120638

What the Heck is Custom Object

Kimberley Hendrix
D3 Technologies

Learning Objectives

- Describe what a custom object is and how it can be used
- Configure a custom object in Vault Professional
- Use Data Standard to create a form to easily populate information for a custom object
- Use Custom Objects to link together various types of data

Description

The focus of this class will be to effectively use Custom Objects in Autodesk Vault Professional to replace spreadsheet based tracking systems. In addition, you will become familiar with other opportunities to put Custom Objects to work in your environment.

In the class I will introduce the use of custom objects to replace spreadsheet type data mining, or desktop database tracking systems, while linking the relevant data reducing the redundancy in searches. I will also introduce briefly how to use Vault Data Standard to make Custom Objects work with ease. Lastly using custom objects to enhance the custom numbering systems inherit in Vault today.

Speaker(s)

Based in Tulsa, Oklahoma, Kimberley Hendrix provides custom solutions for lean engineering using Autodesk, Inc., products and industry knowledge to streamline design and engineering departments. Hendrix has worked in the manufacturing/plant industry for over 30 years and she specialized in automated solutions for the heat exchanger industry. She has worked with Autodesk products since 1984. Hendrix is associated with D3 Technologies as a solutions consultant, focusing on data management, plant, automation, and project management.

Just what is a Custom Object

Autodesk definition:
Custom Objects allow Vault administrators to use an extensible system to create new Vault entities to meet the needs of their team, organization, or company. Right out-of-the-box, the administrator can create a Custom Object definition and assign it categories, lifecycles, and properties. Once the Custom Object definition is created, users can create instances of that custom object directly through the user interface. As with files and folders, users can perform many common Vault tasks with custom objects.

My Definition:
Custom Objects is a table of information residing in the SQL database that allows you to manage lists and collections of information directly in the Vault interface. Allowing for all of your data to be managed in one central location.

Some Example uses that we will explore:
1. Task List
   a. Drawing assignments
   b. Project Review
2. Contacts
   a. List your designers and use to assign the above
   b. Link other files/custom objects to a contact
3. RFQ – containers of files
4. Sequential Drawing numbers

Configuring a Custom Object

Let's start by configuring a custom object called Task. This is a good custom object to learn with because the Data Standard (we will do some data standard setup a little later) settings for this are preconfigured and comes with the download and installation of Data Standard.
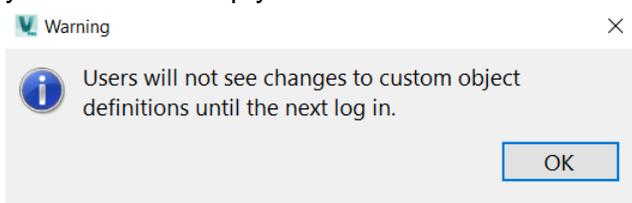
Tools > Administration > Vault Settings
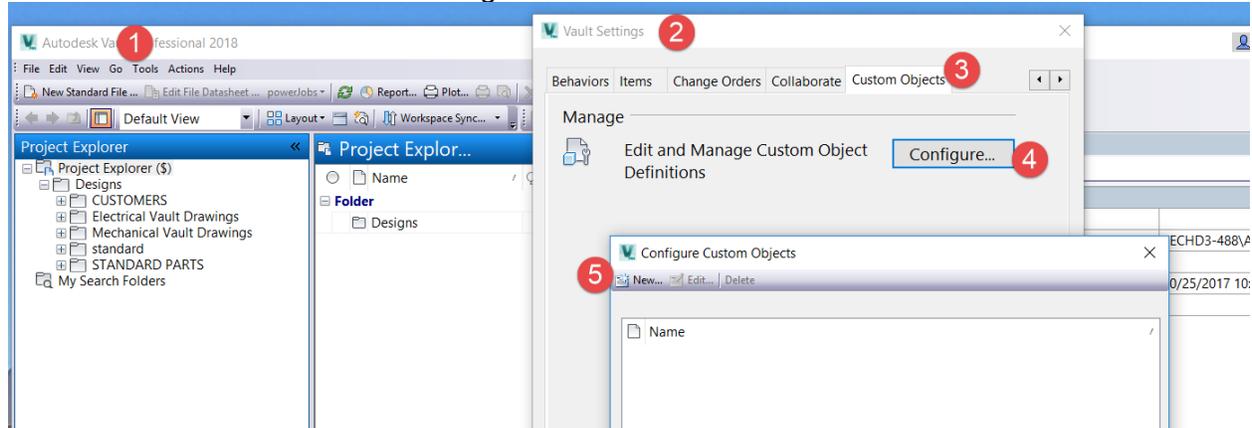
Create a new Custom object and name it Task you can choose any icon (*.ico) file that you have on your system to define it.
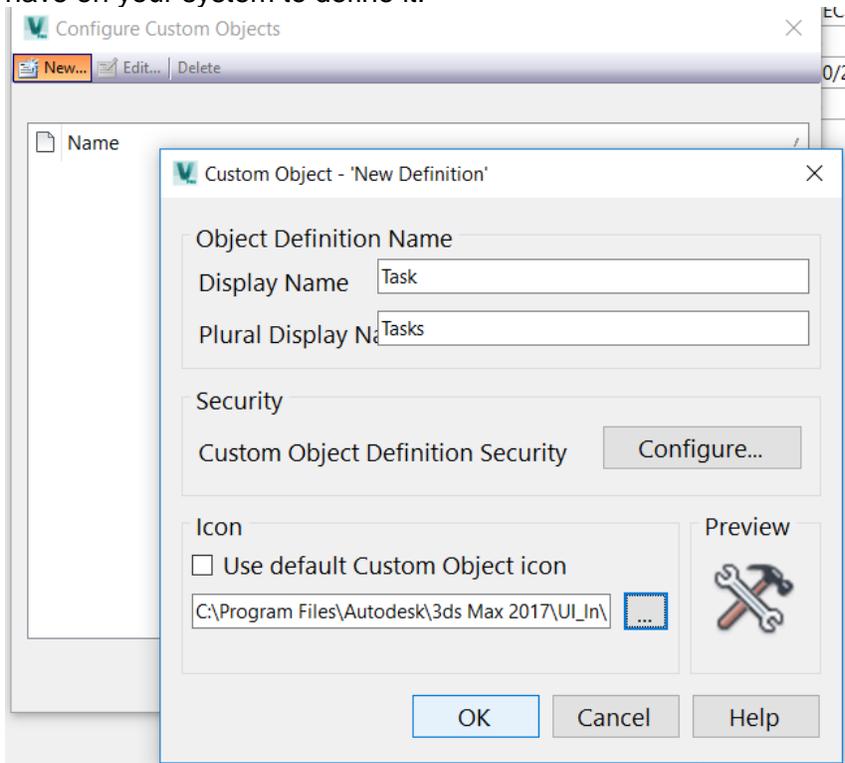
After you hit OK, you will receive this warning, it is normal and before you see the definitions on your Vault desktop you will have to close Vault and reopen it.

When vault is restarted you will see the new custom object named Tasks in the group tabs. Note that if Data Standards is already installed you will have to restart the vault twice. The first time to recognize the new custom object and the second time to load the corresponding Data Standard dialogs after the new custom object is defined.
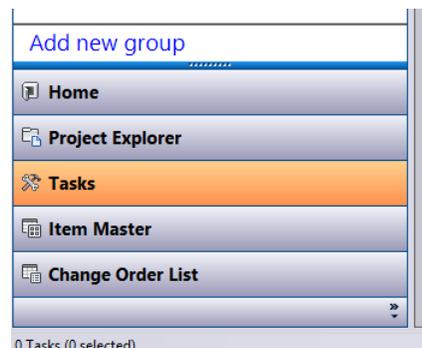
Custom objects can take on categories, lifecycles, properties and securities just like other entities inside vault. Properties can be assigned to the custom object category and can appear in the dialog boxes to edit them properly.

The main window for Custom Objects is customizable the same as the window for files so the properties that are important for you can be shown in the main window.

Creating and Editing a Custom Object

Once the custom object is configured it will function like a file. If you are not using Data Standard, to create a custom object hit the New button just like you would for a file or folder, a dialog asking for the Name of the object will appear.
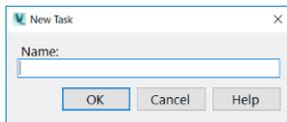
If you are using data standard and have it configured for the custom object both the new and edit dialog boxes will look something like this.
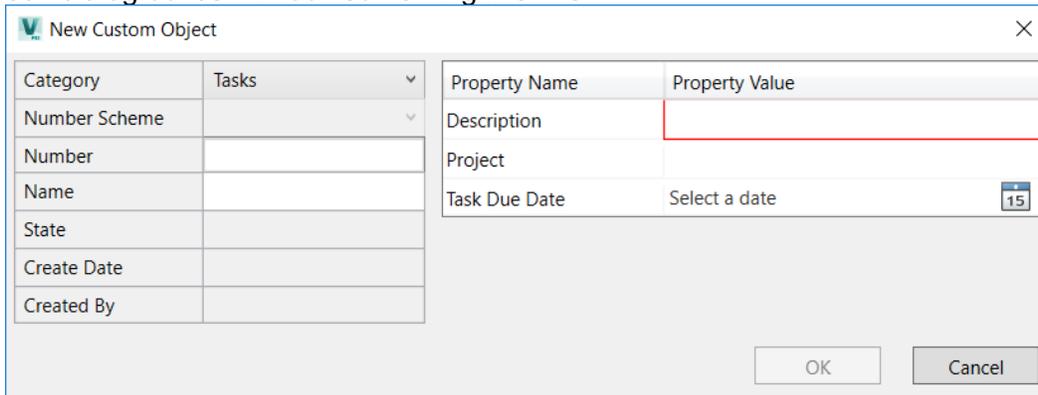
Allowing you to preselect the Category, number schemes and populate the properties that are a mapped to the object.

After the object is created you can then attach links to files or other objects to the record. This is a simple copy / paste as link operation. Select the file or object that you want associated with the record then choose Edit > Copy. Navigate back to your custom object select the Contents tab, then Edit > past as link.

In the Vault Administration settings, Behaviors tab - Categories, Revision Schemes, Lifecycles and Properties can all be configured and mapped to each of your different types of Custom Objects. Taking full advantage of the securities available to you inside Vault.

Benefits of using a Custom Object

After you have configured a new Custom Object, in this instance we will talk about task, and a basic one I created called Contacts. Think of each of these as an Excel spreadsheet that you are filling in to track information and stay updated on. You can 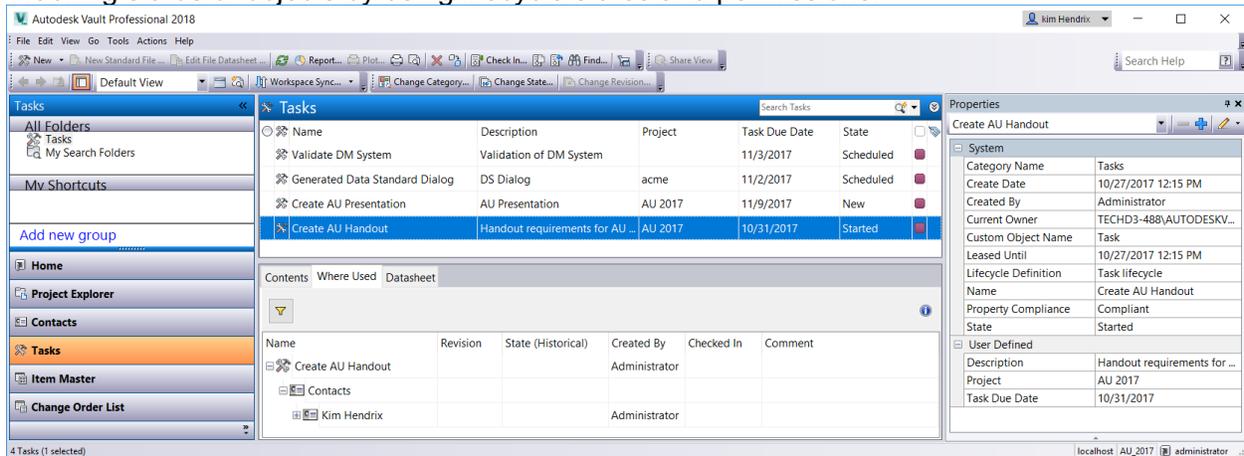certainly do this in Excel, but it won't manage the history of that status, or who did it, or track dates unless you type them in.

With a properly configured Custom Object and an associated lifecycle workflow Autodesk Vault will manage that data for you we can also link the task to the associated contact allowing them to see all of their tasks and status' in one place.
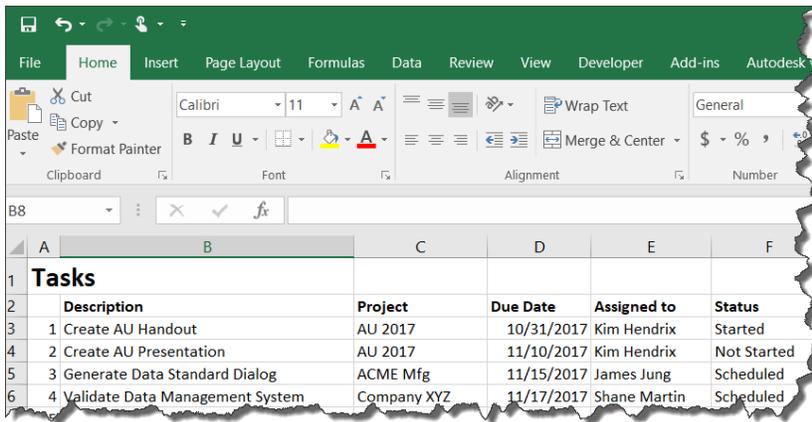
Linking custom objects together like task with contacts, or project objects with drawing files. Tracking status of objects by using lifecycle states and permissions.

When using Custom Objects for a Task/Contacts arrangement the linking works both ways so if a contact has been assigned multiple tasks, going to the contacts list and selecting their name will list all tasks assigned and their associated statuses.

	Name	Revision	State	Created By	Path
	Kim Hendrix				☐
	Contents **Where Used**				
⊟	**Task**				
	Validate DM System		Scheduled	Administrator	Contacts/Kim Hendrix
	Create AU Handout		Started	Administrator	Contacts/Kim Hendrix

Setting up Data Standard for use with Custom Object

Data Standard extension is provided to subscription customers and is a separate download and installation. You must install the extension for Vault, AutoCAD and Inventor each separately. When the Vault extension is installed at *C:\ProgramData\Autodesk\Vault 2018\Extensions\DataStandard* there are two folders under there *Vault* and *CAD* each with subsequent *AddInVault / AddIn* and *Configuration* folders.
In the AddInVault and the menu and powershell script files (ps1) under the Configuration folder are the XAML file that are the dialog box files.

Three components are required for custom object support:

- Menu entries for the create and edit dialog.
- Dialog files (.xaml) for the dialogs and tabs.
- Scripting files (PowerShell) for the dialog logic.

All components required to manage a custom object called "TASK" are pre-configured so if you install Data Standard, and then create a custom object called TASK all if this works, like magic.

Now when you are ready to create a different named custom object you start with the TASK custom object and copy those files to the newly named object. Let's assume you are creating a Custom Object named RFQ the following are the steps required to setup the Data Standard

1. Create the Custom Object inside Vault
2. Menu Entries
    a. *C:\programdata\Autodesk\Vault 2018\Extensions\DataStandard\Vault\MenuDefinitions.xml*
    b. *Copy the definitions MenuItem New Task, MenuItem EditTask*
    c. *Replace the word Task with RFQ in all variables*
    d. *Modify the Descriptions, Hints & Labels*
    e. *Same file create a CommandSite entry for the RFQ*

```
<NewTask Label="CreateRFQ..."
 Description="RFQ"
 Hint="Create a new RFQ object"
 PSFile="CreateRFQObject.ps1"
 Image="NewCustomObject.ico"
 ToolbarPaintStyle="TextAndGlyph"
 NavigationTypes="RFQ"
 MultiSelectEnabled="false"/>

<EditTaskLabel="Edit RFQ..."
 Description="Edit RFQ"
 Hint="Edit RFQ object"
 PSFile="EditRFQObject.ps1"
 Image="EditCustomObject.ico"
 ToolbarPaintStyle="TextandGlyph"
 NavigationTypes="RFQ"
 MultiSelectEnabled="False"/>

</CommandSite>
 <RFQ Label="RFQ" DeployAsPullDown="True" Location="RFQ">
    <Item Name="NewRFQ"></Item>
    <Item Name="EditRFQ"></Item>
 </RFQ>
```

3. Scripting Files
    a. *Folder C:\programdata\Autodesk\Vault 2018\Extensions\DataStandard\Vault*
        i. *Copy Files CreateCustomObject.ps1 and EditCustomObject.ps1 to the names references in the menu steps above*
    b. *These PS1 files point to related XAML files (dialog files) edit the lines in the new PS1 files*
```
...
$xamlFile = New-Object CreateObject.WPF.XamlFile "RFQXaml", "%ProgramData%\Autodesk\Vault 2018\Extensions\DataStandard\Vault\Configuration\RFQ.xaml"
$dialog.XamlFile = $xamlFile
...
```
    c.
4. *XAML File*
    a. *Folder C:\programdata\Autodesk\Vault 2018\Extensions\DataStandard\Vault\Configuration*
    b. *Copy the CustomObject.xaml file and name it RFQ.xaml (must match line is script file)*
    c. *Modify the new XAML file replacing all occurrences of "TaskWindow" with "RFQWindo"*
    d. *Correct the dialog title by by editing the UiStrings.xml and replacing the IDs with the new ID's text strings*
    e. *Add or remove property controls as needed*
5. *DataSheet Tab*
    a. *Copy the Folder C:\ProgramData\Autodesk\Vault 2018\Extensions\DataStandard\Vault\Configuration\Task to C:\ProgramData\Autodesk\Vault 2018\Extensions\DataStandard\Vault\Configuration\RFQ*
    b. *Modify Datasheet.xaml if necessary*

After completion of all steps Autodesk Vault will have to be restarted in order for the configurations to take effect.

Using a custom object to fill in data in another form

(more info to come)