

IM462271

Seven Steps to Building the Ideal Lifecycle

Mike Thomas

Technical Services Manager - Prairie Machine

Learning Objectives

- Learn how to configure lifecycle definitions
- Learn how to apply state changes
- Learn how to apply security settings
- Learn how to manage revisions.

Description

The ideal lifecycle is possible. Everyone can manage change the way they want—and, in fact, they do. Because, the way you manage files now, regardless of whether you like it or not, is what you choose to do. To move past the basics of file management into the area of drawing revisions, approvals, security, and advanced property management, you need to deal with lifecycles. In this class, we'll look at Vault lifecycles and help you decide which is the best course of action. This class will introduce you to the powerful tools that are available in Vault Workgroup and Vault Professional for revision management, approval workflows, and engineering change orders (ECOs). Learn how categories are used to control the flow of your data and how versions and revisions are intertwined. We'll also cover how to apply different revision schemes and how to initiate and track change via a change order.

Speaker(s)

I have been using AutoCAD since r13, cut my solid modeling teeth on Mechanical Desktop, and have been using Inventor since before it was known as Inventor. Data Management has always been a big part of my professional life, for the most part with Autodesk Vault. I've been utilizing Vault for data management for as long as its been an Autodesk product.

Lifecycles

Our drawings and models are the language of engineering and manufacturing. It is how we communicate with others. It is how we go from concept-to-design-to-manufacture. How we manage this language is as important as the documents themselves.

It is important to maintain a history of a document (including drawings) to track what has changed, when it changed, and hopefully why it changed. This history aids in making better decisions, provides crucial customer information, and provides the means to go back to a previous version.

From Vault’s help: “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 7-steps to building the ideal lifecycle:

1. Build Your **Properties**
2. Add a **Category**
3. Define the **Rules**
4. Define the **Revision Schemes**
5. Build the **Transitions**
6. Make it **Secure**
7. **Track the Change**

If you are using **Autodesk Vault Basic** you will not have access to the lifecycle engine. Vault Basic only provides *version control*, no tools to manage the change. **Vault Workgroup** and **Vault Professional** provide the tools for managing change

The *lifecycle engine* is available for files and folders, and additionally within Vault Professional custom objects and items.



A **version** is an *iteration* of a document, something that is different from the previous copy. Versions maintain the previous design in case you need to restore back to a previous copy. All editions of Vault maintain versions of the files checked into Vault.

A **revision** is a *milestone*, the act of making a change and completing the work required for change.

Property Administration

The Oxford Dictionary describes **Properties** as meta data, a set of data describing and providing information about other data. In Vault, properties are meta data describing your data in Vault.



Vault contains two types of properties:

- **System-defined** – preset derived from Vault (aka out-of-the-box)
- **User-defined (UDP)** – you build properties

Properties are available for files, folders, items (Vault Pro), and custom objects (Vault Pro). You associate with entity classes the property is available. Mapping determines where the property gets its information from.

When files are checked into the vault, only the properties that are set enabled are automatically extracted and indexed by default.

Create a new user-defined properties (UDP):

- In the **Property Definitions dialog** select **New**
- Specify the **Name**
- Select the **Property Type**
- Assign the UDP to one or more **Categories**
- Use the Settings to optional further configure the property
- Optional configure Mapping



Map properties so that values from the master file (like a file property) write to the property on file check-in or on item assignment.

Categories

What are Categories?

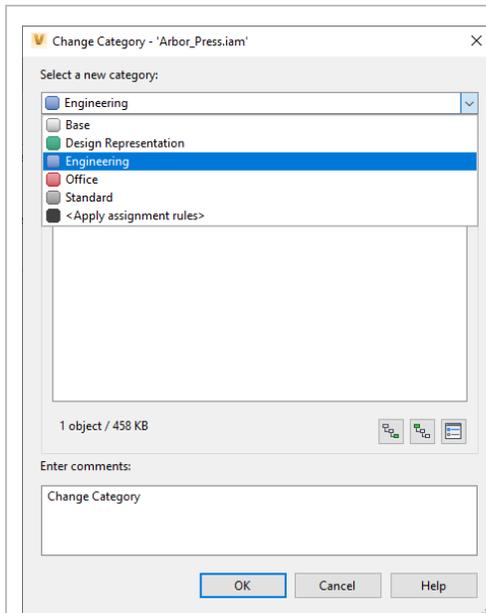


Figure 1: Change Category

Categories are *labels*, providing a method to group objects logically. **Categories** assign user-defined properties to the objects, the available lifecycle definitions, and the available revision schemes.

To change the **Category**, select the object(s) and select **Change Category**. The subsequent dialog allows you to choose from the existing categories.

Building Categories

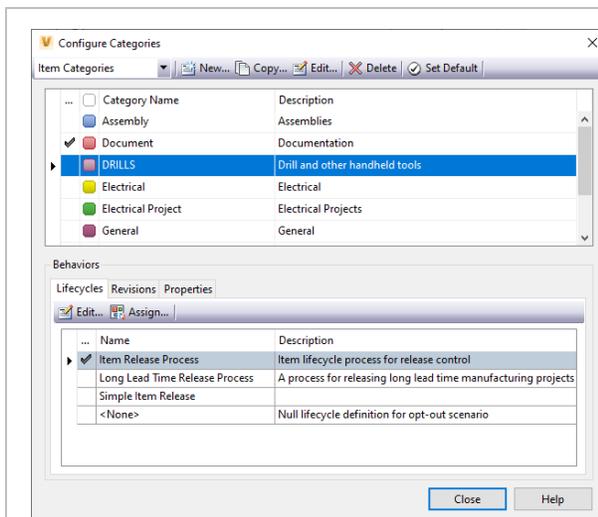


Figure 2: Configure Categories

The steps:

1. Assign the **Name, Colour, & Description**
2. Assign the **Properties**
3. Assign the **Revision Schemes**
4. Assign the **States**

Define the Rules

Rules

Use **Rules** to define conditions where objects automatically assign to a category based on the object's properties. For example, with Files, you could adjust the category manually or establish rules to assign the category automatically based on properties of the files.

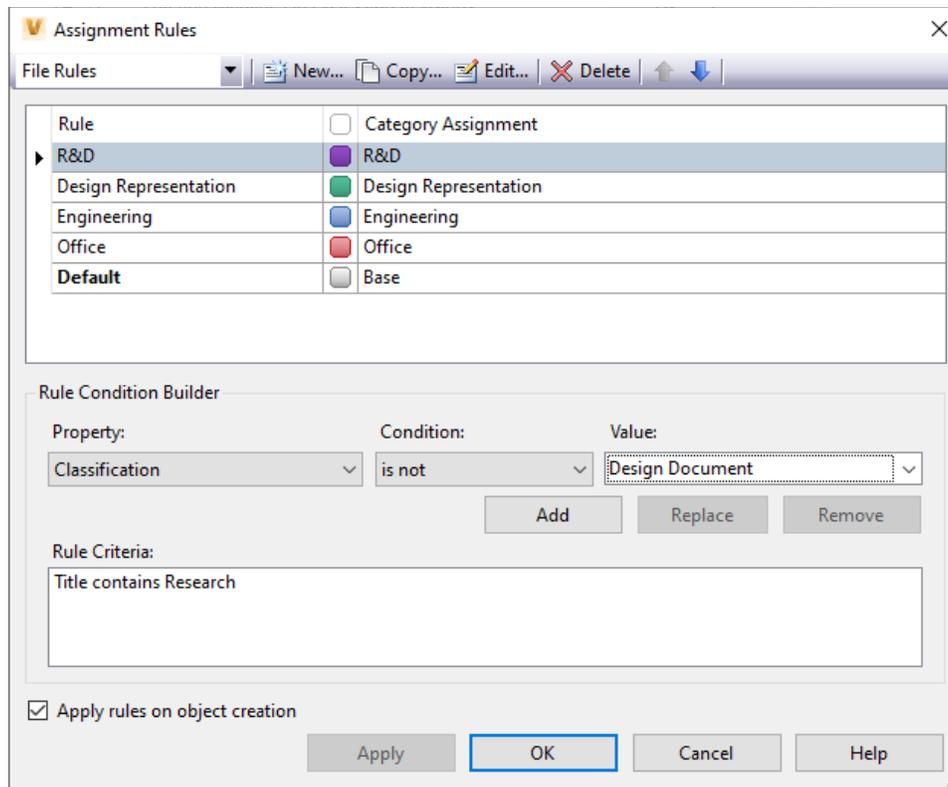


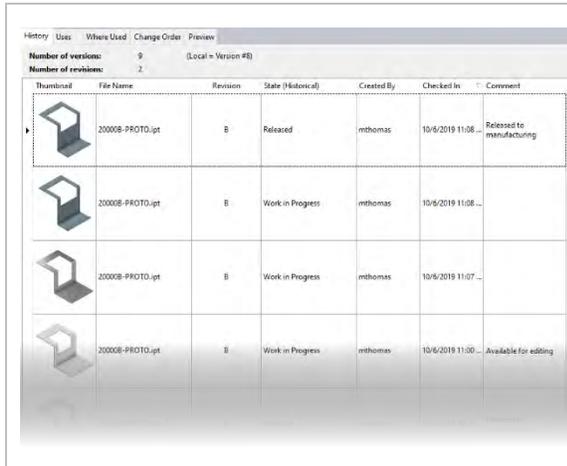
Figure 3: Assignment Rules

Create new rules to enforce how files, items, and custom objects are assigned to categories. Start by setting the Name and selecting the Category to assign it to the new rule.

In the condition variables area, the Property, Condition, and Value lists are displayed. The property list shows the indexed user-defined and system properties for the vault that are in use. The conditions available in the condition list are determined by the selected property. The variable entered in the value box is compared to the property using the selected condition.

Revision Schemes

What is a Revision?



Thumbnail	File Name	Revision	State (Historical)	Created By	Checked In	Comment
	20008-PROTO.ipt	B	Released	mthomas	10/6/2019 11:08 ...	Released to manufacturing
	20008-PROTO.ipt	B	Work in Progress	mthomas	10/6/2019 11:08 ...	
	20008-PROTO.ipt	B	Work in Progress	mthomas	10/6/2019 11:07 ...	
	20008-PROTO.ipt	B	Work in Progress	mthomas	10/6/2019 11:00 ...	Available for editing

A **revision** is the *act of revising*, which is to make new, amended, improved, or up-to-date version.

A revision is a *milestone*. Revisions are *permanent versions*.

Within Vault, you revise your objects by **BUMPING** the revision. Use **Change Revision** to grab the next available value in the current revision scheme. Within the dialog, select the revision level to bump to (i.e. are you going from 1 to 2 or 1 to 1.1) and Vault automatically gets the next available value.

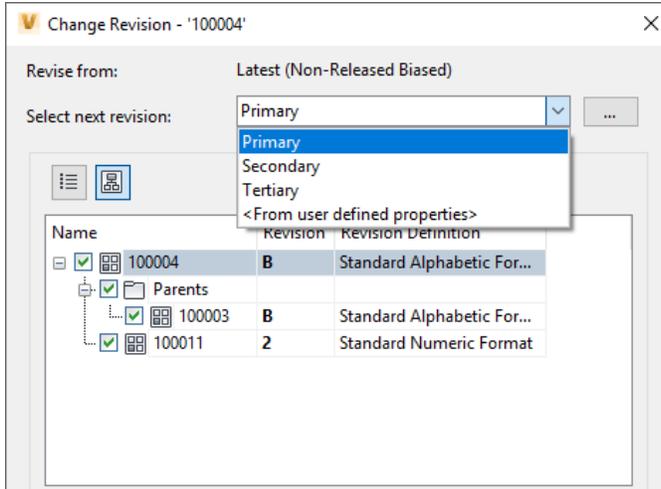


Figure 4: Change Revision



There is **NO** turning back the clock... once a revision level has been used you cannot go back to it.

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.

Revision Schemes

Revisions Schemes define the format and sequence of the revision. They define the format of the primary, secondary, and tertiary revisions. The scheme forces the users into predefined values, so they can not stray away from the standard.

For example, the *Standard Numeric Format* starts at 1 and increments by 1, not skipping any numbers.

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)¹**

Build (and modify) **Revision Schemes** within the **Vault Settings**(Behaviors tab):

1. **Name** the new scheme and provide a **description**
2. Select the **Categories** in which the scheme is available
3. Assign the three **Scheme Formats** (primary, secondary, and tertiary) and the **Delimiter**
4. Create **Scheme Formats** (as required)

The **Scheme Definition** is simply the list of values available to the Revision.



When defining a new scheme, it is much easier to start with a copy of an existing scheme opposed to starting from scratch.

You can mix schemes in the definition. For example, using *Alphabetic* for primary, *Numeric* for secondary, and *ASME* for tertiary.

Once a scheme is used it **CANNOT** be modified. Make sure it is what you want before you start using it.

¹ 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

The **State** identifies the object's status within the lifecycle. The number and type of states is based on the applied lifecycle and its associated **Transition Rules**.

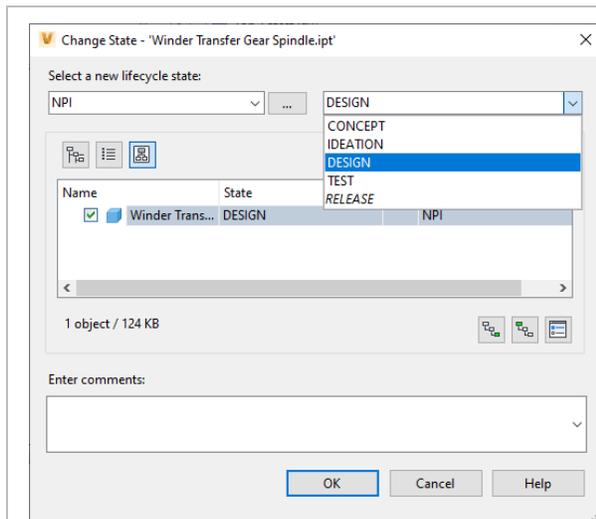


Figure 5: Change State

To change the state, pick the object and select **Change State**.

In the **Change State dialog**, adjust the assigned **Lifecycle definition** and choose the desired **State**.

Depending on the settings you might also notice the revision value change as well.



Folders are like files except you cannot set the revision. After applying the desired **Category**, you can adjust the **State** of that folder. The **State change** configuration may force you to perform a state change on the files contained within the folder before changing the state.

The State determines:

- The **Property Criteria requirements** (Compliance) that must be in place before the state change can occur
- The **Actions** that happen during the change (i.e. revision bump)
- The **Security** enforcing who can make the change and who can see the file in the state

Transitions

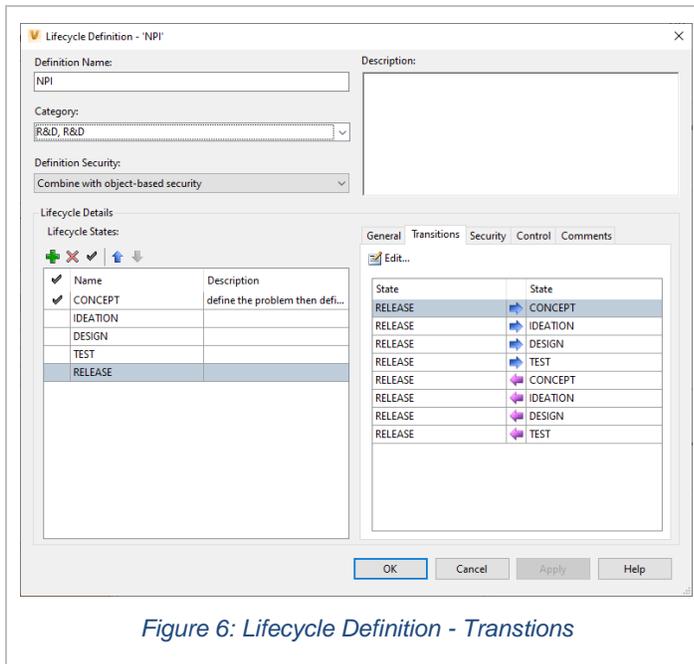


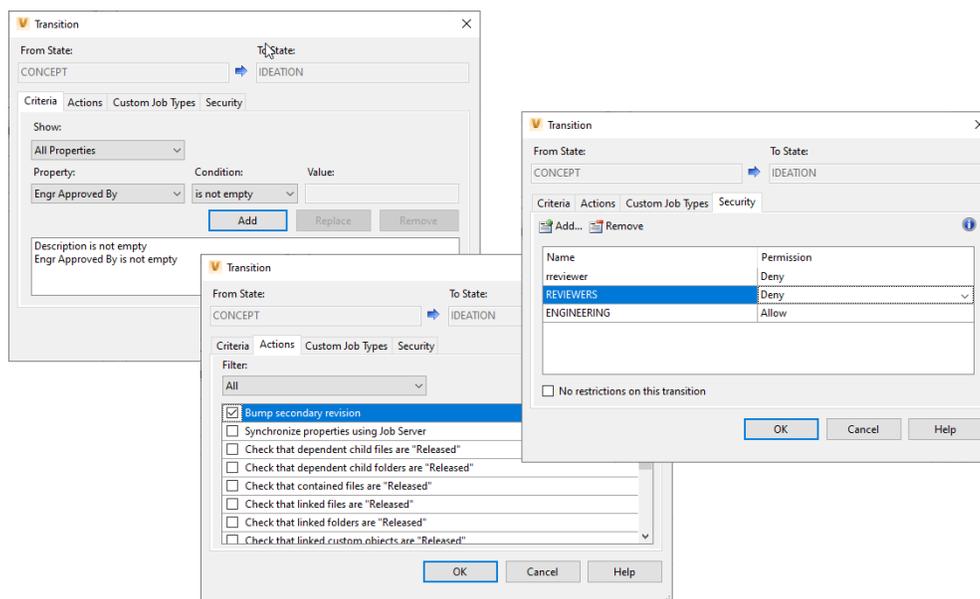
Figure 6: Lifecycle Definition - Transitions

A **transition** occurs when an object moves from one lifecycle state to another.

You configure the criteria on how it moves to the next state. For each state transition in a lifecycle, you can configure

The transition manages:

- **Criteria** – the property details determining when a state transition can occur. For example, a file unable to transition to the next state without a value in the Initial Approver property.
- **Actions** that occur automatically when the transition is made (like a revision bump)
- **Access control permissions** – who can execute the transition



Security

General Transitions Security Control Comments

+ Add... - Remove

	Name	Read	Modify	Delete	Download
	Everyone	Allow	Deny	Deny	Allow
	Administrator	Allow	Allow	Allow	Allow

No state-based security

Options

 Security for associated files of items Configure...

Security for files inside folders Configure...

Figure 7: State Security

Edit the lifecycle's state security to manage which members and groups can read, modify, delete, or download an object assigned to a state

There are three ways to define the security of an object:

- **Object-based security** – security is derived from the object settings. This is the default option.
- **State-based security** – available when a state-based ACL exists and is combined with object-based security.
- **Overridden security** – available when a state-based ACL is overriding object-based security, or when the administrator sets a manual override on the object.



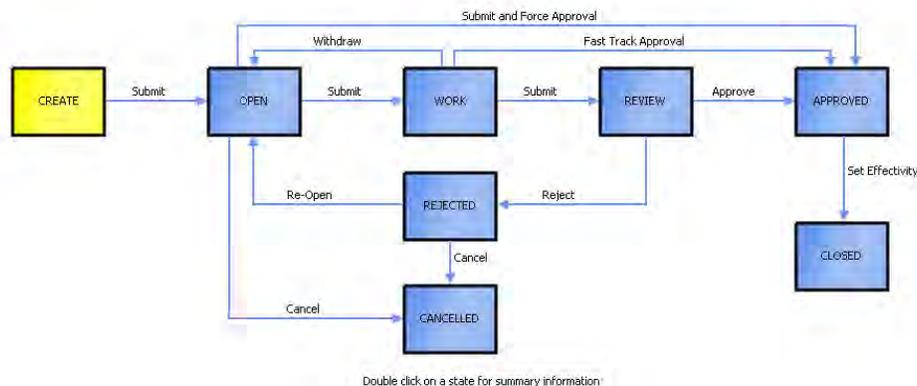
State-based security is either combined with object-based security or overrides object-based security, based on the Definition Security defined for the lifecycle definition.

Tracking Change

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



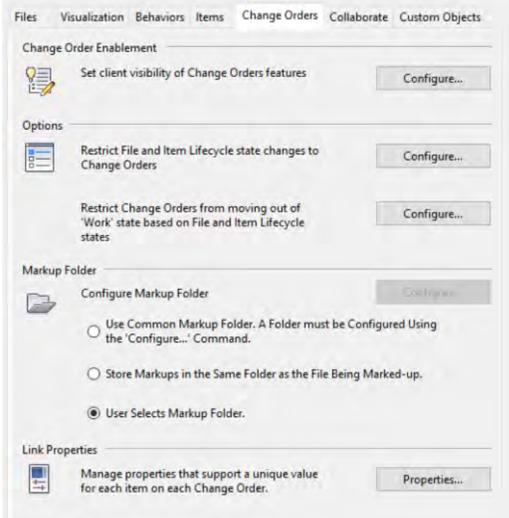
CHANGE ORDER WORKFLOW

Creating New Change Orders (Quick Procedure)

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

Change Order Administration

The **Change Orders** tab contains options for controlling change order behavior, configuring the markup folder, and managing change order link properties.

 <p><i>CHANGE ORDER ADMINISTRATION</i></p>	<p>Change Order Enablement – if you are not going to use Change Orders, might as well make it invisible</p> <p>Restrict File and Item Lifecycle state changes to Change Orders – No Lifecycle changes unless its in a Change Order</p> <p>Restrict Change Orders from moving out of Work state based on File and Item Lifecycle states – no objects in WIP when transitioning out of the “Work” state</p> <p>Markup Folder – configure how and where Vault stores DWF markups created on a Change Order</p> <p>Link Properties – link user-defined properties to change orders</p>
---	---

In addition to the Change Orders tab, define numbering schemes (Behaviors tab) to auto-label the change orders.