

2012  
**DOAG**  
Konferenz + Ausstellung



# Der Widerspenstigen Zähmung - Dokumentation in APEX Projekten

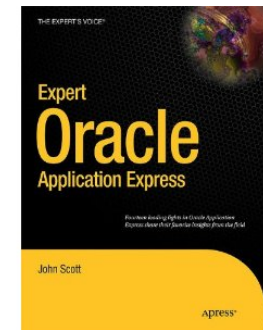
Dietmar Aust

Opal-Consulting, Köln

[www.opal-consulting.de](http://www.opal-consulting.de) / [www.jdd-software.com](http://www.jdd-software.com)

- ▶ Dipl.-Inform. Dietmar Aust
  - Blog: <http://daust.blogspot.com/>
- ▶ Entwicklung von Webapplikationen mit Oracle seit 1997
- ▶ 1997-2000: Consultant at Oracle Germany
  - Oracle Portal, Forms, Reports, OWA Toolkit
- ▶ Seit 09/2000: Freiberuflicher Consultant, seit 2006 – nur noch **APEX!**
- ▶ Regelmäßige Konferenzvorträge (ODTUG, DOAG, OOW)
- ▶ Autor des JasperReportsIntegration Toolkits
  - Open Source Alternative, um druckfähige Berichte für APEX zu erstellen.
  - <http://www.opal-consulting.de/tools>

- ▶ Regelmäßige APEX Trainings zusammen mit Denes Kubicek:
  - Oracle APEX: Knowhow aus der Praxis
  - Oracle APEX New Features
  
- ▶ Co-Autor von „Oracle APEX und XE in der Praxis“
  - Veröffentlicht 21.12.2009
  
- ▶ Co-Autor von „Expert Oracle Application Express“
  - Veröffentlicht 25.05.2011
  - Charity Projekt in Gedenken an Carl Backstrom und Scott Spadafore (ehemalige Mitglieder des APEX Teams)
  
- ▶ JDD-Software UG 2012 gegründet
  - <http://jdd-software.com>



- ▶ Der Hintergrund
- ▶ Vorlagen für die Dokumentation
- ▶ Management eines Delta Releases
- ▶ WordWiki

- ▶ Woher stammen die “Weisheiten” ... ein bisschen Kontext
- ▶ Entwicklung eines Shop-Verwaltungssystems bei der Telekom Shop Vertriebsgesellschaft mbH zur Unterstützung der relevanten Prozesse und operativen Tätigkeiten in 2007
- ▶ Seitdem kontinuierliche Entwicklung mit 4 Releases pro Jahr
- ▶ Begonnen mit APEX 2.2, mittlerweile auf APEX 4.0
- ▶ Gewachsene Applikation, mittlerweile
  - >230 Applikationsseiten,
  - >150 Tabellen,
  - >100 Packages
- ▶ Unterstützung durch ein Test Team, Entwicklung, Abnahme, Produktion
- ▶ Riesiges Thema, Konzentration auf Erfolgsfaktoren aus unserer Sicht

- ▶ Zu viel oder zu wenig Dokumentation, die richtige Granularität zu finden, ist schwer.
  - Tools zur automatischen Generierung liefern zu viel Information
  - Das Wesentliche ist nicht mehr zu erkennen, wird trotzdem gerne als Rechtfertigungsdokumentation genommen!
  
- ▶ Dokumentation wird nicht zeitnah gepflegt
  - Die existierende Dokumentation ist die meiste Zeit über veraltet
  - Die Dokumentation ist nicht am Sourcecode orientiert
  
- ▶ Delta vs. Full
  - Während der Lebenszeit einer Applikation werden üblicherweise die Änderungen dokumentiert (für jedes Release). Die Gesamtbeschreibung wird selten gepflegt.
  
- ▶ Die Dokumentation wird oft im MS Word Format gefordert.
  - Vorlagen im Unternehmensstandard, statischer Text, Datenmodell-Diagramme , eingebettete MS Office Dokumente

## ► Redundanz

- Ist unbedingt zu vermeiden
  - Sehr schwierig synchron zu halten
  - Schnell verwirrend (was ist der aktuelle Stand?) und die gesamte Dokumentation wird als unzuverlässig eingestuft

- ▶ Die Dokumentation **auf das Wichtige reduzieren**. Dazu müssen wir **die relevanten Teile** der Applikation dokumentieren ... mit möglichst **geringem Aufwand**.
- ▶ **Don't make me think!**
  - Wenn der Druck im Projekt zunimmt => hört die Pflege der Dokumentation auf!
  - Prozess für Dokumentation etablieren, dem ich einfach nur zur folgen habe (Vorlagen und Checklisten), fast automatisch. Es dürfen nur wenige und klare Vorgaben existieren.
- ▶ **Delta vs. Full**
  - Prozess etablieren, in dem die Gesamtbeschreibung einfach aktualisiert werden kann, nachdem das Delta-Release ausgeliefert worden ist.
- ▶ **Dokumente im MS Word Format erstellen**



- ▶ Für wen erstellen wir die Dokumentation?
  - Auftraggeber / die Fachabteilung
  - Entwickler
  - Test-Team
  - IT-Betrieb
  - 3<sup>rd</sup> Level Support
  - Endanwender der Applikation
  
- ▶ Für alle diese Zielgruppen ist der wichtigste Aspekt das **Verhalten** und die **Business Rules** der Applikation
  - Das **WAS** und **WARUM** ist viel wichtiger als das **WIE!**
  - Die technischen Details kann ich immer herausfinden, wenn ich mir den Sourcecode ansehe.
  - Aber ich muss wissen, welche Aktionen auf der aktuellen Seite der Applikation ausgelöst werden können und was dann das Ergebnis sein soll.

# Vorlagen und Checklisten ::Demo anhand der APEX Sample Application

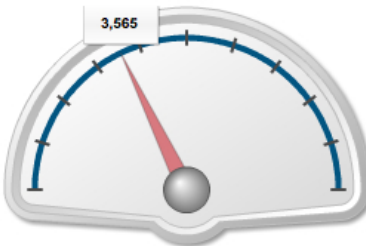
- Konzept vorstellen anhand der APEX Sample Application, die Bestandteil von jedem APEX Release ist.

Sample Database Application Welcome: DIETMAR.AUST Print Feedback Logout

**Home** Customers Products Orders Reports

Home

#### Sales Quota for this Month



3,565

#### Top Customers

<a href="#">Bradley, Eugene - 2 Order(s)</a>	<b>\$2,760.00</b>
<a href="#">Logan, Edward - 2 Order(s)</a>	<b>\$2,420.00</b>
<a href="#">Dulles, John - 1 Order(s)</a>	<b>\$2,380.00</b>
<a href="#">Hartsfield, William - 2 Order(s)</a>	<b>\$2,370.00</b>
<a href="#">LaGuardia, Fiorello - 1 Order(s)</a>	<b>\$1,090.00</b>
<a href="#">OHare, Edward "Butch" - 1 Orde...</a>	<b>\$1,060.00</b>
<a href="#">Lambert, Albert - 1 Order(s)</a>	<b>\$950.00</b>

#### Sample Database Application

Welcome to Sample Database Application. This application is designed to highlight the features of Oracle Application Express.

#### Top Products

<a href="#">Jacket - 18 x \$1,50</a>	<b>\$2,700.00</b>
<a href="#">Bag - 16 x \$1,25</a>	<b>\$2,000.00</b>
<a href="#">Trousers - 21 x \$80</a>	<b>\$1,680.00</b>
<a href="#">Ladies Shoes - 12 x \$1,20</a>	<b>\$1,440.00</b>
<a href="#">Business Shirt - 23 x \$50</a>	<b>\$1,150.00</b>
<a href="#">Skirt - 14 x \$80</a>	<b>\$1,120.00</b>
<a href="#">Mens Shoes - 10 x \$1,10</a>	<b>\$1,100.00</b>

#### Top Orders by Date

4/1/2012	<b>\$2,380.00</b>
3/18/2012	<b>\$1,890.00</b>
4/12/2012	<b>\$1,640.00</b>
4/29/2012	<b>\$1,515.00</b>
4/14/2012	<b>\$1,090.00</b>
5/12/2012	<b>\$1,060.00</b>
4/24/2012	<b>\$950.00</b>

#### Tasks

- > Enter a New Order
- > Add a New Customer
- > Add a New Product

## Scope Dokument

- Übersicht für das Management und den Kunden
- Probleme, Ziele, Haupt-Benutzergruppen
- Verfügbarkeit und SLAs, Übersicht der Schnittstellen und des Datenflusses
- Übersicht der Anforderungen

## Requirements-Detail Dokument

- Erkläre die Anforderungen im Detail
- Unterschiedliche Ansätze möglich, z.B. Use Cases oder funktionale Anforderungen
- Eine eindeutige Nummer zuweisen und dann in einer geeigneten Methodik beschreiben

## System Design Dokument

- GUI: APEX Applikationen
- Module (Business Rules und Logik)
- Physisches Datenmodell
- Jobs und Schnittstellen

## Physische Implementierung

- GUI: APEX Applikationen
- Module (Business Rules und Logik)
- Physisches Datenmodell
- Jobs und Schnittstellen

Sourcecode,  
APEX  
Applikationen

Ähnliche  
Granularität



## 1 Einführung

### 1.1 Hintergrund

#### 1.1.1 Problemstellung

### 1.2 Ziele

### 1.3 Stakeholder

### 1.4 Zeitplan / Wichtige Meilensteine

### 1.5 Ansprechpartner

## 2 Anforderungen (Übersicht)

### 2.1 Systemdiagramm(Übersicht)

Übersichtsdiagramm des Systems und der Schnittstellen zu den Umsystemen.  
Wie ist das System in die Gesamtarchitektur des Unternehmens eingebettet.

### 2.2 Schnittstellen (Übersicht)

Welche Schnittstellen existieren für dieses System / diese Applikation? Welche Informationen fließen in das System hinein und welche hinaus?

Es werden ebenfalls alle manuellen Importe in das System benannt (MS Excel).  
Dies ist wichtig, um den Datenfluß über alle IT-Systeme nachvollziehen zu können.

Nr	Name	Quelle	Ziel	Beschreibung
0001	SAP Daten - Import	SAP	Auftragserfassung	<b>Beschreibung:</b>  <b>Häufigkeit:</b> Einmal pro Tag um 03:00 morgens  <b>Technologie:</b> Synchronisation zwischen zwei Oracle Datenbanken über einen Datenbank-Link.

## 2.3 Benutzer und Rollen

### 2.3.1 Benutzer und Standorte

Wir erwarten 3000 Benutzer von allen Standorten in Deutschland.

### 2.3.2 Rollen

Die Anwender der Applikation werden in drei Gruppen unterteilt, das System unterstützt daher die folgenden Rollen mit unterschiedlichen Berechtigungen:

- **Benutzer-Administrator:** Die Personen in dieser Gruppe können Benutzer einrichten und verwalten, das Passwort zurücksetzen, etc. Sie haben keine weiteren Berechtigungen innerhalb der Applikation.
- **Vertrieb:** Ein Mitarbeiter im Vertrieb kann Aufträge und Kunden erfassen. Er hat nur Zugriff auf die von ihm erfassten Daten.
- **Bereichsleiter:** Ein Bereichsleiter kann übergreifende Auswertungen machen über die Aufträge aller Vertriebsmitarbeiter.

## 2.4 Authentifizierung

Die Anmeldung am System erfolgt über ein Prüfung von User/Passwort gegen das Firmen-LDAP-Verzeichnis.



## 2.5 Technische Rahmenbedingungen

Was sind die Minimalanforderungen für dieses System?

- Oracle Datenbank-Release: Oracle 10.2.0.1 (Oracle XE 10g)
- Oracle APEX Release: Oracle APEX 4.0
- J2EE-Server: Tomcat 6.0
- Client Browser: Internet Explorer 8 und Mozilla Firefox 14

## 2.6 System-Verfügbarkeit / SLAs

Das System wird verfügbar sein während der normalen Geschäftszeiten von 09:00-18:00 sowie Mo.-Fr.

Es wird keine Hochverfügbarkeit benötigt. Wartungsfenster sind außerhalb der Geschäftszeiten einzuplanen.

Da es kein Mission-Critical System ist, erlauben wir ungeplante Downtimes für bis zu 1 Tag und im Falle eines Hardware-Ausfalls 2 Tage.

## 2.7 Anforderungen (Übersicht)

(MoSCoW Priorität: **M**ust, **S**hould, **C**ould, **W**on't)

Nr	Bereich	Anforderung	Beschreibung	MoSCoW
0001	Sicherheit	Authentifizierung	Jeder Benutzer muss sich am System anmelden. Die Login-Daten werden gegen das Unternehmens LDAP-Verzeichnis geprüft.	M
0002	Homepage	Dashboard Übersicht	Personalisierte Übersicht der eigenen Verkaufs-Performance bestehend aus: <ul style="list-style-type: none"> <li>• Sales Quota für diesen Monat</li> <li>• Top Kunden</li> <li>• Top Produkte</li> <li>• Top Aufträge</li> </ul>	M
0003	Kunden	Kunden verwalten	Suchdialog für Kunden, Kunden neu anlegen sowie existierende bearbeiten.	M
0004	Produkte	Produkte verwalten	Suchdialog für Produkte, Produkte neu anlegen sowie existierende bearbeiten. Wir können ebenfalls das Produktbild verwalten.	M
0005	Schnittstellen	Massenimport Kundendaten von	Mit einem CSV-Import können wir mehrere Kunden auf einmal erstellen.	S
0006	Schnittstellen	Synchronisation der Kundeninformationen mit SAP	Implementierung der Schnittstelle SST-0001	S



## 2 APEX – GUI: Sample Database Application

Die Sample Database Application wird von Oracle zusammen mit Oracle APEX ausgeliefert. Sie dient als Showcase, um alle Funktionalitäten von APEX zu demonstrieren. Sie wird jeweils zu jedem neuen Release auf den neuesten Stand gebracht.

### 2.1 Authentifizierung

Wir benutzen das integrierte LDAP Authentifizierungsschema und validieren gegen das unternehmensweite LDAP-Verzeichnis. Wir verwenden https anstelle von http.

### 2.2 Privilegien (Autorisierungen)

### 2.3 Theme / Look & Feel

### 2.4 Build Optionen

#### 2.4.1 Standard Build Optionen

- DEBUG
- DISABLED

#### 2.4.2 Funktionale Build Optionen (Kundeneinstellungen)

## 2.5.1 Seiten-Übersicht

Die Übersicht zeigt nur die Struktur der Applikation (inkl. der Aktionen) ohne irgendwelche Bedingungen oder Berechtigungen zu berücksichtigen. Sie zeigt nur, welche Aktionen grundsätzlich auf den entsprechenden Seiten möglich sind. Die Aktionen sind jeweils den Regionen zugeordnet.

Bereich	ID	Dialog	Dialog (Kommentar)	Region	Region (Kommentar)	Aktionen
<b>Sicherheit</b>	101	Login	Die Loginseite, kann von jedem aufgerufen werden.	Login		<ul style="list-style-type: none"> <li>Login</li> </ul>
<b>Home</b>	1	Homepage	Das ist die Homepage, zeigt die personalisierte Sicht auf das Dashboard für alle angemeldeten Benutzer.	Tasks	In dieser Region werden alle Aktionen für diese Seite zusammengefasst.	<ul style="list-style-type: none"> <li>Neuen Auftrag erfassen</li> <li>Neuen Kunden erfassen</li> <li>Neues Produkt erfassen</li> </ul>
				Sales Quota für diesen Monat		<ul style="list-style-type: none"> <li></li> </ul>
				Top Kunden		<ul style="list-style-type: none"> <li>Drill-down auf den ausgewählten Kunden</li> </ul>
				Top Produkte		<ul style="list-style-type: none"> <li>Drill-down auf das ausgewählte Produkt</li> </ul>
				Top Aufträge		<ul style="list-style-type: none"> <li>Drill-down auf den ausgewählten Auftrag</li> </ul>
<b>Kunden</b>	2	Kunden		Kunden	Liste aller Kunden, es kann nach verschiedenen Kriterien gefiltert werden.	<ul style="list-style-type: none"> <li>Neuen Kunden erfassen</li> <li>Daten importieren</li> <li>Drill-down auf den ausgewählten Kunden</li> </ul>
	7	Kunden-Details		Kunden-Details	Die Details zu einem Kunden werden dargestellt	<ul style="list-style-type: none"> <li>Erstellen</li> <li>Speichern</li> <li>Löschen</li> </ul>

Home Customers **Products** Orders Reports

Home > Products > Product Details

### Product Details

Cancel Delete **Apply Changes**

**\*Product Name**

**Product Description**


**\*Category**

**\*Product Available**  Yes  No

**\*List Price**

**Product Image**

### Product Image



## 2.5.10 Seite-6: Produkte > Produkt-Details

Name	Produkt-Details
Kommentar	Zeigt die Details eines Produktes. Wir können das Produkt hier verwalten (create, update, delete) oder wir können auch das Produktbild verändern.
Call Interface	<ul style="list-style-type: none"><li>• P6_PRODUCT_ID (Den Datensatz mit dem Primärschlüssel fetchen)</li></ul>

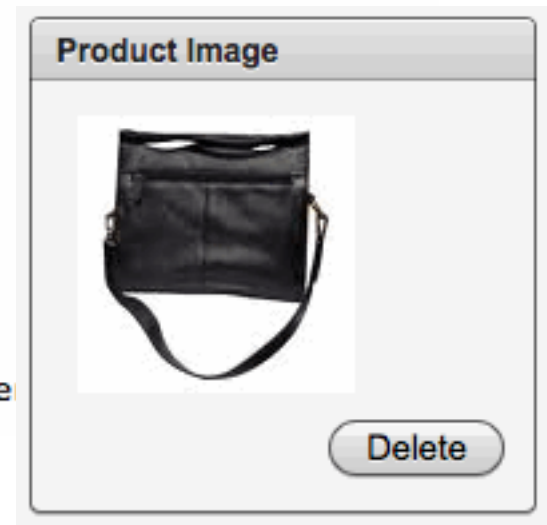
### 2.5.10.1 Regionen

#### Produktbild

Darstellung des Produkt-Bildes.

#### *Aktionen*

- Löschen
  - Beschreibung: Löscht das Bild, Rückkehr auf Übe



**Product Details**

Cancel Delete **Apply Changes**

**\*Product Name**

**Product Description**

**\*Category**

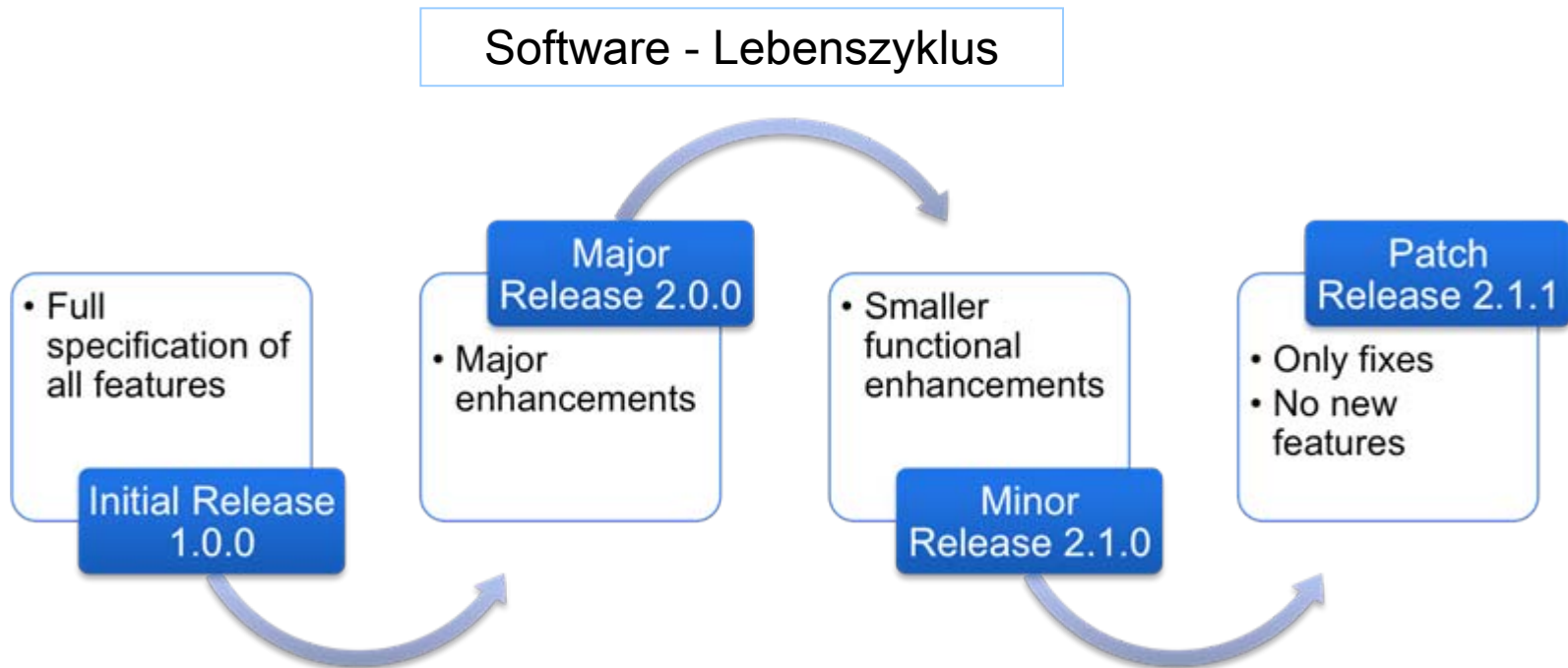
**\*Product Available**  Yes  No

**\*List Price**

**Product Image**   [Download](#)

# Management eines Delta Releases ::Lebenszyklus einer Applikation

- ▶ **Initiales Release** := Gesamtspezifikation aller Funktionalitäten
- ▶ **Delta Release** := Spezifikation aller **Änderungen** an einer **existierenden** Software



## ► Summary Requirement

- Die Anforderung, die der Kunde kommuniziert
  - Üblicherweise grobe Granularität
    - Bsp.: „Kundenverwaltung“
  - oder Vermischung von Sachverhalten / Bundling von Anforderungen
    - Bsp: „Ergonomische Verbesserungen der Applikation“:
      - Zus. Spalten in den Berichten x,y,z
      - Reorganisation der Felder auf mehreren Seiten
- Muss erst analysiert werden und in einzelne, atomare Systemfunktionalitäten (**System Requirements**) übersetzt werden.
- Wird vom Kunden budgetiert und überwacht, ist somit Bindeglied in der Kommunikation.

## ► System Requirement

- Entspricht üblicherweise direkt einer Funktionalität im System, Bsp.:
  - „*Neuen Kunden anlegen*“
  - „*Auftrag erfassen*“ oder
  - „*Kunden im CSV-Format importieren*“
- Granularität: System Requirements sind (fast) **atomar**
  - Create, Update, Delete wird häufig zusammengefasst zu „*Kunde verwalten*“
- Die Liste aller System Requirements **beschreibt vollständig das Verhalten** der Applikation (entweder durch das System oder den Benutzer)
- Bildet oft die Basis für das Testteam



## ► Change Requirement

- Beschreibt die **Veränderung** eines einzelnen existierenden **System Requirements**
- z.B. Kundenanforderung: „Zus. Spalte SAP\_NR in den Kundendaten“ verändert mehrere existierende System Requirements:
  - „Verwalte Kunde (create, update, delete)“
  - „Kunden im CSV-Format importieren“
  - „Kundenbericht x“
  - => **es entstehen 3 Change Requirements**
- Auswirkungen auf die Applikation direkt erkennbar, das Testteam kann die relevanten Teile der Applikation nachtesten.

## ► Non-System Requirement

- **Verändert die Applikation** (die Software) **nicht**
- Oft Bestandteil eines Delta-Releases
- Muss getestet werden
- **Nicht Bestandteil der vollständigen Systembeschreibung**
- Bsp:
  - Einmaliger Import von Stammdaten
  - Befüllung der neuen Spalte SAP\_NR mit Daten aus SAP
  - Skriptbasierte Korrektur von Daten

- ▶ Semantische Differenzierung der Anforderungen ist die Basis für eine saubere Kommunikation der Sachverhalte an den Kunden und das Test Team
- ▶ Extrem wichtiger Faktor, diese semantische Klarheit herzustellen
- ▶ Basis für ein gutes Testen und für die Pflege der Gesamtspezifikation

- Anforderung „Add column SAP\_NO to customer“ als Summary Requirement spezifizieren und anschließend die Anforderung verfeinern:

Q-

REQ ID	Requirement	Requirement Description	Prio	Responsible	Status (Code)	Requirement Type	Changed System Requirement
345	<u>Add column SAP_NO to customer form</u>	<p>Add the column SAP_NO to the customer form.</p> <ul style="list-style-type: none"><li>• Label: SAP-No.</li><li>• The field is freely enterable, there are no validations</li><li>• Position on the form? The field shall be positioned below "Credit Limit"</li></ul>	1	DIETMAR.AUST	NEW	CHANGE	0003: Manage Customers (315)
346	<u>Import current SAP numbers once into customer table</u>	<p>An excel list of the current SAP numbers for the existing customers will be provided and shall be imported into the system.</p>	1	DIETMAR.AUST	NEW	NON-SYSTEM	-
371	<u>Add column SAP_NO to report customer overview</u>	<p>Requirement description: Add the column SAP_NO to report customer overview. The column is visible to everyone who can access the report.</p>	1	DIETMAR.AUST	NEW	CHANGE	0002: Customer Overview (314)

- Änderungen am Datenmodell werden nicht als Anforderung aufgenommen, sondern ergeben sich implizit.

- ▶ Kommunikation der Änderungen an das Test Team und die Fachseite für das Delta-Release:
  - Fragestellung:
    - Was wurde geändert? – Welche System Requirements, Testfälle existieren bereits.
    - Was ist neu, wofür müssen neue Testfälle erstellt werden?
  
- ▶ Erstellung der Release Spezifikation
  - Export aller Anforderungen im HTML Format
  - Import der HTML Datei in die Release Spezifikation (MS Word)
  - Alle geänderten System Requirements werden aus der Gesamtspezifikation in die Release Spezifikation kopiert und auf den neuen Stand aktualisiert.



- ▶ In der Testphase wird oft mehrfach mit dem Test Team und der Fachseite iteriert, dafür werden auf der Abnahme mehrere Versionen bereitgestellt: 1.1.0.0, 1.1.0.1, 1.1.0.2, etc.
  
- ▶ Für jede Iteration werden wieder dieselben Fragen gestellt:
  - Was wurde geändert (in dieser Iteration)?
  - Was müssen wir testen (für diese Iteration)?
  
- ▶ Ansatz:
  - Für jede Iteration wird das Release Dokument aktualisiert und als neue Version gespeichert:
    - Erneuter Export aller Anforderungen im HTML Format
    - Import der HTML Datei in die Release Spezifikation (MS Word)
  - Zwischen den beiden Dateien dann Differenzen aufzeigen mit der MS Word Funktionalität „Extras > Dokumente zusammenführen“
  - Das Differenzdokument an das Test Team und Fachseite kommunizieren

## ► Abschluss des Releases

- Nach der erfolgten Abnahme werden einfach alle System Requirements aus der Release Spezifikation in die Gesamtspezifikation kopiert.
- Ist mit einem Aufwand von 30min-1h zu machen ... einfach Copy/Paste!

- ▶ System-Design Dokument / Feinspezifikation dauerhaft mit geringem Aufwand pflegen ?
  
- ▶ Extraktion der Kommentare und relevanten Inhalte aus unser APEX Applikation über die APEX Dictionary Views:
  - Den APEX Source Code kommentieren:
    - Alles kann mit Kommentaren versehen werden:
    - Applikation, Seite, Region, Prozess, Computation, Item, Branch, Build Option, ...
  
  - Im MS Word Dokument dann Markup einsetzen:
    - Die gesamte Applikation: (:apex.application:)
    - Nur Seite 6: (:apex.application.page page\_id=6:)
    - Tabelle: (:db.table name=demo\_customers:)
    - View: (:db.view name=logger\_logs\_5\_min:)





## ► Makro in MS Word für Generierung ausführen:

- Dokument kopieren
- Markup in Dokument finden:
  - URL erzeugen für Zugriff auf APEX Applikation, z.B:
    - [http://dev-min.opal-consulting.de:8080/apex/f?p=20120618:2:0:::P2\\_CALL\\_INTERFACE,P2\\_TYPE,P2\\_APP\\_ID,P2\\_PAGE\\_ID,P2\\_NAME,P2\\_THEME,P2\\_THEME\\_DETAIL\\_LEVEL:SHOW\\_DB\\_OBJECT,DB.OBJECT,100,,demo\\_orders,demo\\_order\\_items,demo\\_product\\_info,,](http://dev-min.opal-consulting.de:8080/apex/f?p=20120618:2:0:::P2_CALL_INTERFACE,P2_TYPE,P2_APP_ID,P2_PAGE_ID,P2_NAME,P2_THEME,P2_THEME_DETAIL_LEVEL:SHOW_DB_OBJECT,DB.OBJECT,100,,demo_orders,demo_order_items,demo_product_info,,)
  - APEX Applikation erstellt HTML der Metadaten (*flexible Templating Engine, das HTML kann beliebig angepasst werden*)
  - Das generierte HTML wird in das aktuelle Dokument importiert und ersetzt das Markup
  - In MS Word: <H1> wird übersetzt in Überschrift1 **des aktuellen Templates!**

## 2.2 Autorisierung

Autorisierungen definieren, wer in der Applikation zu welchen Aktionen berechtigt ist. Letztlich bilden diese technisch das Berechtigungskonzept ab.

Autorisierung	Kommentar
Admin Users	Administratoren dürfen in der Applikation ALLE Daten verwalten sowie auch die Benutzer administrieren.
Edit Products	Diese Berechtigung ermöglicht einem Benutzer, den Produktkatalog zu bearbeiten.
Verify Order Ownership	Benutzer dürfen NUR ihre eigenen Aufträge bearbeiten, dies wird hier sichergestellt.

## 2.3 Build Options

Build Optionen sind Schalter, um eine Funktionalität in der Applikation konfigurativ ein- oder auszuschalten.

Build Option	Kommentar
DISABLED	This is an easy mechanism to temporarily disable APEX items (pages, regions, tabs, processes, validations, etc.). Just use the build option DISABLED and the item will not be considered by the APEX engine.

## 2.4.2 Page-6 : Product Details

Page name:	Product Details
Kommentar:	
Hilfe:	
Autorisierung:	Edit Products
Nur authentifizierte Benutzer?:	Yes

### 2.4.2.1 Region: 10-Product Details

#### Items

- **Product Id** (P6\_PRODUCT\_ID):
- **Product Name** (P6\_PRODUCT\_NAME):
- **Product Description** (P6\_PRODUCT\_DESCRIPTION):
- **Category** (P6\_CATEGORY):
- **Product Available** (P6\_PRODUCT\_AVAIL):
- **List Price** (P6\_LIST\_PRICE):
- **Product Image** (P6\_PRODUCT\_IMAGE):

#### Actions

- Button: **Cancel** (CANCEL):
- Button: **Delete** (DELETE):
- Button: **Apply Changes** (SAVE):
- Button: **Create Product** (CREATE):

## 3.3.1 Table: JDD\_DOC\_THEMES

All the different themes are stored in this table. A theme contains a set of templates for each object that a theme can render. Typically these are objects of an APEX application or a database object, but can generally be anything in the Oracle database.

### Columns

Column Name	Type	*	Comment
THE_ID	NUMBER	*	Technical ID from sequence
THE_CODE	VARCHAR2(50)		Code for this theme so that it can be referenced by an internal name with doesn't change any more. e.g. DEFAULT
THE_TITLE	VARCHAR2(100)		Title for this theme
THE_DESC	VARCHAR2(400)		Description for this theme
THE_HTML_HEADER	VARCHAR2(4000)		HTML header, this is output first before the rendered html is returned to the client.
THE_HTML_FOOTER	VARCHAR2(4000)		HTML footer, this is output after the rendered html is returned to the client.

- ▶ Das ist erst der erste Schritt, oft werden jedoch zu viele Informationen herausgeneriert.
  - Manche Regionen, dynamic Actions, Prozesse und Items existieren manchmal aus rein technischen Gründen, sollen aber nicht dokumentiert werden, da sie keine fachliche Bedeutung haben.
  - Wie können die relevanten Inhalte gekennzeichnet werden?
  
- ▶ Fokussierung auf:
  - Was sehe ich?
    - Regionen, Berichte (und Spalten), Items
  - welche Aktionen darf ich auslösen?
    - Buttons, Dynamic Actions, Links
  - und was passiert dann?
    - Evtl. Prozesse, oft reicht ein beschreibender Kommentar am Button
  - Eventuell Zustandsmodell beachten, z.B. im Workflow, Bearbeitungsstatus im Datensatz, Freigabe erfolgt oder nicht

- ▶ Dokumentation auf drei verschiedenen Ebenen:
  - Level1: Fachlich
  - Level2: Technisch-grob
  - Level3: Technisch-fein
  - Unterschiedliche Ebenen verwenden Templates mit mehr/weniger Details
  
- ▶ Tags für den APEX Sourcecode (in den Kommentaren):
  - Steuerung der Sichtbarkeit auf den jeweiligen Ebenen
  - Nie sichtbar: `#doc.level.0#`
  - Nur sichtbar auf Level <n>: `#doc.level.1#`, `#doc.level.2#`, `#doc.level.3#`
  - `#doc.exclude_children#`
    - Alle „Kinder“ werden standardmäßig ausgeblendet, nur die explizit benannten (`#doc.level.<n>#`) werden wieder eingeblendet.
    - Z.B. interaktiver Bericht auf Collection hat ca. 55 Spalten, wir verwenden jedoch nur 5: `select * from apex_collections where collection_name=,ABC'`;

► Demo:



- ▶ Unsere Erfahrungen:
  - Kommunikation der Änderungen funktioniert
  - Delta vs. Full funktioniert
  
- ▶ Schlüssel zum Erfolg:
  - Semantische Klarheit der verschiedenen Requirements-Typen unterstützt Kommunikation und Pflege der Gesamtdokumentation
  - Vorlagen, Checklisten und ein klarer Prozess sind Gold wert
  - MS Word – „Dokumente zusammenführen“ ... eine große Hilfe!  
„Änderungsverfolgung“ muss nicht vorher aktiviert sein!
  
- ▶ WordWiki
  - WordWiki Ansatz mit großem Potential, setze ich jetzt in allen Projekten ein
  
- ▶ Gesamter Ansatz Top-Down (Analyse, Design, Implementierung) wie auch Bottom-Up (existierende Applikation übernehmen, dann dokumentieren) verwendbar



Unterstützung bei der Erarbeitung der Konzepte und  
Implementierung der Lösung:

Wolfram Ditzer

Tom Fuhr

# Q&A

Dietmar Aust  
Opal-Consulting, Köln

[www.opal-consulting.de](http://www.opal-consulting.de)  
[daust.blogspot.com](http://daust.blogspot.com)  
[dietmar.aust@opal-consulting.de](mailto:dietmar.aust@opal-consulting.de)

**JDD Spreadsheet Suite**  
<http://jdd-software.com>

