

# Erfahrung nach einem Jahr Fusion Middleware 12c

Jan-Peter Timmermann

JPT-Consulting

Wedel

## Schlüsselworte

Weblogic, Forms, Reports, Administration.

## Einleitung

Ein Jahr Oracle Fusion Middleware 12c. Was bringt die neue Administration? Wie komme ich an die Konfigurationen und wie kann ich das per Skript administrieren? Hier möchte ich aufzeigen, was sich zwischen der Version 11 und 12c verändert hat. Ebenso wird die Repository Situation betrachtet. Wie gehe ich damit um, was ergibt Sinn, und gibt es die Möglichkeit das Repository zu verschieben?

## Ach wie war das in Forms 11g noch schön

Eigentlich sollte es ja heißen: Alles neu macht der Mai, aber das Release Oracle Forms / Reports 12c kam im Oktober 2015 auf den Markt. Mit viel Interesse und Spannung wurde auf das Release gewartet. Ich hatte das große Glück bereits am Erscheinungsdatum einen Download sowie eine Installation zu tätigen. Über diese Erfahrungen sowie die Reaktionen der Kunden möchte ich an dieser Stelle berichten. Die erste Produktive Installation konnte ich bereits im Dezember 2015 verbuchen. Es war nur eine „einfache“ Anwendung, aber es brachte sehr viel an Neuerungen an den Tag. Einen Teil hatte ich bereits in meinem Vortrag vom letztem Jahr erwähnt.

## Die Zertifizierung

In der Version Forms 11gR2 waren 32 und 64 Bit Varianten vorhanden. Es gibt keine 32-Bit Versionen mehr und das JDK muss auf den bekannten Plattformen (Windows, Linux) mindestens 1.8.0\_51+ sein. Es geht nicht mehr mit einem JDK klein 1.8. Das hat evtl. Auswirkungen auf Anwendungen, die mit auf diesen Maschinen laufen. Auch für die Entwickler hat sich an dieser Stelle etwas geändert. Der Forms Developer ist nur noch in der 64-Bit Variante vorhanden.

Bei den Browsern müssen wir jetzt darauf achten, dass nur noch folgende Browser unterstützt werden.

Browser	Version
Microsoft Internet Explorer	11.*
Google Chrome	45+
Mozilla Firefox	31+
Apple Safari	8.*

Im Laufe der nächsten Jahre wird sich das ändern, da viele Browser das Java-Plugin nicht mehr unterstützen werden.

Bei den Datenbanken wird alles ab der Version 11.2.0.4 aufwärts unterstützt.

Die Anforderungen an das Betriebssystem haben sich erhöht.

Operating System	Minimum Physical Memory Required	Minimum Available Memory Required
Linux	4 GB	8 GB
UNIX	4 GB	8 GB
Windows	4 GB	8 GB

Auch hier haben sich die Werte etwas erhöht, war die Empfehlung bei 11g noch 4 GB sind es jetzt bereits 8GB.

## RCU

Was neu hinzugekommen ist die zwingende Voraussetzung einer Datenbank für das Repository. Für das Repository dürfen die Datenbanken in der Version 11.2.0.4 aufwärts genutzt werden.

Die Datenbank sollte UTF8 sowie mit der ORACLE JVM installiert sein.

Character set is AL32UTF8.

If your database does not use the AL32UTF8 character set, you will see the following warning when running RCU: The database you are connecting is with non-AL32UTF8 character set. Oracle strongly recommends using AL32UTF8 as the database character set.

You can ignore this warning and continue using RCU.

The following database parameters are set as specified:

Parameter	Minimum Required Value
SHARED_POOL_SIZE	147,456 KB
SGA_MAX_SIZE	147,456 KB
DB_BLOCK_SIZE	8 KB
session_cached_cursors	100
processes	500
open_cursors	800
db_files	600

Diese Änderungen sollte man machen.

Auf den unterschiedlichen Linux Systemen hat sich nicht viel geändert. Einzig Oracle Linux 7 benötigt anstatt openmotif jetzt:

motif-2.3.4-7 for x86\_64 motif-devel-2.3.4-7 for x86\_64

Was in MOS auch zu diversen Informationen führt.

### Die Architektur

Es hat sich etwas in der Architektur geändert. Wenn wir uns 11gR2 anschauen, dann wird uns folgendes Bild vertraut sein.

Middleware Home = /u00/oracle/Middleware: 11gR2

FRHome	Oracle Software Home
Frinst	Oracle Instance Verzeichnis
Log	Logdateien für die Installation
Modules	
Oracle_common	
User_projects	
Utils	
Wlserver_10.3	Weblogic Software

/u03/oracle/Middleware12: 12c

```
[oracle@aura Middleware12]$ ls
OPatch                coherence              frcommon              jdbc                  ohs                   plssql                root.sh               webgate
ReportsBridgeComponent crs                    guicommon             jdev                 oraInst.loc          plugins              slax                  wlserver
ReportsServerComponent css                    has                   jlib                 oracle_common        precomp              sqlplus              xdk
ReportsToolsComponent dbs                     hs                     ldap                  oracle               procbuilder          srvm                  tools
bin                    domain-registry.xml  install               lib                   ord                   rdbms                ucp
browser                em                     instantclient         network              oui                   relnotes             reports               user_projects
cfgtoollogs            forms                  inventory              nls                   perl
```

Wo ist denn jetzt mein Oracle Home ? Wo ist mein Instance Verzeichnis.

Oracle hat an dieser Stelle mal wieder alles ein bisschen verändert.

So befinden wir uns mit dem Middleware Verzeichnis direkt in dem Oracle Home. Die Instance hat sich auch verschoben. Die ist nun da wo sie eigentlich da wo sie hingehört. Innerhalb der Domäne.

### Erstellen des http – Servers mit einem Skript

Um einen http – Server einzurichten gibt es jetzt im WLST – Bereich einen neuen Befehl. Dieser wird dazu genutzt um die Instance zu errichten.

Aufruf von WLST und anmelden an dem Admin Server, dann ausführen :

```
ohs_createInstance(instanceName = 'ohs1', machine = machines.getName(),
serverName = 'OHS1')
```

Das ganze geht natürlich auch mit der EM-Console

Dazu muss ich mich gegen die EM-Console verbinden um dann in den Administrativen Bereich zu wechseln:



Abb: 1 Create / Delete OHS

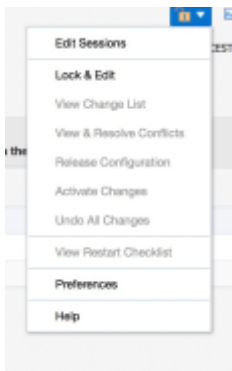


Abb: 2 Lock and Edit

Bevor mit der Verarbeitung begonnen werden kann muss die Oberfläche in den Edit Modus versetzt werden. Danach besteht dann die Möglichkeit einen neuen OHS zu erstellen.

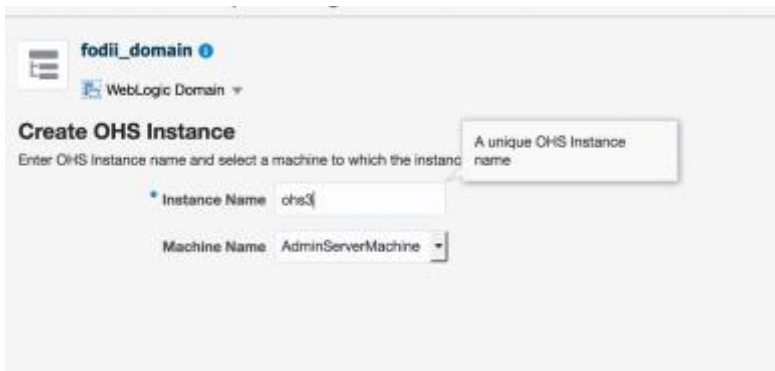


Abb: 3 Create OHS3

Es muss der Name und die Maschine auf der er deployed werden soll eingerichtet werden. Anschließend wird die Konfiguration aktiviert:

Die Port Belegung versucht einen freien Port zu nutzen:

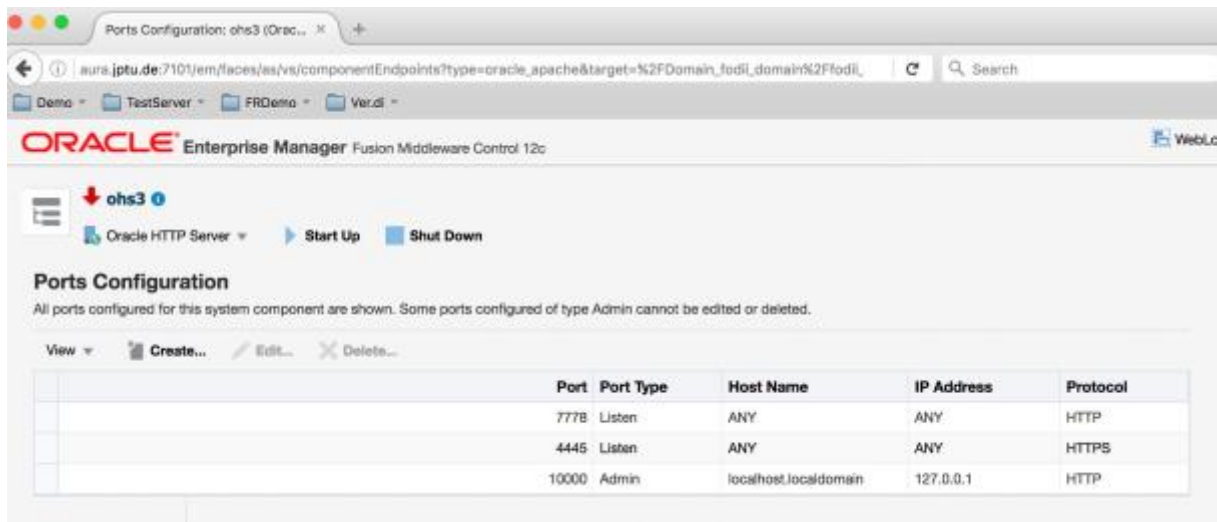


Abb: 4 Port Belegung

Diesen kann ich über diese Oberfläche verändern und dann den HTTP Server starten.



Abb: 5 nicht aktiviert

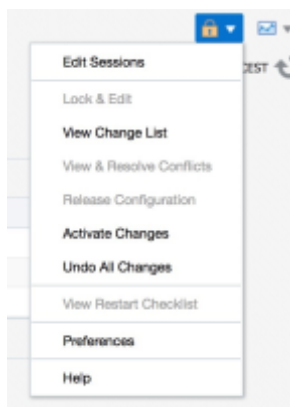


Abb: 6 aktivieren der Änderungen

Danach lässt er sich auch starten.

Zum starten und stoppen kann man auch das Skript nutzen:

```
./startComponent.sh ohs3
oder
./stopComponent.sh ohs3
```

dazu muss aber der Nodemanager laufen.  
Um jetzt eine Anbindung an die Weblogic zu bekommen muss hier noch in der mod\_wl\_ohs.conf der richtige Eintrag gemacht werden.

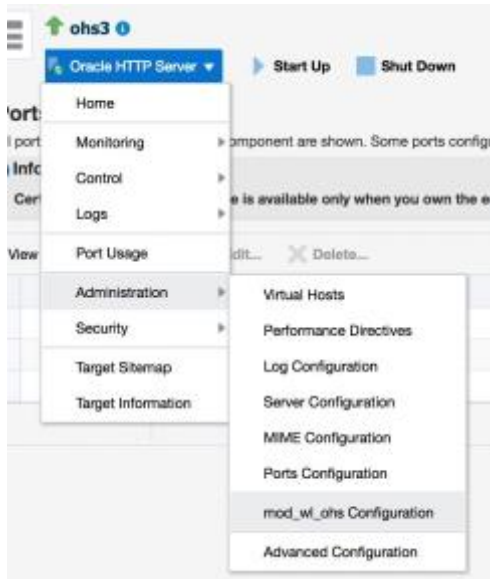


Abb: 7 mod\_wl\_ohs

Dazu kann ich über die Oberfläche die Konfiguration direkt ansprechen.

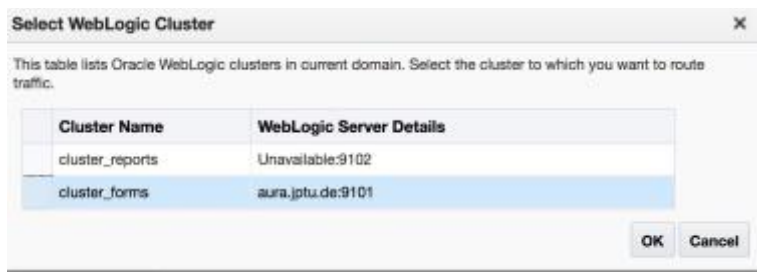


Abb: 8 Cluster Ansprache

Ich kann hier über den Wizard mir den cluster auswählen um dann anschließend die Location (/forms) anzusprechen.

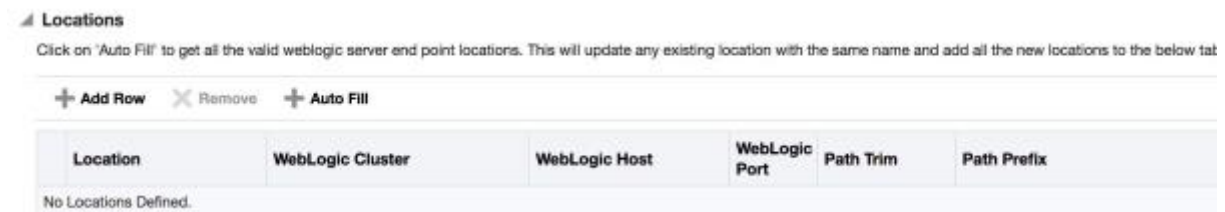


Abb: 9 Auswahl Location

Ich muss allerdings den kompletten Location Namen eintragen.

Ich habe jetzt mit Absicht einmal einen fehlerhaften Eintrag gemacht. Das wird leider nicht überprüft und so eingetragen wie ich es erfasst habe.

```
<IfModule weblogic_module>
#       WebLogicHost <WEBLOGIC_HOST>
#       WebLogicPort <WEBLOGIC_PORT>
#       MatchExpression *.jsp
<Location /forms>
  SetHandler weblogic-handler
  WebLogicHost aura.jptu.de
  WebLogicPort 9101
</Location>
DynamicServerList Off
WebLogicCluster aura.jptu.de:9101
</IfModule>
```

Man sollte hier also sehr vorsichtig sein, was man einträgt.

Das ganze kann ich natürlich auch direkt in die notwendigen Konfigurations-Dateien eintragen.

### Konfiguration Reports Server

Ebenso wie man einen HTTP-Server erstellen kann geht es natürlich auch mit einem Standalone Report Server.

Hierzu gibt es allerdings keine Oberfläche sondern den muss ich mit WLST erstellen.

Dazu gibt es im WLST Umfeld folgenden neuen Befehl:

```
createReportsServerInstance(instanceName=neuerReport,machine=machines.getName())
```

allerdings kann ich den Report Server nur aus Forms heraus ansprechen, wenn ich die Reports Tool Komponenten mit installiert habe.

Dazu muss ich als erstes eine Reports Tool Komponente erstellen.

```
CreateReportsToolsInstance(instanceName=v_reports_tool,machine=machines.getName())
```

mit diesem Befehl werden die notwendigen Verzeichnisse erstellt.

Leider gibt es jetzt keine Möglichkeit über die EM-Console geführt diesen Report Server zu konfigurieren. Dazu muss ich entweder ein Skript benutzen oder aber über den MBean-Browser die einzelnen Komponenten bearbeiten.

Der erste Schritt ist in jedem Falle die „SecurityId“ zu ändern, da ansonsten bei jeder Anfrage an den Report Server eine Sicherheitsabfrage erscheint. Okay ich könnte in der rwserver.conf ein Username/Passwort hinterlegen, müsste dieses aber bei jedem Forms Aufruf mit geben. Alternative ist also die SecurityId auf „null“ zu setzen.

Das kann ich zum einem mit WLST machen:

```
mdsBean = ObjectName('oracle.reports.serverconfig:ReportsServer=rwserver-
'+KomponentName+',name=rwEngrwJaznSec,type=ReportsServer.Job')
mbs.setAttribute(mdsBean, javax.management.Attribute("SecurityId", Value))
```

oder über den Browser:



Abb: 10 Aufruf MBean Browser

Dann muss ich den Reports Server suchen.



Abb: 11 ändern Security ID

Danach wird die Security Abfrage nicht mehr erscheinen.

Ebenso muss der Webzugriff erlaubt werden. Das geht zum einen über die Oberfläche mit dem System MBean-Browser

The screenshot shows the Oracle JConsole interface. On the left, the MBean browser displays a tree structure for 'Server: WLS\_REPORTS' under 'Application: reports'. The 'rwservlet' component is selected. On the right, the 'Show MBean Information' dialog is open, showing the 'Attributes' tab for the 'WebcommandAccess' MBean. The dialog contains a table with the following data:

Name	Description	Access	Value
12 eventTypes	All the event's types emitted by this MBean.	RI	java.util
13 helpurl	Help URL.	RW	
14 imageurl	Image URL.	RW	
15 Inprocess	Inprocess	RW	yes
16 objectName	The MBean's unique JMX name	RI	oracle.r
17 Oldconnection	Get OldConnection	RI	(No value)
18 ReadOnly	If true, it indicates that this MBean is a read only MBean.	RI	false
19 reloadKeymap	Reload KeyMap	RW	
20 RestartNeeded	Indicates whether a restart is needed.	RI	false
21 Server	Server	RW	rep_wls
22 serverKeymap	Server Key map	RW	
23 SingleSignon	SingleSignon	RW	no
24 sysauth	Sys auth	RW	
25 SystemMBean	If true, it indicates that this MBean is a System MBean.	RI	false
26 Visible	If true, it indicates that this MBean is visible to current user.	RI	true
27 WebcommandAccess	Web Command Access Level	RW	

Abb: 12 WebcommandAccess

Da kann man den richtigen Eintrag tätigen wie in 11g :

Oder per Skript :

```

mdsBean = ObjectName(Object);
minfo = mbs.getMBeanInfo(mdsBean)
objs = jarray.array([java.lang.String(Level)], java.lang.Object)
strs = jarray.array(['java.lang.String'], java.lang.String)
repositories = mbs.invoke(mdsBean,"addWebcommandAccess",objs,strs)
print
"=====";
print " WLS REPORTS wird neu gestartet"
print
"=====";
shutdown(name='WLS_REPORTS',entityType='Server',force='true');
start('WLS_REPORTS')

```

danach ist der Zugriff auf die Oberfläche des Report Servers erlaubt. Das kann ich natürlich wie gehabt immer noch in die rwservlet.properties Datei eintragen. In jedem Falle muss der WLS\_REPORTS anschließend neu gestartet werden.

Neue Umgebungen einrichten. Geht ebenso mit Skript wie über den System MBean-Browser

```

mdsBean = ObjectName('oracle.reports.serverconfig:type=ReportsServer,name=rwserver-
'+KomponentName);
minfo = mbs.getMBeanInfo(mdsBean)
objs = jarray.array([java.lang.String(EnvID)], java.lang.Object)
strs = jarray.array(['java.lang.String'], java.lang.String)
repositories = mbs.invoke(mdsBean,"addEnvironment",objs,strs)
save()
activate()

```



In der Praxis hat sich herausgestellt, dass eigentlich so ziemlich jeder Kunde der Oracle Forms nutzt auch Oracle Reports im Einsatz hat. Dies alles über die Oberfläche zu administrieren ist sehr aufwendig. Aus diesem Grunde habe ich mir für alle diese Tätigkeiten ein paar Skripte erstellt, die das in immer derselben Reihenfolge für mich erledigen. Dazu kann man dann eine Vorlagen Konfigurations Datei nutzen.

**Anmerkungen:**

In den vorherigen Versionen gab es immer wieder Patche, die einfach so eingespielt werden konnte. Das ist in 12c bisher nicht der Fall. Alle Patche erfordern ein neues Oracle Home.

Aus diesem Grunde sollte man das User\_project (Domain Verzeichnis) aus dem Oracle Home heraus nehmen, ansonsten läuft man Gefahr das die Domain in einem altem Oracle Home Verzeichnis liegt welches nicht mehr genutzt wird.

**Kontaktadresse:**

Jan-Peter Timmermann  
JPT-Consulting  
Bekstr.4c  
D-22880 Wedel

Telefon: +49 (0) 172-215 10 43  
Fax: +49 (0) 4103 180 371  
E-Mail: [jan.timmermann@jptu.de](mailto:jan.timmermann@jptu.de)  
Internet: [timmis.me](http://timmis.me)