

Apache + ORDS Quickstart für Produktionsumgebungen

Dietmar Aust
Opal-Consulting
Köln

Schlüsselworte:

Oracle APEX, Oracle Rest Data Services, ORDS

Hintergrund – die wesentlichen Use Cases

Um Oracle Application Express nutzen zu können, benötigen wir ein Kommunikations-Gateway, welches die Http-Anfragen des Browsers letztlich in PL/SQL-Aufrufe in der Datenbank übersetzt. Obwohl es unterschiedliche Möglichkeiten gibt, empfiehlt Oracle insbesondere für Produktionssysteme die Nutzung der Oracle Rest Data Services (ORDS, ehemals Oracle APEX Listener).

Wie gehen wir konkret vor, welche Schritte sind notwendig? Irgendwie scheint es doch etwas kompliziert, muss es aber nicht sein. Wir gehen in diesem Vortrag Schritt für Schritt die wesentlichen Use Cases für eine Nutzung in Produktionssystemen durch, unter anderem:

- Einrichtung von https,
- Caching der Bilder im Filesystem,
- Einrichtung von virtuellen Verzeichnissen,
- Konfiguration von AJP mit ORDS,
- Installation im Tomcat,
- Rewrite-Regeln für sprechende URLs, etc.

In diesem Vortrag stelle ich eine Architektur für den performanten und sicheren Betrieb von Oracle APEX auf Basis der Oracle Rest Data Services vor. Diese Architektur kann bestimmt in 80% der Umgebungen direkt eingesetzt werden.

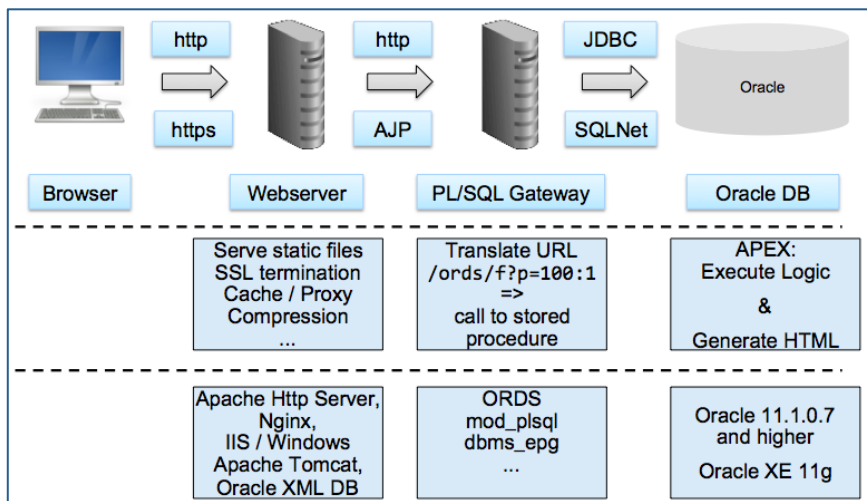
Darüber hinaus ist diese Architektur auch grundsätzlich so erweiterbar, dass viele weitere Anforderungen auf dieser Plattform realisiert werden können. So kann natürlich der Apache Http Server durchaus noch weiter gehärtet werden. Die Integration mit dem Active Directory im Rahmen eines Single-Sign-Ons kann z.B. über `mod_auth_kerb` realisiert werden. Da die Architektur stark auf Standardkomponenten aufsetzt (Apache Http Server / Apache Tomcat), sind die erforderlichen Qualifikationen für den Betrieb dieser Lösung ja oftmals bereits intern vorhanden.

Ich zeige Schritt für Schritt, wie die unterschiedlichen Schichten zu installieren und konfigurieren sind. Je nachdem wie viel Funktionalität man benötigt, reicht es auch aus, nicht alle Schichten zu implementieren und kann dementsprechend früher in der Konfiguration aufhören.

Die Architektur

Wenn wir den Betrieb einer Oracle APEX Umgebung betrachten, dann benötigen wir aus logischer Sicht folgende Komponenten:

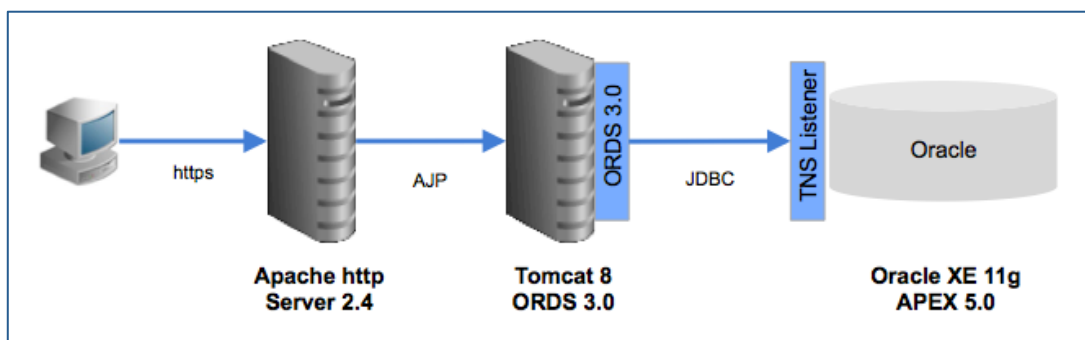
- **Client Browser:** Der Anwender benutzt einen Browser, um die mit APEX erzeugten Seiten anzuzeigen. Dieser kommuniziert über http bzw. https mit einem Webserver.
- **Webserver:** Dieser Webserver kann statische Dateien (z.B. Bilder, CSS-Dateien, Javascript-Dateien) direkt an den Browser zurückliefern. Dieser kann eventuell sogar die Inhalte cachen oder auch on-the-fly komprimieren. Er leitet aber auch Anfragen an das PL/SQL Gateway weiter.
- **PL/SQL Gateway:** Das PL/SQL Gateway übersetzt eine URL des Browsers (z.B. <http://server.domain:port/apex/f?p=100:1>) in einen entsprechenden Aufruf in der Datenbank (in diesem Fall wird die Stored Procedure **f** aufgerufen und als Parameter **p** wird der Wert „100:1“ übergeben).
- **Oracle Datenbank:** Letztlich wird also eine Stored Procedure in der Datenbank aufgerufen. In unserem Szenario wird sich diese Prozedur im Datenbank-Schema APEX_050000 befinden, da dort alle Tabellen und Prozeduren des Frameworks Oracle Application Express liegen.



Für jede dieser logischen Komponenten haben wir jeweils die Möglichkeit, unterschiedliche Softwarekomponenten zu verwenden.

Im Rahmen der Präsentation verwende ich eine virtuelle Maschine und zeige die Umsetzung anhand der folgenden konkreten Komponenten:

- Windows 7 32 bit (virtual machine)
- Apache 2.4
- Tomcat 8
- ORDS 3.0
- Oracle XE 11g
- APEX 5.0.0



Der Übergang von ORDS 2.x auf ORDS 3.x

Zu Beginn waren die Oracle Rest Data Services (ORDS) nur zur Unterstützung von Oracle APEX entwickelt worden, daher war der Name auch ursprünglich „Oracle APEX Listener“ (Version 1.x).

Mit der Version 2.x wurde dann die REST Unterstützung eingeführt, die mit Version 3.x jetzt noch weiter ausgebaut wurde. Um diese REST-Webservices zu verwenden, mussten zwei Datenbank-User konfiguriert werden, diese hatten dann Zugriff auf das Metadaten-Repository von APEX:

- APEX_LISTENER
- APEX_REST_PUBLIC_USER

Die Webservices wurden dann letztlich im SQL –Workshop von Oracle APEX definiert. Es gab also eine klare Abhängigkeit zu Oracle Application Express.

Leider befinden wir uns aktuell in einer Übergangsphase. Mit der Version 3.x wurde diese Abhängigkeit zu Oracle APEX (*leider nicht ganz vollständig*) aufgelöst. ORDS 3.x verwendet nun eigene Schemata, die eingerichtet werden müssen:

- ORDS_METADATA
- ORDS_PUBLIC_USER

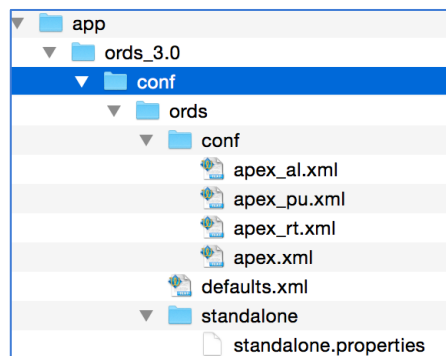
Damit Oracle APEX 5.0 korrekt funktioniert, sind jetzt leider alle Benutzer notwendig und diese müssen auch korrekt konfiguriert werden.

Oracle Rest Data Services 3.0

Wir beginnen die Installation mit dem Download des ORDS (<http://www.oracle.com/technetwork/developer-tools/rest-data-services/downloads/index.html>). Anschließend wird er über die Kommandozeile konfiguriert:

```
java -jar ords.war
```

Im Rahmen der Installation sehen wir uns die Struktur der Konfigurationsdateien an, diese definieren jeweils die Connection Pools für die unterschiedlichen Oracle - Benutzer:



Da die J2EE Applikation `ords.war` auch einen embedded Jetty J2EE Container enthält, können wir ORDS bereits jetzt auf der Kommandozeile starten.

Deployment im Apache Tomcat

Für den Betrieb einer Produktionsumgebung ist es dennoch erforderlich, eine reguläre J2EE Umgebung (Apache Tomcat, Glassfish bzw. Weblogic) zu verwenden. Hier installieren wir einen Apache Tomcat in der Version 8.0 und konfigurieren dort ebenfalls die J2EE Applikation `ords.war`.

Der Tomcat wird nach einem initialen Test dann abgesichert, so dass nur noch über Localhost auf diesen Dienst zugegriffen werden kann und nicht mehr von außerhalb.

Konfiguration des Apache Http Servers

Im nächsten Schritt konfigurieren wir den Apache Http Server, so dass die für APEX benötigten Bilder, CSS, JS (also das /i/ - Verzeichnis) über die URL /i/ zu erreichen ist. Außerdem konfigurieren wir einen Proxy-Zugang, so dass über den URL-Bestandteil /ords/ diese Anfragen dann über das ajp – Protokoll an den Tomcat „durchgereicht“ werden:

```
ProxyPass /ords ajp://localhost:8009/ords timeout=600
ProxyPassReverse /ords ajp://localhost:8009/ords
```

Weitere Anwendungsfälle

Im Rest des Vortrages diskutieren wir ebenfalls noch folgende Anwendungsfälle an:

1. Das Passwort einer der Verbindungen im Connection Pool muss geändert werden
2. Die Datenbank zieht auf einen anderen Server um
3. Analyse der Logfiles im Apache Http Server sowie im Apache Tomcat
4. Performance Monitoring und Dimensionierung des Connection Pools
5. SSO mit MS Windows Domänen
6. REST-style Zugriff auf Bestandteile der APEX Applikation

Fazit

In diesem Vortrag demonstriere ich Schritt für Schritt die Implementierung der Referenzarchitektur für Oracle Application Express auf Basis der Oracle Rest Data Services, eines Apache Tomcats sowie eines Apache Http Servers.

Diese Architektur wurde bereits bei vielen Kunden implementiert und ist sowohl performant als auch erweiterbar.

Kontaktadresse:

Dietmar Aust
Zum Tilmeshof 11
D-50859 Köln

Telefon: +49(0)173-5322 955
Fax: +49(0)221-17099759
E-Mail: dietmar.aust@opal-consulting.de
Internet: <http://www.opal-consulting.de>
BLOG: <http://daust.blogspot.com>
APEX-Tools: <http://www.opal-consulting.de/tools> (zur freien Nutzung und kostenlos)
APEX-Buch: <http://apex-xe-praxis.de/>