AUTODESK UNIVERSITY 2013

Whose Version Should We Believe?—Revising the Story with Autodesk® Vault

Mike Thomas – Prairie Machine & Parts Mfg.

PL1341

This class introduces you to the powerful tools that are available in Autodesk® Vault Workgroup and Autodesk® Vault Professional for revision management, approval workflows, and engineering change orders (ECO). Learn how categories are used to control the flow of your data and versions and revisions are intertwined. We also cover how to apply different revision schemes and how to initiate and track change via a change order.

Learning Objectives

At the end of this class, you will be able to:

- Identify and describe the differences between versions and revisions
- Apply Vault categories and file lifecycles to manage change
- Use ECOs to track revisions
- Explain the differences between file revision management and item revision management

About the Speaker

Mike graduated as the Most Distinguished Graduate from CAD/CAM Engineering Technology at SIAST, Kelsey Campus. He is specialist in the manufacturing industry with strong knowledge on the Autodesk mechanical products supplemented with a solid understanding of document management, hardware, networking, and other Autodesk technologies.

He is now the Technical Services Manager for Prairie Machine & Parts and is responsible for overseeing the engineering department's technical operations and the department's strategic technical growth. Primary duties include the ongoing support of critical computer systems and programs, the interactions between the department and between other departments, providing the engineering department with effective systems and technology, and working with the VP and Engineering Managers on the development and implementation of a cohesive strategic plan for the technical growth of the department

Autodesk Vault Revisions

Autodesk Vault Basic provides *Version* control but no tools to manage and track change. Vault Workgroup and Vault Professional move from a simple data management tool that just tracks versions to a system managing the change of your files, from creation to retirement

I like how Wikipedia describes engineering drawings.

> *Engineering drawing (the activity) produces engineering drawings (the documents). More than just the drawing of pictures, it is also a language—a graphical language that communicates ideas and information from one mind to another. Most especially, it communicates all needed information from the engineer who designed a part to the workers who will make it.*[1]

Our Engineering drawings (and models) are the language that we use to communicate from concept to design to manufacture / construction. How we manage this *language* and its transfer between parties is almost as important as the document itself

Looking at the three versions of Vault we have:

- Vault (Basic) = Version Management "*Organize, manage, and track data creation, simulation, and documentation.*"
- Vault Workgroup = Vault + "*Project lifecycles*": tools to manage change on files; workflows, lifecycles, and other revision management tools
- Vault Professional = Vault Workgroup + Items and tools for managing changes to items (Change Orders)

Webster's dictionary describes a ***revision*** as "*the act of revising, which is to make a new, amended, improved, or up-to-date* ***version***."

So what's the difference between a version and a revision? If I'm maintaining versions why do I care about revisions? Aren't they the same? In this course we're going to explore the differences and explain when and where and why you want to use versions, use revisions, and how they are integrated and work together

[1] From Wikipedia "Engineering drawing" - http://en.wikipedia.org/wiki/Engineering_drawing

Contents

AUTODESK VAULT REVISIONS **2**

WHAT'S THE VERSION OF THAT REVISION? **4**

FILE & FOLDER LIFECYCLES **5**

REVISION SCHEMES **8**

MANUALLY ADJUSTING THE REVISION LEVEL (FILES) **10**

REVISION BUMP 10

USING A FILE PROPERTY 10

USER SPECIFIED REVISION VALUE 11

ADJUSTING A GROUP OF FILES 12

AUTOMATIC REVISION BUMPING **13**

AUTOMATIC REVISIONS 13

CAD AND FILE LIFECYCLES **14**

ITEM LIFECYCLES **15**

WHAT ARE ITEMS? 15

ITEM CATEGORIES & REVISION MANAGEMENT CONFIGURATION 16

USING ITEM LIFECYCLES AND ITEM REVISIONS **18**

ADJUSTING THE REVISION LEVEL OF AN ITEM 18

LIFECYCLES - AUTOMATIC REVISION BUMPING? 19

PURGING ITEMS 19

GETTING THE PREVIOUS VERSION OF A REVISION 20

CHANGE ORDERS **21**

WHAT ARE CHANGE ORDERS? 21

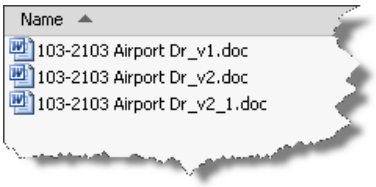
CREATING NEW CHANGE ORDERS (AND SUBMITTING FOR WORK) 22

COMPLETING THE REQUIRED CHANGES (AKA DOING THE WORK) 23

REVIEWING, APPROVING, AND SETTING EFFECTIVITY 23

What's the Version of that Revision?

Within Engineering it is important to maintain a history of a document (usually drawings) to see what has changed, when it was changed, and hopefully why it was changed. This history can aid in making better decisions (*hey look we already tried that*), provide crucial customer information (*oh, they have rev D this is what they need to do*), and provide the means to go back to a previous version of the document in case the change was inadvertent or did not make the improvements desired (aka made things worse)



It is not uncommon for organizations to keep multiple copies of a document labeled with a prefix or suffix identifying the revision of the document. This provides the desired history but is a manual process and prone to user error

A ***Version*** should be looked as an iteration of a document, like having multiple copies of it. Something that is different from the previous copy. Versions maintain the previous design in case you need to restore back to a previous copy at a future time

A ***Revision*** is a milestone, it's the act of making a change and completing the work required for that revision. A revision can be made up of multiple versions: the versions are the steps to get to the revision, the revision is what gets released to everyone else.

So we could maintain history (revisions) within Vault by appending the revision number to the file name. This would be a manual process but would create multiple copies of the document, but this would quickly become a nightmare. This isn't why we all moved to Vault, we did not want more nightmares.

So all flavours of Vault maintain versions of all files checked into Vault. The previous versions can be restored as the current version at any time. The number of versions maintained by Vault is determined by the administrator.

Versions are very important during the design and modification phase but once the project is completed and released they start to lose their importance. Versions represent Work In Progress (WIP) data that is used to keep the previous designs in case new ideas do not work or do not improve the design. As a version is a copy of the file, and files have size, they take up file store space. It's in the Vault Administrators best interest to keep the number of versions to a minimum.

A **revision** is a permanent version, it will never be purged from Vault (unless specifically deleted by an administrator). For example Revision B might be of Version 14 of a file. Revision "B" represents the milestone of completing the necessary changes to the document.

File & Folder Lifecycles

What are Lifecycles?

The Autodesk Wiki describes Lifecycles as

> *A lifecycle definition is an engine that can be configured to automatically assign security, behaviors, and properties to Vault objects based on where the object is in the life of the design process*.

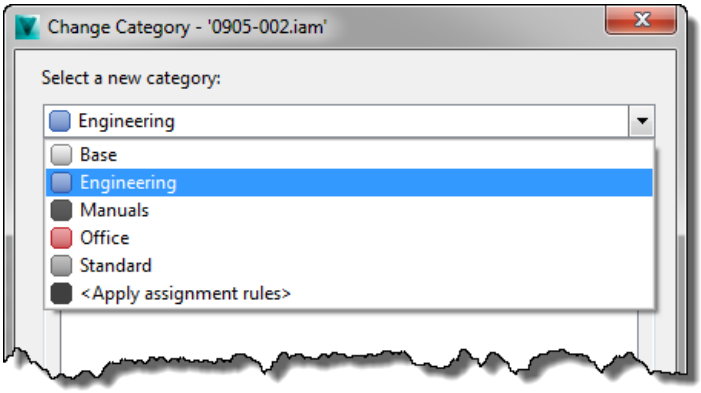
The goal here is not to explain Lifecycles in detail but just highlight their importance with how files & folders are revised and their revisions tracked

General Steps

1. Assign the Category
2. Change the State
3. Revise the File
4. Change the State
5. Repeat

What are Categories?

Vault **Categories** provide a grouping method for a set of files, folders, custom objects or items. They are a key aspect of using Vault lifecycles as they are configured to assign user-defined properties to the objects, the available lifecycle definitions, and **Revision schemes**



To change the **Category** assigned to the file, folder, or custom object select the object(s) and click the Change Category button. The subsequent dialog will allow you to select from the existing categories. [NOTE: Items are discussed later]

The "out-of-the-box" Revision Schemes

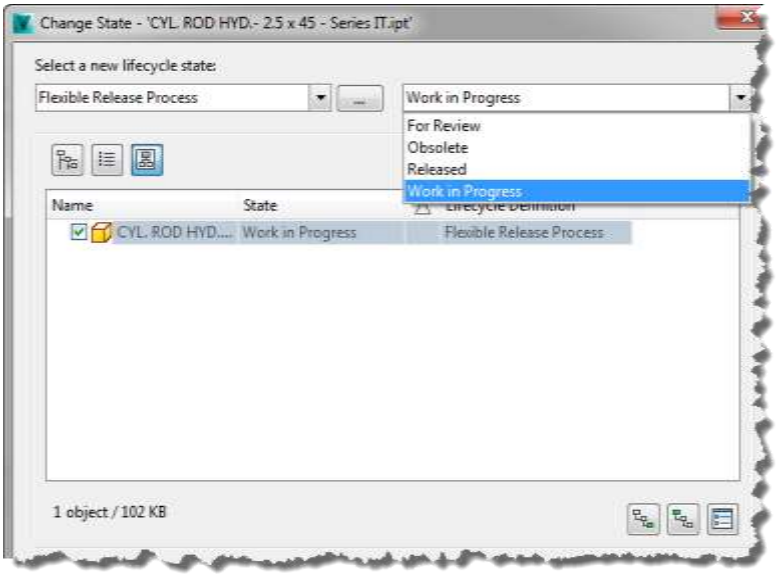
- **Standard Alphabetic (A, B, B.1, B.1.1 to ZZ)**
- **Default Alphabetic (same as standard but starts at rev "-")**
- **Standard Numeric (1, 2, 2.1, 2.1.1 to 99)**
- **Default Numeric (same as standard but starts at rev "-"))**
- **Default ASME Y14.35M (-, A, B, A.1, A.1.1 | no I, O, Q, S, X, Y)**[2]

[2] The American Society of Mechanical Engineering (ASME) standard that defines the practices for revising drawings and associated documentation, includes the methods of identification and revision documentation.

States

With Vault Lifecycles the **state** identifies the current status within the lifecycle, for example Work in Progress (WIP) or Released. The number and type of states the object transitions thru in its lifecycle is based on the **Lifecycle Definition Transition Rules**.

To change the state select the file, folder, or custom object and click the **Change State** button. In the Change State dialog you can adjust the lifecycle state (what's listed is controlled by the category assigned to the object) and the State you want to switch the object to. Depending on the settings you might also notice the revision value change as well



Files

1. Check the Category and adjust as necessary – *Categories* can be set automatically by *Rules* so this step may not be required.
2. Adjust the Revision level, especially when moving legacy data into Vault
3. Change the State when ready to move to the next step

Folders & Custom Objects

Folders are similar to Files except you cannot set the revision. After applying the correct Category you can adjust the state of the folder. The state change configuration may force you to perform a state change on the files contained within the folder before changing the state

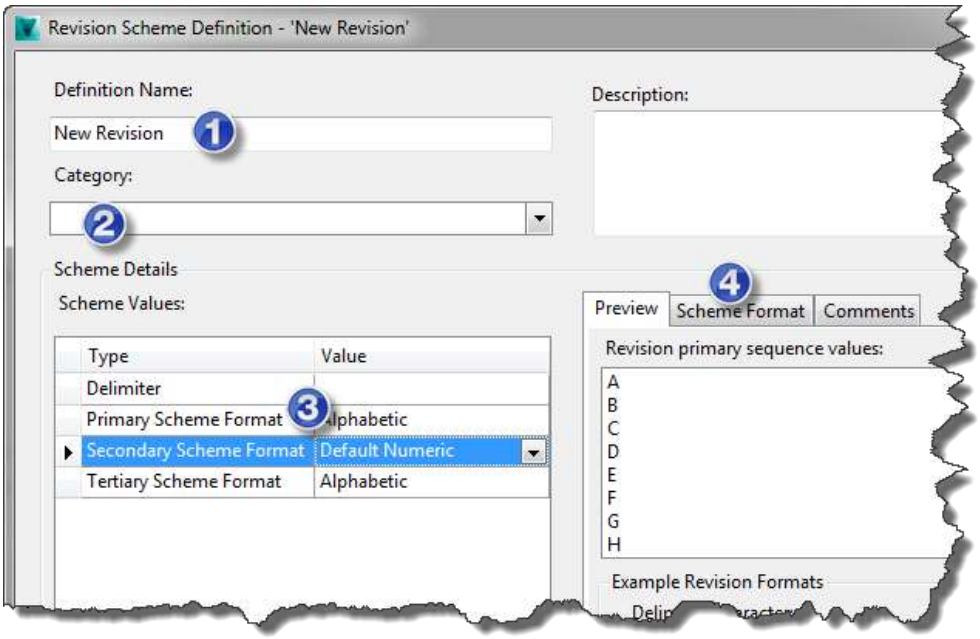
Custom Objects are similar to Folders in that you cannot set the revision, only adjust categories and states. Categories do not have sub-content rules to worry about like folders and files.

Revision Schemes

Revision Schemes define the Revision value applied to the file and the sequence of subsequent values. You specify how the primary, secondary, and tertiary revisions appear. The scheme forces the users into predefined values, so they can not stray away from the standard.

For example the *Standard Numeric Format* (one of the ones that is out-of-the-box) starts at 1 and increments by 1, not skipping any numbers.

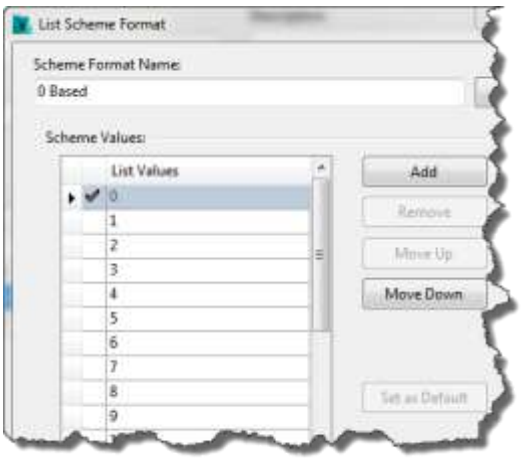
To get started access the Vault Settings dialog (Tools > Administration > Vault Options) and navigate to the **Behaviors** tab, "*Edit and Manage Lifecycles and Revisions*" section, and click the **Revisions…** button. [NOTE: *You can either copy and existing scheme or start from scratch*]



1. Name the new Revision Definition and enter a description
2. Select the Categories in which you want the new scheme to be available. This can include files and items.
3. Assign the three(3) Scheme Formats (Primary, Secondary, and Tertiary) and the Delimiter
4. If desired Scheme Format is not available, create a new Scheme Format

The Scheme Definition is simply the list of values available to the Revision. In most cases it's the easiest to copy an existing scheme and tweak it, otherwise if you start from scratch you'll be defining each and every value in the scheme

> Once a scheme is used it CANNOT be modified so make sure it's what you want before you start using it



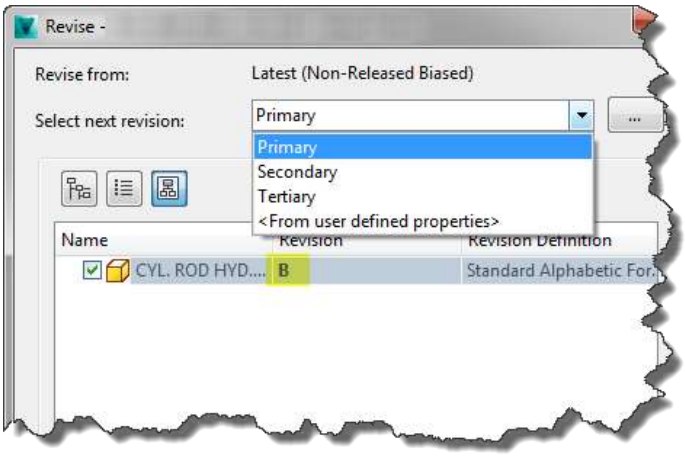
For example, we start at Rev 0 which is the initial release to manufacturing. The "out-of-the-box" numeric options either start at – or 1. We created a copy of the Default Numeric called 0-BASED which is identical but starts at 0 instead of –

You can mix schemes in the Revision Scheme Definition, for example, using "Alphabetic" for Primary, "Numeric" for Secondary, and "0-BASED" for Tertiary.

Manually Adjusting the Revision Level (Files)

Revision Bump

To “bump” the revision to the next available value in the current revision scheme select the file(s) and click the **Revise** button. Select the level you want to Bump (i.e. are you going from 1 to 2 or 1 to 1.1) and the Revision value will automatically go to the next level.



> The ability to adjust the revision of the file will depend on the permissions assigned to your Vault User Account, the category the files belongs to, and the current state of the file.

Using a File Property

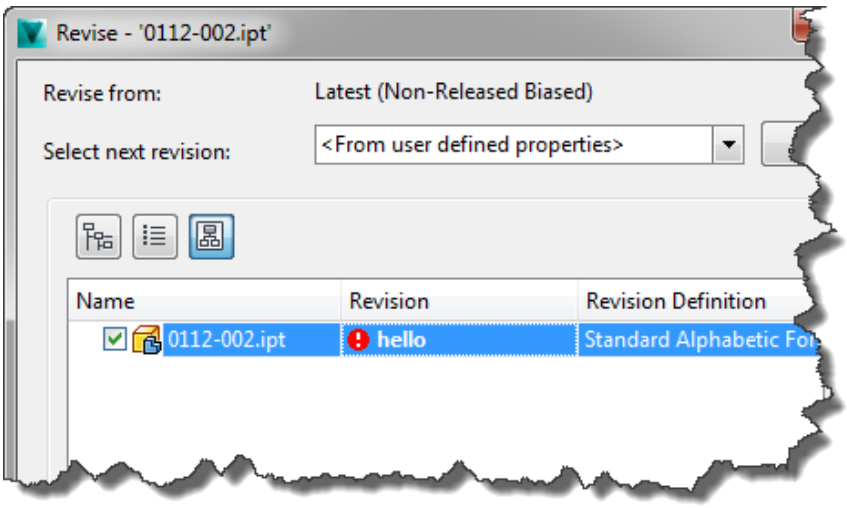
Any property of the selected file can be used to populate the Revision value IF the current property value fits within the Scheme Definition. For example I couldn’t use the description if my current scheme is A, B, C.

This can be useful when migrating data into the Vault. If the current revision is set to a property I can use that to set the initial Vault revision value.

User Specified Revision Value

To adjust the revision manually click in the Revision field and type the value you want. Again this value needs to “fit” within the scheme definition, I couldn’t enter “hello” if my revision scheme was A, B, C or 1, 2, 3.

In the image shown below note the red exclamation mark beside the revision level as the current entered value “hello” does not fit within the Standard Alphabetic Format.

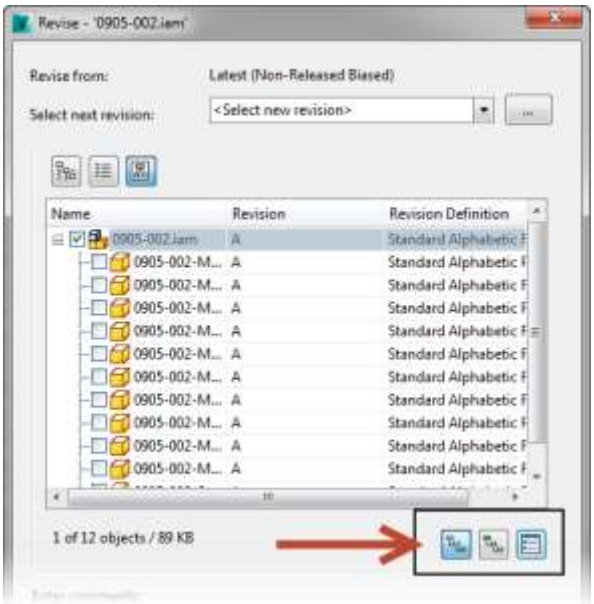


This is extremely useful with files with historical revisions occurring before they were put into Vault. If I add a drawing to Vault that is currently at Rev 14 I don’t want to have to bump the revision 14 times. I can Revise the file and set the revision to 14

> Once a Revision level is achieved there is no going backwards. If I take a file and manually adjust from 1 to 11 I will NOT be able to assign revision levels 2 thru 10 to it.

Adjusting a Group of Files

As Vault maintains file relationships of the files checked into Vault you can adjust the revision level on a file and all of it dependents or parents. Use the buttons at the bottom right-corner to Include Parents, Include Children, or other optional files like attachments.

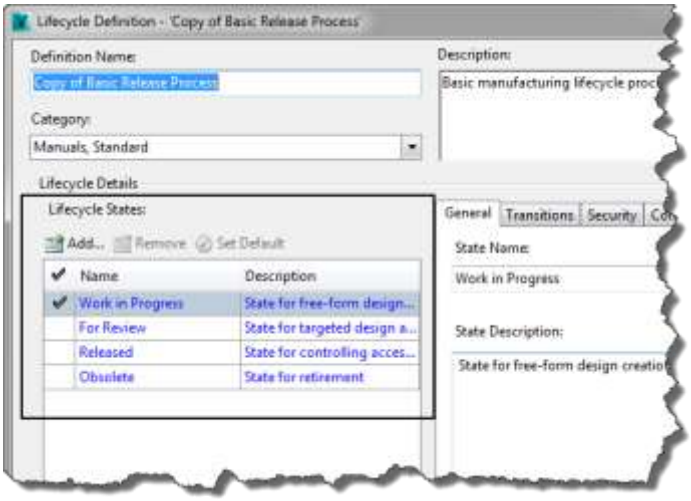


You can also select multiple files in the Main Grid view and adjust the revision of all the selected files in one step. This will “bump” each file’s revision, NOT set them all to be the same

Automatic Revision Bumping

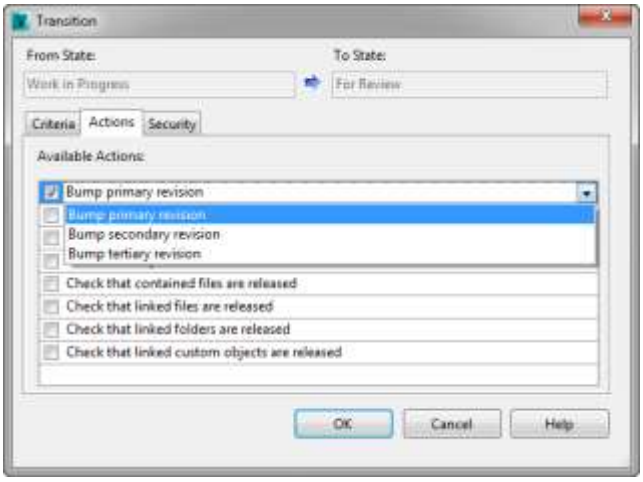
Automatic Revisions

Lifecycle **States** identify the current status of your file within the Lifecycle Definition. A very basic workflow would be two states (like Work in Progress and Released). By themselves the Lifecycle States are simply labels which represent various stages that you want your files to transition through.



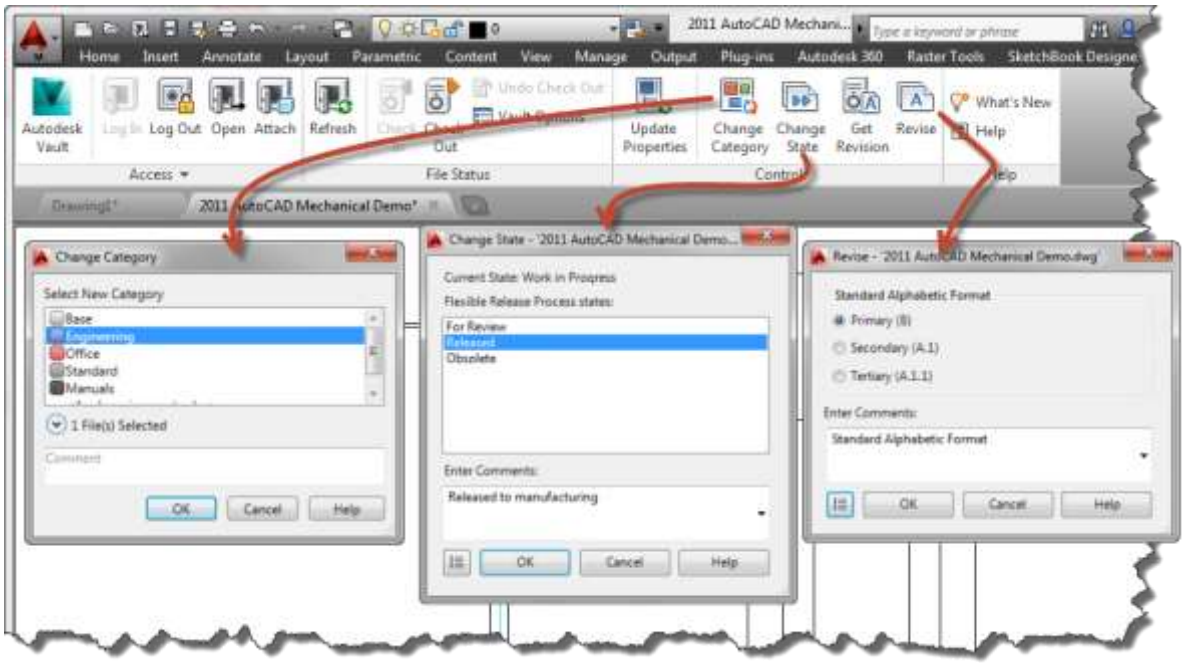
The **Transition** of a State determines many things, but with Revisions it manages the Actions that happen during the state change (like revision bump), and the Security, as in who can make the specific state change

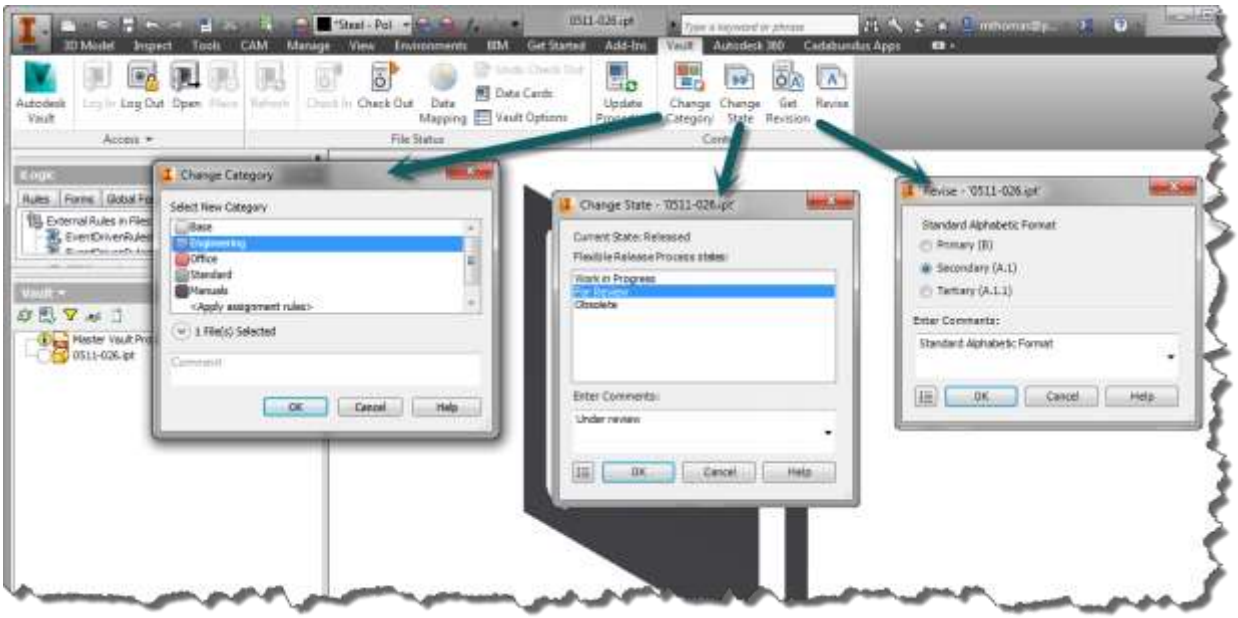
In the State configuration dialog use the *Actions* tab to define what occurs during a state change and the Security tab to specify who can make the transition state change.



CAD and File Lifecycles

When using the Vault CAD add-in you can perform lifecycle functions from within certain Autodesk applications. This includes changing Categories, Changing States, and Revising





The CAD Applications will also understand when files are Released and locked from edit

Item Lifecycles

What are Items?

Like how a recipe or business card represents information and provides a method for organization and quickly locating, items represent things within our business. **Items** represent the components that the company manage, assemble, sell and manufacture.

Each item is identified by its own unique item number (or sometimes referred to as part number). Items can represent a variety of things including parts and assemblies, instructional guides, consumable goods (fluids, lubricants, etc), and any related Bill of Material. Some items might be purchased, while others are manufactured internally.

Within ***Vault Professional***, items are records stored within an item master. The item master is a master list of everything within the engineering system. Each item is assigned an unique item number (identifier) which is used to locate, edit, update, and track changes to an item.

Items are used to:

- Provide access to the data for everyone who needs access. This will also include non-engineering staff without access to a CAD application.
- Provide a mechanism for the exchange of data between systems
- Issue new numbers (i.e. stock numbers)
- Release Management (lifecycles)

NOTE: Items and the Item Master are ***ONLY*** available within Vault Professional

Should you use items? Ask yourself these questions…

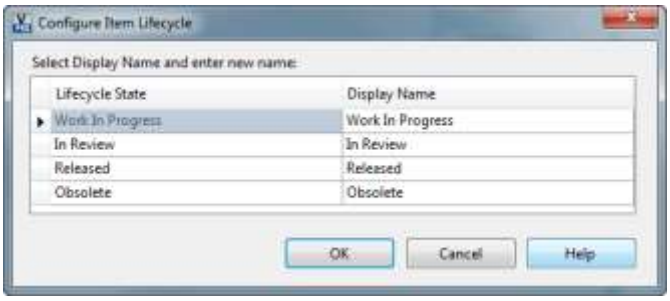
- Do I need to deliver data to a downstream application, like MRP, ERP, PLM, etc, etc?
- Do non-engineering or non-cad people need access to data about my parts and assemblies? Possibly manipulate Bill Of Material data?
- Do I want a method to manage revisions? And control what downstream users see of my data? Do I want to use Change Orders to control and track revisions?
- Do I want to manage CAD AND non-CAD things? Do I want to release data locking it from access?

Item Categories & Revision Management Configuration

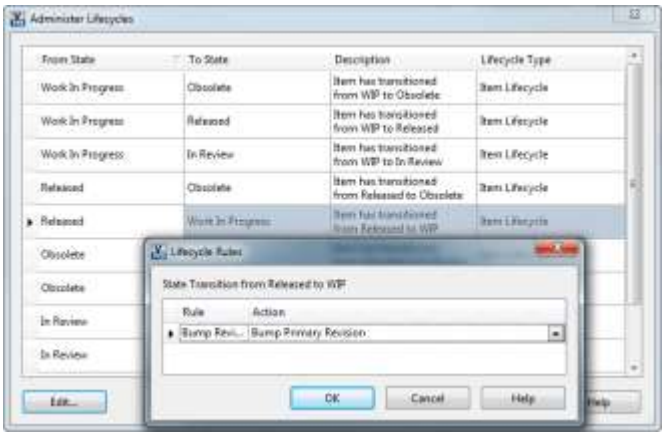
Categories provide a mechanism to classify items to make it easier to identify what the component belongs too. For example it can be difficult if I only look at a plate with some holes in it to know what the plate belongs to. Categories also provide another mechanism for filtering and searching.

NOTE: Categories do NOT control lifecycle behaviour…. that is a file-level only option

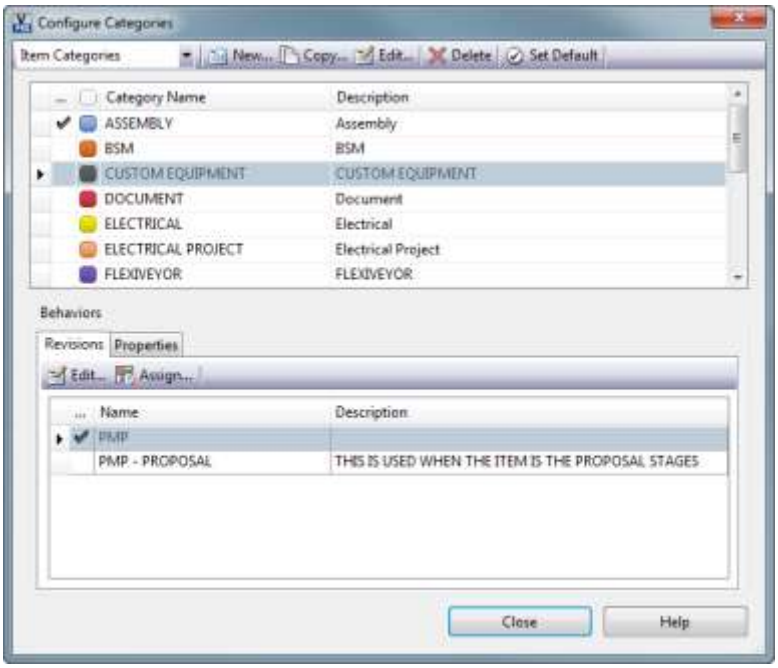
With items the "out-of-the-box" lifecycle states provided are *Work In Progress, In Review, Released,* and *Obsolete.* The names of these states can be adjusted but cannot be deleted nor can additional states be created



The **Item Lifecycle Rules** are used to control when the revision value of the item is automatically "bumped" and what revision level it is bumped to. Bumped simply refers to the action of automatically increasing the revision level, like revision 3 to revision 4. The starting revision value and the level of the revision is completely controlled by the revision scheme.



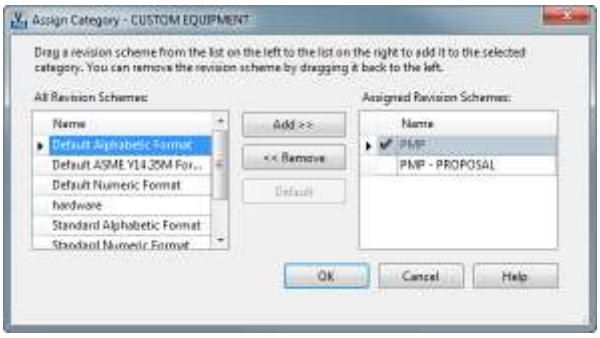
In the example shown here the state change of moving the item from Released to Work in Progress is configured to Bump the Primary Revision. For example depending on the revision scheme applied to the item this would mean that the item could go from 1 to 2 or from A to B



In the Vault Settings dialog box the ***Behaviors*** tab provides category management options.

Use the ***New*** button to define a new category and specify the name and colour of the category icon. This dialog is also used to set the default category. Using the drop-down in the upper left corner you can filter the list to only show item categories

You assign Revision Schemes to the Category making these revision schemes the only available options to items assigned to that category. You also set the default scheme to use with that category



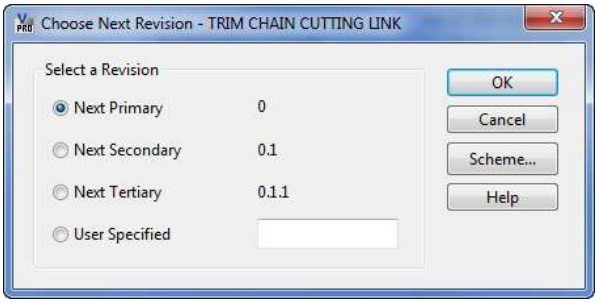
Category Revision Schemes

Using Item Lifecycles and Item Revisions

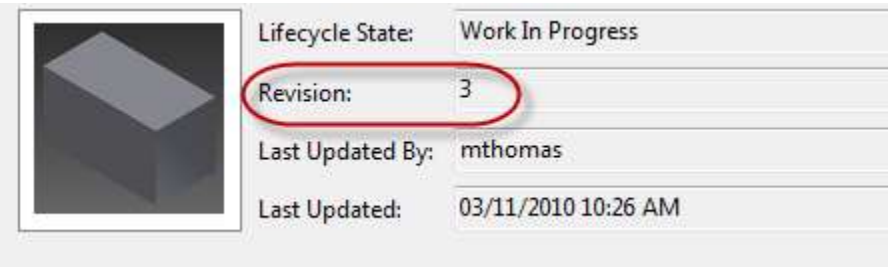
Adjusting the Revision Level of an Item

Revisions are used to track the history of changes applied to an item. Once the item is created you can use item lifecycles to manage the revisions. Unfortunately there is no current method of having the file set the initial revision level of the item. However once the item is created the item revision will then be used to set the file's revision.

To adjust the revision of the item select it, right-click, and select **Change Revision…**



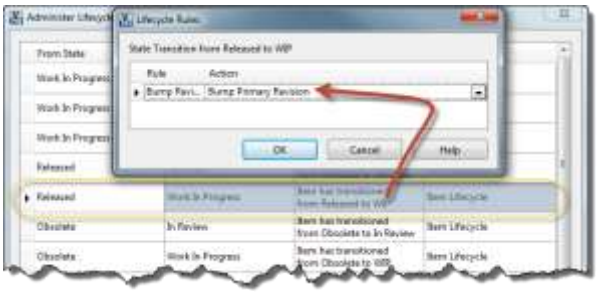
You can use the **Scheme** button to switch the Revision Scheme to a different type



There is NO turning back the clock… once a revision level has been used you cannot go to a lower value!

Lifecycles - Automatic Revision Bumping?

The lifecycle state of an item tracks the state of the item through the manufacturing process. This shows the state from the item creation (inception) to when the item is no longer required (retirement). To change the lifecycle state of an item select the item, right-click, and select Change State. Select the desired state and click OK.



Item lifecycle changes can be configured to auto-bump the revision level

Item Editor Level 2 Users will have the ability to “Skip Revision Bump” which does exactly that

Lifecycle changes can be rolled back to a previous state using **Roll Back Lifecycle State Change**

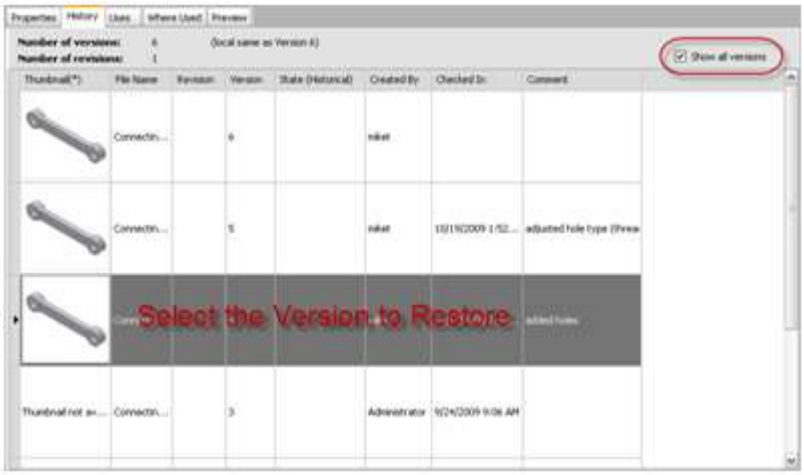
Purging Items

Vault maintains a history of every version of an item, which is generated each time the item is updated, edited or changed state… this version history can be purged. Most common case of using this is that an item is renamed in a later WIP state, Vault will hold onto the previous item number. By purging you can reuse this number.

Getting the previous version of a revision

Vault tracks versions of a file and the revision of an item which is of a particular version of that file. To make a previous version of a file the current version the Item needs to be in WIP (Work in Progress). Then:

- Switch to the Project Explorer view and locate the file.
- Check out the file (use the Vault Client, not the CAD application).
- Insure that the “Show all versions” is toggled.
- Select the version you want to restore, right-click, and select Get / Checkout. Click OK to restore that version of the file. You will be prompted to replace the existing file, click yes to all
- Open the file in the CAD application. Make sure you open the file locally and not from the Vault.
- From the CAD application check the file back into the Vault which will become the new version.

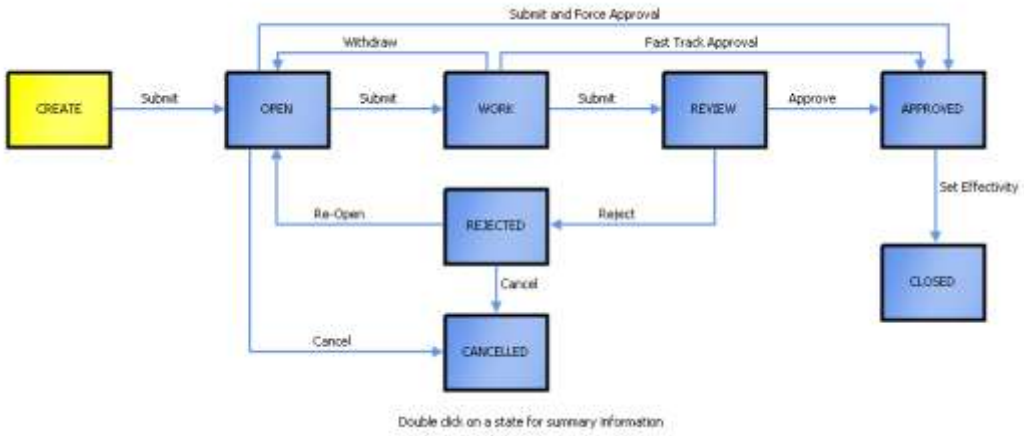


Change Orders

What are Change Orders?

When a design requires a change or modification you want to put controls in place to help manage the change. A record of the changes provides a history "trail" of what was changed, why it was changed and when the change occurred. Vault Professional provides **Change Orders** to capture the changes and manage the change as the modifications are completed, reviewed, and released to be manufactured. The change order is the historical "paper trail" of the why, how, when, who, and what of the design modifications

Change Orders are not available to everyone. You need to have at least basic level privileges to the items to create or participate in change orders. If you are unsure of your privileges talk to your Vault Administrator



Change Order Workflow

Creating New Change Orders (and Submitting for Work)

Quick Procedure

1. Locate the ***item*** or ***file*** requiring the change
2. Create the Change Order
3. Describe the change required
4. Markup the drawing (as required)
5. Assign the Routing
6. Submit it

Creating the Change Order

The first step is to create the change order is to locate the item requiring the change, right-click, and select **New Change Order.** This puts the change order into the ***Create*** stage

Creating a New Change Order

At this point you will put in information about the required change. On the ***General*** tab the change order title and summary (description) are entered as well as the expected due date of the required changes.

The ***Items*** tab lists all the items requiring changes. Additional items that are not directly related / tied to the active item can be added to the Change Order using the **Add** button. This can include other assemblies or components requiring changes

Completing the Required Changes (aka doing the work)

Once the change order enters the Work stage the Responsible Engineer can start making the required changes and modifications.

Quick Summary of the steps:

1. Add related files (the files requiring changes)
2. Change the lifecycle state of the files to Work In Progress
3. Adjust item revision levels as necessary
4. Check-out the files and make the changes
5. Check-in the files
6. Update the items within the change order
7. Change the lifecycle state of any WIP items to “In Review’
8. Submit the change order for review

Once the items are updated and you are satisfied with the changes you can change the lifecycle of the items to In Review. Submit the change order to move to the next stage (Review)

Reviewing, Approving, and Setting Effectivity

In the **Review** stage **Reviewers** can view the change order, make markups, and add comments.

In the **Review** stage **Approvers** can either Reject or Approve the changes. Rejecting the changes moves the change order back to a **Rejected** state where further decision can be made. Approving the changes moves the change order into the **Approved** state where the change order will await the setting of effectivity.