	See the documentation on MSDN .
PartialRollbackUnit	Instantiates the PartialRollbackUnit class.
RegisteredUser	Instantiates the RegisteredUser class.
SplitProcessParameter	Instantiates the SplitProcessParameter class.
String	See the documentation on MSDN .
WFAccessRights	Instantiates the WFAccessRights class.
WFAny	Instantiates the WFAny class.

Name	Description
WFEvent	Instantiates the WFEvent class.
WFProcessInstance	Instantiates the WFProcessInstance class.
WFQueryExpr	Instantiates the WFQueryExpr class.
WFTimeDuration	Instantiates the WFTimeDuration class.
WFTimeUnit	See the documentation on MSDN .