

# Forms Wissen trifft auf ADF

Jan-Peter Timmermann  
Pitss GmbH  
Stuttgart

## Schlüsselworte

Forms Weblogic ADF.

## Einleitung

Mit diesem Vortrag wird eine Gegenüberstellung von Forms zu ADF getätigt. Es wird aufgezeigt wie sich ein Datenblock in ADF wiederfindet. Es werden anhand von Beispielen die Unterschiede / Gemeinsamkeiten von beiden Welten gegenübergestellt umso dem Forms Entwickler ein Gefühl für den Einstieg in ADF zu geben.

Wie erstelle ich einen Canvas in Forms => Was heißt das in ADF

Wie erstelle ich einen Datenblock in Forms => Wie in ADF

Relationen in Forms, wie schaffe ich das in ADF

Trigger oder Programm Units sind in Forms recht einfach zu erstellen, wie mach ich das jetzt in der ADF Welt. Das sind die Fragen die dieser Vortrag versucht zu beantworten

## Die Entwicklungs- Umgebung

Als erstes möchte ich einmal die beiden Entwicklungs-Umgebungen vorstellen. Der FormsBuilder in der Version 11gR2 kommt fast mit demselben Look and Feel wie die Vorgänger.

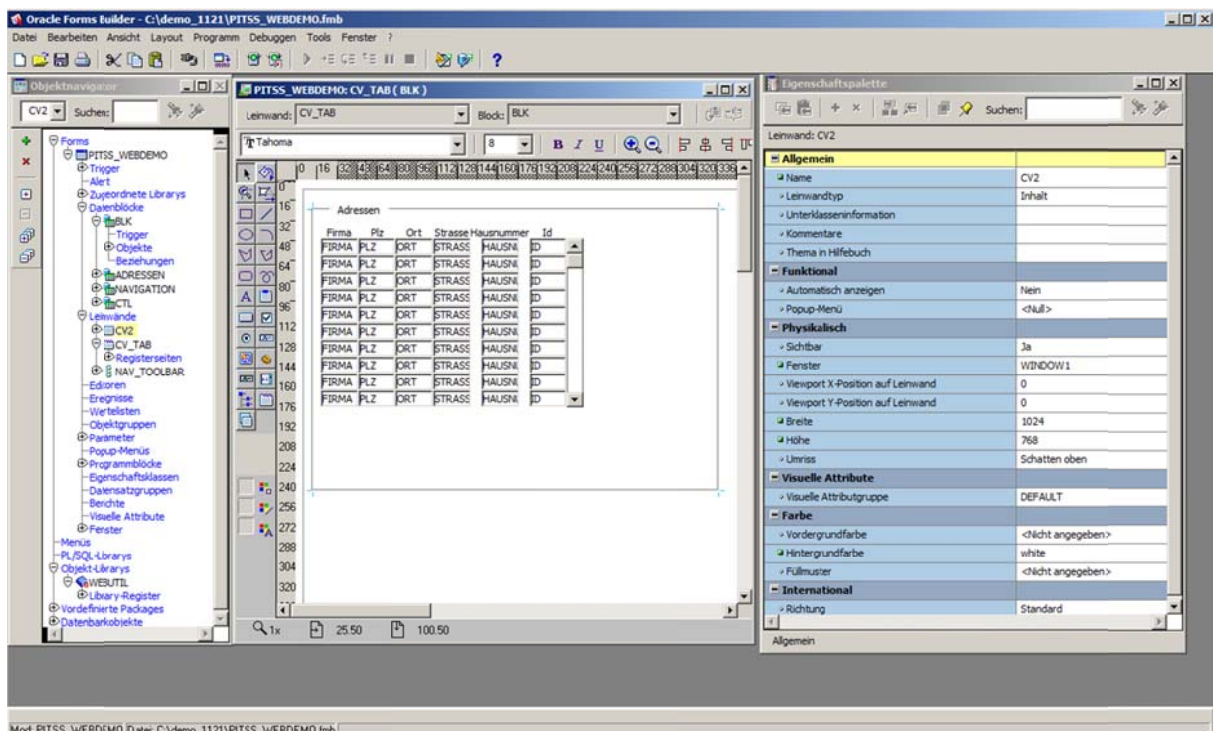


Abb. 1: Forms Bulder 11gR2 11.1.2.1

Wie gewohnt befindet sich auf der linken Seite unser Navigator in der Mitte der IDE sehen wir den Canvas Bereich und auf der rechten Seite die Property-Palette.

Wenn wir uns dazu jetzt einmal den JDeveloper näher anschauen.

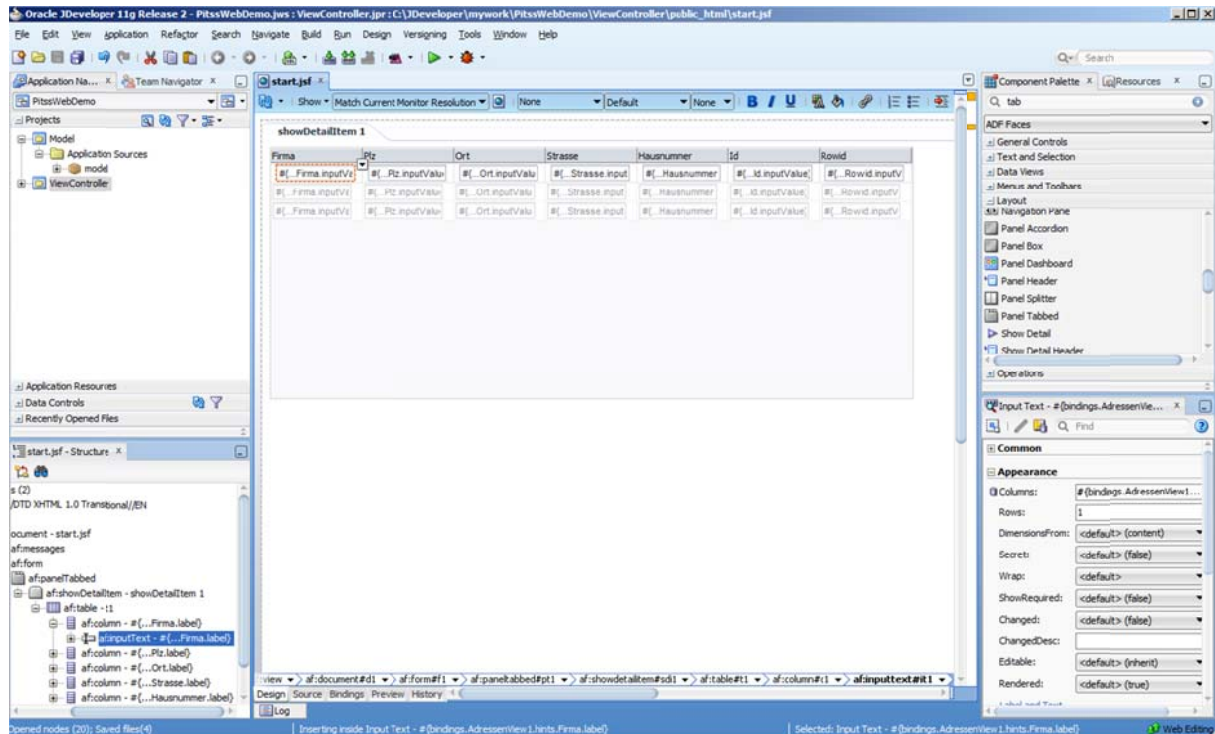
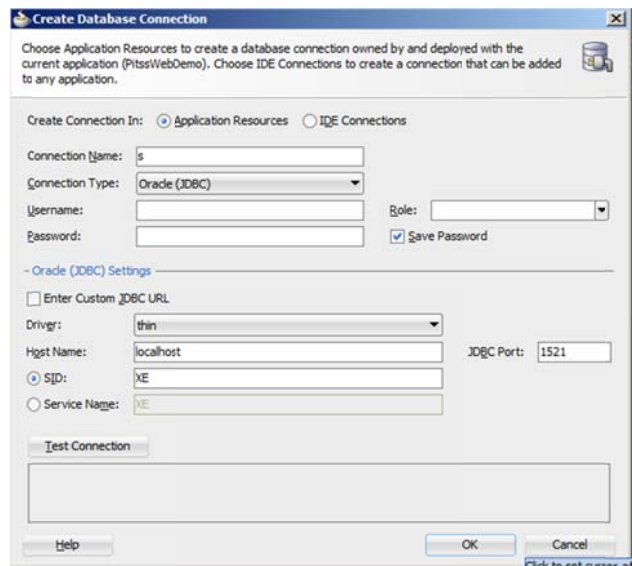


Abb. 2: Oracle JDeveloper 11.1.2.3.0

Genauso wie in dem FormBuilder haben wir auf der linken Seite einen Navigations-Baum der uns durch unsere ganze Anwendung führt. In der Mitte wie gewohnt der "Canvas" für unsere Objekte die wir darstellen wollen. Und genauso wie im FormBuilder die dazu gehörige Property Palette.

In beiden Umgebungen finden wir eine fast gleiche Struktur wieder. Der erste Schritt einer Anwendung ist das Anmelden an eine Datenbank. Hier wird im FormBuilder

Abb. 3: FormBuilder anmelden



Beide Anmelde Routinen führen uns durch den Anmelde Prozess. Beim JDeveloper ist es das Erstellen der Datenbank Verbindung, beim FormsBuilder der Anmelde Prozess.

Was an dieser Stelle als erstes auffällt ist: Beide IDE's sind stark WIZARD geführt. Dies betrifft fast alle Objekte.

### Eine neue Anwendung

In Forms ist eine „Anwendung“ in den meisten Fällen eine \*.fmb (mmb,pll,olb). In der ADF-Welt besteht eine Anwendung aus mehreren Teilen, da hier das MVC-Model (Trennung BC, Controller und Layout) zum Tragen kommt.

Zum Erstellen wird in Forms auf (+) oder Neu gedrückt. Dann wird ein neues Modul erstellt.

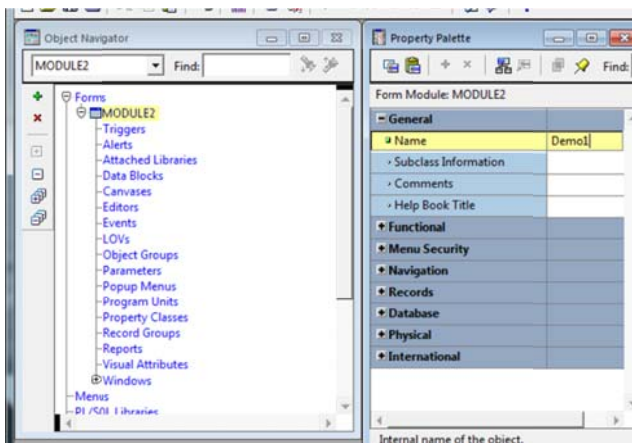


Abb.4: FormBuilder neues Modul

In diesem Modul sind das Layout sowie die Logik hinterlegt. Im vereinfachten Sinne werde ich im ADF Umfeld bei der Erstellung einer neuen „Applikation“ geführt.

Es fängt an mit der Auswahl was möchte ich erstellen.

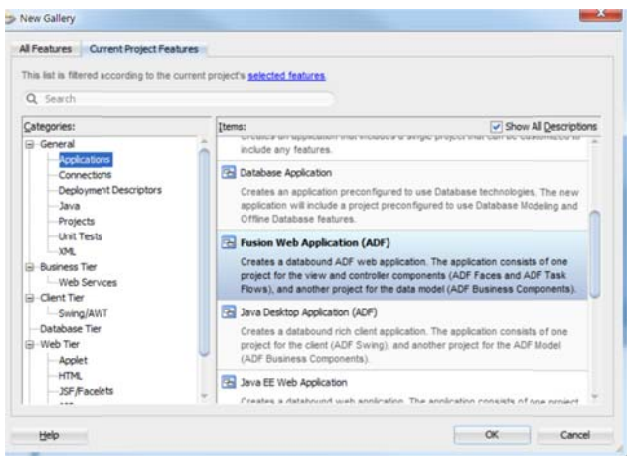


Abb.5: JDeveloper neues Modul

Auch hier muss ich meiner Anwendung einen Namen mit geben.

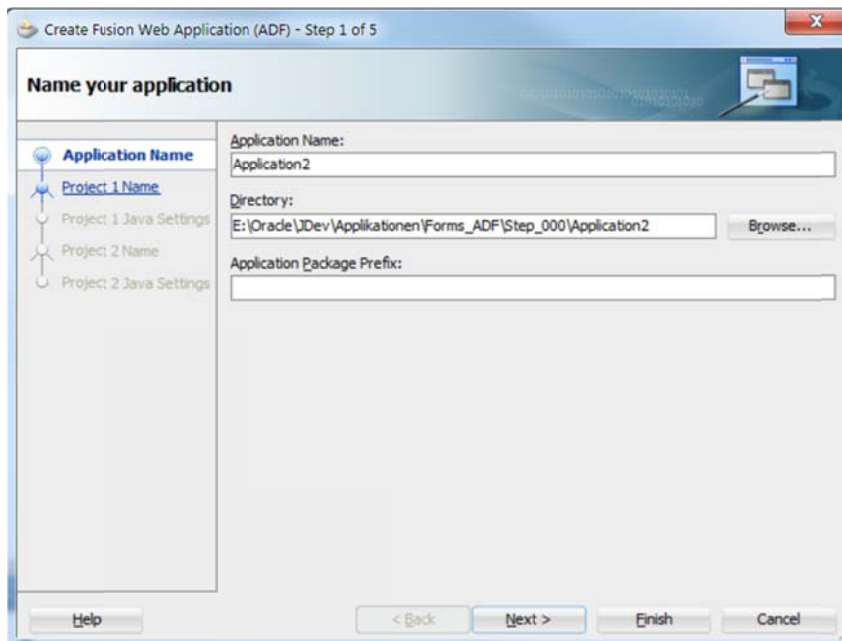


Abb.6: JDeveloper Application Name

Da wir hier die Trennung zwischen Daten/Logik/Layout haben müssen ein paar mehr Fragen für den Wizard beantwortet werden.



Abb.7: JDeveloper Model Name

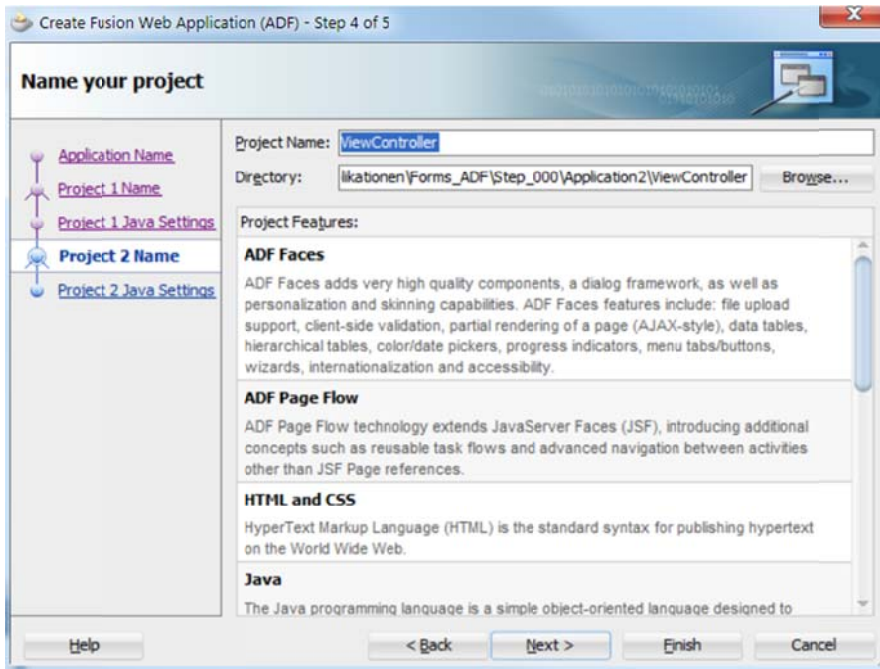


Abb.8: JDeveloper ViewController Name

## Daten in der Anwendung

Beide Anwendungen leben von und mit Daten. Hier verhalten sich beide Umgebungen recht identisch. Im FormsBuilder nutze ich den Datenblock Wizard zum Erstellen der Blöcke die mir die Informationen bereitstellen sollen.

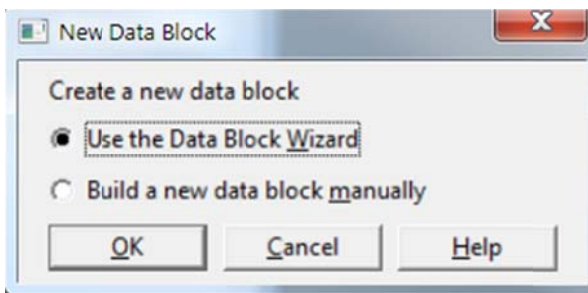
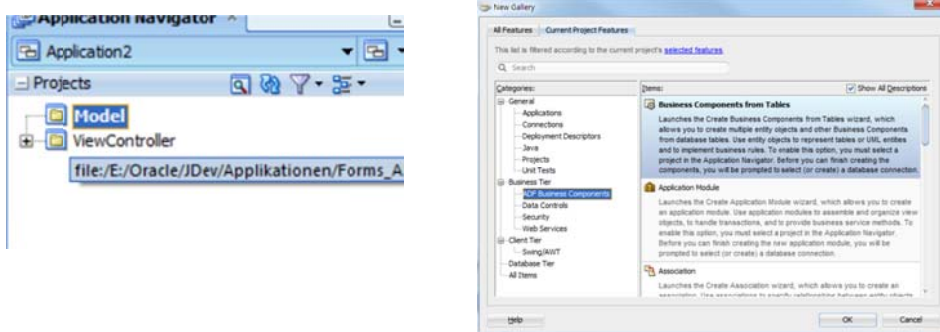


Abb.9: FormsBuilder Datenblock Wizard starten

Abb.10: BC Komponente erstellen





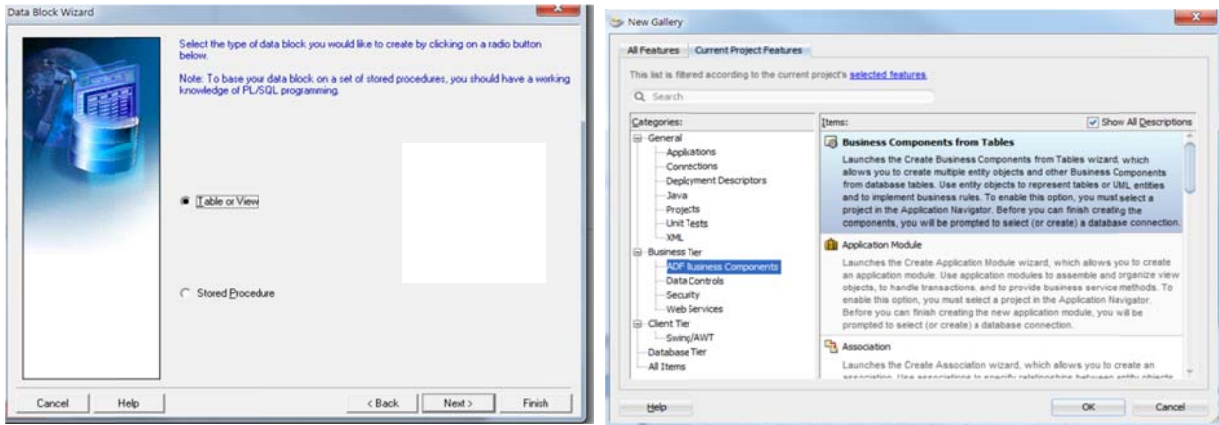


Abb.11: Auswahl was möchte ich erstellen

Über den Wizard werden dann die Tabellen angezeigt auf die der User Zugriff hat. Und genauso wie in der Forms-Welt kann man sich die Tabellen und die Spalten mit denen man arbeiten möchte auswählen.

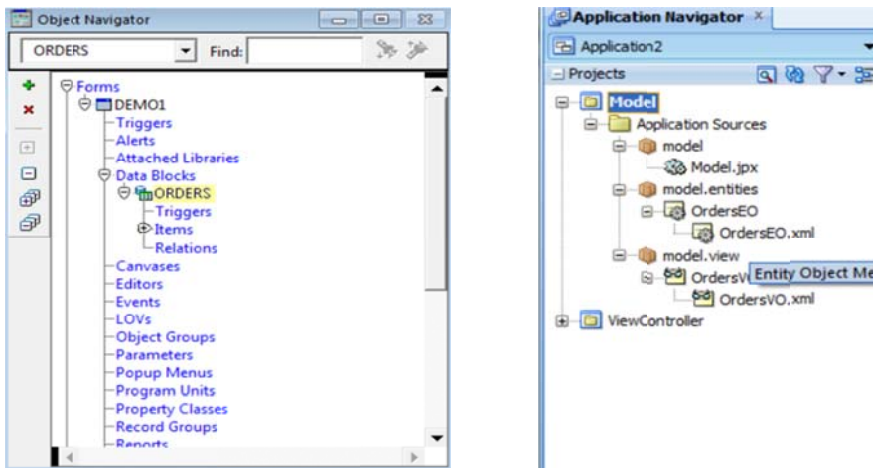


Abb.12: Datenblöcke / Model Sicht

Im Navigations-Bereich stehen mir der „Datenblock“ in Forms sowie das „Model“ im JDeveloper zur Verfügung. Beide Seiten bieten mir diverse Eigenschaften an die man über Property Paletten verändern kann.

Das was ich aus Forms heraus gewohnt bin, die Data Source Columns zu betrachten, bietet mir auch der JDeveloper an.

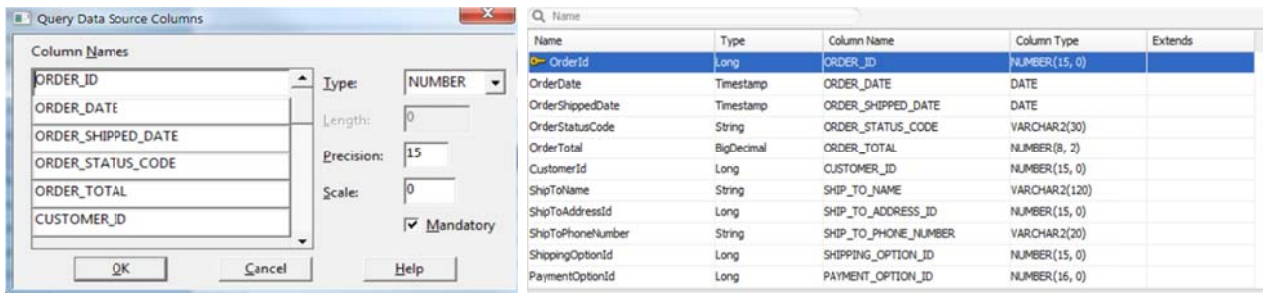


Abb.13: Quellen der Datenbanken

## List of Values in der Anwendung

In beiden Welten wird man sich enge Verbundenheit mit der Datenbank zu Nutze machen um dem Anwender Auswahl Möglichkeiten bei Eingabe Feldern zur Verfügung zu stellen. In Forms nennen wir das LOV's (List of Values).

Jeder der mit Forms arbeitet kennt das Vorgehen für so eine Liste.

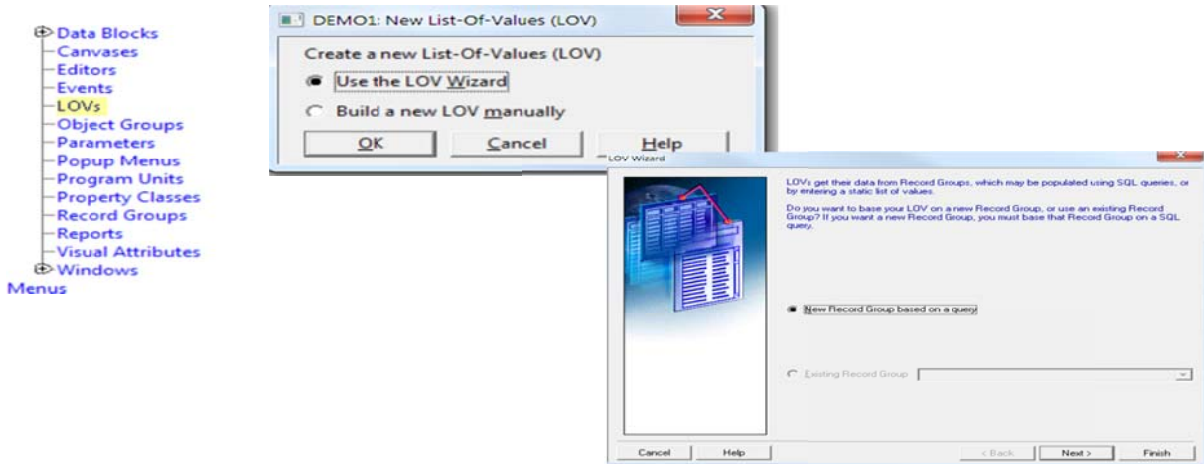


Abb.14:LOV-Wizard in Forms

Ich habe die Möglichkeit die Daten per SQL-Query einzugeben oder per Query-Builder mir das Select zusammen zu stellen.

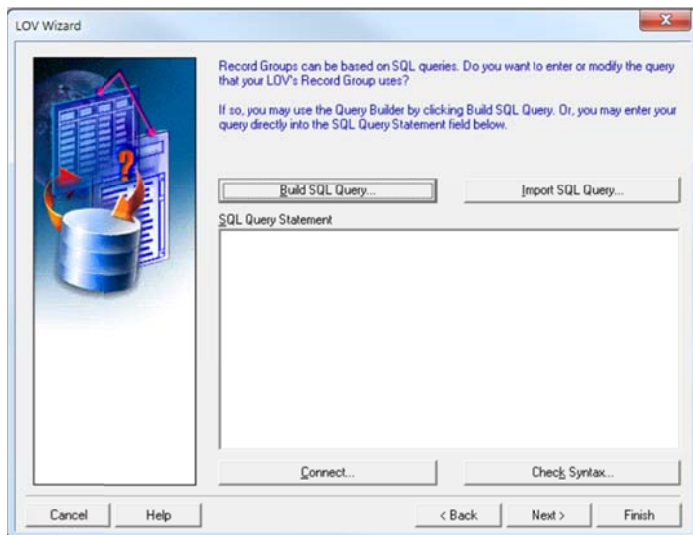


Abb.15:Query für die LOV in Forms

Ebenso sind wir es gewohnt in dem Wizard die Felder die wir anzeigen wollen auszuwählen, sowie die Felder zu zuweisen in denen wir die Werte eintragen wollen.

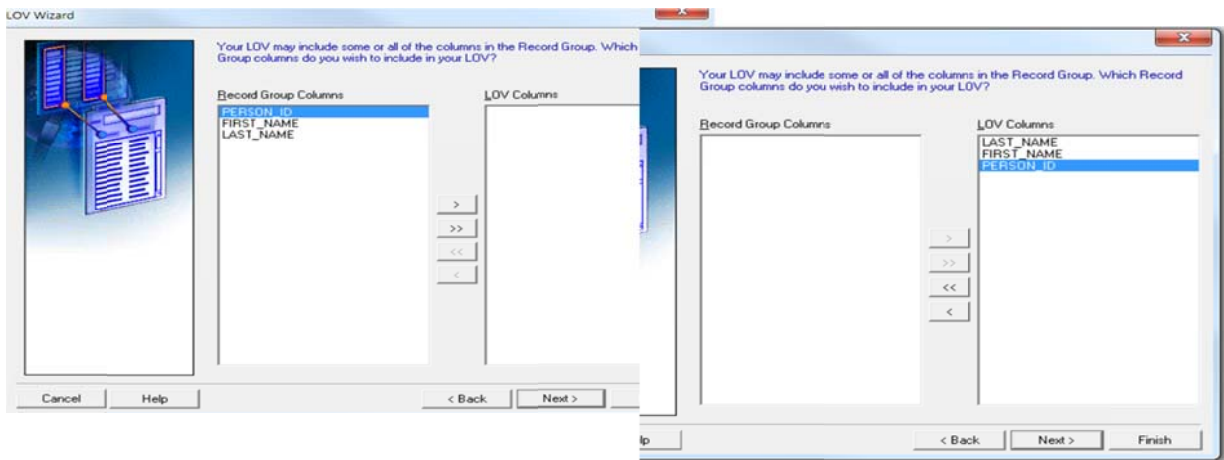


Abb.16: Auswahl der Felder

Das ganz basiert auf einer Record Group die mit erstellt wird, oder im Vorwege erstellt worden ist.

Im JDeveloper haben wir nun ein ähnliches Verhalten. Wir bewegen uns zu dem Objekt aus welchem heraus wir eine Liste erstellen wollen. In unserem Falle ist es das PersonVO (Personen View Object).

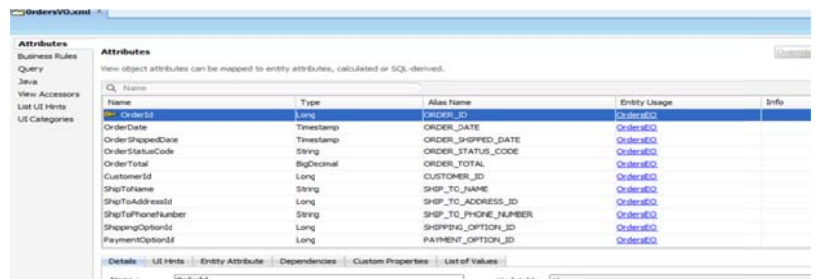
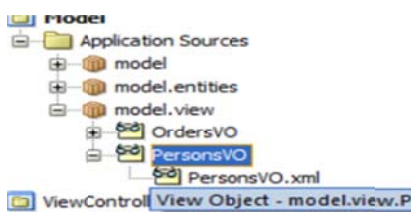


Abb.17: Auswahl des Objektes

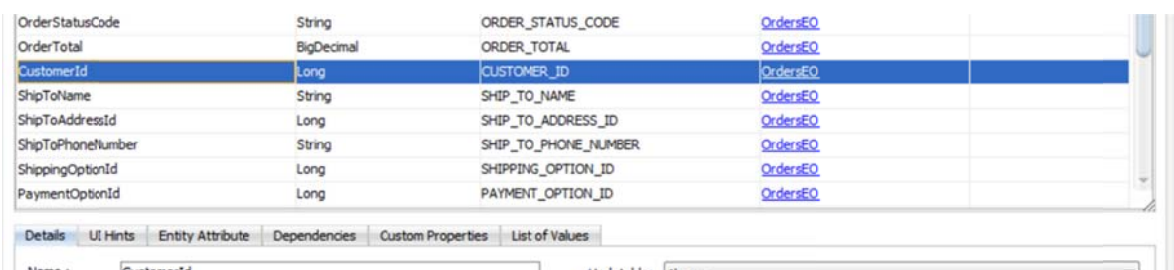


Abb.18: Auswahl des Attributes



Über den Tab-Reiter List of Values, haben wir dann die Möglichkeit hier eine LOV zu beschreiben.

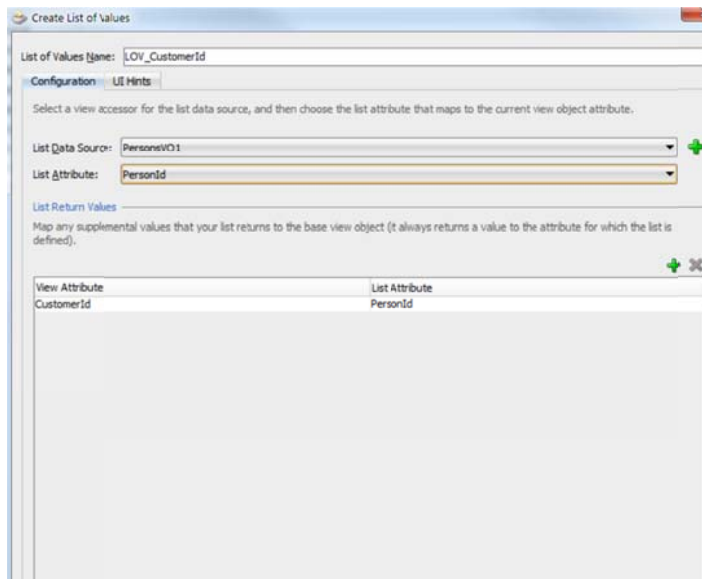


Abb.19: Beschreibung der List of Values

## Fazit?

Für mich als Forms – Entwickler gibt es sehr viele Bereiche in denen ich mich sehr schnell wieder finde. Aber „Es ist nicht Forms“. Es bedarf hier einer vernünftigen Ausbildung, Einarbeitung. Und man sollte nicht versuchen „Alleine“ so ein neues ADF-Projekt zu bewältigen. Es hat sich gerade für Forms – Entwickler als sehr positiv herausgestellt mit ADF- Entwicklern zusammen zu arbeiten. Schön ist es auch immer wieder, wenn man seine Forms Module migriert und dann schon mit einem bestehendem ADF-Set anfangen kann.

Ich möchte Mut machen sich mit dem Thema zu beschäftigen und keine Angst vor dem „neuen“ Aussehen.

## Kontaktadresse:

Jan-Peter Timmermann  
Pitss GmbH  
Zettachring , 2  
D-70567 Stuttgart

Telefon: +49 (0) 1722151043  
E-Mail: jtimmermann@pitss.de  
Internet: www.pitss.de