



# R12 Upgrade With Configurator Attributes to Fusion Configurator Engine

Siva Pola Managing Partner Cerebra Consulting Inc siva.pola@cerebra-consulting.com Jayaram Ampolu Vice President Cerebra Consulting Inc jayaram.ampolu@cerebra-consulting.com





## **Siva Pola Profile**

- Bachelor of Technology in Computer Science and Systems Engineering
- 16 Years Experience in Information Technology
- 10+ Years of Experience with Oracle Configurator and Oracle EBS





## Jayaram Ampolu Profile

- Bachelor of Technology in Computer Science and Systems Engineering
- 16 Years Experience in Information Technology
- 4+ Years of Experience with Oracle Configurator and Oracle EBS





## References

Oracle Configurator Methodologies Release 11i (Part No. B10618-01)

Oracle Configurator Installation Guide Release 12.1 (Part No. E14323-02)

Oracle Configurator Fusion Configurator Engine Guide Release 12.1 (Part No. E14325-02)

Oracle Configurator Release Notes, Release 12.1.1 [ID 729984.1]





# Agenda

- Oracle Configurator Attributes Methodology
- Fusion Configurator Engine Key Features
- Upgrading to Oracle Fusion Configurtaor Engine
- Key Considerations while upgrading models which have implemented configurator attributes





## **Configurator Attributes**

- Attributes of an configuration produced by the runtime configurator
- Predefined data items that record qualities of something like an Item or ps\_node
- For example, a sheet of aluminum (Item) can have attributes called length, width, gauge, and the percentages of the various components that make up the metal





## **Solution Overview**

- A configurator model is specially modified to contain configuration attribute data.
- A configurator extension captures the configuration attribute data from the model and inserts it in the table CZ\_CONFIG\_ATTRIBUTES.
- A downstream application can access the data from CZ\_CONFIG\_ATTRIBUTES.





## **Elements of Configurator Attributes Solution**

- Imported BOM Nodes
- Attribute Features
- Properties
- Descriptive Flexfield Definition
- CZ\_CONFIG\_ATTRIBUTES.
- Configurator Extension





## An Example

- Configurator Extension
- Search for BOM nodes with property ATTR\_n\_PATH
- Search for attribute node as specified by ATTR\_n\_PATH
- Get values of ATTR\_CONTEXT, ATTR\_NAME
- Get Runtime value of attribute feature
- Query DFF Tables using ATTR\_CONTEXT, ATTR\_NAME
- Get the column ATTRn in CZ\_CONFIG\_ATTRIBUTES
- Write data in to CZ\_CONFIG\_ATTRIBUTES table







## **Setup Task List for Configurator Attributes**

Task	Original Configurator Engine Fusion Configurator Eng				
1	Setup Descriptive	Flex fields			
2	Add attribute features				
	Associating the Attribute Features to Flex field Segments				
4	Associating the BOM Nodes with Attribute Features				
5	Compile CX Class and Create CX Archive.	Create CX Rule			
	Create CX Rule				





# **Setup Step 1 : Setup Descriptive Flexfields**

Setup the descriptive flex field contexts and segments

Configurator (Application), Configurator (Table Application) Table Name (CZ\_CONFIG\_ATTRIBUTES) Flexfields: Association with Contexts and Segments Contexts (in Model, value of AttributeFeatureName.ATTR CONTEXT) Name Internal Custom Standard Name Column Name Column Name Column Size ATTRIBUTE1 Duration ATTRIBUTE1 Customer ATTRIBUTE1 ATTRIBUTE2 Weight ATTRIBUTE2 ServicePlan ATTRIBUTE3 Color

Segments (in Model, value of AttributeFeatureName.ATTR\_NAME )





## **Setup Step 2 : Add attribute features**

**Color:** to capture the color of the BIKE\_FRAME being ordered

**Frame\_Size:** to capture the frame size of the BIKE\_FRAME being ordered

**Customer:** to capture the name of the customer placing the BIKE\_ORDER

**Service\_Plan:** to capture whether that customer elected a service plan for the BIKE\_ORDER

Model Structure: Added Attribute Structure					
BIKE_ORDER					
Color					
:					





#### **Setup Step 3** : Associating the Attribute Features to Flex field Segments

Add the Properties ATTR\_CONTEXT and ATT\_NAME to all attribute features

Model Structure: Association with Flexfield Segments					
BIKE_ORDER					
LBIKE_FRAME					
_FRAME_TYPE					
_MOUNTAIN					
_ColorATTR_CONTEXT = Standard, ATTR_NAME = Color					
_Red					
_Blue					
_Green					
_Frame_SizeATTR_CONTEXT = Standard, ATTR_NAME = Size					
_CustomerATTR_CONTEXT = Internal, ATTR_NAME = Customer					
Service_Plan ATTR_CONTEXT = Custom, ATTR_NAME = ServicePlan					
:					
:					





#### **Setup Step 4** : Associating the BOM Nodes with Attribute Features

Add the Properties ATTR\_n\_PATH to BOM Nodes Model Structure: Association with Attribute Features

BIKE_ORDER ATTR_1_PATH = Customer, ATTR_2_PATH = Service_Plan
[BIKE_FRAMEATTR_1_PATH = Color
_FRAME_TYPEATTR_1_PATH = Frame_Size
_MOUNTAIN
_Color
_Red
_Blue
_Green
_Frame_Size
_Customer
_Service_Plan
:





#### **Setup Step 4** : Associating the BOM Nodes with Attribute Features

- The Property ATTR\_n\_PATH should be unique within the scope of the current node.
- Property value has to be set to the node path to the attribute feature, starting from the BOM Model node that is nearest to the node.
- The nodes in the node path are delimited by a dot ('.')
- Multiple ATTR\_n\_PATH properties can be created for the same node to assign multiple attribute features to the node.
- You can use the same ATTR\_*n*\_PATH name in different nodes without conflict.





- Get WriteAttributes.java from Oracle Configurator Methodologies Guide.
- Modify the class name to WriteAttributesCX from WriteAttributes.
- Do not extent the class WriteAttributesCX from AutoFunctionalCompanion.
- Modify afterSave method to add a parameter rootNode.
- Compile the WriteAttributesCX.java











Create Configurator Extension Archive

Configurator Developer ×
← → C (③ r1211.cerebraebs.com:8000/OA_HTML/OA.jsp?page=/oracle/apps/cz/developer/repository/main/web
Repository Workbench
Main   Item Master   Publications
Repository, main < Configurator Extension Archive, wheekindules Archive <
Description WriteAttributes Archive Archive File wa.zip Expand All Collapse All
WriteAttributes Archive WriteAttributesCX\$Attribute WriteAttributesCX
Return to Archive Details
Repository Workbench Home Logout Preferences H Oracle Configurator Developer Version 12.1.3.31.3 (Schema 31a), VIS2, Oracle Configurator Administrator Privacy Statement





#### Create Configurator Extension Rule

General	Structure   Rules   User Inter	face
Workbench	: Rules >	
Effectivity		
Create Rul	le Entities: Define Configurator Ext	tension Rule
Model <b>'Re</b> * Indicates	pository Main'/'Cerebra User Fol required field	ders'/'Siva Pola'/SP_TEST_MODEL
	* Name	Write Attributes CX Rule
	Path	'SP_TEST_MODEL Rules'/'Write Attributes CX Rule'
	Description	Write Attributes CX Rule
		Disable
Definition	n	Disable
Definition Associate a	n model node with a Java class in an A Model Node*SP TEST MODEL	Choose Node
Definition	n model node with a Java class in an A Model Node*SP_TEST_MODEL	Disable  rchive, by binding method signatures to configuration events.  Choose Node  Choose Configuration configuration events.
<b>Definition</b> Associate a	n model node with a Java class in an A Model Node* <b>SP_TEST_MODEL</b> Java Class <sup>®</sup> WriteAttributesCX	Disable  archive, by binding method signatures to configuration events.  Choose Node  Choose C
<b>Definitio</b> Associate a	n model node with a Java class in an A Model Node* <b>SP_TEST_MODEL</b> Java Class* WriteAttributesCX Java Class <sub>*</sub> With Model Node Instantiation	Disable  archive, by binding method signatures to configuration events.  Choose Node  Choose C  Instance
Definition Associate a Event Bind	n model node with a Java class in an A Model Node* <b>SP_TEST_MODEL</b> Java Class* WriteAttributesCX Java Class <sub>*</sub> With Model Node Instantiation	Disable  archive, by binding method signatures to configuration events. Choose Node  Choose C  Instance
Definition Associate a Event Bind	n model node with a Java class in an A Model Node* <b>SP_TEST_MODEL</b> Java Class* WriteAttributesCX Java Class <sub>*</sub> With Model Node Instantiation	Disable  rchive, by binding method signatures to configuration events. Choose Node  Choose C  Instance  Create Binding
Definition Associate a Event Bind Event	n model node with a Java class in an A Model Node <sup>*</sup> SP_TEST_MODEL Java Class <sup>*</sup> WriteAttributesCX Java Class <sub>*</sub> With Model Node Instantiation Scope Method	Disable   archive, by binding method signatures to configuration events.   Choose Node   Choose C     Instance •     Create Binding   Edit Delete





#### **Create Event Binding**







aenerdi	Structure   Rules   User Interfa	ace	
Workbench	n: Rules >		
) Effectivity	Notes		
Create Ru	le Entities: Define Configurator Exte	ension Rule	
Model <b>'Re</b> * Indicates	pository Main'/'Cerebra User Foldes required field	ers'/'Siva Pola'/sp_fce_model	
	* Name	Write Attributes for FCE CX Rule	
	Path	'sp_fce_model Rules'/'Write Attributes for FCE CX Rule'	
	Description	Write Attributes for FCE CX Rule	
		Disable	
Definition	a		
Associate a	model node with a Java class in an Ar	chive by hinding method signatures to configuration events	
looverate a	Model Node*sp fce model C	Choose Node	
	Java Class* oracle.apps.cz.cx.C	ZAttributeCX Choose Class	
	Java Class <sub>*</sub> With Model Node I Instantiation	Instance 👻	
Event Bind	dinas		
		Create Binding	
Event	Scope Method	Edit Delete	

#### **Create CX Rule**





#### **Create Event Binding**

	🔒 Navigate
Repository Workbench	
Seneral   Structure   Rules   User Interface	
Norkbench: Rules > Create Rule Entities: Define Configurator Extension Rule > Add Event Binding: Select Event and	Binding >
Add Event Binding: Bind Method Arguments	

Configurator Extension Write Attributes for FCE CX Rule Model Node sp\_fce\_model Java Class oracle.apps.cz.cx.CZAttributeCX Event postConfigInit Command Name Event Scope Global Method Name postConfigInit

#### Argument Bindings

Argument Type Argument Name		Argument Specification	Binding	Select Node or Property		
oracle.apps.cz.core.IRuntimeNode	Arg1	System parameter 🔹	<basenodeofrule> 👻</basenodeofrule>	÷		





#### CZ\_CONFIG\_ATTRIBUTES

#### Reading Data from the CZ\_CONFIG\_ATTRIBUTES Table

#### CZ\_CONFIG\_ATTRIBUTES Table

#### CONFIG CONFIG ATTRIBUTE CONFIG HDR ID REV NBR ITEM ID PS NODE NAME CATEGORY ATTRIBUTE1 ATTRIBUTE2 ATTRIBUTE3 ----------4175522 1 1923850 BIKE SHOP Internal 4175522 1 1923850 BIKE SHOP Custom 0.0 FALSE 4175522 1 1923850 BIKE SHOP Standard 20 Alloy 1 1923851 BIKE FRAME 4175522 Standard Red 4175522 1 1923852 FRAME TYPE Standard 18 Runtime value for: Attribute Feature: Frame Size Where: Flexfield Context: Standard Flexfield Segment: Size

#### Down Stream Applications can query the CZ\_CONFIG\_ATTRIBUTES table





#### **Special Considerations for Attribute Design**

- Referenced Models
- Location of attribute features
- Multiple Component Instances in the Node Path
- Reusing attribute value for multiple items
- Required Items
- Effects of auto complete and adjust mode.





## **Fusion Configurator Engine Key Features**

- Rules can now be classified as constraint, defaults, or search decision.
- You can specify the order in which rules evaluate if they are classified as defaults or search decisions.
- The absolute and relative quantity of a BOM node can now be controlled by defining constraints.
- Statement rules can now be used to control the number of instances of a multi instantiable model component.
- Accumulator rules to replace old numeric rules.
- New ways to handle multiple components using copy, remove and reuse from a pool.
- A new feature called Auto Complete Configuration that allows for a model to complete automatically with just a few user inputs.





### Upgrade to R12.1 Fusion Configurator Engine

- OCE Vs FCE
- Upgrade Vs Reimplemenation
- Rewriting CX with core API vs Use CIO Emulation





### **Upgrade to R12.1FCE Steps**

- Set the Profile Options for FCE
- Run Model Conversion Utility
- Migrate Configurator Extensions and CX Rules





### **R12.1 FCE Configurator Profile Options**

- CZ: Configurator Engine for New Models: Fusion
- CZ: Default Max Quantity Decimal : Fusion or Both
- CZ: Default Max Quantity Integer
- CZ: Enable Configurator Engine
- CZ: Use BOM Default Quantity as Domain





**Repository** - Main

- Select the Model and pick 'Convert Model to FCE' and click Go Button.
- Click on Convert Selected Models on next screen

Views				
View Defa	ult 🔻 Go Personalize			
Select Obje	ects: Actions Convert Model(s) to use FCE  Go			
Select All	Select None Expand All Collapse All			
÷				
Repository	/ Main > Cerebra User Folders >			
Select Focu	is Name	Edit	Create	Description
	CA Demo	1	1	CA Demo
	⊗ Previous			
	WriteAttributes Archive	1		WriteAttributes Archive
	sp_fce_model	I		sp_fce_model
	SP_TEST_MODEL	1		SP_TEST_MODEL
	⊗Next			





- Note Down the conversion set id.
- Run the concurrent program 'Process a Single Model Conversion' for the conversion set id.

ORA	CLE.	Oracle Configurator Administrator	
Repository	Workbench		
Main   Iter	m Master   P	Publications	

#### Information

Created a conversion set with ID 1023. You must now run the model conversion concurrent process. Please review the output of the concurrent process for important messages about the conversion.





• Check Model Conversion Report - Conversion Set 1024.

ORACLE <sup>®</sup> Concurrent Programs	🛱 Navigator 🔻 👒 Fa	vorites 🔻	Diagnostics Home	Logout F	Preferences	s Help
Requests View Last 24 hours  GO Requests Summary Table				Searc	h Subm	iit Request
Refresh						
Request ID Name	Phase	Status	Scheduled Date	Details	Output	Republish
5819450 Model Conversion Report - Conversion Set 1024 (Model Conversion XML Publisher Concurrent Program)	Complete	d Normal	01-Mar-2012 14:48:35		æ	Pa-
5819449 Process a Single Model Conversion	Complete	d Normal	01-Mar-2012 14:48:13			
5819448 Process a Single Model Conversion	Complete	d Error	01-Mar-2012 14:41:03			

		Diagnostics Home Logout Preferences Help	
About this Page	Privacy Statement	Copyright (c) 2006, Orade. All rights rese	rved





• Check that a new model with [FCE] suffix is created .

lect Focus Name			Create	Description
	CA Demo	1	<b>*</b>	CA Demo
	⊘ Previous			
	WriteAttributes Archive	1		WriteAttributes Archive
	sp_fce_model	1		sp_fce_model
	SP_TEST_MODEL	1		SP_TEST_MODEL
	SP_TEST_MODEL-[FCE]	Ì		SP_TEST_MODEL
	⊗Next			





### **Step 3 : Configurator Extensions and Rules**

- The OCE Configurator Extensions use CIO API whereas FCE Fusion Configurator Engine uses *core* API.
- Re writing the configurator extensions using core API Vs Modify configurator extension to use CIO *Emulation*.
- Use a perl script provided by *Oracle* to modify the source code of configurator extension.
- Compile the modified code and resolve any errors. Review the code and make changes as necessary.
- Create a configurator extension archive with modified code and attach it to the model and
- Use Oracle Configurator Developer to convert existing Configurator Extension Rules that use the converted classes





## Specific Considerations for Configurator Attributes Update

- Set FCE Profile Options
- Run Model conversion Utility
- Delete the old Write Attributes
   CX Rule
- Create a new CX Rule as shown

General   Structure   Rules   User Interface				
Workbench: Rules >				
Effectivity     O Notes				
Create Rule Entities: Define Configurator Extension Rule				
Model 'Repository Main'/'Cerebra User Folders'/'Siva Pola'/sp_fce_model * Indicates required field				
* Name Write Attributes for FCE CX Rule				
Path 'sp_fce_model Rules'/'Write Attributes for FCE CX Rule'	'sp_fce_model Rules'/'Write Attributes for FCE CX Rule'			
Description Write Attributes for FCE CX Rule	A.			
Disable				
Definition				
Associate a model node with a Java class in an Archive, by binding method signatures to configuration events.           Model Node*sp_fce_model         Choose Node				
Java Class <sup>®</sup> oracle.apps.cz.cx.CZAttributeCX Choose Class				
Java Class <sub>∞</sub> With Model Node Instance ▼				
Event Bindings				
Create Binding				
Event Scope Method Edit Delete				
postConfigInit Global postConfigInit(oracle.apps.cz.core.IRuntimeNode Arg1) 🥜 🍵				
Pin Bank				

Argument Bindings										
Argument Type	Argument Name	Argument Specification	Binding	Select Node or Property						
oracle.apps.cz.core.IRuntimeNode	Arg1	System parameter 🔹	<basenodeofrule> -</basenodeofrule>	4						





#### **Questions ???**