

Introduction to the Navisworks .NET API

Objectives

After completing this module, you will be able to:

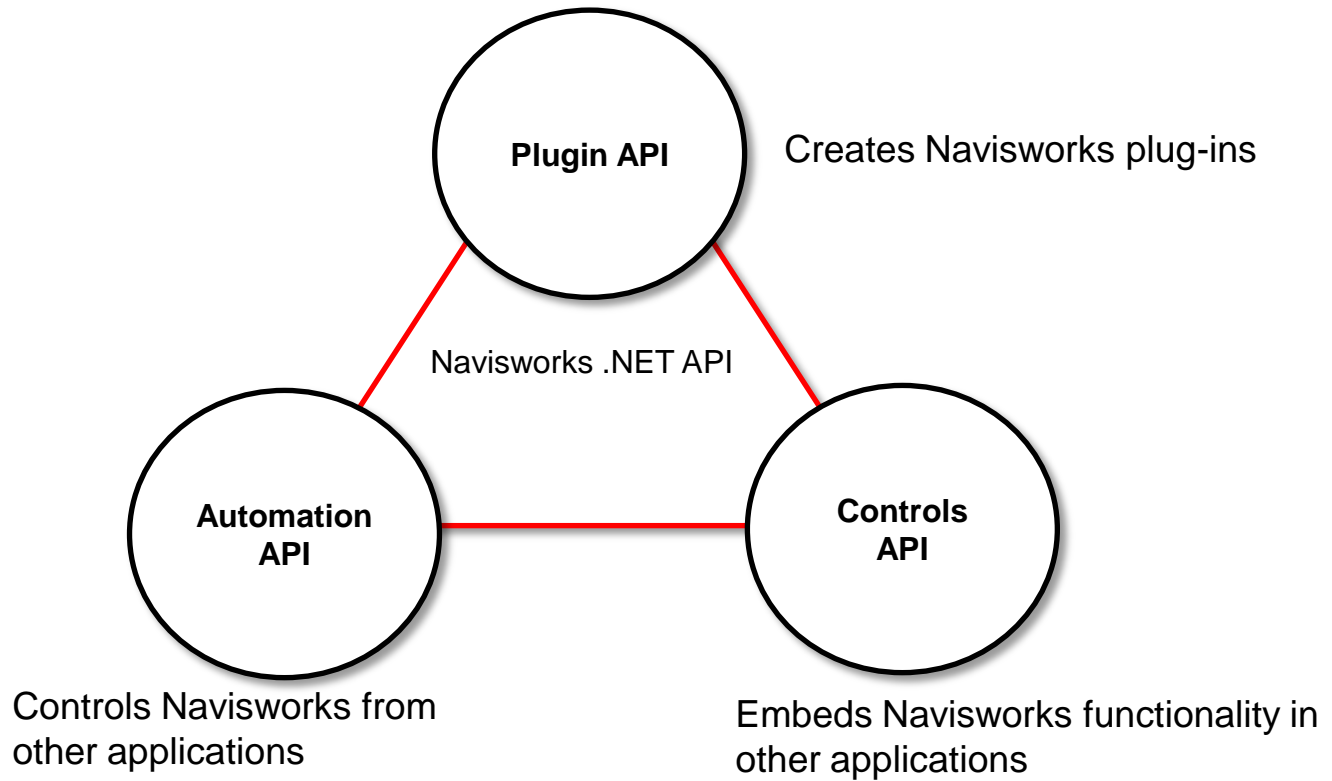
- Describe the different parts of the Navisworks .NET API and the types of applications they are used for
- Identify the libraries used for creating automation, plug-in and controls applications
- Identify the core classes in the .NET API

Prerequisites

You need the following to use the Navisworks .NET API:

- Working knowledge of Navisworks
- Good knowledge of a .NET programming language, preferably C# or VB.NET
- Microsoft Visual Studio or Visual Studio Express 2010
- Microsoft .NET Framework 4.0, if not already installed with Visual Studio

Navisworks .NET API Components



Navisworks Plugin API Objects

Most objects in the .NET API can be used in creating plug-ins

- Exceptions:

- Controls-related objects ([ApplicationControl](#), [DocumentControl](#), [ViewControl](#)) – used only in creating controls-based applications
- Automation application object ([NavisworksApplication](#)) – used only in creating automation-based applications

Navisworks Automation API

Only one API object is accessible via Automation:

- Automation application object ([NavisworksApplication](#)) – used in creating automation-based applications. No access to other objects in the API model
- Object has methods for opening, saving and appending files, and for executing plug-ins

Navisworks Control Objects

Navisworks .NET API has three control classes, used together in developing control-based applications:

- Autodesk.Navisworks.Controls.[ApplicationControl](#) – static class, used to initialize and terminate the API
- Autodesk.Navisworks.Controls.[DocumentControl](#) – a component class that hosts a Navisworks document, provides entry into the API hierarchy through the [Document](#) property
- Autodesk.Navisworks.Controls.[ViewControl](#) - a Windows Forms Control that displays the Document within a DocumentControl

Navisworks .NET API Assemblies

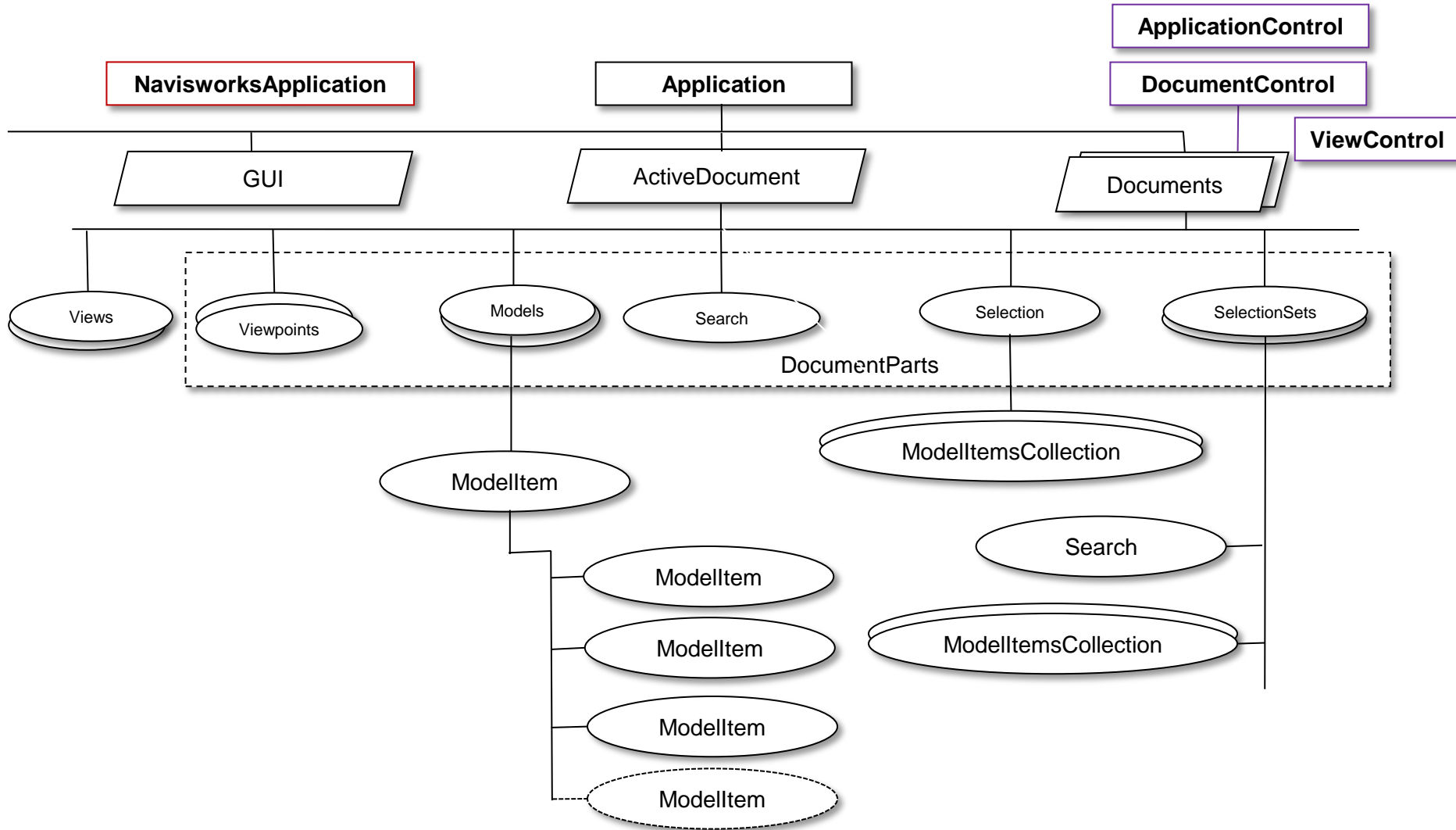
The .NET API is implemented in the following assemblies:

- [Autodesk.Navisworks.Api.dll](#) – core API, used for plug-in development and for controls-based applications
- [Autodesk.Navisworks.Automation.dll](#) – automation API
- [Autodesk.Navisworks.Controls.dll](#) – required for controls GUI

If the .NET API is to be combined with the COM API, you require:

- [Autodesk.Navisworks.Api.Interop.ComApi.dll](#) - .NET interop assembly containing all the COM API interfaces
- [Autodesk.Navisworks.ComAPI.dll](#) – assembly containing the [ComApiBridge](#) class, which supports conversion between COM and .NET API objects

Navisworks .NET API Objects



Automation

Controls

Application Classes

Navisworks .NET API has three application classes used in plug-ins and automation:

- Autodesk.Navisworks.Api.[Application](#) – static class, used only within plug-ins
- Autodesk.Navisworks.Api.Automation.[NavisworksApplication](#) – used in automation
- Autodesk.Navisworks.Api.ApplicationParts.[ApplicationAutomation](#) – same interface as [NavisworksApplication](#), but cannot be instantiated, and callable only within plug-ins

External (automating) applications call [NavisworksApplication](#), which in turn calls [ApplicationAutomation](#).

The automation application objects have no members for accessing documents and models

```
using Autodesk.Navisworks.Api.Automation;  
  
//Start Navisworks externally  
NavisworksApplication NWApp = new NavisworksApplication ();
```

Document Object

Document object represents the contents of a data file (NWD, NWC, etc.)

It allows opening, merging, appending, saving of files.

It is a container for models collection, current search, current selection, views and viewpoints.

Access the **Document** object through the application object:

- as the **ActiveDocument** property of **Application**
- as an element in the documents collection of **Application**

```
//Get the active document object  
Document doc = Autodesk.Navisworks.Api.Application.ActiveDocument;  
  
//Append a file with those already in the document  
doc.AppendFile("c:\\newFile.nwd");
```

DocumentParts

Classes representing objects stored in a document are grouped together in the [DocumentParts](#) namespace:

- [DocumentCurrentSelection](#) object containing the current selection
- [DocumentCurrentSearch](#) object containing the current search
- [DocumentCurrentViewpoint](#) object containing the current viewpoint data

[DocumentParts](#) objects are not created directly, but accessed as a property of a [Document](#) object.

```
//Get the active document object
Document doc = Autodesk.Navisworks.Api.Application.ActiveDocument;

//Get the current selection object
Autodesk.Navisworks.Api.DocumentParts.DocumentCurrentSelection curSel =
doc.CurrentSelection;

//Select all items in model
curSel.SelectAll();
```

Model and Model Collection objects

The collection of models in a document is held in the [DocumentModels](#) object.

[DocumentModels](#) object has property members for accessing individual models as well as item collections within the models.

```
//Get the active document object
Document doc = Autodesk.Navisworks.Api.Application.ActiveDocument;

//Get the model collection in the document
DocumentModels models = doc.Models;

//Get the first and second models
Model firstModel = models.First;
Model secondModel = models.Item(1);

//Get the file name of the first model
String fileName = firstModel.FileName;

//Get the collection of root items in all models
ModelItemEnumerableCollection rootItems = models.RootItems;
```

ModelItem Objects

A **ModelItem** object represents an object within a model hierarchy, corresponding to an item in the Navisworks selection tree.

It may have parent and child **ModelItem** objects.

ModelItem objects are accessed through the model's root item.

```
//Get the active document object
Document doc = Autodesk.Navisworks.Api.Application.ActiveDocument;

//Get the model collection in the document
DocumentModels models = doc.Models;

//Get a model - first model
Model model = models.First;

//Get the root item in the model
ModelItem rootItem = model.RootItem;

//Get a collection containing the root item and its descendants i.e. all ModelItem objects in the
model
ModelItemEnumerableCollection modelItems = rootItem.DescendantsAndSelf;

//Get individual model items by iterating through the model item collection
foreach(ModelItem modelItem in modelItems)
{ }
```

Combining .NET with COM API

The .NET API often has to be combined with the COM API, depending on the scope of application.

In addition to the .NET API assemblies, you require:

- [Autodesk.Navisworks.Api.Interop.ComApi.dll](#) - .NET interop assembly containing all the COM API interfaces
- [Autodesk.Navisworks.ComAPI.dll](#) – assembly containing the [ComApiBridge](#) class, which supports conversion between COM and .NET API objects

Summary

The .NET API is made up of plug-in, automation and controls components.

The root class of the plug-in API is [Application](#), through which document and model objects are accessed.

[Document](#) object provides access to [Model](#) object, which in turn provides access to [ModelItem](#) objects.

Create instance of the [NavisworksApplication](#) class to use the automation API.

Navisworks control API has three classes; [ApplicationControl](#), used for initializing and terminating the API, [DocumentControl](#), which hosts the Document, and [ViewControl](#), used for displaying the Document in the DocumentControl.

Further Information

Reference .NET API documentation is installed in <Navisworks>\api\net\documentation:

- Combined Navisworks API Reference and Developers Guide (NET API.chm)
- Code samples in <Navisworks>\api\net\examples