



# Navisworks API Training

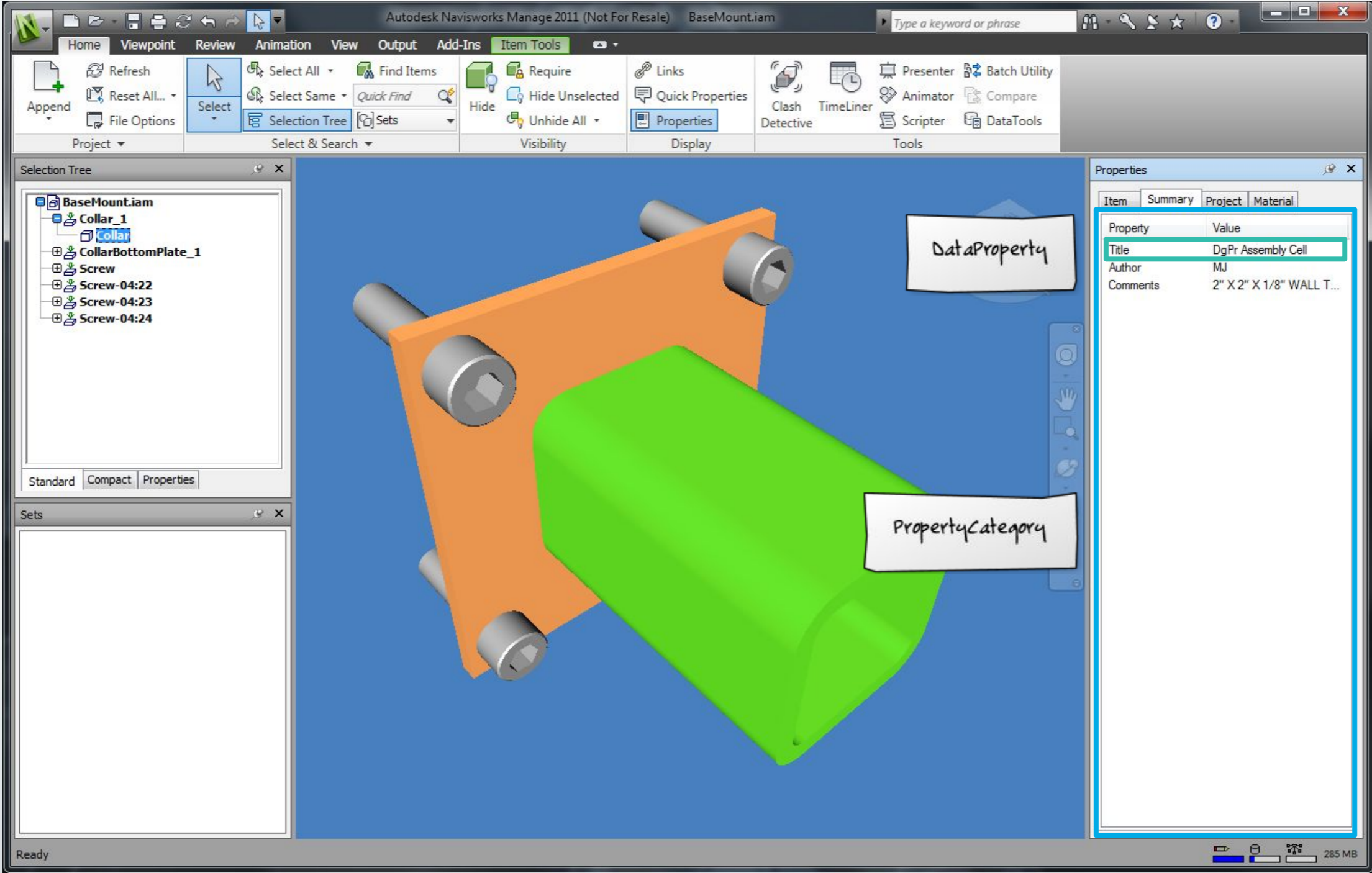
## Lab 5 – Properties



# Agenda

- Object Properties
- Find Properties

# Object Properties



# .NET API

- **ModelItem.PropertyCategories**
  - PropertyCategoryCollection
    - PropertyCategory
- **PropertyCategory.Properties**
  - DataPropertyCollection
    - DataProperty



- Current .NET API cannot modify/add custom properties. Need COM API

# Properties Name

- **Meaningful name**
  - Independent of languages
  - Pre-define strings, easy to use
- **Internal name**
  - Independent of languages
  - Named from the class definition of API
- **Display name**
  - What the end users can see in
  - Localization

Meaningful Name	Internal Name	Display Name (English)
PropertyCategoryNames.Item	LcOaNode	Item
PropertyCategoryNames.AutoCadEntityHandle	LcOpDwgEntityAttribs	Entity Handle
DataPropertyNames.Layer	LcOaNodeLayer	Layer
DataPropertyNames.AutoCadEntityHandleValue	LcOaNat64AttributeValue	Value

# NamedConstant & VariantData

- **NamedConstant**
  - Identifier for a value combining the internal constant's name and a localized display name equivalent
- **VariantData**
  - A value type that can store data of one of several different types
  - **DataProperty.Value**
    - Check type before print out the value

# Iterate Properties Demo

```
Document oDoc = Autodesk.Navisworks.Api.Application.ActiveDocument;

if (oDoc.CurrentSelection.SelectedItems.Count > 0)
{
    StringBuilder output = new StringBuilder(1000);
    output.Append("Dump Property Category of Current Selected Item\n");
    //dump the first item only
    ModelItem oItem = oDoc.CurrentSelection.SelectedItems[0];
    foreach (PropertyCategory oPC in oItem.PropertyCategories)
    {
        //each category
        output.Append(" Display Name: " + oPC.DisplayName + " Internal Name" + oPC.Name + "\n");
        output.Append(" Properties\n");
        foreach (DataProperty oDP in oPC.Properties)
        {
            //each property
            output.Append(" [Display Name]: " + oDP.DisplayName + "[Internal Name]:" + oDP.Name);
            if (oDP.Value.IsDisplayString)
                //if the value is display string
                output.Append("[Value]: " + oDP.Value.ToString() + "\n");
            else if (oDP.Value.IsDateTime)
                //if the value is a date
                output.Append("[Value]: " + oDP.Value.ToDateTime().ToShortTimeString() + "\n");
            else
                //other types
                output.Append("<Other types of values>" + "\n");
        }
    }
}
```

# Get Property by Method

- PropertyCategoryCollection
- Various methods to get property category and property, e.g.
  - FindCategoryByDisplayName
    - find property category by display name only
  - FindCategoryByName
    - find property category by internal name only
  - FindPropertyByDisplayName
    - find property by display name only
  - FindPropertyByName
    - find property by internal name only



# Demo: Get Categories by Display Name

```
ModelItem oSelectedItem = oDoc.SelectedItems.ElementAt<ModelItem>(0);

//get property category by display name
PropertyCategory oPC_DWGHandle =
    oSelectedItem.PropertyCategories.FindCategoryByDisplayName("Entity Handle");

//by internal name
PropertyCategory oPC_DWGHandle1 =
    oSelectedItem.PropertyCategories.FindCategoryByName
    (PropertyCategoryNames.AutoCadEntityHandle);

//by combined name
PropertyCategory oPC_DWGHandle2 =
    oSelectedItem.PropertyCategories.FindCategoryByCombinedName(new
    NamedConstant(PropertyCategoryNames.AutoCadEntityHandle, "Entity Handle"));
```

# Demo: Get Categories by Methods

```
ModelItem oSelectedItem = oDoc.SelectedItems.ElementAt<ModelItem>(0);
```

```
//get property category /by display name
```

```
PropertyCategory oPC_DWGHandle =  
    oSelectedItem.PropertyCategories.FindCategoryByDisplayName("Entity Handle");
```

```
//by internal name
```

```
PropertyCategory oPC_DWGHandle1 =  
    oSelectedItem.PropertyCategories.FindCategoryByName  
    (PropertyCategoryNames.AutoCadEntityHandle);
```

```
//by combined name
```

```
PropertyCategory oPC_DWGHandle2 =  
    oSelectedItem.PropertyCategories.FindCategoryByCombinedName(new  
    NamedConstant(PropertyCategoryNames.AutoCadEntityHandle, "Entity Handle"));
```

# Demo: Get Properties by Methods

**//by display name (property category and property)**

```
DataProperty oDP_DWGHandle =  
    oSelectedItem.PropertyCategories.FindPropertyByDisplayName("Entity Handle", "Value");
```

**//by internal name**

```
DataProperty oDP_DWGHandle1 =  
    oSelectedItem.PropertyCategories.FindPropertyByName(PropertyCategoryNames.AutoCadEntityHandle,  
        DataPropertyNames.AutoCadEntityHandleValue);
```

**//by combined name**

```
DataProperty oDP_DWGHandle2 =  
    oSelectedItem.PropertyCategories.FindPropertyByCombinedName(  
        new NamedConstant(PropertyCategoryNames.AutoCadEntityHandle, "Entity Handle"  
        new NamedConstant(DataPropertyNames.AutoCadEntityHandleValue, "Value")));
```

**//display the value of the DWG handle**

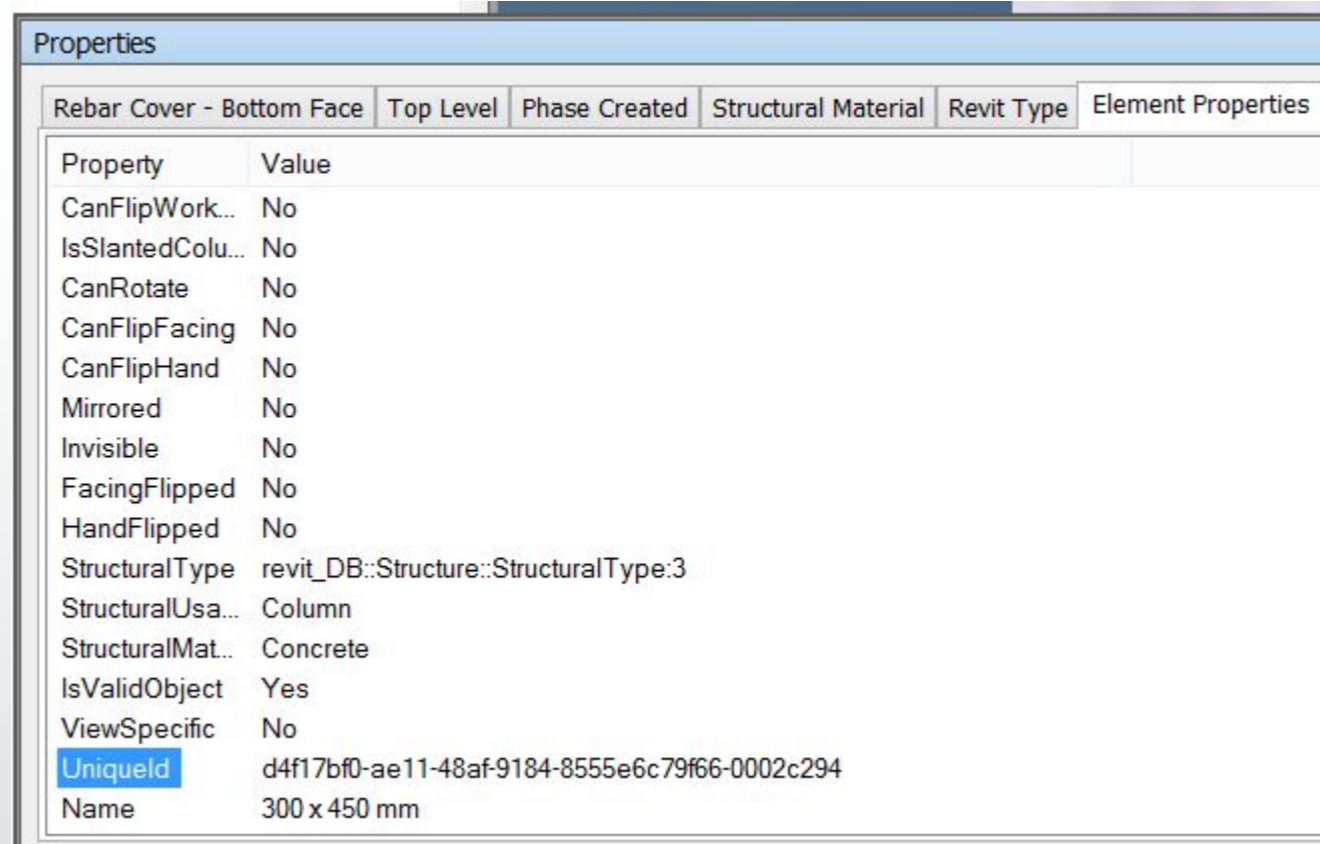
```
System.Diagnostics.Debug.Write(oDP_DWGHandle.Value.ToString());
```

# Add Custom Properties

- See Lab 09 [COM Interop]

# Unique Identifier of Object

- **ModelItem.InstanceGuid**
  - Same to Item>>Guid in UI
  - Available for some file formats only (Revit, AutoCAD)
- **Properties from source CAD file**
  - Some format provides a kind of ID that could be an identifier such as AutoCAD Handle, Guid of Microstation.
- **Revit file**
  - Recommend with Elements Properties>>Uniqueld.



The screenshot shows the Revit Properties palette for a rebar cover. The 'Element Properties' tab is selected, and the 'Uniqueld' property is highlighted. The table below lists the properties and their values.

Property	Value
CanFlipWork...	No
IsSlantedColu...	No
CanRotate	No
CanFlipFacing	No
CanFlipHand	No
Mirrored	No
Invisible	No
FacingFlipped	No
HandFlipped	No
StructuralType	revit_DB::Structure::StructuralType:3
StructuralUsa...	Column
StructuralMat...	Concrete
IsValidObject	Yes
ViewSpecific	No
<b>Uniqueld</b>	d4f17bf0-ae11-48af-9184-8555e6c79f66-0002c294
Name	300 x 450 mm

# Exercise

- Create a plugin
- Check the language of Navisworks.
- Use the display name of local language to find some properties

