Configuration Management 101: Best Practices for using Baselines, Effectivity and Configurations

Jeff Zemsky
PTC
Configuration Management 101

Configuration Basics –
- Why we need these tools

Baselines

Effectivity

Configurations

Appendix
- Preferences
Which Verticals depend on Configuration Management?

<table>
<thead>
<tr>
<th>Aerospace/Defense</th>
<th>Automotive</th>
<th>High Tech/Electronics</th>
<th>Industrial Equipment</th>
<th>Consumer Products</th>
<th>Life Sciences</th>
</tr>
</thead>
</table>

**But....**
There are differing levels of configuration control needed based upon:
1. Product Complexity
2. Customer requirements / demands
3. Regulatory demands
4. Service models
5. And so on....
Business Strategies that use these Configuration techniques

Assemble-To-Stock
- Products fully designed with identified options
- ERP plans production of pre-defined options
- Product combinations manufactured based on forecast

Industries:
Consumer Products, Hi-Tech

Configure-To-Order
- Products fully designed with rules that allow creation of custom parts
- Each order configured & validated by PLM using rules established by Engineering

Industries:
Automotive OEMs, Hi-Tech, Industrial

Engineer-To-Order
- Significant custom engineering added to generic product
- Each order validated by Engineering and PLM

Industries:
Industrial OEMs, Auto Suppliers, Aero

Contract Product
- Designed and produced to meet customer or contract specific requirements.
- Each order designed and validated by PLM and Engineering

Industries:
Defense, Prototypes, Custom Tooling

Amount of Order-Specific Custom Engineering

Low

High

No Involvement by Engineering in Individual Orders

Many

One

Number of Products in an Order
Configuration Basics

Why we need these tools?
- Designs change
- Latest alone is insufficient
- We need to understand these impacts over time
- We need to capture the configurations for downstream needs
  - Supply Chain
  - Manufacturing
  - Safety and Certification
  - Field Service
  - Customers

<table>
<thead>
<tr>
<th>Part Number</th>
<th>Name</th>
<th>QTY</th>
</tr>
</thead>
<tbody>
<tr>
<td>44-33252-A</td>
<td>Brake Assembly</td>
<td></td>
</tr>
<tr>
<td>23-2311235</td>
<td>Bracket</td>
<td>2</td>
</tr>
<tr>
<td>32-97302-2</td>
<td>Bolt-1/4-UNF</td>
<td>6</td>
</tr>
<tr>
<td>73-428083</td>
<td>Spindle</td>
<td>1</td>
</tr>
<tr>
<td>77-298379</td>
<td>Housing-1</td>
<td>1</td>
</tr>
<tr>
<td>23-83276</td>
<td>Washer-960</td>
<td>6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Part Number</th>
<th>Name</th>
<th>QTY</th>
</tr>
</thead>
<tbody>
<tr>
<td>66-23908</td>
<td>Shoe support</td>
<td></td>
</tr>
<tr>
<td>23-138930</td>
<td>Spring</td>
<td></td>
</tr>
<tr>
<td>88-237398</td>
<td>Pad Assy-HT</td>
<td></td>
</tr>
<tr>
<td>48-2398908</td>
<td>Pad-HT-L</td>
<td></td>
</tr>
<tr>
<td>78-323980</td>
<td>Mtg Sup-LT</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Part Number</th>
<th>Name</th>
<th>QTY</th>
</tr>
</thead>
<tbody>
<tr>
<td>44-33252-B</td>
<td>Brake Assembly</td>
<td></td>
</tr>
<tr>
<td>23-2311235</td>
<td>Bracket</td>
<td>2</td>
</tr>
<tr>
<td>32-97302-38</td>
<td>Bolt-3/8-UNF</td>
<td>4</td>
</tr>
<tr>
<td>73-4280822</td>
<td>Spindle-HF</td>
<td>1</td>
</tr>
<tr>
<td>76-5694824</td>
<td>Housing-2</td>
<td>1</td>
</tr>
<tr>
<td>23-83276</td>
<td>Washer-860</td>
<td>4</td>
</tr>
<tr>
<td>66-23908</td>
<td>Shoe support</td>
<td>2</td>
</tr>
<tr>
<td>23-138930</td>
<td>Spring</td>
<td>2</td>
</tr>
<tr>
<td>77-238392</td>
<td>Pad Assy</td>
<td>2</td>
</tr>
<tr>
<td>38-2398908</td>
<td>Pad-LR-M</td>
<td>1</td>
</tr>
<tr>
<td>68-323340</td>
<td>Mtg Sup-LT</td>
<td>1</td>
</tr>
</tbody>
</table>
Tools to help understand our Configurations

Baselines

“A managed baseline is essentially a snapshot of an evolving collection of product data objects.”

Effectivity

“Effectivity is the planned date, lot, or serial number at which old part versions are replaced with new part versions in production.”

Configurations & Instances

“A configuration identifies the versions of parts used to build the part instance.”

“An instance identifies, by serial number, a unique, manufactured instance of a part, built according to a specific configuration.”
Configuration Management 101

Configuration Basics –
- Why we need these tools

Baselines
Effectivity
Configurations
Appendix
- Preferences
Baselines can help with Pancakes
Pancake Recipes

The weekends are time to make pancakes....but.....

So – which pancake recipe (or BOM) to use?

Decisions, decisions?

- Baking soda
- Baking powder
- Eggs mixed in
- Eggs separated and the whites whipped and folded in
- Add butter or not
- Milk
- Buttermilk
- Sour cream
- Cast iron pan
- Teflon griddle
But if we had Baselines for our Pancakes

Problem is – we update the recipes all the time

If we had baselines to reference we could see

- What changes we made
- Compare the differences
- Use the baselines to make sure we had the right BOM
Pancakes – great but what does this mean to me?

Capture snapshots of important Configurations

- Design review configurations
- Production Releases
- Significant configurations to share with partners, suppliers and customers

Enable the organization to see the configuration and its related data
Baselines basics

Baselines may include:
- Parts
- Documents
- Dynamic documents
- CAD documents
- Requirements
- Change Objects

Baseline Types
- Windchill Created – for example: Promotion Requests
- User created: Managed Baselines

Managed Baselines can be:
- **Protected**: Objects in that baseline cannot be deleted without deleting baseline
- **Locked**: Locking makes the baseline private to the person who locked it
Creating and Updating Baselines

You can create baselines:
- Manually (i.e. Folder ⇒ File ⇒ New ⇒ Baseline)
- During Check In of CAD Documents
- During Import from Spreadsheet

Updating can include:
- Changes to attributes
- Add, remove or change content
  - Add or Paste items
  - Update Revisions of existing items

Primary Object
- Indicates which object is the reason for creating the Baseline

Note: As-Stored Configuration for CAD Document is not a Baseline, it is a dynamic configuration. It can change!
What can you do with Baselines

Add to or Check out to Workspace

Use a Configuration Criteria to show product structure
- In PSB and PSE
- Multi-level BOM Compare
- Information Compare

For Collecting items
- Any place the Collector is used
- Changes, Packages, etc.
- Set Configuration to Baseline
Configuration Management 101

Configuration Basics –
- Why we need these tools

Baselines

Effectivity

Configurations

Appendix
- Preferences
Effectivity
Effectivity and the Kitchen plan

Ever start a project with a simple goal
Like…”Let’s repaint the cabinets”
From there the project grew and grew to
Let’s add

○ Lights
○ New Hardware
○ Rebuild the pantry

With Effectivity I can know what versions of Parts are effective on specific dates
Effectivity – let’s get deeper

Effectivity is the planned date, lot, or serial number at which old part versions are replaced with new part versions in production.

- Let’s view what versions of parts are in the product structure based upon Date, Serial number or Lot

Effectivity is set or updated from a change notice or a change task.

- When the change notice has been approved, the planned effectivities get copied over as actual effectivities on the given object.
- Effectivity can also be updated manually on the Part properties page

Effectivity can be applied to:
- Parts
- MPMLink objects including
  - Process plans (manufacturing process plans, sequences, and operations)
  - Manufacturing resources (plants, resource groups, skills, process materials, tooling, and work center)
What did you just say?
Effectivity Concepts

Effectivity Type

- Effectivity applicability is determined by the Trace Code of the Part
- If it is set to Serial Number one can apply Serial Number, MSN or Date effectivity
- If it is set to Lot or Lot/Serial one can apply Lot, Block or Date effectivity

Context

- Effectivity has a context so that a part can be effective in one assembly on one date or serial number and in another assembly on a different date or serial number
  
  – Example – The new Cooling Fan, Part 123 should be Effective in the Desktop Computer, Part 500 from July 5, 2010 and will be Effective in the Server, Part 600 on September 21, 2010
Effectivity Concepts - Propagation

Propagation Type

- STRUCTURE PROPAGATION would propagate the new effectivity settings throughout the selected structure; to all the children
- SIBLING PROPAGATION will close the open-ended date ranges on the previous revision of the assembly based on our new settings for the new revision
- RECURSIVE SIBLING PROPAGATION will propagate the closure of effectivity on the previous revision of the assembly down including all of its components.

Propagation can only be done using a Change Notice
# Effectivity format examples

## Date

<table>
<thead>
<tr>
<th>Effectivity Format</th>
<th>Means</th>
</tr>
</thead>
<tbody>
<tr>
<td>6/23/2010</td>
<td>Effective only on June 23, 2010</td>
</tr>
<tr>
<td>6/23/2010 -</td>
<td>Effective from June 23, 2010 onwards</td>
</tr>
</tbody>
</table>

(Note – Date Effectivity format is controlled by Locale of the server)

## Serial and Lot

<table>
<thead>
<tr>
<th>Effectivity Format</th>
<th>Means</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Effective for serial number (or lot) 5 only</td>
</tr>
<tr>
<td>5, 23</td>
<td>Effective for serial number (or lot) 5 and 23 only</td>
</tr>
<tr>
<td>5 -</td>
<td>Effective from serial number (or lot) 5 onwards</td>
</tr>
<tr>
<td>5 - 23</td>
<td>Effective serial number (or lot) 5 through 23</td>
</tr>
</tbody>
</table>
Effectivity – some examples

Date Effectivity Example

Effectivity for 7/15/2010

Desktop Computer, PN 500

CPU, PN 501
1/1/2010 – 7/6/2010

A
B

Case, PN 502
7/7/2010 – 6/24/2010

A
B

Cable, PN 503

A
B
Effectivity – some examples

Serial Effectivity Example

- Gas Turbine, PN 100
- Turbine, PN 101
- Housing, PN 102
- Compressor, PN 103
- Blades, PN 104

Effectivity for Serial #23
Applying Effectivity

Set on the Resulting Objects for a Change Notice

Add or View Effectivity
Applying Effectivity

- Specify Effectivity on all selected Items
- Specify Effectivity one at a time
- Copy from one row to another
Specifying Effectivity

Set the Context (determines if Date, Serial or Lot is available)

Select Effectivity type & set range

Information attribute (Exact or No Later than)

If Propagate preference is set, determine whether to use it and where to stop (i.e. Stop at the next lower level Serial traced Part)
Release the Change and Effectivity is applied

Applied in the Workflow

Note Multiple Effectivity statements
Effectivity Sibling Propagation

Update the Effectivity on the next Revision and it closes the previous

In this example –

PANTRY CABINET, PN 4067 Rev C specified Serial Number 8 and higher

PANTRY Cabinet, PN 4067 Rev B was then set to Serial Number 1 – 7
Be aware - Sibling Propagation ...

New rev of assembly, swapping in a new component.

Effectivity of these 2 new items set for 20/12/2007 - with NO structure propagation.

Rev A of assembly now has it's effectivity close down to the previous day.

Rev B of assembly is effective as specified in the change notice.
... and the effect of Recursive Sibling Propagation ... be careful ...

NOTE: The closure of effectivity on version A of the assembly has propagated down to it's components, which are also re-used in version B

So if you were wondering why propagation is only available as part of the change process, here is a good example why – the result of effectivity propagation need to be considered and hence this why we use a change process to manage this ...
Ad-hoc Effectivity

Effectivity can be edited, added or remove on the Part Info Page

- Why?
  - Last minute changes in delivery or production schedule
  - Typo in Change Notice

Add new Effectivity or Remove existing Effectivity

Edit Effectivity
Using Effectivity

Similar to Baselines, Effectivity can be used to:

- Set the Configuration Specification to show product structure in PSE, PSB
- Determine the items to gather using the Collector
- Determine what to compare in Multi-Level BOM Compare
- Change Management Baseline Report
Configuration Management 101

Configuration Basics –
- Why we need these tools

Baselines

Effectivity

Configurations

Appendix
- Preferences
Configurations & Instances

Configurations are used to define the versions of the parts used to create a Part Instance.

- The Part configuration allows you to track the versions of parts that are used to manufacture a specific line of end item parts for record keeping and ongoing maintenance of that end item part in the field.

An instance identifies, by serial number, a unique manufactured instance of a part, built according to a specific configuration. It allows you to identify exactly which part is used in that instance. Once the end item is built, the instance becomes a record that can be used to indicate changes to that particular copy of the end item part while it is in service. If the instance is changed (for example, one part is replaced by another), the replacement part is recorded in the instance.

Part instances can be created for parts that are traced by:

- Serial Number
- Lot Number
- Lot and Serial Number
Easier to show....

Sedan, Rev A
- Chassis, Rev A
- Engine, Rev B
- Transmission, Rev A
- Interior, Rev C

Sedan, VIN XYZ10
- Chassis, Rev A
- Engine, SN-10028
- Transmission, SN-10
- Interior, Rev C
But wait – what about Effectivity?

Effectivity tells you what ranges of Serial or Lot numbers can be used for a given revision.

Instances track exactly which Serial or Lot number are used for a specific manufactured item.

Example uses for Configurations and Instances:

- There is a need to trace which components are in which fielded item.
- There is a need to trace components for warranty, regulatory or customer requirement.
Steps in Configurations and Instances

1. **Step 1:** Create the design structure.
2. **Step 2:** Create Configurations for all traceable Parts within the design structure.
3. **Step 3:** Create Instances from Configurations.
4. **Step 4:** Incorporate the bottom most instances.
5. **Step 5:** Allocate lower instances to higher level instances.
6. **Step 6:** Incorporate next higher level instances.

Repeat Incorporations until all required components are specified.
Creating the Configuration

**BASIC:** Only populates parts into the configuration that are traceable.

**FULL:** Populate all parts irrespective of trace code.

Create a Configuration from a specific traceable Part.
Creating the Instance

Create a new part instance from the info page of a Part or Configuration.
Part Instances – Support Documents

Use Cases
- Test Results
- Conformance Certification
- Repair Records
- Calibration Records

Also relate to Change objects:
- Variances
- Problem Reports
- Change Requests
- Change Notices
Allocating a Part

Add New or use an Existing Serial Number
Incorporating the Part

What date did the part go into service

- Start: Current or earlier date when part was incorporated
- Estimated: Planned date to incorporate (Can be future)
Instance Details

Instances Can support Soft Attributes

Parts in Instances can be updated:

- With new Revisions
- With Alternate and Substitutes
Wrap-Up

We covered three ways to help manage Configurations
- Baselines
- Effectivity
- Configurations and Instances

There is no one size, one answer solution
- Evaluate and Plan for your needs
- War game real world examples in test environments
- Consider downstream and upstream needs
Customer Visits

Every year, PTC Product Management visits hundreds of customers

Purpose of Visits

- Listen to your needs
- Understand your processes and products

You can help shape the future of PTC products!

Interested in hosting a PTC customer visit?

- Email your Product Manager
- Drop off a business card with customer visit written on the back

Email me at:
- jzemsky@ptc.com

"Courage is what it takes to stand up and speak; Courage is also what it takes to sit down and listen”
- Winston Churchill
Questions?
Configuration Management 101

Configuration Basics –
- Why we need these tools
Baselines
Effectivity
Configurations

Appendix
- Preferences
## Preferences for Effectivity

<table>
<thead>
<tr>
<th>Preference</th>
<th>Values</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Recursive Sibling Propagation</td>
<td>Yes/No</td>
<td>Enables propagation of effectivity statements down a product structure for statements which result from a sibling propagation operation.</td>
</tr>
<tr>
<td>Enable Sibling Propagation</td>
<td>Yes/No</td>
<td>Enables propagation of effectivity statements to siblings.</td>
</tr>
<tr>
<td>Enable Structure Propagation</td>
<td>Yes/No</td>
<td>Enables propagation of effectivity statements down a product stucture.</td>
</tr>
<tr>
<td>Include Work in Process Parts for Part Configuration Specification for Structure Propagation</td>
<td>Yes/No</td>
<td>Includes work in process parts for the part configuration specification used in structure propagation of effectivity statements.</td>
</tr>
<tr>
<td>Lifecycle State of Part Configuration Specification for Structure Propagation</td>
<td>Released</td>
<td>Lifecycle state of the part configuration specification used in structure propagation of effectivity statements.</td>
</tr>
<tr>
<td>Lot number format</td>
<td>wt.eff.format.DefaultUnitEffFormat</td>
<td>Lot number format</td>
</tr>
<tr>
<td>Serial number format</td>
<td>wt.eff.format.NumericUnitEffFormat</td>
<td>Serial number format</td>
</tr>
<tr>
<td>View of Part Configuration Specification for Structure Propagation</td>
<td>(Select from Views or blank)</td>
<td>View of the part configuration specification used in structure propagation of effectivity statements.</td>
</tr>
</tbody>
</table>
Effectivity Format - wt.eff.format.NumericUnitEffFormat

This format supports fixed-length (max. length of 18) effectivity statements consisting of only numerical digits (0-9).

Use this format if your unit effectivity statements consist of only numbers and have a fixed length. This format implements the various comparisons and range arithmetic logic by converting the range value to an equivalent number. Hence all range value characters must be digits (0-9).

This format accepts the following configurable parameters:

- **Length**
  - Description: Enforces effectivity statements adhering to this format, to have a fixed length. This format supports a maximum length of 18 digits.
  - Type: Positive Integer (Number), less than 18.
  - Special Values: 0. This value states that the effectivity range values do not have a fixed length.
  - Required: Yes
  - Default: N/A
  - Examples: 3, 6, 8.

Effectivity formats in this package are driven by parameters that dictate operations they support. See the Windchill Customizer's Guide for explanation on how to specify the format a traceable business object supports along with the parameters it accepts.

**Example:**

wt.eff.format.NumericUnitEffFormat| length=3

Gives Serial effectivity of length 3 – numbers 000 - 999
Effectivity Format  - wt.eff.format.DefaultUnitEffFormat

This format represents the default format for unit effectivity statements if no other effectivity format is supplied and thus acts like a placeholder. Since it is unaware of the actual format of effectivity statements, it provides rudimentary implementations of the various format-related operations.

This format accepts the following configurable parameters:

• length
  • Description: Enforces effectivity statements adhering to this format, to have a fixed length.
  • Type: Positive Integer (Number).
  • Special Values: 0. This value states that the effectivity range values do not have a fixed length.
  • Required: No (If not specified, range values can have an arbitrary length)
  • Default: 0
  • Examples: 3, 6, 8.

• contains-whitespace
  • Description: Specifies whether effectivity range values contain space(s).
  • Type: Boolean.
  • Required: No (If not specified range values can contain space characters)
  • Default: true. Allows spaces to be part of the effectivity range values.
  • Other values: false. Does not allow spaces to be part of effectivity range values.

Effectivity formats in this package are driven by parameters that dictate operations they support. See the Windchill Customizer's Guide for explanation on how to specify the format a traceable business object supports along with the parameters it accepts.

The only parameter this format supports is the length of effectivity range values.
Use this effectivity format, if you do not intend to take advantage of any of the format-related capabilities like range value validation, comparisons and range arithmetic.