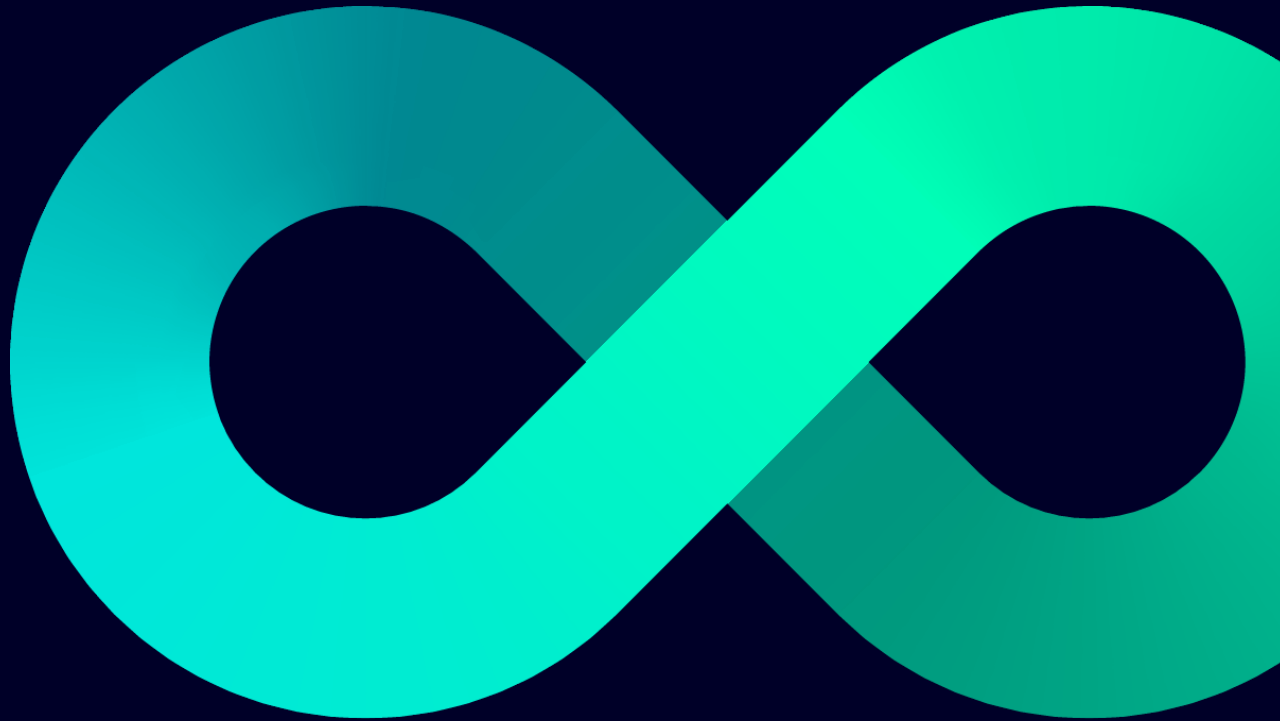


Realize
LIVE

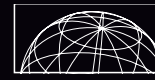
Teamcenter BMIDE Conditions for Workflows

Jesse Binford
Sr PLM Architect, Manager
Applied CAx



What is Applied CAx?

FINITE ELEMENT ANALYSIS
Predictive Engineering



APPLIED CAx

A Siemens Software Partner built from a joint venture of Predictive Engineering (FEA consulting services) and Applied CAx, a systems integrator.

Focused on A&D and Renewable Energy, Applied was recognized as Siemens North American Partner of the Year: 2023, 2020, and 2016.

2023 North America
PARTNER OF THE YEAR
APPLIED CAx



NX CAE FEMAP LS-DYNA STAR-CCM+ NX TEAMCENTER FIBERSIM PLANT SIM PROCESS SIM

- 5 simulation engineers (PhDs, MEs, PEs)
- FEA consultants
- CFD consultants
- Training for FEMAP & NX Nastran
- Training for LS-DYNA Implicit & Explicit

- 9 veteran Teamcenter architects
- 4 Senior NX Application Engineers
- Training for NX CAD/CAM & Teamcenter
- Deep design experience
- PDM experience

Agenda

Old vs New

How does it work?

BMIDE Condition Formatting

Workflow Condition Examples

Old vs New

Out with the old, in with the new...

Old vs New

Workflow Template Filter

- No longer available, deprecated in 11.2
- These lists were maintained within the Workflow Designer
- Not flexible
- No required outage for changes

BMIDE Conditions

- Conditions are created and maintained within the Teamcenter BMIDE
- Conditions are used in various areas across Teamcenter
- Conditions are sets of logic that are executed by the system to determine a result
- Conditions are extremely flexible
- Changes require outage

How does it work?

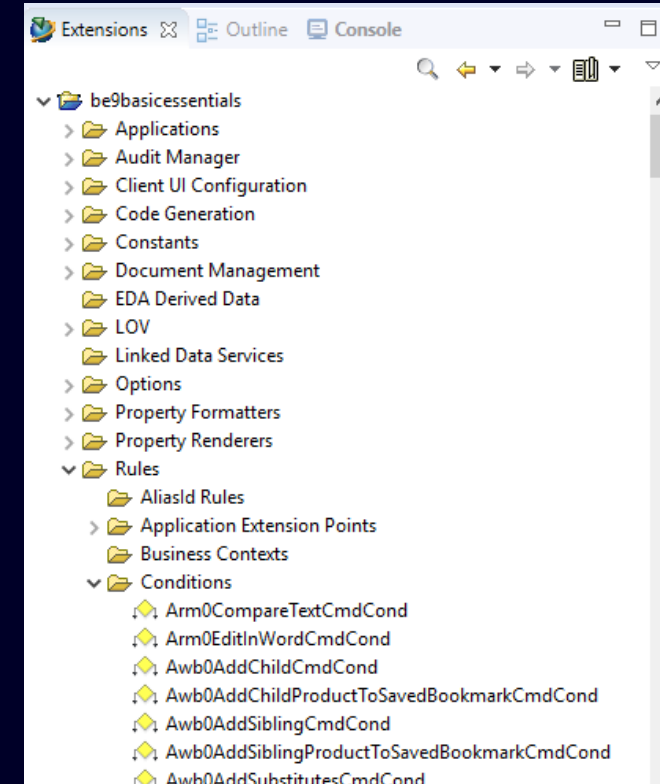
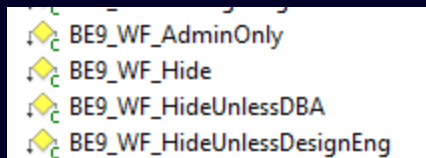
Setup Steps

1. Setup BMIDE conditions
2. Perform BMIDE deployment
3. Open Workflow Designer and set workflow template condition
4. (Optional) Configure Preference

Setup BMIDE Conditions

All BMIDE conditions can be found in Extensions under Rules -> Conditions

I recommend using a prefix or suffix to keep your conditions organized for easier visibility and management.



Setup BMIDE Conditions

All condition you wish to use on workflows must match the signature below. The alias's do not need to match mine (o, t, u).

WorkspaceObject o, ImanType t, UserSession u

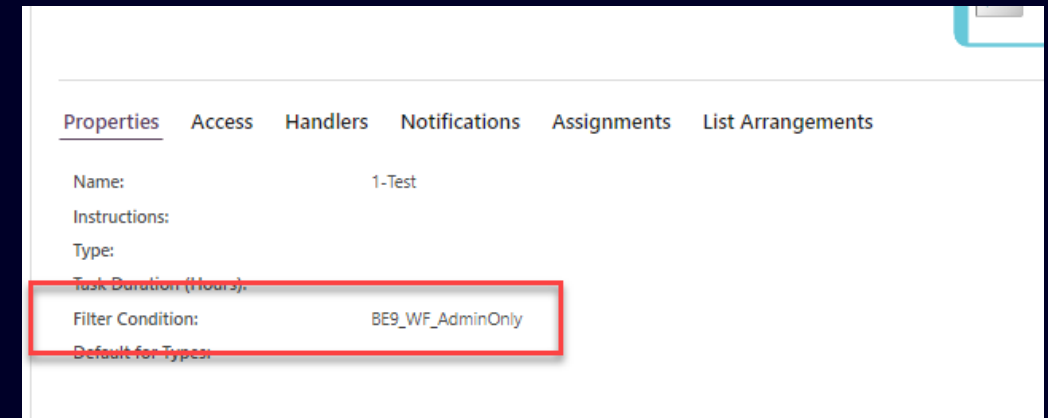
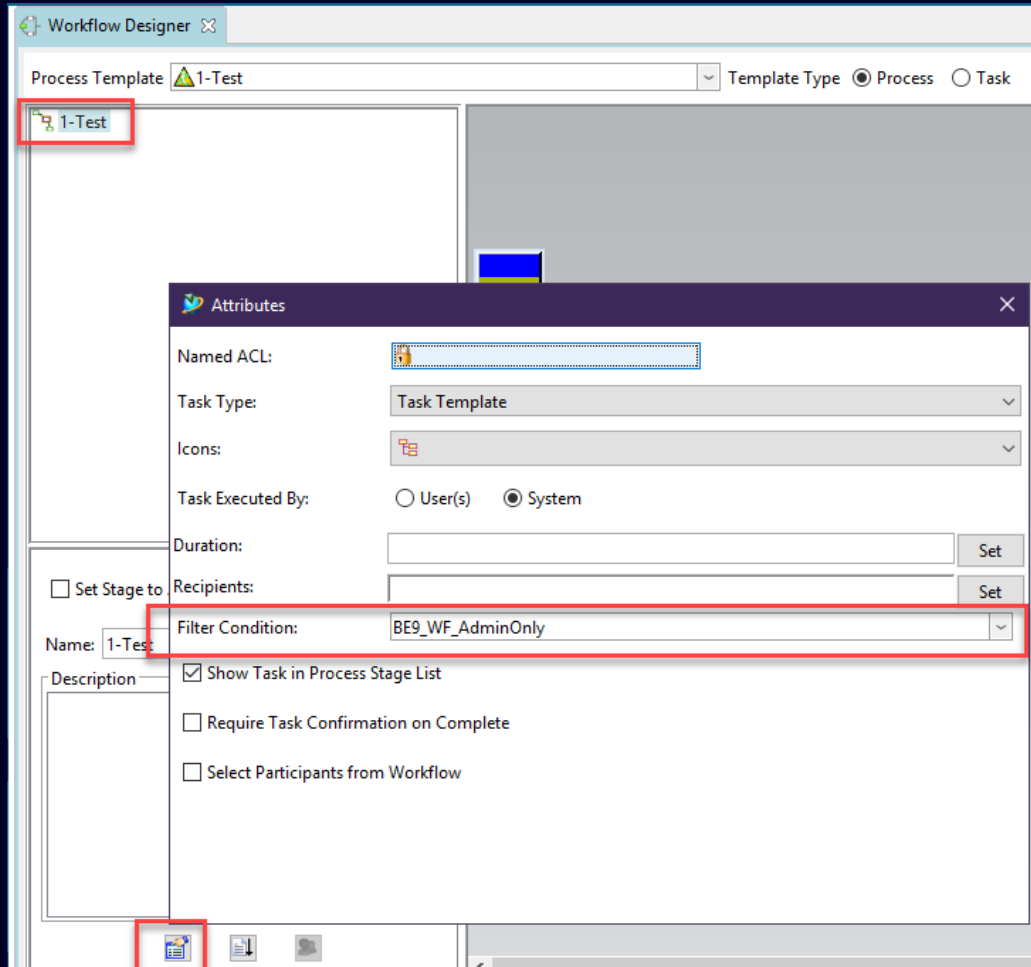
Input parameters:	☐ Business Object ☐ Business Object and User Session ☒ Custom
Signature:	BE9_WF_Hide (WorkspaceObject o , ImanType t , UserSession u)

Perform BMIDE Deployment

Perform your standard BMIDE deployment via Deployment Center.

You can also deploy in TEM if you really want or until it disappears.

Set Condition on Workflow Template



(Optional) Configure Teamcenter Preference

You can use the following preference to limit visibility in the New Workflow window to only show templates with valid conditions.

Name: CR_allow_alterate_procedures

Value: none

Name:	CR_allow_alterate_procedures
Product Area:	Workflow
Description:	Manages the visibility of workflow templates in the "New Process" panel of the Workflow application in the rich client. Valid Values: * any : The "All" option is pre-selected in the "All/Assigned" radio button group, which means that all templates are listed. * Assigned: The "Assigned" option is pre-selected in the "All/Assigned" radio button group, which means that only the templates assigned to the group and selected target object type are listed. This is the default value. * none: The options "All/Assigned" are not visible. Furthermore, only the templates assigned to the group and selected target object type are displayed (behaviour similar to the "Assigned" option). When no templates are assigned for the target group/type, the displayed list is empty. If no object was selected when initiating the workflow, all the templates assigned to the group are listed. Process templates assigned to a group and a target object type are defined through the "Process designer->Edit->Template Filter" command.
Protection Scope:	User
Environment:	Disabled
Location:	Site
Type:	String
Multiple Values:	No

▼ VALUES

Values:

none 

(Optional) Configure Teamcenter Preference

With the new preference applied, any new workflow windows will now only allow the user to select workflows with valid conditions.

Submit to Workflow

Workflow Assignments

Template: 1-Test

* Name:

1-Test : 1051001^-^TEST PART

Description:

||

▼ TARGETS

⊕ Add ✓ Select All

 **TEST PART**
1051001
Revision: -

BMIDE Condition Tips

BMIDE Condition Tips

BMIDE conditions are sets of logic that are evaluated at time of execution and provide a true or false result value.

For workflows, a true result from a condition will result in the workflow showing for the user.

Conditions can call other conditions (Nesting).

Conditions can be one or more statements to get to a result.

Parentheses “(“ “)” are your friend for grouping statements together.

You can use the term “AND” and “OR” between statements similar to Excel or other logic.

Conditions use the alias’s you configured in the signature.

BMIDE Condition Tips

`u.group_name != "dba"`

User's current session group name does not equal dba.

`u.group_name != "dba" AND u.group_name != "Engineering"`

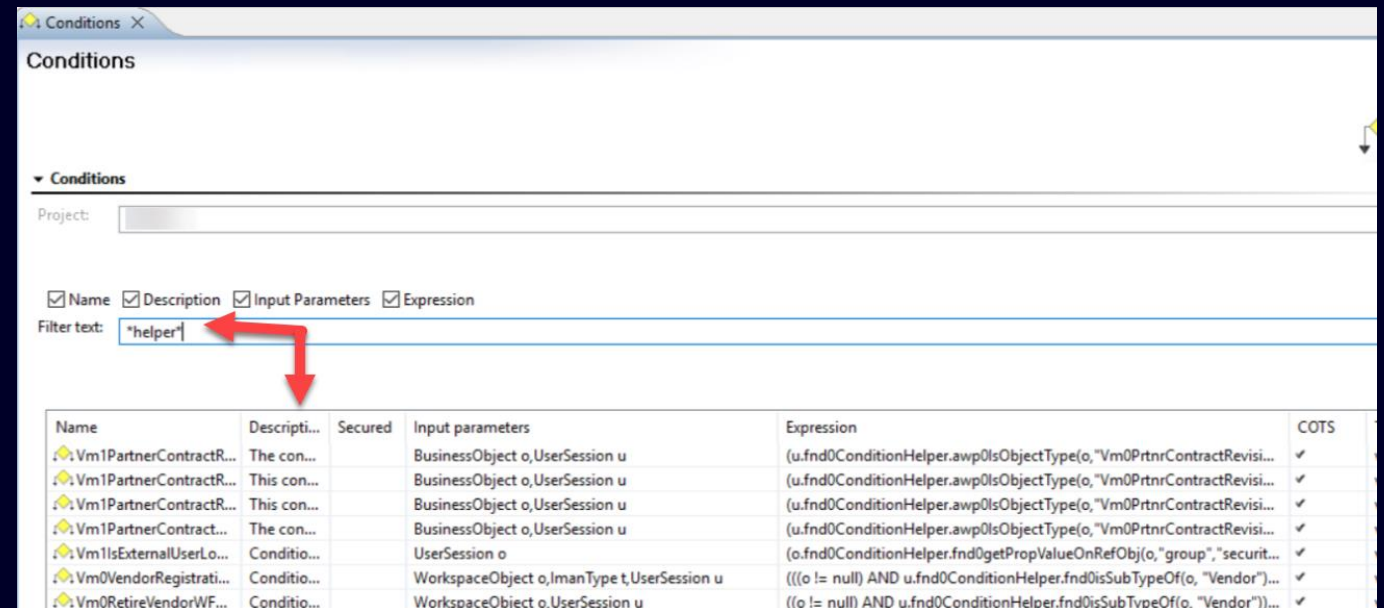
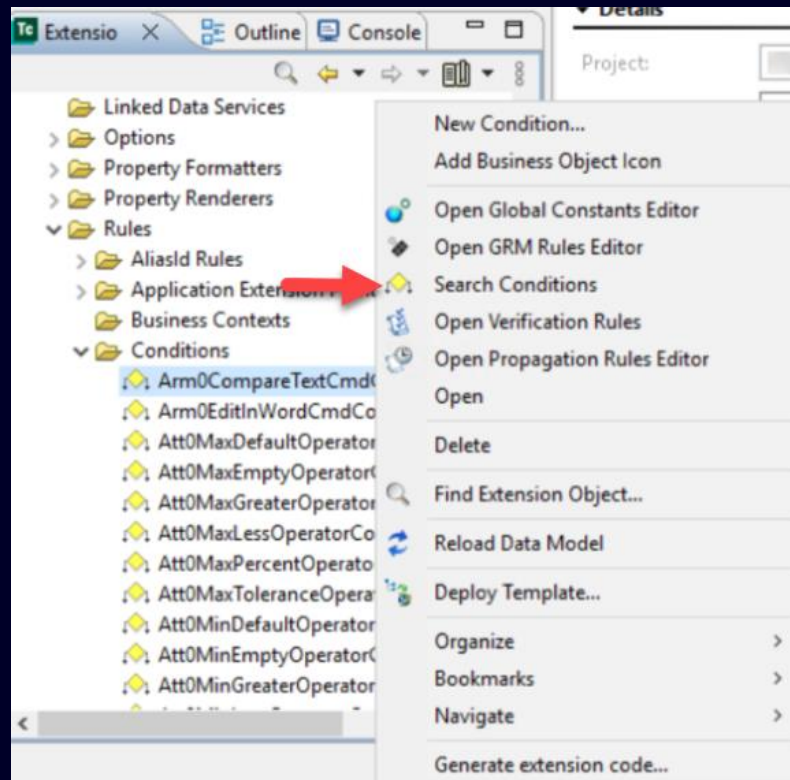
User's current session group name does not equal dba or Engineering.

`(u.group_name = "dba" AND u.role_name != "DBA") OR (u.group_name = "Engineering" AND u.role_name != "Approver")`

User's current session must be in group DBA and not role DBA or group Engineering and not role Approver.

BMIDE Condition Tips

Remember, you can search preferences in BMIDE!



Workflow Condition Examples

Workflow Condition Examples

Condition to hide the workflow from all users.

	☐ Secured
Input parameters:	☐ Business Object ☐ Business Object and User Session ☒ Custom
Signature:	BE9_WF_Hide (WorkspaceObject o , lmanType t , UserSession u)
Expression:	Condition::isFalse()

Workflow Condition Examples

Condition to only show the workflow to a user in the dba group.

(o != null) is restricting the workflow from starting without any WorkspaceObject selected in this example:

Input parameters:	☐ Business Object ☐ Business Object and User Session ☒ Custom
Signature:	BE9_WF_HideUnlessDBA (WorkspaceObject o , ImanType t , UserSession u)
Expression:	(o != null) AND (u.group_name = "dba")

Workflow Condition Examples

Condition to only allow the owning user of the object to start the workflow.

Input parameters:	☐ Business Object ☐ Business Object and User Session ☒ Custom
Signature:	<code>_WF_ArchiveOwn (WorkspaceObject o , ImanType t , UserSession u)</code>
Expression:	<code>((o != null) AND (o.owning_user=u.user))</code>

Workflow Condition Examples

Condition to only allow Design revisions with a property value matching “Priority”. The user must also be in the dba group or the Design_Engineering subgroup.

```
o != null AND o.object_type = "BE9_DesignRevision" AND (u.group_name="dba" OR  
u.fnd0getGroupName() = "/*.Design_Engineering.*") AND  
u.fnd0ConditionHelper.awp0GetPropertyValue(o,"be9_PartType","Priority",false)=true
```

☐ Secured

Input parameters: ☐ Business Object ☐ Business Object and User Session ☒ Custom

Signature: BE9_WF_PriorityParts (WorkspaceObject o , lmanType t , UserSession u)

Expression: `o != null AND o.object_type = "BE9_DesignRevision" AND (u.group_name="dba" OR u.fnd0getGroupName() = "/*.Design_Engineering.*") AND u.fnd0ConditionHelper.awp0GetPropertyValue(o,"be9_PartType","Priority",false)=true`

Workflow Condition Examples

You can use the following helper to validate property values:

awp0GetPropertyValue

Property Name

Result if the property cannot be found or null

u.fnd0ConditionHelper.awp0GetPropertyValue(o,"be9_PartType","Priority",false)=true

Input object from the condition

Value you are expecting in the property

```
graph TD; A[Property Name] --> C["u.fnd0ConditionHelper.awp0GetPropertyValue(o,\"be9_PartType\",\"Priority\",false)=true"]; B[Result if the property cannot be found or null] --> D[false]; E[Input object from the condition] --> F[o]; G[Value you are expecting in the property] --> H["Priority"];
```

Workflow Condition Examples

Condition to only allow Design revisions with a value in the Item Revision ID field that exists in a custom preference. The user must also be in the dba group or the Design_Engineering subgroup.

```
o != null AND o.object_type = "BE9_DesignRevision" AND (u.group_name="dba" OR  
u.fnd0getGroupName() = "*.Design_Engineering.*") AND  
u.fnd0ConditionHelper.fnd0isValuePresentInPref("BE9_AllowedRevisionCodesForProduction",  
u.fnd0ConditionHelper.awp0GetStringPropertyValue(o,"item_revision_id"))=true
```

Input parameters:	☐ Business Object ☐ Business Object and User Session ☒ Custom
Signature:	BE9_WF_ProdRevisionOnly (WorkspaceObject o , lmanType t , UserSession u)
Expression:	<pre>o != null AND o.object_type = "BE9_DesignRevision" AND (u.group_name="dba" OR u.fnd0getGroupName() = "*.Design_Engineering.*") AND u.fnd0ConditionHelper.fnd0isValuePresentInPref("BE9_AllowedRevisionCodesForProduction",u.fnd0ConditionHelper.awp0GetStringPropertyValue(o,"item_revision_id"))=true</pre>

Workflow Condition Examples

You can use the following helper to check a value against value(s) in a preference:

fnd0isValuePresentInPref

This will be the preference
name we want to check from
Teamcenter



```
u.fnd0ConditionHelper.fnd0isValuePresentInPref("BE9_AllowedRevisionCodesForProduction",  
u.fnd0ConditionHelper.awp0GetStringPropertyValue(o,"item_revision_id"))=true
```



This will get the string value for the
field 'item_revision_id'

[Overview](#) [Template Data Model](#) [Whats new](#) [Glossary](#) [Deprecated](#)

Copyright 2020 Siemens Product Lifecycle Management Software Inc. All Rights reserved.
This documentation is proprietary and confidential to Siemens Product Lifecycle Management Software Inc.

Business Object: *Fnd0ConditionHelper*

Secondary Business Object

Owning Template: *foundation*

Description: *Fnd0ConditionHelper* is a singleton runtime business object that represents a common placeholder to define generic operations that can be used by condition expressions. A typed reference property on Use operations include: *fnd0isNullDate* to check if a given date property value is NULLDATE, *fnd0now* to return the current date time, and *fnd0isValuePresentInPref* to check if an input value is present in the given preference.

- + [BusinessObject](#)
 - + [RuntimeBusinessObject](#)
 - + [Fnd0ConditionHelper](#)

Field Summary	
Name	Fnd0ConditionHelper
Description	Fnd0ConditionHelper is a singleton runtime business object that represents a common placeholder to define generic operations that can be used by condition expressions. Examples of generic operations include: <i>fnd0isNullDate</i> to check if a given date property value is NULLDATE, <i>fnd0now</i> to return the current date time, and <i>fnd0isValuePresentInPref</i> to check if an input value is present in the given preference name.
Package	null
Parent	RuntimeBusinessObject
Abstract?	false
Artifact Name	Fnd0ConditionHelper

Contact

Published by Siemens

Jesse Binford

Sr PLM Architect, Manager | Digital Innovation Services
Applied CAX

E-mail jbinford@appliedcax.com