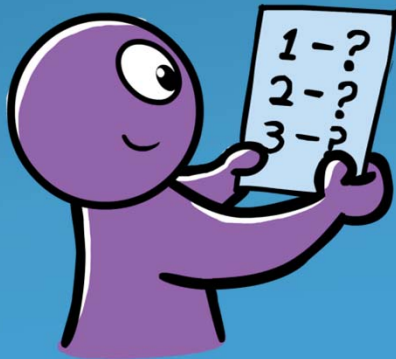


Creating Applicability Statements that Work for the CCT!

Or, How to Succeed at Conditional Processing
Without Killing Yourself



Presenter: **Mike Cook**

Title: **Senior S1000D Analyst**

Company: **SDL**

For: **S1000D User Forum – San Diego**

September 21st through 23rd

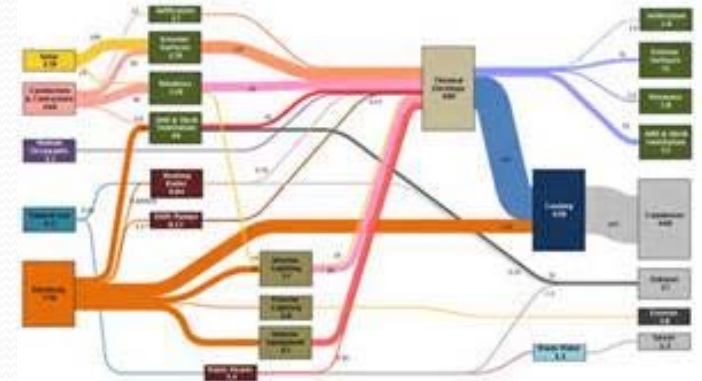
Applicability is all about...

- Getting to “TRUE”
- **And**, figuring out how conditionals work in the Conditions Cross-reference Table (CCT) – which still requires getting to “TRUE”
- Making PCT based applicability work is fairly easy, so let’s focus on the CCT and also how it works with the PCT since that can be fairly hard
- Since the CCT isn’t all that easy to understand, let’s start there
- It is all about “conditions”, but there is more than one type of condition



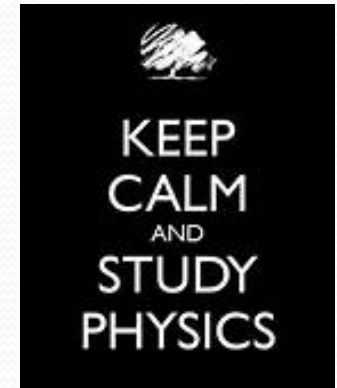
Conditions Cross-reference Table (CCT)

- Conditions come in two or three flavors and are known as **Condition Types**:
 - Environmental Factors
(you won't find this definition in the spec)
 - Events or Time dependencies
(like PRE and POST for service bulletins and TCTOs)
 - And Global Variables (aka Global Properties)
Global variables are for various uses – but mostly in *process.xsd* based data modules
 - Sorry, we're not getting into global variables today



What is a condition?

- Basically, it's one of the following:
 - An environmental factor or **threshold** for a value based on physics as it pertains to the product:
 - Freezing or boiling point of water
 - Barometric pressure points
 - PH values
 - Hydraulic pressure tolerance points
 - High and low refrigerant gas values for a specific system
 - etc
 - An **event** or **artificial threshold** of the product based on changes to maintenance procedures due to engineering changes:
 - PRE or POST
 - PRE and POST-001, POST-002, POST-003, etc
- ❖ A variable for use in a process data module
 - As of 4.1, there is a way to define global variables in the CCT using the *valueDataType* attribute. Are these, by themselves, conditions? No not really, but when used in a process data module they can be tested for values that are compared against predefined norms.



What is an Event or Time dependency?

- Any threshold or condition requires something to compare a value against
- Any value being tested for a decision path must be provided by the end user
- The value provided by the end user requires the IETP to manage that value in something called a “state” table (not the CCT – a table in the IETP for variables and their values)
- That variable generally has either the condition name or product attribute name
- That variable is then compared against the various conditions or product attributes with the same name to identify if it is TRUE or FALSE



How an IETP uses Conditional Applicability

```
<applic id="APPLIC-0006"><!-- If temperatue is below freezing -->  
<assert applicPropertyIdent="temperatureF" applicPropertyType="condition" applicPropertyValues="-300~32"/>  
</applic>  
<applic id="APPLIC-0005"><!-- If temperature is above freezing -->  
<assert applicPropertyIdent="temperatureF" applicPropertyType="condition" applicPropertyValues="33~212"/>  
</applic>
```

If, in the state table, the property identifier (variable) has no value, then request it from the end user

```
<proceduralStep id="PS-0004">
```

```
<para>When draining the condensate drain pan, observe the following:</para>
```

```
<proceduralStep applicRefId="APP-005"><!-- water is liquid -->
```

```
<para>Pull the drip line free of the condenser plenum drain port.</para>
```

```
<para>Inspect the condenser drain port for dust and debris. Clean as necessary.</para>
```

```
<para>Reinsert drip line into the condenser plenum drain port.</para>
```

```
</proceduralStep>
```

```
<proceduralStep applicRefId="APP-006"><!-- water is frozen-->
```

```
<para>Heat the base of the condenser plenum with a heat gun set to 150 degrees for one minute.</para>
```

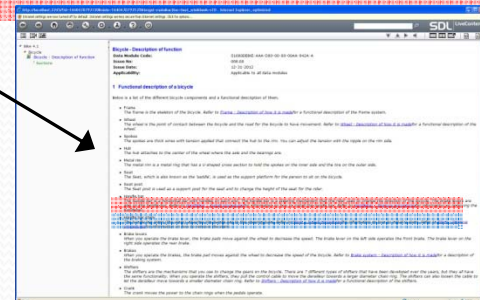
```
<para>Pull the drip line free of the condenser plenum drain port.</para>
```

```
<para>Inspect the condenser drain port for dust and debris. Clean as necessary.</para>
```

```
<para>Warm the drip line with the heat gun until pliable and reinsert into the condenser plenum drain port.</para>
```

```
</proceduralStep>
```

```
</proceduralStep>
```



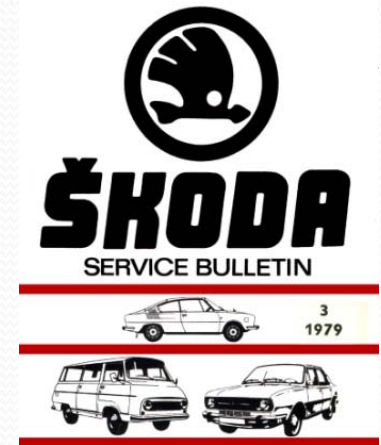
Evaluating and Resolving applicability

- When the value is evaluated, knowing where to look to resolve the condition requires looking in either the PCT or CCT data modules
- Resolving uses the ***applicPropertyType*** attribute of the <assert> element to know where to look – either the PCT (prodattr) or the CCT (condition)
- If the end user wants to configure the IETP to provide only the information about the product being worked on, the user would enter the unique identifier of that product instance, like a production number, tail number, or serial number
- The IETP would then, using applicability, display only the information **without** applicability markup **and** markup with applicability that resolves to TRUE



Why Does a Service Bulletin or TCTO Need to be Managed as a Condition?

- Service Bulletins and TCTOs impose an event *in time* as well as *per product instance* as to which procedures must be followed to assure proper maintenance
 - There is generally a PRE condition of the product – what was there originally as a service procedure prior to the service bulletin/engineering change
 - Followed by a POST – what is new due to a configuration change in the product imposed by the service bulletin/engineering change that may require new servicing/maintenance procedures
 - Depending on the manufacturer, they may choose to use a simple PRE/POST paradigm, or impose the slightly more complex idea of an aggregated single Service Bulletin that can have multiple POST scenarios (POST-001, POST-002, POST-003, etc)



Why do we Need PRE and POST?

- Without the ability to use PRE and POST as an applicability assertion, we must show the maintenance technician **all** the possible PRE and POST conditions and make them choose what to do.
- If we **know** the configuration of the product when they begin a maintenance task, the IETP can identify what to show the technician – either PRE for a product that has not had the Service Bulletin applied, or the POST configuration knowing that the Service Bulletin has been applied
- However, this requires the holy grail of knowledge **reported back** to the managers of the applicability that the service bulletin was applied to the product so the appropriate **incorporation status** can be configured for that service bulletin



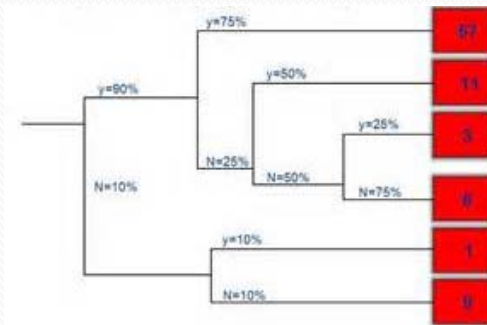
Configuring a Condition Type

Examples of a **condition type** definition:

```
<condTypeList>
  <condType id="CCT-001">
    <name>SB</name>
    <descr>Service Bulletin</descr>
    <enumeration applicPropertyValues="PRE"/>
    <enumeration applicPropertyValues="POST"/>
  </condType>
  <condType id="CCT-002">
    <name>Water</name>
    <descr>Freezing and Boiling point of water</descr>
    <enumeration applicPropertyValues="32"/>
    <enumeration applicPropertyValues="212"/>
  </condType>
```

- OR -

```
<condType id="CCT-001">
  <name>SB</name>
  <descr>Service Bulletin</descr>
  <enumeration applicPropertyValues="PRE|POST"/>
</condType>
<condType id="CCT-002">
  <name>Water</name>
  <descr>Freezing and Boiling point of water</descr>
  <enumeration applicPropertyValues="32|212"/>
</condType>
```



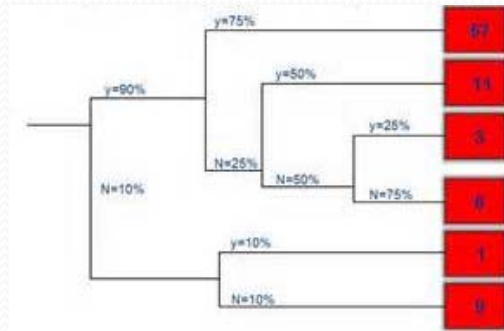
For each <condType> you create, you must include an *id* attribute value. Without it, references won't work – and you need to be able to reference the definitions. The two examples here should be easy to follow, the second example is just a duplicate of the first using the alternate “|” (or) paradigm.

Using a Condition Type in the Condition List

```
<condType id="CCT-001">  
  <name>SB</name>  
  <descr>Service Bulletin</descr>  
  <enumeration applicPropertyValues="PRE|POST"/>  
</condType>  
<condType id="CCT-002">  
  <name>Water</name>  
  <descr>Freezing and Boiling point of water</descr>  
  <enumeration applicPropertyValues="32|212"/>  
</condType>
```

...

```
<condList>  
  <cond id="SB20019" condTypeRefId="CCT-001">  
    <name>SB20019 - Brake retrofit</name>  
    <displayName>Brake retrofit Bendix B20019</displayName>  
    <descr>Brake retrofit kit for SB2000</descr>  
    <refs>  
      <pmRef>  
        <pmRefIdent>  
          <pmCode modelIdentCode="SB2000" pmlIssuer="10295" pmNumber="00001" pmVolume="00"/>  
        </pmRefIdent>  
      </pmRef>  
    </refs>  
  </cond>
```



Creating Applicability Assertions Based on Condition Types

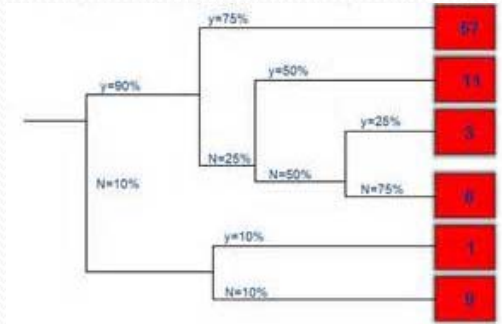
This is for creating applicability assertions in a data module where it will be used (inside the <referencedApplicGroup>).

For solid and liquid water:

```
<applic id="APPLIC-0006"><!-- If temperatue is below freezing -->
  <assert applicPropertyIdent="temperatureF" applicPropertyType="condition" applicPropertyValues="-300~32"/>
</applic>
<applic id="APPLIC-0005"><!-- If temperature is above freezing -->
  <assert applicPropertyIdent="temperatureF" applicPropertyType="condition" applicPropertyValues="33~212"/>
</applic>
```

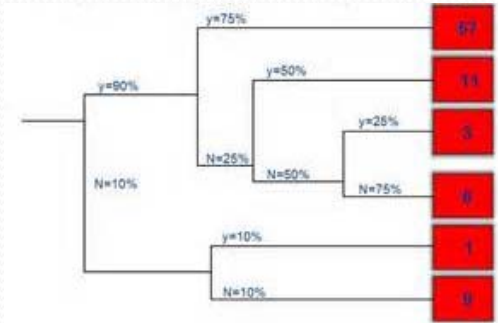
For gas pressure in an R22 refrigeration system:

```
<applic id="APPLIC-0007"><!-- If R22 pressure is low -->
  <assert applicPropertyIdent="R22PressureHighSide" applicPropertyType="condition" applicPropertyValues="0~49"/>
</applic>
<applic id="APPLIC-0008"><!-- If R22 pressure is normal -->
  <assert applicPropertyIdent="R22PressureHighSide" applicPropertyType="condition" applicPropertyValues="50~180"/>
</applic>
<applic id="APPLIC-0009"><!-- If R22 pressure is high -->
  <assert applicPropertyIdent="R22PressureHighSide" applicPropertyType="condition" applicPropertyValues="181~500"/>
</applic>
```



Creating Applicability Assertions Based on Conditions in the Condition List

- PRE and POST is a much more difficult assertion type to master
- It requires knowing much more about the product and specifically each product instance



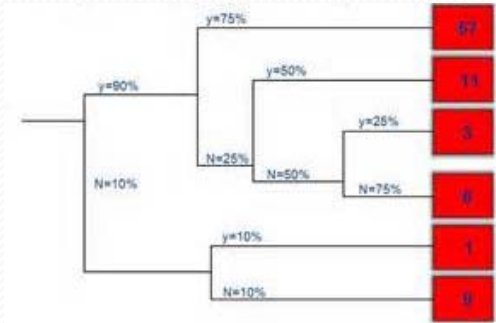
```
<applic id="APPLIC-0012">  
  <evaluate andOr="and">  
    <assert applicPropertyId="model" applicPropertyType="prodattr" applicPropertyValues="SB2000"/>  
    <assert applicPropertyId="PR-20592011" applicPropertyType="condition" applicPropertyValues="PRE"/>  
    <assert applicPropertyId="PR-25931722" applicPropertyType="condition" applicPropertyValues="POST001"/>  
  </evaluate>  
</applic>
```

Look to resolve this from the PCT data module

Look to resolve this from the CCT data module

How Incorporation Status Affects PRE and POST Applicability Assertions

- When all the product associated with a Service Bulletin have an “incorporationStatus” of “incorporated”, all authored content associated with that service bulletin is now a ***candidate*** for removal.
- The PRE content can be removed and the POST content can now become “regular” unmarked content with or without applicability (depending on whether or not all instances of product use the POST procedures or only specific instances of product are associated with that content).



Example PRE and POST removal

```
<applic id="APP-001">
  <assert applicPropertyId="PR-20592011" applicPropertyType="condition" applicPropertyValues="POST"/>
</applic>
<applic id="APP-002">
  <assert applicPropertyId="PR-20592011" applicPropertyType="condition" applicPropertyValues="PRE"/>
</applic>
...
<proceduralStep id="PS-0004">
  <para>There are two Warp Nacelles, one on the left warp pod and one on the right warp pod.</para>
  <proceduralStep applicRefId="APP-002"><!-- Shows when PR-20592011 is POST and PR-25931722 is PRE -->
    <para>The steps to remove and install the port warp nacelle must be performed in space dock.</para>
    <para>The steps to remove and install the starboard warp nacelle can only be accomplished in the field after a
      barion sweep of the entire ship. See task: <internalRef internalRefId="MSFP-1950A"/>.</para>
  </proceduralStep>
  <proceduralStep applicRefId="APP-001"><!-- Shows when PR-20592011 is PRE and PR-25931722 is POST -->
    <para>The steps to remove and install the warp nacelles requires a hydro-spanner with a warp nulling anode. See
      task: <internalRef internalRefId="MSFP-1953B"/>.</para>
  </proceduralStep>
</proceduralStep>
```

Everything in blue can be removed. Everything in red can remain, after Service Bulletin PR-20582011 is fully incorporated across all instances of product affected by the Service Bulletin.

Examples of Environmental Factors Assertion Statements

```
<applic id="APP-008"> <!-- If temperatue is below freezing -->
  <assert applicPropertyIdent="temperatureF" applicPropertyType="condition" applicPropertyValues="-300~32"/>
</applic>
<applic id="APP-007"> <!-- If temperatue is above freezing -->
  <assert applicPropertyIdent="temperatureF" applicPropertyType="condition" applicPropertyValues="33~212"/>
</applic>
...
<proceduralStep id="PS-0004">
  <para>When draining the condensate drain pan, observe the following:</para>
  <proceduralStep applicRefId="APP-007"><!-- water is liquid -->
    <para>Pull the drip line free of the condenser plenum drain port.</para>
    <para>Inspect the condenser drain port for dust and debris. Clean as necessary.</para>
    <para>Reinsert drip line into the condenser plenum drain port.</para> </proceduralStep>
  <proceduralStep applicRefId="APP-008"><!-- water is frozen-->
    <para>Heat the base of the condenser plenum with a heat gun set to 150 degrees for one minute.</para>
    <para>Pull the drip line free of the condenser plenum drain port.</para>
    <para>Inspect the condenser drain port for dust and debris. Clean as necessary.</para>
    <para>Warm the drip line with the heat gun until pliable and reinsert into the condenser plenum drain port.</para>
  </proceduralStep>
</proceduralStep>
```

The assumption in this markup is the property identifier **temperatureF** (which in this case is actually a variable) has a value in it. Usually, this is set by the user from a dialog presented by the IETP and then tested by the transform at display time against variables listed in the “state” table.

Examples of Service Bulletin Assertion Statements

- To properly use applicability, you need to understand how it was designed to be used. The main idea is to point to the service bulletin of interest and then indicate whether it is PRE or POST (basically meaning there is a predefined “condition type” of PRE and POST).
- The spec also tends to make you think it’s okay to create a condition in the PCT without telling you why you really shouldn’t
- For example:

```
<applic id="APP-002">  
  <evaluate andOr="and">  
    <assert applicPropertyIdent="PR-20592011" applicPropertyType="condition" applicPropertyValues="POST"/>  
    <assert applicPropertyIdent="PR-25931722" applicPropertyType="condition" applicPropertyValues="PRE"/>  
  </evaluate>  
</applic>  
<applic id="APP-001">  
  <evaluate andOr="and">  
    <assert applicPropertyIdent="PR-20592011" applicPropertyType="condition" applicPropertyValues="PRE"/>  
    <assert applicPropertyIdent="PR-25931722" applicPropertyType="condition" applicPropertyValues="POST001"/>  
  </evaluate>  
</applic>
```

There is no reference to instances of product in these applicability statements because the CCT has this information in its <referencedApplicGroup> and is tied to each service bulletin

Example (mutually exclusive)

```
<applic id="APP-002">
  <evaluate andOr="and">
    <assert applicPropertyIdent="PR-20592011" applicPropertyType="condition" applicPropertyValues="POST"/>
    <assert applicPropertyIdent="PR-25931722" applicPropertyType="condition" applicPropertyValues="PRE"/>
  </evaluate>
</applic>
<applic id="APP-001">
  <evaluate andOr="and">
    <assert applicPropertyIdent="PR-20592011" applicPropertyType="condition" applicPropertyValues="PRE"/>
    <assert applicPropertyIdent="PR-25931722" applicPropertyType="condition" applicPropertyValues="POST001"/>
  </evaluate>
</applic>
...
<proceduralStep id="PS-0004">
  <para>There are two Warp Nacelles, one on the left warp pod and one on the right warp pod.</para>
  <proceduralStep applicRefId="APP-002"><!-- Shows when PR-20592011 is POST and PR-25931722 is PRE -->
    <para>The steps to remove and install the port warp nacelle must be performed in space dock.</para>
    <para>The steps to remove and install the starboard warp nacelle can only be accomplished in the field after a
      barion sweep of the entire ship. See task: <internalRef internalRefId="MSFP-1950A"/>.</para>
  </proceduralStep>
  <proceduralStep applicRefId="APP-001"><!-- Shows when PR-20592011 is PRE and PR-25931722 is POST001 -->
    <para>The steps to remove and install the warp nacelles requires a hydro-spanner with a warp nulling anode. See
      task: <internalRef internalRefId="MSFP-1953B"/>.</para>
  </proceduralStep>
</proceduralStep>
```

PCT Based Applicability vs. CCT Centric Assertions

Let the <referencedApplicGroup> in the CCT drive this out – you normally don't need to indicate the instances of product in this way in a regular data module. You should be defining product level applicability in the CCT for service bulletin related dependencies.

```
<applic id="APPLIC-0012">  
  <evaluate andOr="and">  
    <assert applicPropertyId="prodno" applicPropertyType="prodattr" applicPropertyValues="15~52|109~141"/>  
    <assert applicPropertyId="PR-20592011" applicPropertyType="condition" applicPropertyValues="PRE"/>  
    <assert applicPropertyId="PR-25931722" applicPropertyType="condition" applicPropertyValues="POST"/>  
  </evaluate>  
</applic>
```

```
<applic id="APPLIC-0001">  
  <assert applicPropertyId="prodno" applicPropertyType="prodattr" applicPropertyValues="25"/>  
</applic>  
<applic id="APPLIC-0002">  
  <assert applicPropertyId="prodno" applicPropertyType="prodattr" applicPropertyValues="1~24"/>  
</applic>
```

```
<applic id="APPLIC-0010">  
  <evaluate andOr="or">  
    <assert applicPropertyId="prodno" applicPropertyType="prodattr" applicPropertyValues="1~7"/>  
    <evaluate andOr="and">  
      <assert applicPropertyId="prodno" applicPropertyType="prodattr" applicPropertyValues="22~50"/>  
      <assert applicPropertyId="engine" applicPropertyType="prodattr" applicPropertyValues="3.2"/>  
      <assert applicPropertyId="trans" applicPropertyType="prodattr" applicPropertyValues="M"/>  
    </evaluate>  
  </evaluate>  
</applic>
```

Example – inside a CCT data module

```
<applic id="APP-001">  
  <assert applicPropertyIdent="prodno" applicPropertyType="prodattr"  
    applicPropertyValues="8|10~17|19|21|23|24|31~55|66|71|99|101|105~107|110|112|125~138|145|151|169|188"/>  
</applic>
```

...

```
<condList>
```

```
<cond id="SB20019" condTypeRefId="CCT-001">  
  <name>SB20019 - Brake retrofit</name>  
  <displayName>Brake retrofit Bendix B2001</displayName>  
  <descr>Brake retrofit kit for SB2000</descr>  
  <refs>  
    <pmRef>  
      <pmRefId>  
        <pmCode modelIdentCode="SB2000" pmIssuer="10295" pmNumber="00001" pmVolume="00"/>  
      </pmRefId>  
    </pmRef>  
  </refs>  
</cond>
```

...

```
<incorporation>
```

```
<condIncorporation condRefId=" SB20019 ">  
  <documentIncorporation>  
    <incorporationStatus incorporationStatus="inprogress" applicRefId="APP-001"/>  
    <incorporationStatus incorporationStatus="incorporated" applicRefId="APP-002"/>  
  </documentIncorporation>  
</condIncorporation>
```

Points to PRE|POST definition

Two *mutually exclusive* applicability assertions

Because the applicability in the <incorporation> element points to the applicability in the CCT <referencedApplicGroup>, you don't need to use an <applic> in the data module that uses conditions (for PRE|POST conditions) to call out which instances of product the service bulletin applies to.

PCT Dependent Applicability Assertions

This...

```
<displayText>For production numbers 1 through 25</displayText>  
<assert applicPropertyIdent="prodno" applicPropertyType="prodattr" applicPropertyValues="1~25"/>
```

Is the same as...

```
<assert applicPropertyIdent="prodno" applicPropertyType="prodattr" applicPropertyValues="1~25">For  
production numbers 1 through 25</assert>
```

These are mutually exclusive...

```
<applic id="APPLIC-0001">  
  <assert applicPropertyIdent="prodno" applicPropertyType="prodattr" applicPropertyValues="21~25"/>  
</applic>  
<applic id="APPLIC-0002">  
  <assert applicPropertyIdent="prodno" applicPropertyType="prodattr" applicPropertyValues="1~20"/>  
</applic>
```

PCT Dependent Applicability Assertions

```
<applic id="APPLIC-0010">
  <evaluate andOr="or">
    <assert applicPropertyId="prodno" applicPropertyType="prodattr" applicPropertyValues="1~7">Product numbers 1 to 7</assert>
  <evaluate andOr="and">
    <assert applicPropertyId="prodno" applicPropertyType="prodattr" applicPropertyValues="22~50">Product numbers 22 to 50</assert>
    <assert applicPropertyId="engine" applicPropertyType="prodattr" applicPropertyValues="3.2">Product using a 3.2 liter engine</assert>
    <assert applicPropertyId="trans" applicPropertyType="prodattr" applicPropertyValues="M">Product using a manual transmission</assert>
  </evaluate>
</evaluate>
</applic>
```

```
<applic id="APPLIC-0011">
  <evaluate andOr="and">
    <assert applicPropertyId="prodno" applicPropertyType="prodattr" applicPropertyValues="1~20|16~32">Prods 1 to 20 or 16 to 32</assert>
  <evaluate andOr="or">
    <assert applicPropertyId="trans" applicPropertyType="prodattr" applicPropertyValues="M6">Prods with manual 6 gear tranny</assert>
    <assert applicPropertyId="rim" applicPropertyType="prodattr" applicPropertyValues="15">Prods with 15 inch rims</assert>
  <evaluate andOr="and">
    <assert applicPropertyId="submodel" applicPropertyType="prodattr" applicPropertyValues="Z25">Product sub models Z25</assert>
    <assert applicPropertyId="top" applicPropertyType="prodattr" applicPropertyValues="soft">Product top soft</assert>
  </evaluate>
</evaluate>
  <evaluate andOr="and">
    <assert applicPropertyId="submodel" applicPropertyType="prodattr" applicPropertyValues="Z24">Product sub models Z24</assert>
    <assert applicPropertyId="top" applicPropertyType="prodattr" applicPropertyValues="gull">Product top gull</assert>
  </evaluate>
</evaluate>
</applic>
```

If you're going to try and use "computed display text", you'll want to use the assert text format versus the <displayText> element since it allows a finer degree of control of the text displayed to the end user.



Thanks for Attending!