Name: Customizing Multiple Inventory Dispositions

<INTERNAL>

By: Saul Lempirikany – Change Management

Change History:

|  |  |  |
| --- | --- | --- |
| **Date** | **Author** | **Description** |
| 11/24/09 | Saul Lempirikany | Created |
| 04/08/2010 | Saul Lempirikany | Make changes |
|  |  |  |

</INTERNAL>

# Best Practice Name and Classification

## Customizing Multiple Inventory Dispositions

# Objective

To customize Multiple Inventory Disposition types and values

## Problem Statement

You want to customize Multiple Inventory Dispositions on the AffectedActivityData link.

## Background

As of X-20 release, Multiple Inventory Dispositions on the AffectedActivityData Link shall be available OOTB. This document details steps on how to customize Multiple Inventory dispositions by;

* Adding new inventory disposition types
* Adding new inventory disposition values to be used by inventory disposition types
* Setting a default disposition value.

## Scope/Applicability/Assumptions

The action to set inventory dispositions is exposed on the toolbar or on the actions menu drop-down of the Affected Objects table in the create/edit Change Task wizard. It is assumed that you have the ability to navigate to that table.

New inventory disposition values can be added using Enumerated Type Customization Utility (enumCustomize tool). It is assumed you have the basic knowledge on how to use the tool, if you are not familiar with the tool see Windchill Customizer’s Guide ([link](http://www.ptc.com/WCMS/files/89337/en/WCCustomizersGuide.pdf)).

## Intended Outcome

* You shall be able to introduce new inventory disposition values
* You shall be able to specify a default inventory disposition value
* You shall be able to introduce new types of inventory disposition types

# Solution

## Solution Statement

* Use Enumerated Type Customization Utility (**enumCustomize tool**) to introduce new inventory disposition values for use in the inventory disposition types
* Use Enumerated Type Customization Utility (**enumCustomize tool**) to specify a default inventory disposition value
* Use Type manager to introduce new types of inventory dispositions.

## Prerequisite knowledge

To apply this best practice, you need to have an understanding of the following:

* Basic knowledge of change management process.
* The management of RBINFO files customizations.
* Basic knowledge on how to use the new (X-20) Type Manager UI
* Java Language

## Solution Elements

|  |  |  |
| --- | --- | --- |
| **Element** | **Type** | **Description** |
| inventoryDispositionRB. rbInfo | rbInfo | Contains the master list of inventory disposition values.  **ClearCase Location:** \Windchill\DevModules\Foundation\src\wt\change2  **Runtime Location:**  \Windchill\src\wt\change2 |
| change2ClientResource.java | Class | Resource bundle for inventory disposition types  **ClearCase Location:**  \wcEnterprise\ChangeManagement\src\com\ptc\windchill\enterprise\change2  **Runtime Location:**  Windchill\srclib\wnc\ChangeManagement.jar |
| AffectedActivityData.xml | xml | Loader for AffectedActivityData Link. Contains OOTB dispositions and their values.  **ClearCase Location:**  \wcEnterprise\ChangeManagement\src\loadFiles\type  **Runtime Location:**  \Windchill\loadFiles\type\ |
| standardX20.dtd | dtd | DTD for disposition elements  **ClearCase Location:**  Z:\Windchill\DevModules\Foundation\src\loadXMLFiles  **Runtime Location:**  \Windchill\loadXMLFiles |
| Coreobjects.dtd | dtd | The IXB DTD for disposition elements  **ClearCase Location:**  \Windchill\DevModules\wtMisc\src\cb\_registry\ixb\dtds\standardX20.dtd  **Runtime Location:**  \Windchill\codebase\registry\ixb\dtds\standardX20.dtd |
| AffectedActivityData.java | Java class | OOTB link class  **ClearCase Location:**  \Windchill\DevModules\Foundation\src\wt\change2  **Runtime Location:**  \Windchill\srclib\wnc\Foundation.jar |
| DefaultDispositionHandler.java | Class | Default disposition handler. You will need to extend this handler or implement DispositionHandler Interface  **ClearCase Location:**  **\**wcEnterprise\ChangeManagement\src\com\ptc\windchill\enterprise\change2\handler  **Runtime Location:**  \Windchill\srclib\wnc\ChangeManagement.jar |
| DispositionHandler.java | Interface | If you are not extending DefaultDispositionHandler, implement this interface.  **ClearCase Location:**  **\**wcEnterprise\ChangeManagement\src\com\ptc\windchill\enterprise\change2\handler  **Runtime Location:**  \Windchill\srclib\wnc\ChangeManagement.jar |
| ChangeTaskAffectedItemsTableViews.java | Class | Contain configurable views for affected items table  **ClearCase Location:**  \wcEnterprise\ChangeManagement\src\com\ptc\windchill\enterprise\change2\tableViews  **Runtime Location:**  \Windchill\srclib\wnc\ChangeManagement.jar |
| AffectedItemsTableBuilder.java | Class | Builder for affected items table  **ClearCase Location:**  \wcEnterprise\ChangeManagement\src\com\ptc\windchill\enterprise\change2\mvc\builders\tables    **Runtime Location:**  \Windchill\srclib\wnc\ChangeManagement.jar |
| ChangeManagement-actions.xml | xml | Defines actions related to inventory dispositions  **ClearCase Location:**  \wcEnterprise\ChangeManagement\src\_web\config\actions  **Runtime Location:**  \Windchill\codebase\config\actions |
| DispositionDataUtility.java | Class | Sets the value of an inventory disposition type which is a column in the affected items table  **ClearCase Location:**  \wcEnterprise\ChangeManagement\src\com\ptc\windchill\enterprise\change2\dataUtilities  **Runtime Location:**  \Windchill\srclib\wnc\ChangeManagement.jar |
| ChangeManagement-service.properties.xconf | xconf | Links the disposition handler with inventory disposition types  **ClearCase Location:**  \wcEnterprise\ChangeManagement\src\com\ptc\windchill\enterprise\change2  **Runtime Location:**  Windchill\codebase\service.properties |
| ChangeManagement-components.dataUtilities.properties.xconf | xconf | This is where to register the data utilities  **ClearCase Location:**  \wcEnterprise\ChangeManagement\src\com\ptc\windchill\enterprise\change2  **Runtime Location:**  \Windchill\codebase\com\ptc\core\components\components.dataUtilities.properties |
| ChangeManagement-configs.xml | xml | This where the builder is registered so the MVC framework can find it.  **ClearCase Location:**  \wcEnterprise\ChangeManagement\src\_web\config\mvc  **Runtime Location:**  \Windchill\codebase\config\mvc |

## Procedure

### **Introduce new values for use in the disposition types:**

Out-of-the-box (OOTB) inventory disposition attribute values are fixed lists of values. Customers who want to customize an available OOTB disposition type by adding a new entry must define an entry in the InventoryDispositionRB.rbinfo file.

The InventoryDispositionRB.rbinfo file contains a fixed set of master lists of inventory disposition values. To add an entry in the InventoryDispositionRB.rbinfo file, use the enumCustomize tool to edit the file.  For more information, see The Enumerated Type Customization Utility section in the Enumerated Types chapter of the Windchill Customization Guide. ([link](http://www.ptc.com/WCMS/files/89337/en/WCCustomizersGuide.pdf))

After you have customized the disposition type, use the Type and Attribute Management utility.  On the Affected Activity Data Type, select the disposition attribute and edit the Enumerated Value List from the Constraints tab.

</INTERNAL>

* Copy InventoryDispositionRB.rbinfo to Windchill\src\wt\change2
* Start the tool and browse to InventoryDispositionRB.RB.ser in codebase

</INTERNAL>

### **Specify a default value for disposition**

Use enumCustomize tool described in the “**The Enumerated Type Customization Utility**” section of the Windchill Customizer’s Guide ([link](http://www.ptc.com/WCMS/files/89337/en/WCCustomizersGuide.pdf)) to select a default disposition value.

</INTERNAL>

enumCustomize tool will display all possible inventory disposition values available for inventory disposition types. Select the value you want to set as a default and click “Default Value” box

A ‘default value’ means the value that will be preselected in the inventory disposition UI drop down. Unfortunately, the tool only allows setting one default value per master list of inventory disposition values instead of setting a default value per inventory disposition type.

</INTERNAL>

If the default disposition value is not added to the constraint set of an inventory disposition type, a blank shall be shown as the preselected value in the inventory disposition UI drop down

### **Introduce new types of dispositions**

Inventory disposition types are enumerated columns that are added to the AffectedActivityData Link. There are inventory dispositions types that are added OOTB; customizers can add more inventory disposition types as soft attributes on the AffectedActivityData Link to meet their business needs by using the Type Manager. Here are the steps to follow

#### Customization points

* Extend AffectedItemsTableBuilder and Override getDispositionComponentIds API to account for your new inventory disposition types. Also, if you decide to extend ChangeTaskAffectedItemsTableViews as opposed to just changing it, then you will need to override buildConfigurationTable to return your new table views.

*@ComponentBuilder("changeTask.affectedItemsTable")*

*public class YourNewTableBuilder extends AffectedItemsTableBuilder {*

*/\*\**

*\* Returns the list of supported disposition types.*

*\*/@Override*

*public List<String> getDispositionComponentIds() {*

*}*

*/\*\**

*\* Returns the configurable table.*

*\*/@Override*

*public ConfigurableTable buildConfigurableTable(String id) throws WTException {*

*}*

*}*

* Register your builder.

The builder will be discovered dynamically by the MVC framework. In order for the framework to find your builder, you will need to register either the base package location of the builder or the builder class in the xml configuration file under the mvc package in the src\_web variant. Here is an example for AffectedItemsTableBuilder as registered in ChangeManagement-configs.xml

* How to register the base builder package. Add this to the xml configuration file

*<mvc:builder-scan base-package="com.ptc.windchill.enterprise.change2.mvc.builders"/>*

* Or register the builder class. Add this to the xml configuration file

*<bean class=" com.ptc.windchill.enterprise.change2.mvc.builders.tables.AffectedItemsTableBuilder "/>*

The @ComponentBuilder annotation connects the builder with other components in the system such as JSP’s. The string you put in the @ComponentBuilder(“changeTask.affectedItemsTable

“) annotation in the builder must be the same builder ID you include in the JSP.

For example affectedItemsTable.jsp has the following to request the change task affected item component.

*<jsp:include page="${mvc:getComponentURL('changeTask.affectedItemsTable')}"/>*

* Extend ChangeTaskAffectedItemsTableViews to add your new inventory disposition types to the link

*public class YourTableViews extends ChangeTaskAffectedItemsTableViews*

*{*

*/\*\**

*\* Get the disposition columns*

*\*/*

*@Override*

*public List<?> getSpecialTableColumnsAttrDefinition(Locale locale) {*

*}*

*}*

* Extend DefaultDispositionHandler to get/set the values of your new inventory disposition types. At a minimum, you will need to override the API’s given below:

*public class YourNewDispositionHandler extends DefaultDispositionHandler {*

*/\*\**

*\* Get disposition value for a given disposition type.*

*\*/*

*@Override*

*public InventoryDisposition getDispositionValue(String componentId, BinaryLink link) {*

*}*

*/\*\**

*\* Set the disposition value for a given disposition type. \*/*

*@Override*

*public boolean setDispositionValue(String componentId, BinaryLink link, InventoryDisposition disposition) throws WTException {*

*}*

*/\*\**

*\* Get the default disposition value for a given disposition type*

*\*/*

*@Override*

*public InventoryDisposition getDefaultInventoryDisposition(ChangeLinkAttributeBean linkBean, String component\_id) {*

*}*

*/\*\**

*\* Get the disposition value set of a given disposition type*

*\*/*

*@Override*

*public ArrayList<InventoryDisposition> getInventoryDispositionSet(ChangeLinkAttributeBean linkBean, String component\_id) {*

*}*

*}*

Register your new disposition handler in the service properties file (ChangeManagement-service.properties.xconf). The selector attribute is your new inventory disposition type and the serviceClass is your new disposition handler.

<! -- Handler for processing the disposition types -->

<Service context="default" name="com.ptc.windchill.enterprise.change2.handler.DispositionHandler"><Option serviceClass="com.ptc.windchill.enterprise.change2.handler.YourNewDispositionHandler"

selector="YourNewDispositionType"

requestor="null"

cardinality="duplicate"/>

</Service>

* Register your disposition types against DispositionDataUtility. You may also choose to extend DispositionDataUtility to meet your business need. The service property file to register the dispositions is ChangeManagement-components.dataUtilities.properties.xconf.

Here is the example;

<Option **serviceClass=**"com.ptc.windchill.enterprise.change2.dataUtilities.DispositionDataUtility"

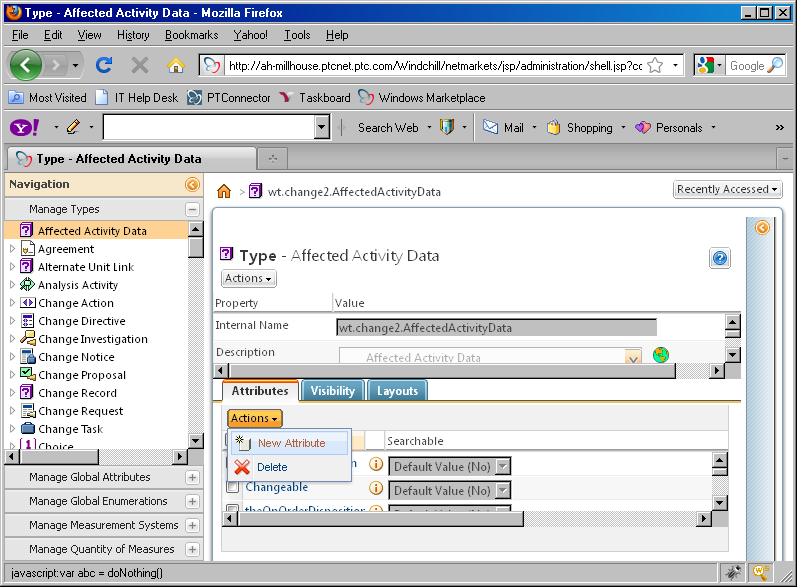
**requestor=**"[java](http://ah-opengrok.ptcnet.ptc.com/s?path=java&project=/x-20).[lang](http://ah-opengrok.ptcnet.ptc.com/s?path=java/lang&project=/x-20).[Object](http://ah-opengrok.ptcnet.ptc.com/s?path=java/lang/Object&project=/x-20)"

**selector=**"YourNewDisposition"

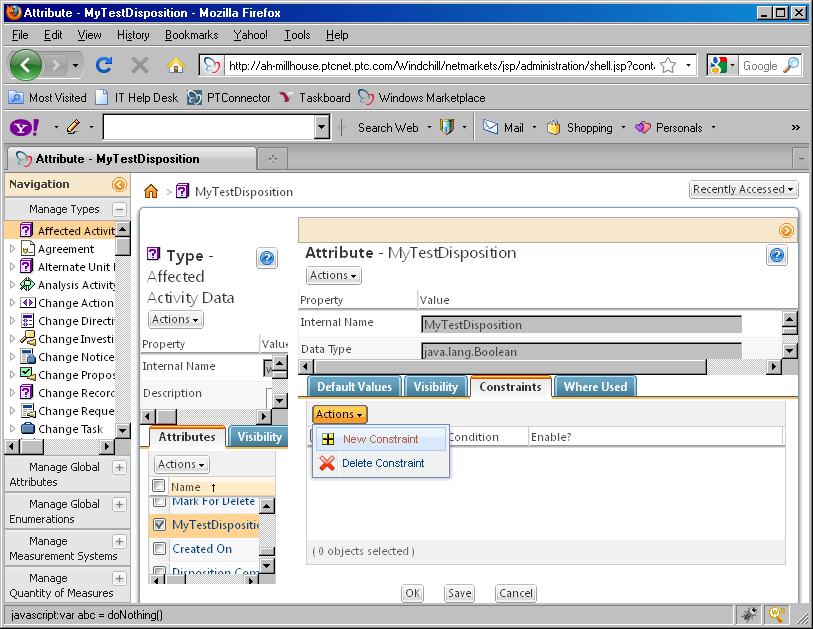
**cardinality=**"duplicate"/>

#### The UI

* As the site administrator, go to Type Manager and create a new soft attribute for AffectedActivityData Link, this is the inventory disposition type



* Constrain the attribute to the new disposition values you created in section 3.3.1



* <<<Question for Nate/John>> How about the DTD’s

Now that we have configured multiple disposition types and values, it is time to create an instance of multiple dispositions on the AffectedActivityData link.

* Create a change notice and update the default change task by going to “Select Affected/Resulting Objects” step
* Your new disposition type(s) should show as column(s) on the affected objects table
* The drop-down under your new disposition type (column) should show your new disposition values constrained to that type
* To change the disposition values for a disposition type, click Actions drop-down and select “Set Dispositions”.

