



# Oracle Forms und APEX

Gerd Volberg + Jan Winkels

---

OPITZ CONSULTING Deutschland GmbH

# Agenda

1

Oracle Forms

2

APEX

3

Live-Demos

4

Known Problems

5

Resümee



# Oracle Forms

- Wie kann man Forms modernisieren, ohne alles wegzuschmeissen?
- Wie kann man ohne viel Arbeit an andere Entwicklungsumgebungen andocken?
- Solche und viele ähnliche Fragen waren die Grundlage dieses Vortrags



# Oracle Forms

- Forms und APEX miteinander zu verheiraten, funktioniert ab Forms 11g.
- Die Live-Demos arbeiten in einer VM mit
  - Forms 11g Release 2 (11.1.2.2.0)
  - DB 11g Release 2 (11.2.0.4)
  - Weblogic Server 11g Release 1 (10.3.6)
  - APEX 5 (5.0.4)
  - TomCat 8 (8.5.5)



# APEX

## ■ Region-Plugin


- Parameter und Einstellungen
- Browserabhängige Probleme
- Java Probleme

## ■ Java-Script API



# Region-Plugin

## ■ Parameter und Einstellungen

▼ Settings	
Forms Application URL	<input type="text"/>
Separate	<input type="text" value="true"/>
Width	<input type="text"/>
Height	<input type="text"/>
Look&Feel	<input type="text" value="- Select -"/> 
Modulname	<input type="text"/>

# Region-Plugin

## ■ Browserabhängige Probleme

- IE und Firefox integrieren das Java-Plugin für Forms mit unterschiedlichen HTML Tags

```
92 <EMBED SRC="" PLUGINSOURCE="http://java.sun.com/products/archive/j2se/6u12/index.html"
93     TYPE="application/x-java-applet"
94     java_codebase="" || l_server_url || '/forms/java"
95     java_code="oracle.forms.engine.Main" |
96     java_archive="frmall.jar"
97     WIDTH="" || l_width || ""
98     HEIGHT="" || l_height || ""
99     ID="" || l_modulname || ""
```



```
49 <OBJECT classid="clsid:CAFEEFAC-0016-0000-FFFF-ABCDEFEDCBA"
50     codebase="http://java.sun.com/update/1.6.0/jinstall-6-windows-i586.cab#Version=1,6,0,12"
51     WIDTH="" || l_width || ""
52     HEIGHT="" || l_height || ""
53     HSPACE="0"
54     VSPACE="0">
```



[http://www.w3schools.com/tags/tag\\_object.asp](http://www.w3schools.com/tags/tag_object.asp)

# Region-Plugin

## ■ Lösung

```
49 <OBJECT classid="clsid:CAFEEFAC-0016-0000-FFFF-ABCDEFFEDCBA"  
50     codebase="http://java.sun.com/update/1.6.0/jinstall-6-windows-i586.cab#Version=1,6,0,12"  
51     WIDTH="" || l_width || ""  
52     HEIGHT="" || l_height || ""  
53     HSPACE="0"  
54     VSPACE="0">
```



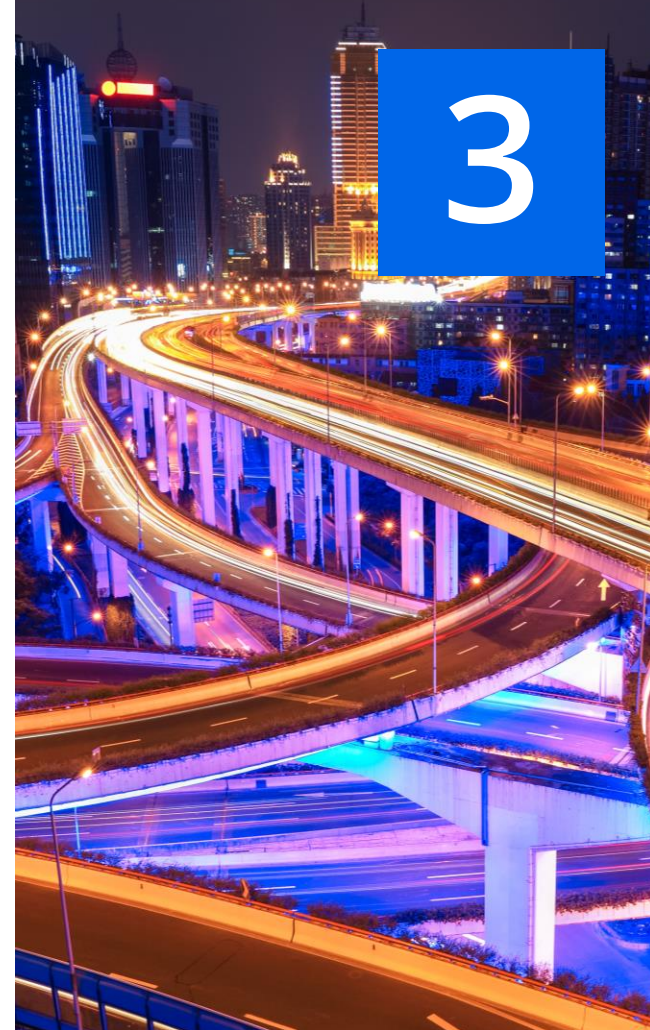
```
BrowserDetect.init();  
if (BrowserDetect.browser=="Explorer"){  
    $("object[classid='clsid:CAFEEFAC-0016-0000-FFFF-ABCDEFFEDCBA']")[0].setAttribute("id",id);  
}
```

```
34 APEX_JAVASCRIPT.ADD_ONLOAD_CODE(  
35     p_code=>'frm_apex.installObject('' || l_modulname || '' )'  
36 );
```



# Live-Demos

- Grundlagen
- APEX steuert Forms
- Forms steuert APEX
- Forms im externen Fenster



# Grundlagen - Allgemein

### Forms

Fenster

DOAG 2016

Allgemein | Daten abfragen | Daten senden

HTML Item

P3\_TEST

### APEX

Test

Event name

Event value

Raise

# Grundlagen – Raise Event

- Mittels „raiseEvent“ können Daten von APEX an Forms übergeben werden.
- Die Payload besteht aus einem Datenpaket mit Name und Wert.
  - Event\_Name => Eindeutiger Name zur Weiterverarbeitung in Forms
  - Event\_Value => Payload, die dann in Forms genutzt werden kann

# Grundlagen – Raise Event

The image shows two side-by-side panels. The left panel, titled 'Forms', displays a 'Fenster' (Window) titled 'DOAG 2016'. It has three tabs: 'Allgemein', 'Daten abfragen', and 'Daten senden'. The 'Allgemein' tab is active, showing an 'HTML Item' with the text 'P3\_TEST'. Below this, a code block is visible: 

```
*** WHEN-CUSTOM-JAVASCRIPT-EVENT ***  
Event: MY_EVENT  
Value: Hallo Forms
```

. The right panel, titled 'APEX', shows a configuration form with three input fields: 'Test' (empty), 'Event name' (containing 'MY\_EVENT'), and 'Event value' (containing 'Hallo Forms'). A 'Raise' button is located to the right of these fields. Blue arrows point from the 'Event name' and 'Event value' fields to the 'Raise' button, and from the 'Raise' button to the code block in the 'Forms' panel.

# Grundlagen – Raise Event

## ■ APEX



Dynamic Action: OnClick

```
$("#get_set")[0].raiseEvent($("#P3_EVENT_NAME").val()),$("#P3_EVENT_VALUE").val())
```

## ■ Forms

```
*** WHEN-CUSTOM-JAVASCRIPT-EVENT ***  
Event MY_EVENT  
Value: Hallo Forms
```

WHEN-CUSTOM-JAVASCRIPT-EVENT :

DECLARE

```
V_Name varchar2 (300) := :system.javascript_event_name;
```

```
V_Value varchar2 (300) := :system.javascript_event_value;
```

BEGIN

```
:DOAG.TI_Event_Value := '*** WHEN-CUSTOM-JAVASCRIPT-EVENT ***' || CHR (10) ||  
                        'Event: ' || V_Name || CHR (10) ||  
                        'Value: ' || V_Value;
```

END;

# Grundlagen – Daten abfragen (native JavaScript)

### Forms

Fenster

DOAG 2016

Allgemein | **Daten abfragen** | Daten senden

JavaScript

```
document.getElementById("P3_TEST").value
```

Returnwert

Get Hallo Forms

### APEX

Test

Event name

Event value



# Grundlagen – Daten abfragen (native JavaScript)

## ■ Forms

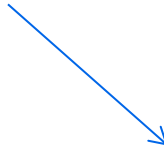
JavaScript
<code>document.getElementById("P3_TEST").value</code>
Returnwert
Get

WHEN-BUTTON-PRESSED :

BEGIN

`:DOAG.TI_Return := web.javascript_eval_function (:DOAG.TI_Get);`

END;



## ■ APEX

Test	<input type="text" value="Hallo Forms"/>
Event name	<input type="text"/>
Event value	<input type="text"/>

JavaScript	
<code>document.getElementById("P3_TEST").value</code>	
Returnwert	
Get	Hallo Forms

# Grundlagen – Daten abfragen (jQuery)

The image shows two side-by-side screenshots from an Oracle Forms and APEX environment. The left screenshot, titled 'Forms', shows a 'Fenster' (Window) titled 'DOAG 2016' with three tabs: 'Allgemein', 'Daten abfragen', and 'Daten senden'. The 'Daten abfragen' tab is active, displaying a 'JavaScript' section with the code `frm_apex.getTextboxById("P3_TEST").getValue()`. Below the code is a 'Returnwert' (Return value) section with a 'Get' button and the value 'Hallo Forms via jQuery'. The right screenshot, titled 'APEX', shows a form with a 'Test' field containing 'Hallo Forms via jQuery', two empty 'Event name' and 'Event value' fields, and a 'Raise' button.

# Grundlagen – Daten abfragen (jQuery)

## ■ Forms

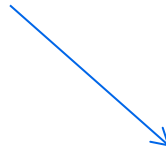
JavaScript
<code>frm_apex.getTextboxById("P3_TEST").getValue()</code>
Returnwert
Get

WHEN-BUTTON-PRESSED :

BEGIN

`:DOAG.TI_Return := web.javascript_eval_function (:DOAG.TI_Get);`

END;



## ■ APEX

Test	Hallo Forms
Event name	
Event value	

JavaScript	
<code>frm_apex.getTextboxById("P3_TEST").getValue()</code>	
Returnwert	
Get	Hallo Forms

# Grundlagen – Daten senden (native JavaScript)

The image shows the Oracle Forms Designer interface, divided into two main panels: "Forms" and "APEX".

**Forms Panel:**

- Tab: "Daten senden" (Send Data)
- Section: "Data" (Data)
- Field: "Set" (Set) with value "Hallo APEX"
- Section: "JavaScript" (JavaScript)
- Code: `document.getElementById("P3_TEST").value="Hallo APEX"`

**APEX Panel:**

- Field: "Test" (Test) with value "Hallo APEX"
- Field: "Event name" (Event name)
- Field: "Event value" (Event value)
- Button: "Raise" (Raise)

# Grundlagen – Daten senden (native JavaScript)

## ■ Forms

Data	
Set	Hallo APEX
JavaScript	
document.getElementById("P3_TEST").value="Hallo APEX"	

WHEN-BUTTON-PRESSED :

BEGIN

web.javascript\_eval\_expr (:DOAG.TI\_Set);

END;

## ■ APEX

Test	<input type="text"/>	→	Test	<input type="text" value="Hallo APEX"/>
Event name	<input type="text"/>		Event name	<input type="text"/>
Event value	<input type="text"/>		Event value	<input type="text"/>

# Grundlagen – Daten senden (jQuery)

## ■ Forms

Data	
Set	Hallo APEX

JavaScript
frm_apex.getTextboxById("P3_TEST").setValue("Hallo APEX")

WHEN-BUTTON-PRESSED :

```
BEGIN
  web.javascript_eval_expr (:DOAG.TI_Set);
END;
```

## ■ APEX

```
setValue: function(value){
  this.item.value=value;
  this.object.change();
}
```

Test

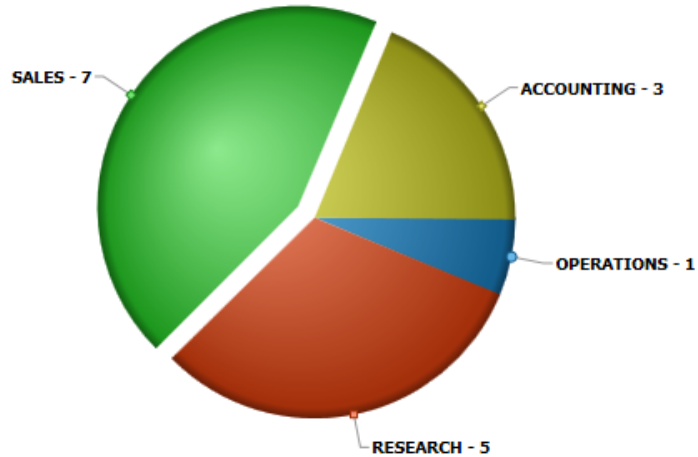


Test



# APEX steuert Forms

Departments



Employees

Fenster

DOAG 2016

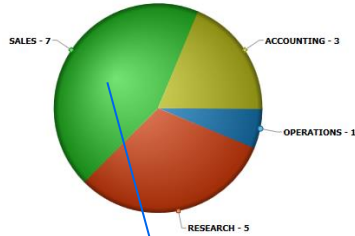
Employees

Empno	Ename	Job	Mgr	Deptno
7499	ALLEN	SALESMAN	7698	30
7521	WARD	SALESMAN	7698	30
7654	MARTIN	SALESMAN	7698	30
7698	BLAKE	MANAGER	7839	30
7844	TURNER	SALESMAN	7698	30
7900	JAMES	CLERK	7698	30

\*\*\* WHEN-CUSTOM-JAVASCRIPT-EVENT \*\*\*  
Event: QUERY\_EMP  
Value: 30

# APEX steuert Forms

## ■ APEX



## ■ Forms

Empno	Ename	Job	Mgr	Deptno
7499	ALLEN	SALESMAN	7698	30
7521	WARD	SALESMAN	7698	30
7654	MARTIN	SALESMAN	7698	30
7698	BLAKE	MANAGER	7839	30
7844	TURNER	SALESMAN	7698	30
7900	JAMES	CLERK	7698	30

SQL-Query auf Chart-Region

```
select 'javascript:${"#emp_dept"}[0].raiseEvent(''QUERY_EMP'', ''||d.deptno||'')' LINK,  
       d.dname          LABEL,  
       Count(e.empno) ANZ_EMP  
from emp e  
join dept d on (e.deptno=d.deptno)  
group by 'javascript:${"#emp_dept"}[0].raiseEvent(''QUERY_EMP'', ''||d.deptno||'')',  
         dname;
```

WHEN-CUSTOM-JAVASCRIPT-EVENT :

DECLARE

```
V_Event_Name  varchar2 (300) := :system.javascript_event_name;  
V_Event_Value varchar2 (300) := :system.javascript_event_value;
```

BEGIN

```
IF V_Event_Name = 'QUERY_EMP' THEN  
  :Parameter.PA_DEPTNO := V_Event_Value;  
  Go_Block ('EMP');  
  Execute_Query;
```

```
END IF;
```

END;

# Forms steuert APEX

## Departments

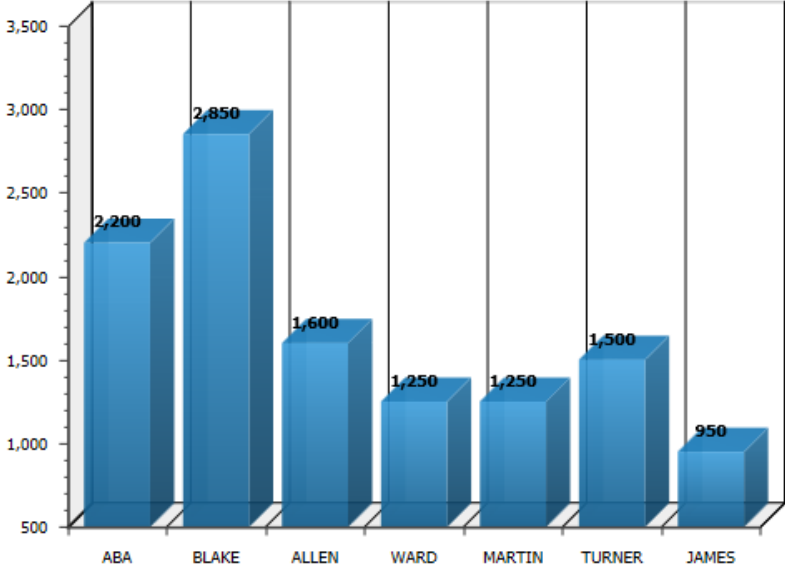
Fenster

DOAG 2016

Deptno	Department	Location
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

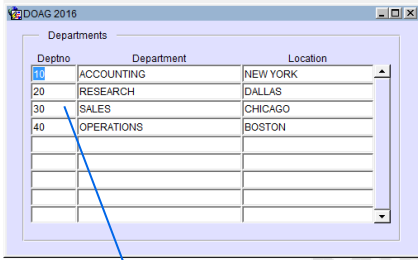
## Salaries

Deptno SALES



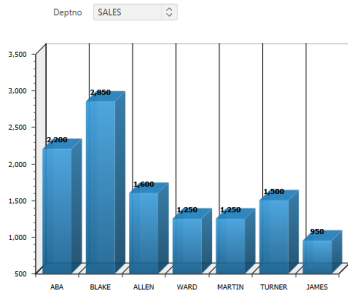
# Forms steuert APEX

## ■ Forms



Deptno	Department	Location
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

## ■ APEX



WHEN-BUTTON-PRESSED :

BEGIN

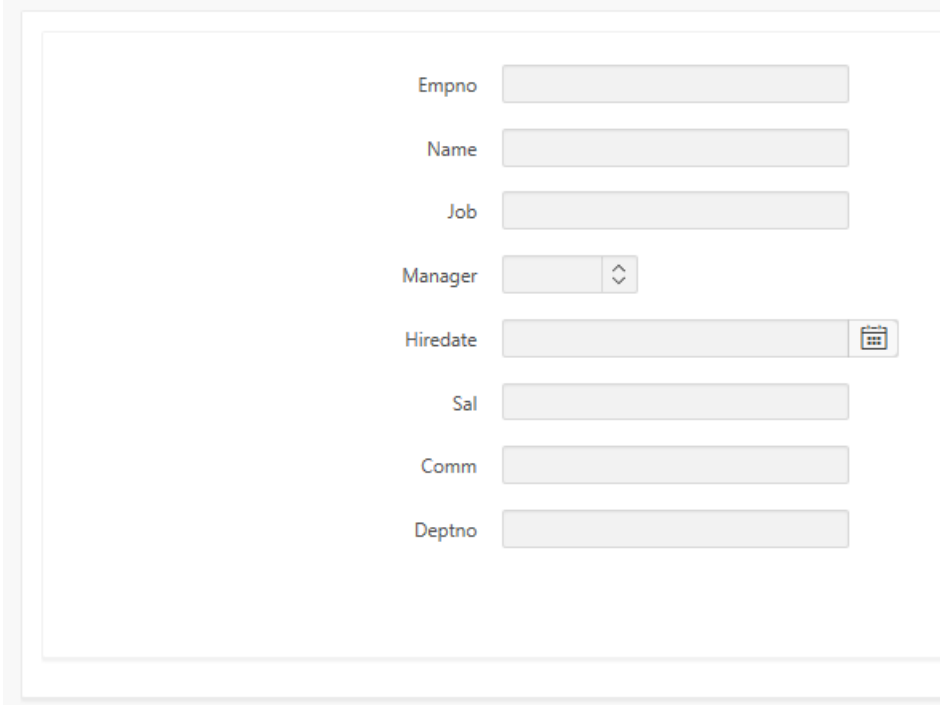
```
web.javascript_eval_expr  
('frm_apex.getSelectListById(''P6_DEPTNO'').setValue(' || :DEPT.DEPTNO || '));  
END;
```

Userdefined JavaScript-Function

```
setValue: function(value){  
  apex.item(this.item.id).setValue(value);  
}
```

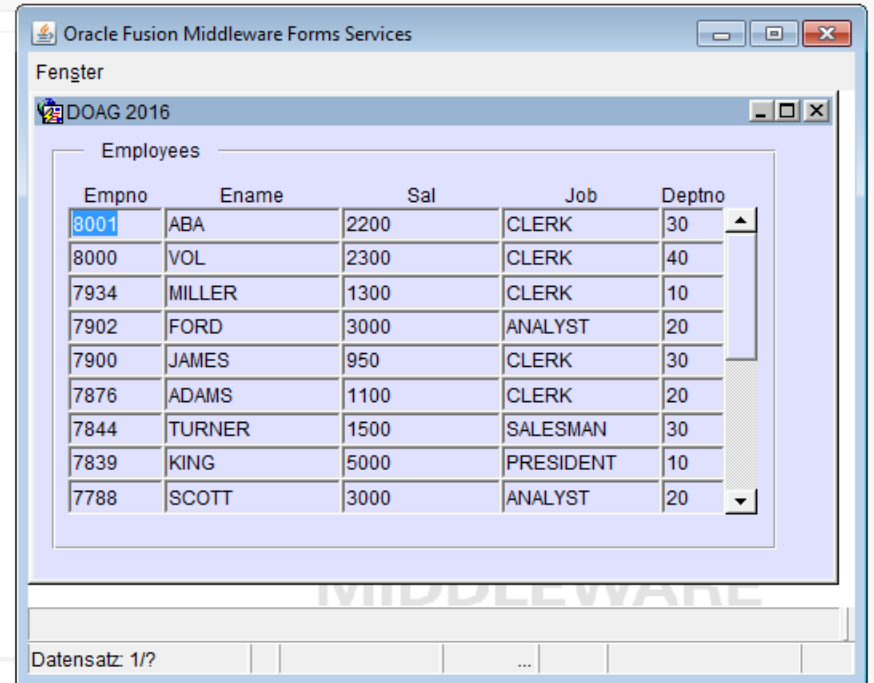
# APEX / Forms-Synchronisation

## APEX-Formular



The image shows a screenshot of an APEX form layout. It consists of several input fields arranged vertically on the left side of a light gray container. The fields are labeled as follows: Empno, Name, Job, Manager, Hiredate, Sal, Comm, and Deptno. The Manager field includes a small dropdown arrow icon. The Hiredate field includes a calendar icon. The form is designed to capture employee information.

## Forms im „separateFrame“



The image shows a screenshot of a window titled "Oracle Fusion Middleware Forms Services". Inside the window, there is a sub-window titled "Fenster" which contains a table titled "Employees". The table has five columns: Empno, Ename, Sal, Job, and Deptno. The first row is highlighted in blue. Below the table, there is a status bar that reads "Datensatz: 1/?".

Empno	Ename	Sal	Job	Deptno
8001	ABA	2200	CLERK	30
8000	VOL	2300	CLERK	40
7934	MILLER	1300	CLERK	10
7902	FORD	3000	ANALYST	20
7900	JAMES	950	CLERK	30
7876	ADAMS	1100	CLERK	20
7844	TURNER	1500	SALESMAN	30
7839	KING	5000	PRESIDENT	10
7788	SCOTT	3000	ANALYST	20

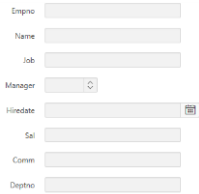
# APEX / Forms-Synchronisation

- Formsmasken, die mit APEX zusammen auf einer HTML-Seite laufen, können auch im eigenen Fenster arbeiten.
- Mit „`separateFrame=true`“ lassen sich Legacy-Forms sehr schnell mit APEX-Funktionalität anreichern.
- Die Live-Demo zeigt, wie in einem APEX-Formular Daten erfasst und danach in einem externen Fenster in Forms angezeigt werden. Nach jedem Speichervorgang wird Forms einen Auto-Query starten.



# APEX / Forms-Synchronisation

## ■ APEX



Empno  
Name  
Job  
Manager  
Hiredate  
Sal  
Comm  
Deptno

Save

Dynamic Action: OnClick

```
$("#emp")[0].raiseEvent("QUERY_EMP","EMP");
```

## ■ Forms

Empno	Ename	Sal	Job	Deptno
8001	ABA	2200	CLERK	30
8000	VOL	2300	CLERK	40
7934	MILLER	1300	CLERK	10
7902	FORD	3000	ANALYST	20
7900	JAMES	950	CLERK	30
7876	ADAMS	1100	CLERK	20
7844	TURNER	1500	SALESMAN	30
7839	KING	5000	PRESIDENT	10
7788	SCOTT	3000	ANALYST	20

WHEN-CUSTOM-JAVASCRIPT-EVENT :

DECLARE

```
V_Event_Name varchar2 (300) := :system.javascript_event_name;  
V_Event_Value varchar2 (300) := :system.javascript_event_value;
```

BEGIN

```
IF V_Event_Name = 'QUERY_EMP' THEN  
  Go_Block (V_Event_Value);  
  Execute_Query;
```

```
END IF;
```

END;

# Known Problems

- Unterschiedliche HTML-Templates
- Forms und JavaScript-Aufrufe
- Java

4



# Herausforderungen + Known Problems

## ■ Unterschiedliche HTML-Templates

- IE und Firefox reagieren unterschiedlich auf die Java-Plugin-Definition, die im Template genutzt wird.
- Um jQuery in APEX sinnvoll nutzen zu können, musste eine Browserweiche genutzt werden.

## ■ Forms und JavaScript-Aufrufe

- Werden in einer Formsmaske mehrere JavaScript-Befehle nacheinander an APEX übergeben, führt dies zu merkwürdigem Verhalten
- Manchmal läuft Forms in das bekannte Focus-Lost-Problem
- Ab und zu wird der Cursor in andere Felder zurückgesetzt und der zuvor gestartete Sourcecode erneut ausgeführt (kann sogar zu Endlosschleifen führen)

# Herausforderungen + Known Problems

## ■ Java

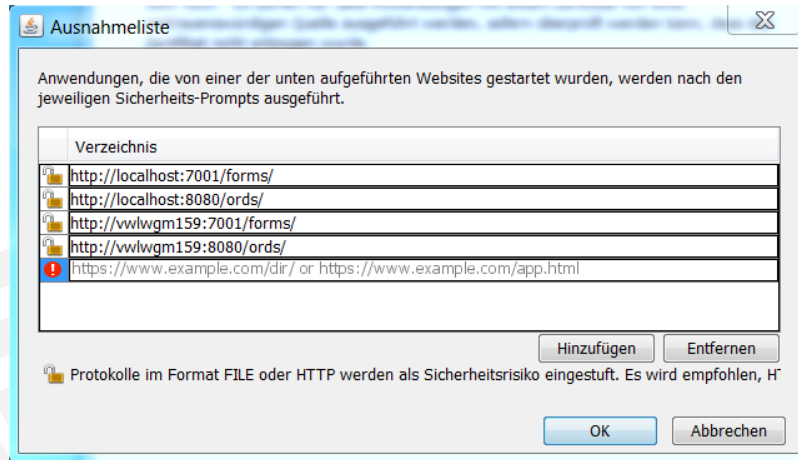
❌ SCRIPT16389: Liveconnect call for Applet ID 1 is not allowed in this JVM instance  
Datei: f, Zeile: 318, Spalte: 236



❌ ▶ Error: Error calling method on NPObjekt!



## ■ Lösung



Offene Frage: Gibt es eine serverseitige Lösung für das Problem?

# Resümee

- Viele Probleme können durch Workarounds behoben werden.
- jQuery ermöglicht eine einfache Integration in APEX.

~~Forms vs. APEX ?~~

Forms + APEX !

5





Q & A



**Gerd Volberg**

Solution Architect

Kirchstraße 6  
51647 Gummersbach

[gerd.volberg@opitz-consulting.com](mailto:gerd.volberg@opitz-consulting.com)

+49 2261 6001 0



**Jan Winkels**

Consultant

Kirchstraße 6  
51647 Gummersbach

[jan.winkels@opitz-consulting.com](mailto:jan.winkels@opitz-consulting.com)

+49 2261 6001 0



[WWW.OPITZ-CONSULTING.COM](http://WWW.OPITZ-CONSULTING.COM)



[@OC\\_WIRE](https://twitter.com/OC_WIRE)



[OPITZCONSULTING](https://www.youtube.com/OPITZCONSULTING)



[opitzconsulting](https://www.linkedin.com/company/opitzconsulting)