ADF Code Corner

89. How-to conditionally switch model driven LOV in ADF forms and tables



Oracle ADF Code Corner is a loose blog-style series of how-to documents that provide solutions to real world coding problems.

Disclaimer: All samples are provided as is with no guarantee for future upgrades or error correction. No support can be given through Oracle customer support.

Please post questions or report problems related to the samples in this series on the OTN forum for Oracle JDeveloper: <u>http://forums.oracle.com/forums/forum.jspa?forumID=83</u>

Introduction

Some use cases require a list of values to show different list data based on an external condition. In the example created for this article, the condition is the value of the *PaymentId* attribute, which, if set to 1, means a customer pays cash, and, if set to 2, that a customer pays with credit card.

Dependent on the type of payment chosen for an order, the list of values shows a list of accepted currencies or a list of accepted credit cards to choose from. The database schema used in this sample is **ADF Summit**, which you can download from here:

http://download.oracle.com/otn_hosted_doc/jdeveloper/11gdemos/SummitADF/SummitADFV1_0_08072011.zip

The ADF Summit schema can be installed on the Oracle XE and enterprise database.

The image below shows an order table with a list-of-values defined for the **PaymentType** column. The LOV data, as explained before, depends on the value of the **PaymentId** column. This means that the data – currencies or credit cards – displayed in the LOV changes on a row-by-row basis.

Form	Table			
d	CustomerId	PaymentT	Search and Select:	PaymentDetail
97	201	2	≥ Search	Advance
98	202	1	Match 🔘 All 🦳 An	
99	203	2	Id	
100	204	2	CardName	
101	205	2		
102	206	1		Search) Res
103	208	1	Id	CardName
104	208	2	AMEX	American Express
105	209	2	VISA	VISA
106	210	2	BIFUE	Carte Bleue
107	211	2	MASTER	Master Card International
108	212	2	JCB	Japan Credit Bureau
100	213	2		
109	213	1		
110	214	1		
	204	1		
112	210	2		
113	202	1		
114	202	1		
115	202	1		
116	202	1	•	III
117	202	1		
118	206	1		OK Car
				(dd)

ADF CODE CORNERHow-to conditionally switch model driven LOV in ADFforms and tables

The image above shows the list-of-values for a **PaymentId** value of 2 (credit card). The image below shows the same LOV for a PaymentId value of 1 (cash). Instead of showing the payment id values, I could have used a single select one choice, which I did not for clarity reasons.

d	CustomerId	PaymentTypeId	PaymentDetail	DateOrdered		DateShipped		SalesRepId	Total	OrderFilled	
97	201	2	Q	6/12/2011	陸	6/14/2011	- B	12	84000	Y	
8	202	1	Search an	nd Select: Payı	ment	Detail					
99	203	2									
100	204	2	⊻ Searc	n						Ag	Ivance
101	205	2	Match (All O Any							
102	206	1	Isc								
103	208	1	Territory	/							
04	208	2	Currency	/							
05	209	2								Search	Res
06	210	2	The second secon			Tamitan					J
07	211	2	FURO			Furopean Unic	ND.				
00	211	2	ARS			Argentina	///				
08	212	2	AUD			Australia					
09	213	2	RUB			Russia					
10	214	1	GBP			Great Brittain					
.11	204	1	INR			India					
12	210	2	BRL			Brazilia United States					
13	202	1	CAD			Cannada					
14	20.2	1	CNY			China					
	202	-	DKK			Denmark					
.15	202	1	HKD			Hong Kong					
16	202	1	JPY			Japan					
17	202	1	SGD			Singapore					
18	206	1	ZAR			South Africa					
19	206	1	CHF			Switzerland					
20	205	1		III	_						
20	200	1									

In the following, this article explains how the use case of varying data lists can be implemented declaratively using model driven list-of-values, which I think is a great proof point for how ADF simplifies web development, as the same use case would require many lines of code to be written in other programming languages or frameworks.

The Oracle JDeveloper 11.1.1.4 example workspace used in this article can be downloaded as sample 089 from the ADF Code Corner website:

http://www.oracle.com/technetwork/developer-tools/adf/learnmore/index-101235.html

Database

The default **ADF Summit database schema** needs to be patched for this used article, for two tables, *S_CREDIT_CARDS* and *S_CURRENCIES* to be added. The *S_ORD* table that holds orders also needs to be enhanced with an additional *PAYMENT_DETAIL* column.

The sample download contains a SQL script that you can run against the **SUMMIT_ADF** schema to create these changes. The script is in the **SUMMIT_ADF_SCHEMA-PATCH** folder of the sample workspace.

ADF CO<u>DE CORNER</u>

Connections × Reports ×	S_ORD × adfSummit ×	
🕂 🕅 🏹	olumns Data Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL	
Connectors adSummit adSummit adSummit adSummit Source addition Source addition	Johnne Data Constraints Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL. Image: Source Discourse Disc	
ter-time Packages		

The image below shows the ADF Summit schema with the patch script installed.



The ADF Summit *S_PAYMENT_TYPE* table knows of two values, CASH and CREDIT for which we need to create list of values for an ADF Business Components view object.



ADF BC Model

As shown in the image below, an ADF Business Components model needs to be created for the three tables *S_ORD*, *S_CURRENCIES* and *S_CREDIT_CARDS*. Only the *S_ORD* view object is later exposed to application developers. The other two entity objects and view objects are used internally to populate the LOV.

🚖 Create Business Compor	nents from Tables - Step 1 of 6
Entity Objects	
Entity Objects Updatable View Objects Read-Only View Objects Application Module Diagram Summary	Specify the package to contain your new entity objects and associations. Package: [adf.sample.model Prowse Package: [adf.sample.model Prowse Priter the types of schema objects to display as available, then select the schema object(s) and dick '>' to create entity objects. Schema: [SuMMIT_ADF Type Filter: OFF Filter Types Schema: [SuMMIT_ADF Schema: [SuMMIT_ADF Schema: [Schema: [Sch
< >>>	Entity Name:
Help	< Back Next > Einish Cancel

The ADF Business Component model is built choosing **New | Business Tier | ADF Business Components | Business Components from Tables** from the context menu that opened by a right mouse click onto the Model project for an application that was created using the Fusion Web Application (ADF) template. In the opened wizard, a database connection needs to be defined to access the ADF Summit schema - summit_adf.

After creating the entities, which is shown in the image above, view objects are created for each entity. The application module then is edited to only expose the *S_ORD* view object. The view object instance in the example also was renamed to **AllSummitOrders**.

SOrdView.xml	AppModule.xml		6
		(٥
Seneral Data Model	Data Model Components		
ava 38 Session Bean	Select a view object from the tree of available view objects, select the instance or application module to be its parent in the data model tree, and click >' to create a named instance of object in the data model.	the view	
ervice Interface			
onfigurations	The data model contains a list of view object and view link instance. disnlaring master datal relationships		
	Available View Objects: Data Model: Subtyper	s Edit	
	영화 af sample-model Model 응 · · · · · · · · · · · · · · · · · · ·		
	New View Instance: SOrdView1 View Instance: AllSummitOrders	1	
	New View Link Instance: View Link Instance:	1	
	Vew Definition: <u>adf.sample.model.vo.SDrdVew</u> Vew Link Definition:	1.	
	Application Module Instances		
verview Source H	History <	2	5

To create the model driven LOVs, the **SOrdView** view object is opened in the ADF Business Components editor, which is done by a double click on the view object definition in the JDeveloper Application Navigator.

SOrdView.xml							(Ŧ
							?	^
General Entity Objects	Attributes					Override S	et Source Order	
Attributes	View object attribute	es can be mappe	d to entity attributes, ca	alculated or SQL-deriv	ed.	🔶 Set Source Order		
Query Java View Accessors	Name			.	Ð) Optionally control the order in which the attributes appear in the associated XML and, if applicable, Java source.	🕂 - 🖉 🗙	
List UI Hints	Name	Type	Alias Name	Entity Usage	In	Attributes:	-	
	🖙 Id	Number	ID	SOrd		Id(SOrd:ID)		
	CustomerId	Number	CUSTOMER_ID	SOrd		DateOrdered(SOrd:DATE_ORDEDED)		
	DateOrdered	Date	DATE_ORDERED	SOrd		DateShipped(SOrd:DATE_SHIPPED)		
	DateShipped	Date	DATE_SHIPPED	SOrd		SalesRepId(SOrd:SALES REP ID)		
	SalesRepId	Number	SALES_REP_ID	SOrd		Total(SOrd:TOTAL)		
	Total	Number	TOTAL	SOrd		PaymentTypeId(SOrd:PAYMENT_TYPE_ID)		
	PaymentTypeId	Number	PAYMENT_TYPE_ID	SOrd		PaymentDetail(SOrd:PAYMENT_DETAIL)	election Up (Alt U	h
	OrderFilled	String	ORDER_FILLED	SOrd		OrderFilled(SOrd:ORDER_FILLED)	election op (Alt-0)	1
	PaymentDetail	String	PAYMENT_DETAIL	SOrd				
	Custom Propert List of Values: P	ties: Payment PaymentDetai	Detail			Help OK Cancel	+ / ×	

The image above shows how the order attributes in a generated view object can be changed using the **Set Source Order** button.

In the example, reordering is used to show the **PaymentTypeId** and the **PaymentDetail** columns next to each other.

To create the list-of-values, the **PaymentDetail** attribute is selected and the green plus icon that is next to the **List of Values: PaymentDetail** label pressed. To define the model driven list the green plus icon next to **List Data Source** is pressed in the **Create List of Values** dialog.

	lues
ist of Values Name:	CASH
Configuration	IIT Hints
object a view acce	essor for the list data source, and then choose the list attribute that maps to the current view
List Data Source:	<none specified=""></none>
List <u>A</u> ttribute:	
List Return Values	
Map any suppleme	ental values that your list returns to the base view object (it always returns a value to the
attribute for which	n the list is defined).
🕂 🗙	
View Attribute	List Attribute

In the list of available view objects the **SCurrenciesView** is selected and moved to the list of selected **View Accessors**. This is how model driven list-of-values are created.

6	Create List	st of Values	23
	List of Values	Name: CASH	
	Configura	🥹 View Accessors	
	Select a v object at	Select a view object or shared view instance and shuttle it to the selected list to reate a view accessor.	
	List <u>D</u> ata		
	List <u>A</u> ttrib	Available view Objects: view Accessors: Cott Overnde So adf.sample.model.Model So crdView	
	List Retur	G - @ adf.sample.model.vo	-
	Map any : attribute	ScurrenciesView	
	+ ×		
	View Attr	<	
		Name: SCurrenciesView2 Accessor Name: SCurrenciesView1 Definition: adf sample model v	
		Help OK b Capel	
	Help	OK Cancel	

Next, as shown in the image below, the **PaymentDetail** attribute is mapped to the **Iso** attribute of the currencies view object. This ensures the currency ISO value of the selected list value is copied to the **PaymentDetail** attribute at runtime.

👈 Create List of Va	lues
List of Values <u>N</u> ame:	CASH
Configuration	JI Hints
Select a view acce object attribute.	ssor for the list data source, and then choose the list attribute that maps to the current view
List Data Source:	SCurrenciesView1
List <u>A</u> ttribute:	Iso
Map any suppleme attribute for which	intal values that your list returns to the base view object (it always returns a value to the the list is defined).
View Attribute	List Attribute
PaymentDetail	lso
Help	OK Cancel

ADF CODE CORNERHow-to conditionally switch model driven LOV in ADFforms and tables

It is important to note that the **List of Values Name** is chosen to be **CASH** to match with the payment type name for payment type Id 1. This way we don't need to further transform the payment type value to the name of a matching LOV.

SOrdView.xml					
General					
Entity Objects	Attributes		_	🖕 Create List of Values	x
Attributes	View object attribute	s can be mappe	d to entity		
Query	(AB) 11		-	List of Values Name: CASH	
Java			-	Configuration UI Hints	
View Accessors					
List UI Hints	Name	Type	Alias Ni	Default List Type: Input Text with List of Values	-
	id and its and	Number	ID	Display Attributes	
	Customerid	Number	CUSTON DATE O	Select display attributes for the list of values and combo box. Optionally show a subset in the combo box	
	DateOrdered	Date	DATE O	(multiple values are separated by white space).	
	SalesRenId	Number	SALES P	Available: Selected:	
	Total	Number	TOTAL	Currency	
	PaymentTypeId	Number	PAYMEN	150 Territory	T
	PaymentDetail	String	PAYMEN		I
	OrderFilled	String	ORDER		
				Show in Combo Box: All 👻 🛛 🗣	
				List Sparsh	
	Custom Proper	ties: Payment	Detail	Table Scott Parises all provide and the	
			- 1	indude search Region: All Queryable Attributes	1
	List of Values: P	aymentDetai	- I	Query List Automatically	
	Enable this attribute	to display a list	of values t	Choice List Options	
				✓ Query Limit: 10	
				Most Recently Lised Count:	
				Eliter Combo Box Using: <pre></pre> <pre></pre> <pre></pre> <pre></pre> <pre></pre> <pre></pre> <pre></pre>	
Overview Source	History <			Include "No Selection" Item: Blank Item (First of List)	
	[1.1.500.7]				
Imessages - Log				Help OK Can	icel

Next, a second LOV needs to be created for the same attribute – *PaymentDetail* – but this time for the CREDIT payment type.

	b Create Lis	t of Values	23
	List of Values	Name: CREDIT	
	Configura	View Accessors	
	Select a v object att	Select a view object or shared view instance and shuttle it to the selected list to create a view accessor.	view
	List <u>D</u> ata	Ausibhle View Objecter	-
	List <u>A</u> ttrib	Available view Objects. View Accessors: Edit Overhoe	
F	List Retur	⊡	
	Map any s attribute	SurrenciesView	e
	+ ×		
	View Attr	٩	
		Name: SCreditCardsView2 Accegsor Name: SCreditCardsView1	
		Definition: adf.sample.model.v	
		Help OK Cancel	
	Help	OK Car	ncel

In both cases, the LOV is built as **Input Text with List of Values**. Note that you cannot have two different list types.

💩 Create List of Val	lues
List of Values <u>N</u> ame:	CREDIT
Configuration	JI Hinte
Select a view acce object attribute.	ssor for the list data source, and then choose the list attribute that maps to the current view
List <u>D</u> ata Source:	SCreditCardsView1
List <u>A</u> ttribute:	Id
List Return Values Map any suppleme attribute for which	ntal values that your list returns to the base view object (it always returns a value to the it he list is defined).
View Attribute	List Attribute
PaymentDetail	Id
Help	OK Cancel

The **List of Values Name** is set to **CREDIT** and the **PaymentDetail** attribute is mapped to the Id value of the **CreditCard** view object. This time, at runtime, when selecting a value of the LOV, the payment detail attribute is updated with the abbreviation of a credit card.

🆕 Create List of Values		×
List of Values Name: CREDIT		
Configuration UI Hints		
Default List Type: Input Text w	ith List of Values	
Display Attributes		
Select display attributes for the l	ist of values and combo box. Opti	onally show a subset in the combo box
(multiple values are separated b) Available:	y white space). Selected	l:
	CardNar Id	me
	Show in	
List Search		
Include Search Region: All Que	ryable Attributes	•
Query List Automatically		
Choice List Options		
Query Limit:		0
Most Recently Used Count:		0
Eilter Combo Box Using:	< No View Critieria Defined >	- /
☑ Include "No Selection" Item:	Blank Item (First of List)	· · · · · · · · · · · · · · · · · · ·
Help		OK Cancel

The *S_ORD* table only has a **FK reference** to the *S_PAYMENT_TYPE* table *ID* column. To switch between the different LOV data lists, we need a discriminator column to return a value CASH or

CREDIT. This can be achieved easily with a transient attribute in the View Object as shown in the image below.

SOrdView.xml						Resour	rce Paletti
						2 🗖 🖪 I 📌	· 📴 1 🍙
Attributes	View object attribute	s can be mappe	d to entity attributes, ca	lculated or SQL-deri	ved.		
Query	(**					Түрс	
Java	(👸 Name			4	1)	Name	*:
View Accessors						🖶 - 🦯 🗙 👘 Display	y Name:
List UI Hints	Name	Туре	Alias Name	Entity Usage	Info	New A <mark>t</mark> ribute	
	🕬 Id	Number	ID	SOrd		Add At Shute from Entity	
	CustomerId	Number	CUSTOMER_ID	SOrd		Create new attribu	te
	DateOrdered	Date	DATE_ORDERED	SOrd		Proper	ty Set:
	DateShipped	Date	DATE_SHIPPED	SOrd		Value	Serializer (
	SalesRepId	Number	SALES_REP_ID	SOrd		Dis	criminator
	Total	Number	TOTAL	SOrd		Su	ubtype Va
	PaymentTypeId	Number	PAYMENT_TYPE_ID	SOrd		··· Date E	ffectivity
	PaymentDetail	String	PAYMENT_DETAIL	SOrd		Start D	ate: fak
	OrderFiled	String	ORDER_FILLED	SOrd		5.40	in Cit
						End Da	ite: mais

The transient attribute is named **PaymentTypeIdTranslated** and uses a Groovy expression to return a value of "CASH" for PaymentTypeId 1 and "CREDIT" for PaymentTypeId 2.

Note that the view attribute's **Value Type** property must be set to **Expression** for Groovy strings to be executed at runtime.

Attribute			_	-Updatable -
Name:	PaymentTypeIdTranslated			
Type:	String	 <u>B</u>rowse 		
Property Set:	<none></none>		-	
Value Type:	◯ Literal () Expression			
<u>V</u> alue:	PaymentTypeId == 1? 'CASH'	: 'CREDIT'	Edit	Always
				O While Ne
Mapped	to <u>C</u> olumn or SQL	Key Attribute		○ Ne <u>v</u> er
Discrimin		V Queryable		
Default \	/alue:	Start O End		
Passivat	ie			
Query Column-				
	VIEW ATTR	Type: VARCHAR2(255)		
Expression:				

What if you need this expression to be more generic - feeling groovy ?

The image above has the value strings CASH and CREDIT hard-coded, which is not optimal. To define the Groovy string more generic, we want to read the translated payment type from a method exposed on a view object created for the S_PAYMENT_TYPE table in the ADF Summit schema.

Note: If you are okay with the hard-coded values, then you may skip the following section and continue with "What we've done so far " on page 12.

The **SPaymentTypeView** view object is also created using the **ADF Business Components from Table** entry in the Oracle JDeveloper **New Gallery**.

The **SPaymentTypeView** view object is opened in the ADF Business Components editor by a double click on its entry in the Application Navigator. A Java implementation class is created for the view object by selecting the Java menu option in the view object editor and pressing the **pencil** icon next to the **Java Classes** label.

The dialog shown in the image below is displayed to create the SPaymentTypeViewImpl class.

✓ Projects	■ ₩ 7 = № -	General
B	Select Java Options	
B		
SCreditCardsView	Optionally generate Java classes to ad	dd custom behavior.
€ SCurrenciesView	Generate View Object Class: SPay	/mentTypeViewImpl
i SOrdView		
SPaymentTypeView		
		urce methods
Application Sources	Generate View Row Class	
adf.sample.view	✓ Include accessors	
migrageDefs	Expose accessors to client	
DataBindings.cox	Generate View Object Definition Cla	226
Application Resources		Channe Extrand
▽ Data Controls		Classes Extend
AppModuleDataControl	Service Data Object	
allOrders	Generate Service Data Object Clas	IS
allPayment i ypes Operations	Name: SPaymentTypeVi	ewSDO
	Namespace: /adf/sample/mod	lel/vo/common/
	Support warnings	
	Help	OK Cancel

In the view object implementation class, a single public method is defined that, when invoked, queries the **SPaymentTypeView** RowSet for the **PaymentType** name matching the ID passed as an argument.

🗢 Projects 💽 🖓 🖓 🕶 🔚 🕶	1 package adf.sample.model.vo;
ia@ adf.sample.model	^ 2
	3 B import;
🗄 📲 AppModule	6 ///
	7 // File generated by Oracle ADF Business Components Design Time.
	8 // Tue Aug 09 09:09:40 CEST 2011
🗈 😁 SCreditCardsView	9 // Custom code may be added to this class.
SCurrenciesView	10 // Warning: Do not modify method signatures of generated methods.
😥 🚵 SOrdView	11 //
SPaymentTypeView	👕 🕆 🖃 public class SPaymentTypeViewImpl extends ViewObjectImpl {
SPaymentTypeView.xml	13 🖃 /**
SPaymentTypeViewImpl.java	14 * This is the default constructor (do not remove).
- Model.jpx	15 */
🖻 🗝 🛅 ViewController	16 🖃 public SPaymentTypeViewImpl() {
Application Sources	✓ 17 }
Application Resources	18
▼ Data Controls 🕅 🍸	19 //query type name from ID
E	20 🖃 public String getPaymentTypeTranslated(Long id)(
🗉 📒 allOrders	<pre>21 this.getRowSet().first();</pre>
🕀 📃 allPaymentTypes	22 Row rw = null;
😟 🛅 Operations	<pre>23 rw = this.getRow(new Key(new Object[](id)));</pre>
	<pre>24 return (String) rw.getAttribute("PaymentType");</pre>
	25
	26]
	27

The **PaymentTypeIdTranslated** value expression (Groovy) is then changed to query the public method on the "allPaymentTypes" View Object, which is the instance of the **SPaymentTypeView** view object configured in the Data Model of the Application Module.



```
//query type name from ID
public String getPaymentTypeTranslated(Long id){
    this.getRowSet().first();
    Row rw = null;
    rw = this.getRow(new Key(new Object[]{id}));
    return (String) rw.getAttribute("PaymentType");
}
```

What we've done so far:

- Create a View Object for the S_ORD table entity
- Create two LOV for the PaymentDetail attribute
- Create a transient attribute to hold the values CASH, CREDIT based on the *PaymentTypeId* attribute value
- Create a public method to translate the PaymentTypeId to its type name. A groovy string is used to update the *PaymentTypeIdTranslated* transient attribute

PaymentTypeId	Number	PAYMENT_TYPE_ID	SOrd		
PaymentDetail	String	PAYMENT_DETAIL	SOrd		
OrderFilled	String	ORDER_FILLED	SOrd		
PaymentTypeIdTransla	String	VIEW_ATTR		Transient	
Custom Properties:	Pavmenti	Detail			
List of values: Paym	entDetail				
Enable this attribute to dis	play a list o	of values to use in the u	ser interface.		
List Type UI Hint:	Input Te	ext with List of Values			
List of Values Switcher:	Payment	TypeIdTranslated			
	Custome	rId			
	DateOrd	ered			
Lists of Values	DateShip	ped			
Default	Id				
Default	OrderFill	ed			
۲	Payment	Detail			
0	Payment	TypeId			
	Payment	TypeIdTranslated		N	
				1	

In a last step, the **PaymentTypeIdTranslated** attribute is configured as the value of the **List of Values Switcher** property. This defines the **PaymentTypeIdTranslated** as the discriminator that – at runtime – decides which list data is displayed in the LOV.

	_		c	ملد
Model.jpx		Open AppModu <u>l</u> e		
wController		Ref <u>a</u> ctor	•	
	×	<u>D</u> elete		
		<u>F</u> ind Usages		
		Configurations		
		Business Components Deploymen		_
	\triangleright	Run	Ctrl-F11	
	۲	De <u>b</u> ug K		

To test the LOV switcher, select the **AppModule** object in the Application Navigator and choose **Run** from the context menu.

🛃 Oracle Business Com	nponent Browser (AppModuleLocal-Local)	
<u>File View Create Data</u>	abase <u>H</u> elp	
62		
AppModule AllSummitOrders	AllSummitOrders	. –
	K « » » + × • • • •	V 9
	Id	98
	CustomerId	202
	DateOrdered	2011-06-17 16:26:13.0
	DateShipped	2011-06-27 16:26:13.0
	SalesRepId	14
	Total	595
	PaymentTypeId	1
	PaymentDetail	
	OrderFilled	Y
	PaymentTypeIdTranslated	CASH
Name:AppModule.AllSumm	itOrders Definition:adf.sample.model.vo.SOrdVi	ew

In the ADF BC tester, you can now test the LOV, which, dependent on the **PaymentTypeId** attribute value, shows credit card or currencies values. The list-of-values data also changes when you change the PaymentTypeId from a value of 1 to 2.

ADF Faces View

We did the obligatory short program, now it is time for free skating.

The real work has been done in the model already and the ADF Faces UI defined next is only needed to show how to display the LOV switcher in a web environment.

Application Run Ma	IstOfValuesSwitcher.jspx
🕞 AdfBcListOfValuesSwitch 🔹 🕞 🔹	🔞 🔹 Show 👻 Full Screen Size 🕶 🙆 None 🔹 Default
Y Projects Nodel WewController Web 2:NF Web 2:NF Page Flows UistOfValuesSwitcher.;spx	Form Table
Application Resources	Create ⊕: Carousel Form → ADF Form Gantt → ADF Regd-only Form
▼ Data Controls ● ● MailsummiOnders ● ● 303 Customerid ● ● 303 DateOrdered ● ● 303 DateShipped ● ● 303 Id ● ● 303 PaymentTypeId ● ● 303 PaymentTypeId ● ● 303 SalesRepId ●	Gauge I ADF Search Form Geographic Map Graph Multiple Selection > Navigation > I able Trge > Cancel

To create an ADF form, we drag the *AllSummitOrders* collection onto the JSF page and choose **ADF Form** from the context menu.

The **PaymentTypeIdTranslated** attribute is internal detail information and not needed to display in the form. It can be deleted from the view as shown in the image below.

Configure the components that y components after you click OK.	you want to display in your form. No You can also add more components d	te that you can remove or edit the res irectly to the layout later.	ulting
Fields:		4	×
Display Label	Value Binding	Component To Use	Delet
xxa <default></default>	📼 Id	DF Input Text w/Label	Delet
💴 <default></default>	CustomerId	🛱 ADF Input Text w/ Label	
xx∎ <default></default>	DateOrdered	🔲 ADF Input Date w/ Label	
💴 <default></default>	DateShipped	🖽 ADF Input Date w/ Label	
💴 <default></default>	SalesRepId	🖆 ADF Input Text w/ Label	-
xx■ <default></default>	📼 Total	撞 ADF Input Text w/ Label	- C
💴 <default></default>	PaymentTypeId	🖆 ADF Input Text w/ Label	1
আ <default></default>	📼 PaymentDetail	ADF List of Values Input	
💴 <default></default>	OrderFilled	🖆 ADF Input Text w/ Label	<
xxa <default></default>	PaymentTypeIdTranslated	🚽 🛑 ADF Input Text w/ Label	4
Include Ivavigation Controls			
Include Submit Button			
Include <u>o</u> donie odetoni			

How-to conditionally switch model driven LOV in ADFADF CODE CORNERforms and tables

The **PaymentTypeId** field in the form needs to have its **AutoSubmit** property set to **true**. This ensures that a change of the payment Id type is immediately saved in the model so that the list-of-values data list can switch accordingly.

ld	#{ld.inputValue}				
Customerid	#{Customerld.inputValue}				
DateOrdered	#{DateOrdered.inputValue}	120			
DateShipped	#{DateShipped.inputValue}	20	🛱 Input Text - #{	bindings.PaymentTypeId.hints.label}	- Pro
SalesRepld	#{SalesRepld.inputValue}		🖳 I 📌 🖹 I 🥒	🛃 🍘 Find	↓ ĵ
Total	#{Total.inputValue}		Horizontal Align:		• ~
mentTypeld	#{PaymentTypeld.inputValue}		Decoration:		• •
ymentDetail	#{PaymentDetail.inputValue}	Q	Vertical Align:	▼ %	• ~
OrderFilled	#{OrderFilled.inputValue}				
First Pre	evious Next Last				
			Behavior		
			Required:	#{bindings.PaymentTypeId.hints.mandate	ory} ~
			Benavior Required: ReadOnly:	#{bindings.PaymentTypeId.hints.mandato <default> (false)</default>	ory} ~
			Behavior Required: ReadOnly: Disabled:	#{bindings.PaymentTypeId.hints.mandato <default> (false) <default> (false)</default></default>	ory} ~
			Benavior Required: ReadOnly: Disabled: AutoSubmit:	#{bindings.PaymentTypeId.hints.mandato <default> (false) <default> (false) true</default></default>	ory} ~
			Benavior Required: ReadOnly: Disabled: AutoSubmit: AutoTab:	#{bindings.PaymentTypeId.hints.mandato <default> (false) <default> (false) true <default> (false)</default></default></default>	• • •

Next, we change the **ChangeEventPolicy**, on the **AllSummitOrdersIterator** in the PageDef file of the JSPX document from **PPR** to **None**.

ChangeEventPolicy set to PPR fires a partial refresh on each change of the row currency, which we don't need for this sample. The PageDef file can be accessed from the **Bindings** tab shown at the bottom of the JSPX document in the visual page editor. The image below shows the PageDef view.

Definition File: adf/	sample/view/pageDefs/Li	stOfValuesSwitcherPa	<u>geDef.xml</u>		
ings and Executable	s Contextual Events	Parameters			
Bindings	+ / ×	Executables	+ / X	Data Control	
Custon	nerId rdered	AllSu	ibles mmitOrdersIterator	AppModuleDataControl	
SalesR	epId				
Payme	ntTypeId ntDetail iilad			Find	4A)
First	lieu		Sortable:	<default></default>	• ~
Reviou	s		RSIName:		~
Last			RowCountThreshold:	0	~
🕵 AllSum	mitOrders		viewdefName:		~
			BindingClass:		~
			DefClass:		~
			RenderHint:	<default> (immediate)</default>	• •
			Active Events	. , ,	
			ChangeEventPolicy:	none N	• •
				inorite 2	

Now it is time for a first test of list-of-value switching in an ADF Faces web environment. Select the JSPX document in the JDeveloper Application Navigator and choose run from the context menu.

When the page comes up in a browser, navigate the ADF form data and click the LOV button for different **PaymentTypeId** values to see the change in the list data shown in the two images below.

* Id 97			
* Customerid 201			
DateOrdered 6/12/2011			
DateOrdered 0/12/2011			
Dateshipped 6/14/2011 dG			
SalesRepid 12	Fearch and Felect: Payr	nentDetail	
Total 84000	Search and Select Payl		· · · · · · · · · · · · · · · · · · ·
PaymentTypeId 2	✓ Search		Advanced
PaymentDetail	Match (All Any		
OrderFilled Y	Id		
First Previous Next Last	CardName		
Submit			
			Search Reset
	Id	CardName	
	AMEX	American Express	
	VISA	VISA	
	BIFUE	Carte Bleue	
	MASTER	Master Card International	
	JCB	Japan Credit Bureau	
	•	m	•
			OK Cancel
http://127.0.0.1:state=620x7env8	4 +		
	-		
Form Table			
* Id 98			
* CustomerId 202			
* CustomerId 202	k		
* CustomerId 202 DateOrdered 6/17/2011	b		
* CustomerId 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22	b		
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesReptd 14	6		
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesReptd 14 Total 595	Search and Sele	ct: PaymentDetail	
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesReptd 14 Total 595 PaymentTypedd 1	Search and Sele	ct: PaymentDetail	
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesRepid 14 Total 595 PaymentTypeld 1	Search and Sele ⊻Search	ct: PaymentDetail	Advanced
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesRepid 14 Total 595 PaymentTypeld 1 PaymentDetal	Search and Sele ⊻Search Match @ Al @	c t: PaymentDetail	Advanced
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesReptd 14 Total 595 PaymentTypeld 1 PaymentDetai OrderFilled Y	Search and Sele Search Match @ Al (Iso	s ct: PaymentDetail) Any	Advanced
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesRepId 14 Total 595 PaymentTypeId 1 PaymentDetail OrderFilled Y First Previous Next Last	Search and Sele Search Match @ All (Iso Territory	ct: PaymentDetail) Any	Advanced
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesRepid 14 Total 595 PaymentTypeld 1 PaymentDetai OrderFilled Y First Previous Next Last Submit	Search and Sele ✓ Search Match ④ Al Iso Territory Orreory	c t: PaymentDetail) Any	Advanced
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesRepid 14 Total 595 PaymentDypeld 1 PaymentDypeld 1 Prist Previous Next Last Submit	Search and Selo Search Match @ Al Iso Territory Currency	x ct: PaymentDetail) Any	Advanced
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesRepId 14 Total 595 PaymentTypeId 1 PaymentDetai OrderFilled Y First Previous Next Last Submit	Search and Sele Search Match @ All Iso Territory Currency	:ct: PaymentDetail) Any	Advanced Search Reset
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesReptd 14 Total 595 PaymentTypeld 1 PaymentDetail OrderFilled Y First Previous Next Last Submit	Search and Sele Search Match @ All Iso Territory Currency	c t: PaymentDetail	Advanced Search Reset
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesRepid 6/27/2011 22 SalesRepid 14 Total 595 PaymentTypeld 1 PaymentDetail CrderFilled Y First Previous Next Last Submit	Search and Sele Search Match @ Al Iso Territory Currency Iso	cc: PaymentDetail) Any Territory	Advanced Search Reset
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesRepId 14 Total 595 PaymentTypeId 1 PaymentDetai OrderFiled Y First Previous Next Last Submit	Search and Sele Search Match @ Al Iso Territory Currency Iso EURO Acc	ct: PaymentDetail	Advanced Search Reset
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesReptd 14 Total 595 PaymentTypeId 1 PaymentDetail OrderFilled Y First Previous Next Last Submit	Search and Sele Search Match @ All Iso Territory Currency EURO ARS AUD	Ct: PaymentDetail	Advanced Search Reset
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesRepid 14 Total 595 PaymentTypeld 1 PaymentDetail OrderFilled Y First Previous Next Last Submit	Search and Selo	ct: PaymentDetail) Any Territory EuropeanUhion ArgentIna Russia	Advanced Search Reset
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesReptd 14 Total 595 PaymentDypeid 1 PaymentDypeid 1 Prirst Previous Next Last Submit	Search and Select Search Match @ Al @ Iso Territory Currency Iso EURO ARS AUD RUB GB	ct: PaymentDetail	Advanced Search Reset
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesRepId 14 Total 595 PaymentTypeId 1 PaymentTypeId 1 Prirst Previous Next Last Submit	Search and Sele Search Match @ All @ Iso Terrtory Currency Iso EURO ARS AUD RUB GP IIR	Ct: PaymentDetail	Advanced Search Reset
* Customerid 202 DateOrdered 6/17/2011 22 SalesRepid 6/27/2011 22 SalesRepid 14 Total 595 PaymentTypeld 1 PaymentDetail CrderFilled Y First Previous Next Last Submit	Search and Selo Search Match @ Al Iso Territory Currency Iso ELRO ARS AUD RUB GBP INR BRL	ct: PaymentDetail) Any Territory EuropeanUnion Argentinas Australa Russia Great Brittain India Brazilia	Advanced Search Reset
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesRepid 14 Total 595 PaymentDypeid 1 PaymentDypeid 1 PaymentDypeid 1 First Previous Next Last Submit	Search and Sele Search Match @ Al (Iso Territory Currency EURO ARS ALD RUB BR USD	ct: PaymentDetail	Advanced Search Reset
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesRepId 14 Total 595 PaymentTypeId 1 PaymentDetai OrderFilled Y First Previous Next Last Submit	Search and Sele Search Match @ All Iso Territory Currency Iso EURO ARS ALD RUB GBP INR BRL USD CAD	Ct: PaymentDetail	Advanced Search Reset
* Customerid 202 DateOrdered 6/17/2011 22 SalesRepid 6/27/2011 22 SalesRepid 14 Total 595 PaymentDypeld 1 PaymentDypeld 1 PaymentDetail Q OrderFilled Y First Previous Next Last Submit	Search and Sele Search Match @ Al Iso Territory Currency Iso ELRO ARS ALD RUB GBP INR BRL USD CAD ONY	Ct: PaymentDetail	_Advanced SearchReset
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesRepid 14 Total 595 PaymentDypeid 1 PaymentDypeid 1 PaymentDypeid 1 First Previous Next Last Submit	Search and Sele Search Match @ Al Iso Territory Currency Currency EURO ARS AUD RUB BRI USD CAD CAD CAP	Ct: PaymentDetail	Advanced Search Reset
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesReptd 14 Total 595 PaymentTypeId 1 PaymentDetai OrderFiled Y First Previous Next Last Submit	Search and Sele	Ct: PaymentDetail	Advanced Search Reset
* Customerid 202 DateOrdered 6/17/2011 2 SalesNepid 6/27/2011 2 SalesNepid 14 Total 595 PaymentDypeld 1 PaymentDypeld 1 PaymentDetail Criefrilled Y First Previous Next Last Submit	Search and Sele Search Match @ Al Iso Territory Currency Iso ELRO ARS ALD RUB GBP INR BRL USD CAD CNY DKK HKD JPY SRD	ct: PaymentDetail	_Advanced
* Customerid 202 DateOrdered 6/17/2011 22 SalesRepid 6/27/2011 22 SalesRepid 14 Total 595 PaymentDypeld 1 PaymentDypeld 1 Prist Previous Next Last Submit	Search and Sele Search Match @ Al Iso Territory Currency EURO EURO RUB GBP INR BRL USD CAD CAD CAD CAD CAD CAD CAD CA	Ct: PaymentDetail) Any European Union ArgentinkS Australia Russia Great Brittain India Brazilia United States Carnada China Denmark Hong Kong Japan Singapore South Africa	Advanced
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesReptd 14 Total 595 PaymentTypeId 1 PaymentDetal 0 OrderFiled Y First Previous Next Last Submit	Search and Sele Search and Sele Iso Territory Currency Iso EURO ARS AUD RUB GPP INR BRL USD CAD CAD CAD CAD CAN CAN CAN CAN CAN CAN CAN CAN	Ct: PaymentDetail	Advanced Search Reset
* Customerid 202 DateOrdered 6/17/2011 22 SalesRepid 6/27/2011 22 SalesRepid 14 Total 595 PaymentDypeld 1 PaymentDypeld 1 Prist Previous Next Last Submit	Search and Sele	Ct: PaymentDetail) Any	Advanced Search Reset
* Customerid 202 DateOrdered 6/17/2011 22 SalesRepid 6/27/2011 22 SalesRepid 14 Total 595 PaymentTypeld 1 PaymentDetal OrderFilled Y First Previous Next Last Submit	Search and Sele Search Match @ Al (Iso Territory Currency EURO ARS ALD RUB GBP INR BRL USD CAD CHY DKK HKO JPY SGD ZAR CHF TWY 4	ct: PaymentDetail	Advanced Search Reset
* Customerid 202 DateOrdered 6/17/2011 22 DateShipped 6/27/2011 22 SalesReptd 14 Total 595 PaymentDypeid 1 PaymentDypeid 1 PaymentDypeid 7 First Previous Next Last Submit	Search and Sele Search and Sele Iso Territory Currency Iso EURO ARS AUD RUB GPP INR BRL USD CAD CAD CAD CAD CAN CAN CAN CAN CAN CAN CAN CAN	Ct: PaymentDetail	Advanced Search Reset
* Customerid 202 DateOrdered 6/17/2011 2 SalesReptd 6/27/2011 2 SalesReptd 14 Total 595 PaymentDypeld 1 PaymentDypeld 1 Prist Previous Next Last Submit	Search and Sele Search Match At Iso Territory Currency Iso ELRO ARS ALD RUB GBP INR BRL USD CAD CAD CAT SCO ZAR CHP T C	Any Territory European Union Argentinal & Australia Russia Great Brittain India Brazilia United States Carnada China Dermark Hong Kong Japan Singapore South Africa Switzerland Tr	Advanced Search Reset

The same LOV switch works in ADF tables too. Drag the **AllSummitOrders** collection to the JSF page and choose **Table** | **ADF Table** from the context menu.

ADF CODE CORNER



Again, the **PaymentTypeIdTranslated** attribute can be deleted from the display as this is not needed in the table.

Edit Table Columns Edit Table Columns Row Selection O None Single Row Multiple Rows] Enable Sorting Enable Eiltering	
Columns:		🕂 💥 Group Ungroup
Display Label	Value Binding	Component To Melete
<pre>xxa <default></default></pre>	Id Id	DF Input Text w/ Label
Image: second secon	CustomerId	DF Input Text w/ Label
<pre>xxa <default></default></pre>	DateOrdered	ADF Input Date w/ Label
<pre>xxa <default></default></pre>	DateShipped	ADF Input Date w/ Label
<pre>xma <default></default></pre>	SalesRepId	🛱 ADF Input Text w/ Label 🛛 🙀
<pre>xxa <default></default></pre>	📼 Total	n ADF Input Text w/ Label
<pre>xma <default></default></pre>	PaymentTypeId	학 ADF Input Text w/ Label 🍟
<pre>xma <default></default></pre>	PaymentDetail	🗐 ADF List of Values Input
<pre>xma <default></default></pre>	OrderFilled	لَّ ADF Input Text w/ Label
₩ <default></default>	PaymentTypeIdTranslated	🚽 🛑 ADF Input Text w/ Label 🥂
Help		OK Cancel

Note: Make sure the ADF table is configured for **single row selection** after dragging the collection onto the page. For this set the **Row Selection** property shown in the image above from **None** to **Single Row**.

Set the **AutoSubmit** property of the af:inputText component in the PaymentTypeId column to **true**.

d	Customerid	PaymentTypeld	PaymentDetail	DateOrdered	DateShipped	Sales
#{ld.inputValue]	#{Customerld.in	#{PaymentType	#{PaymentDe	🔍 #{DateOrdere 🖄	#{DateShippe	#{
#{ld.inputValue]	#{CustomerId.in	#{PaymentType	#{PaymentDe	🗼 #{DateOrdere 🖄	#{DateShippe	#{5
#{ld.inputValue]	#{Customerld.in	#{PaymentType	#{PaymentDe	#{DateOrdere 🖄	#{DateShippe	#{\$
		R P	1 🖻 🥒 📑 (Find	\$ \$)?
		Require	ed: #{bin	dings.AllSummitOrders.hi	nts.PaymentType	~
		ReadO	only: <def< td=""><td>ault> (false)</td><td>-</td><td>~</td></def<>	ault> (false)	-	~
		Disable	ed: <def< td=""><td>ault> (false)</td><td>• .</td><td>~</td></def<>	ault> (false)	• .	~
		AutoSu	ubmit: <def< td=""><td>ault> (false)</td><td>~ .</td><td>~</td></def<>	ault> (false)	~ .	~
		AutoTa	ab: <defa< td=""><td>ault> (false)</td><td></td><td>-</td></defa<>	ault> (false)		-
		Partial	Triggers: true		N	-
		Refres	hCondition:			~ 1
		···· Validat	tion			
		🗐 Maximi	umLength: #	{bindings.AllSummitOrder	s.hints.PaymentT	~
		Immed	liate:	defaults (falce)	•	

Run the JSPX document to test the LOV for different table rows.

Form	Table				
Id	CustomerId	PaymentT	Search and Select: Pay	mentDetall	
97	201	2	Search		A <u>d</u> vanced
98	202	1	Match (All Anv		
99	203	2	Id		
100	204	2	CardName		
101	205	2			
102	206	1			Search Reset
103	208	1	Id	CardName	
104	208	2	AMEX	American Express	
105	209	2	VISA	VISA Diperer Sub International	
106	210	2	BLEUE	Carte Bleue	
107	211	2	MASTER	Master Card International	
108	212	2	JCB	Japan Credit Bureau	
109	213	2			
110	214	1			
111	204	1			
112	210	2			
113	202	1			
114	202	1			
115	202	1			
116	202	1			
117	202	1	•	m	+
118	206	1			OK Cancel
	200	-			

Conclusion

Model driven LOVs in AD Business Components can be configured to conditionally show different sets of data. This article explained how to change the LOV data lists dependent on whether the payment type is CASH or CREDIT. The database schema used in this sample is ADF Summit. You can download the ADF Summit sample and schema scripts from

http://download.oracle.com/otn_hosted_doc/jdeveloper/11gdemos/SummitADF/SummitADFV1_0_08072011.zip

You can download the Oracle JDeveloper 11.1.1.4 sample workspace from ADF Code Corner website where it is sample 89

http://www.oracle.com/technetwork/developer-tools/adf/learnmore/index-101235.html

Make sure you run the SQL script contained in the SUMMIT_ADF_SCHEMA-PATCH folder to add the credit cards table and currencies table used in this sample. Configure the database connect information in the sample to point to a database of yours that has the ADF Summit schema installed.

RELATED DOCOMENTATION

ADF Summit application and schema
http://download.oracle.com/otn_hosted_doc/jdeveloper/11gdemos/SummitADF/SummitADFV1_0_ 08072011.zip