

# Cloud Computing für Entwickler leicht gemacht: APEX als Entwicklerplattform "As A Service"

Autor: Carsten Czarski, ORACLE Deutschland B.V. & Co KG

*Nahezu alle Unternehmen stehen heute vor der gleichen Herausforderung: Geschäftsprozesse ändern sich immer schneller - und so ergeben sich fast täglich neue Anforderungen für die IT, die diese Prozesse unterstützen soll. Der "klassische" Prozess der Anwendungsentwicklung kann vielfach nicht mehr Schritt halten - bis die Projektumgebung aufgesetzt ist und die Entwickler produktiv arbeiten können, haben sich die Anforderungen meist schon wieder geändert.*

*So besteht der Bedarf nach einer flexiblen Plattform, auf der neue Anwendungen schnell und unkompliziert bereitgestellt werden, bestehende einfach geändert und Entwickler eigenverantwortlich agieren können.*

*Oracle Application Express (APEX) kommt diesem Bedürfnis in geradezu idealer Weise entgegen. Der Artikel stellt vor, wie eine Entwicklungsplattform mit APEX nach den Konzepten des Cloud Computing bereitgestellt und flexibel genutzt werden kann.*

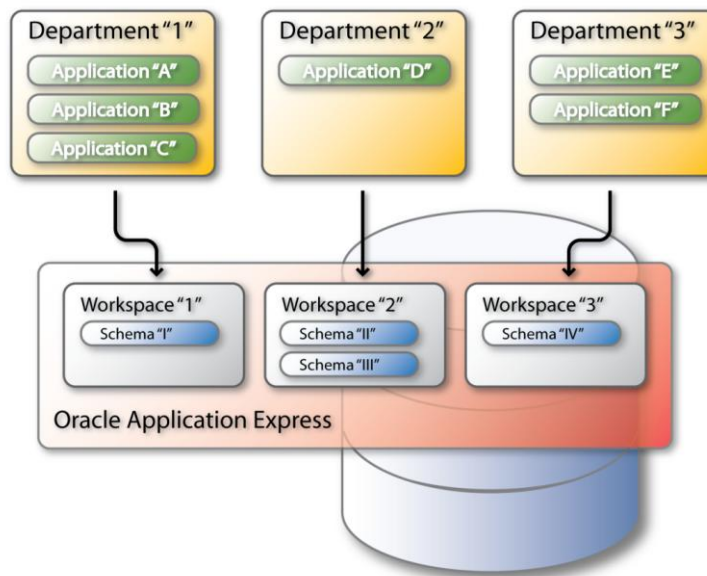


Abbildung 1: Anwendungsentwicklung in der Cloud: Application Express

## Einleitung: Cloud Computing

Von Cloud Computing spricht das amerikanische National Institute of Standards and Technology (NIST), wenn ein System folgende Kriterien erfüllt.

- Selbstbedienung
- Zugang mittels üblicher Netzwerktechnik und Geräte
- Ressourcen werden je nach Bedarf (und nicht fest) zugeordnet
- Ressourcen stehen flexibel, quasi unbegrenzt zur Verfügung
- Abrechnungsfähiger Service

Eine Entwicklungsplattform, welche diese Kriterien erfüllt und unternehmensweit bereitgestellt wird, erleichtert die Entwicklung neuer Anforderungen ganz erheblich. So stellt sich bei jeder neuen Anwendung, und sei sie noch so klein, die Frage nach der Laufzeitumgebung - es wird ein Datenbankschema, ein Anwendungsserver und ein Konzept zum laufenden Betrieb benötigt. Mitunter reicht ein Multiprojektserver aus, aber auch hier sind die Prozesse zum Hinzufügen einer neuen Anwendung meist kompliziert und langwierig. Das gleiche gilt für das Einspielen eines neuen Release oder eines Bugfixes.

Komplizierte und langwierige Prozesse sind leider auch teuer. Je mehr Personen sich aktiv mit der Anfrage beschäftigen müssen und je individueller die Prozesse sind, desto teurer wird das Projekt für die Fachabteilung und desto länger dauert es, bis die nötige IT-Unterstützung für den Geschäftsprozess implementiert ist.

Hier können die Konzepte des Cloud Computing helfen. Wenn die Ressourcenbeschaffung standardisiert ist, automatisiert abläuft und vom Projektverantwortlichen in Eigenverantwortung durchgeführt werden kann, beschleunigt das den ganzen Prozess, denn automatisierte Vorgänge sind stets kostengünstiger als manuelle. Daher sollte am Nutzen einer Cloud-Umgebung mit APEX kein Zweifel bestehen - allein die Tatsache, dass die Umgebung für eine benötigte Anwendung automatisiert und in wenigen Minuten bereitgestellt werden kann, spricht für sich.

Die Ressourcen in einer Cloud werden in Form von Diensten (*Services*) in Anspruch genommen, wobei im wesentlichen drei Serviceformen unterschieden werden:

- INFRASTRUCTURE AS A SERVICE (IAAS)

Hier werden Infrastruktur-Komponenten, wie zum Beispiel Storage, Netzwerk oder Rechnerkapazität zur Verfügung gestellt

- PLATFORM AS A SERVICE (PAAS)  
Die angebotene Plattform beinhaltet zum Beispiel Datenbank- und Applikationsserver-Ressourcen. Eine solche Plattform kann dann Grundlage für eine konkrete Anwendung sein.
- SOFTWARE AS A SERVICE (SAAS)  
Hier wird eine komplette Anwendung als Dienst angeboten.

APEX wäre nach diesem System "Platform-as-a-Service (PaaS)", denn eine Entwicklungsumgebung (Plattform) wird als Dienst bereitgestellt.

## Cloud Computing mit APEX: Selbstbedienung

*Selbstbedienung* ist einer der wichtigsten Aspekte beim Cloud Computing: die Tatsache, dass Nutzer sich, ohne manuelle Eingriffe durch einen Administrator, selbst bedienen können, macht die Systeme flexibel und effizient.

In APEX arbeiten Entwickler in *Workspaces*. Ein APEX-Workspace ist ein abgegrenzter Bereich mit eigener Benutzerverwaltung, der wie eine *Sandbox* funktioniert. Der Eigentümer eines Workspace kann diesen völlig unabhängig verwalten: So lassen sich neue Benutzer einrichten, bestehende ändern, entfernen oder neue Anwendungen erstellen - und zwar ohne dass ein Datenbankadministrator aktiv werden muss. Der wird nur dann gebraucht, wenn die Rechte oder Ressourcen eines Workspace erweitert werden müssen. Und auch hierfür gibt es einen definierten Prozess: Abbildung 2 zeigt die Anfrage eines Workspace nach mehr Speicherplatz. Diese muss (vom Administrator) genehmigt werden - danach wird der Speicherplatz bereitgestellt (die Tablespace-Quota wird erweitert) und der Entwickler kann weiterarbeiten.

Identify Request   **Define Request**   Confirm

← Cancel Next >

Amount of storage to add (MB): 10 ▾

Schemas   **Tablespaces**

GEOLOCNEW USERS

**Tasks**

- [Detailed Tablespace Utilization Report \(may take several seconds\)](#)
- [Sign up for another workspace](#)

Abbildung 2: Anfrage eines Workspace nach mehr Speicherplatz

Je nachdem, wie ein APEX-Server aufgesetzt wurde, werden neue Workspaces - und damit neue Entwicklerbereiche - entweder manuell oder vollautomatisch eingerichtet. Abbildung 3 zeigt die Konfiguration der APEX-Instanz durch den Datenbankadministrator.

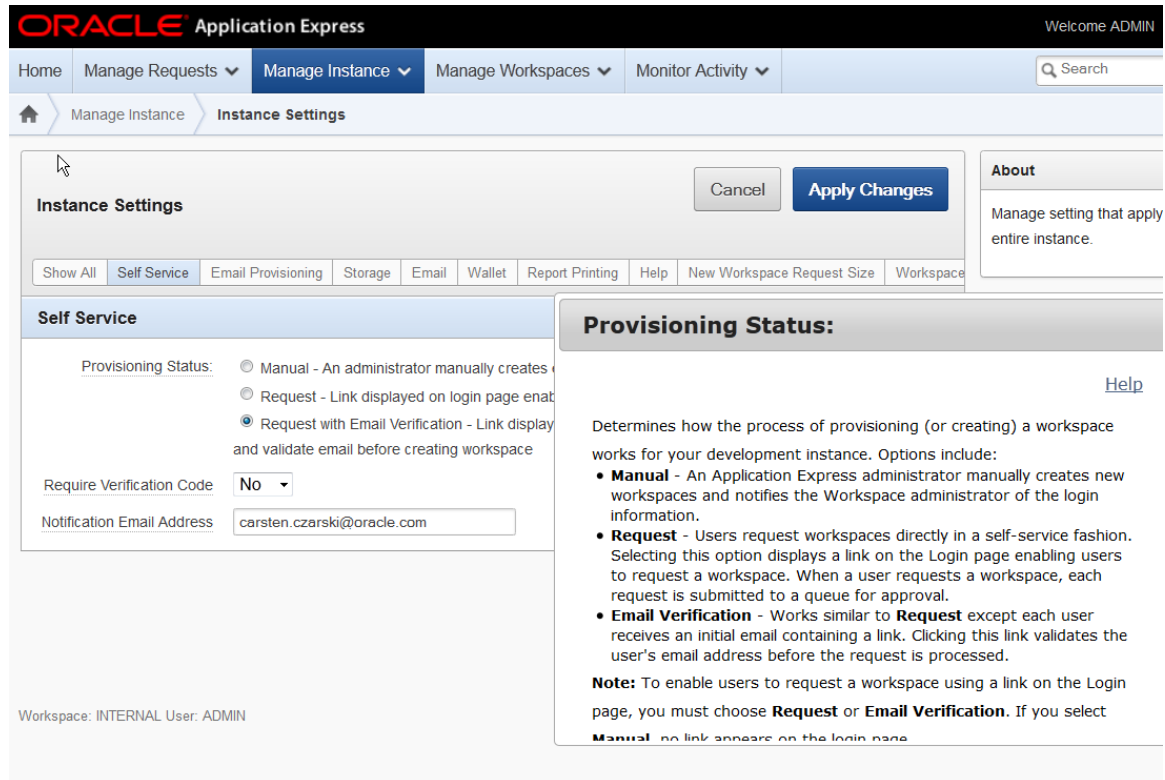


Abbildung 3: Einstellen des Bereitstellungsmodus für neue Workspaces

Wichtig bei einer solch flexiblen Umgebung ist der Umgang mit inaktiven Workspaces. Wie beim öffentlichen APEX-Demoserver **apex.oracle.com** melden sich die meisten Nutzer nicht formal ab, wenn der Workspace nicht mehr gebraucht wird. APEX bringt daher einen Standardprozess zum "Aufräumen" inaktiver Workspaces mit. Es wird eingestellt, wie lange ein Workspace inaktiv sein muss, damit er zum Löschen vorgemerkt wird und wie oft der Eigentümer vorher gewarnt werden soll. Den Rest macht Application Express alleine.

## Anwendungsentwicklung mit dem Browser: von Überall

Eine Umgebung für einen Anwendungsentwickler aus der Cloud bereitzustellen, ist eine echte Herausforderung: Schließlich arbeiten Entwickler mit typischen Werkzeugen (Visual Studio, Eclipse oder JDeveloper) - und das sind Desktop-Applikationen, die zunächst installiert werden müssen. Das bedeutet allerdings, dass der Entwickler nur noch mit seinem PC arbeiten kann. Eine schnelle Änderung, von einem anderen PC aus, ist kaum machbar.

In APEX sind sowohl Entwicklungs- als auch Laufzeitumgebung webbasiert. Die Entwicklungsumgebung ist selbst wiederum eine APEX-Anwendung, läuft also in der gleichen Architektur ab, wie die entwickelten Anwendungen.

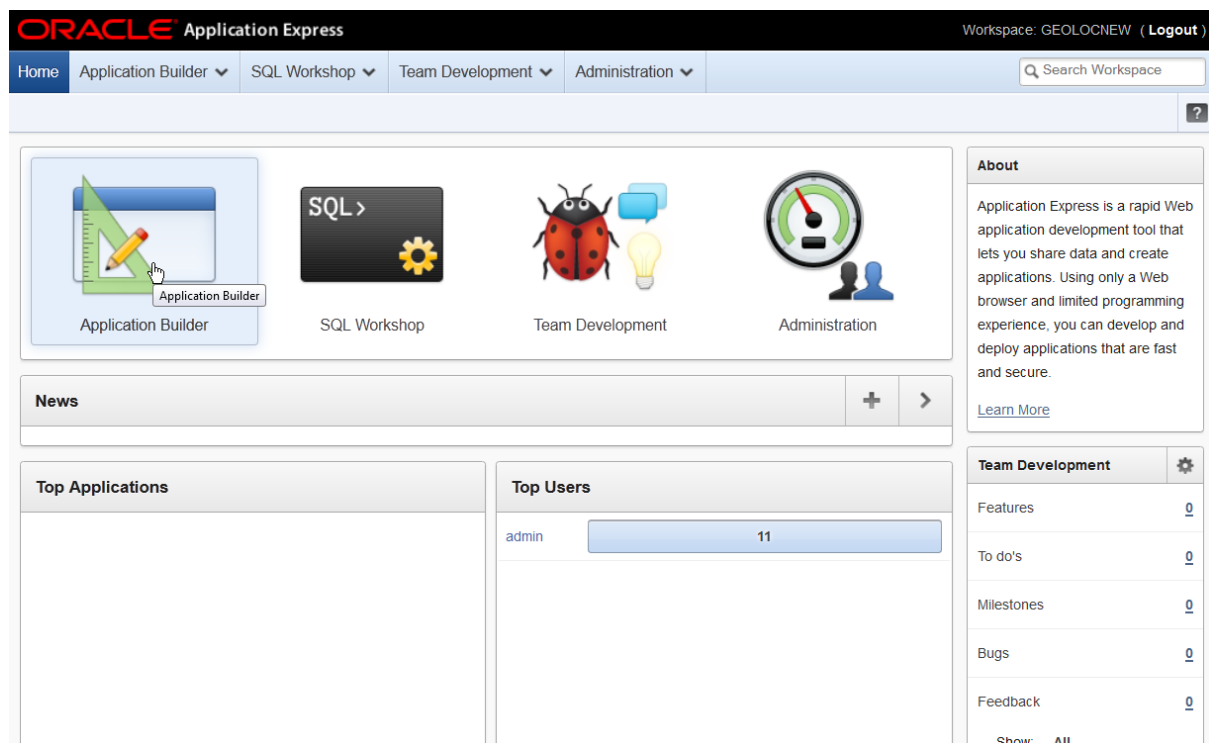


Abbildung 4: APEX ist vollständig webbasiert - der Entwickler arbeitet allein mit dem Browser

Für die Arbeit mit APEX muss also keine Software installiert werden. Auch Arbeiten am Datenbankschema, wie das Erstellen von Tabellen oder die Arbeit an PL/SQL-Prozeduren, können mit dem Browser erledigt werden.

Das *Deployment* einer Anwendung findet nicht statt, da APEX ein metadatengetriebenes Werkzeug ist. Der Entwickler verändert durch seine Arbeit die im APEX-Repository gespeicherte Anwendungsdefinition und der Nutzer sieht die daraus generierten Webseiten sofort.

Eine APEX-Anwendung kann also von jedem Ort aus sowohl genutzt als auch bearbeitet werden: Auf welchem Server APEX läuft, spielt keine Rolle. Ein Entwickler mit VPN-Zugang kann eine APEX-Applikation zur Not auch von unterwegs oder von zu Hause aus bearbeiten.

## Ressourcenverwaltung durch die Datenbank

Ein APEX Workspace wird stets mit einem oder mehreren Datenbankschemas verknüpft - in den Schemas liegen die Datenbankobjekte wie Tabellen, Views oder PL/SQL-Funktionen, mit denen die Anwendung arbeitet. Bei Bedarf kann diese

Zuordnung geändert werden. Dem Datenbankschema kann nun (wie immer) eine Speicherplatz-Quota auf einem oder mehreren Tablespaces zugeordnet werden: APEX stellt speziell hierfür einen Anfragemechanismus bereit (siehe auch Abbildung 2). Solange diese Speicherplatz-Quota eingehalten wird, kann der Entwickler also beliebig neue Objekte anlegen oder bestehende erweitern.

Wenn viele APEX-Workspaces gemeinsam auf einem System arbeiten, entstehen desto mehr Konfliktsituationen, je höher die Last auf dem System ist. Hierfür bietet die Oracle-Datenbank geeignete Mittel an: Wird der *Ressourcen-Manager* eingeschaltet, so lassen sich die verschiedenen Workspaces oder Anwendungen priorisieren. Wiederum merkt der Entwickler davon nichts - solche Aspekte muss er auch nicht bei der Entwicklung seiner Anwendungen berücksichtigen.

Reichen die Ressourcen eines Servers nicht aus, kommen Technologien wie Real Application Clusters zum Einsatz - in diesem Fall greifen mehrere Server auf ein- und dieselbe Datenbank zu. Wird APEX auf einer RAC-Datenbank installiert, so bemerkt der Entwickler davon ebenfalls nichts - APEX funktioniert genauso wie in einer Single-Instance-Datenbank.

Der Datenbankadministrator kann also anhand des Gesamt-Anforderungsprofils ...

- Wie viele APEX-Anwendungen sind vorhanden?
- Wie viele Nutzer greifen parallel zu?
- Welche Verfügbarkeit wird benötigt?

... Ressourcen wie Server, CPU-Kerne und I/O-Bandbreiten bereitstellen. Die Oracle-Datenbank sorgt dafür, dass die APEX-Entwickler diese möglichst effizient nutzen können. Das wird am öffentlichen APEX-Demoserver **apex.oracle.com** wiederum besonders deutlich: Welche Hardware dort tatsächlich läuft, ist dem APEX-Entwickler in keinsten Weise ersichtlich - in einem APEX-Workspace arbeitet er, als hätte er den Server für sich alleine. Die Datenbank sorgt jedoch mit dem Ressourcen Manager dafür, dass keine Applikation die Server-Ressourcen für sich alleine allokiert kann.

## **Kosten verteilen: Abrechnungsverfahren mit APEX**

APEX selbst bringt kein spezielles Modul zum Abrechnen von Kosten mit. Gerade hier hat wahrscheinlich jedes Unternehmen eigene Anforderungen und auch eigene Prozesse. Während die Kosten bei Unternehmen A nach der Belegung von Plattenplatz verrechnet werden, kommt bei Unternehmen B vielleicht eine pauschale Verteilung nach Workspaces in Betracht. Unternehmen A möchte die Kennzahlen

direkt von einem ERP-System verarbeiten lassen - Unternehmen B nimmt vorher eine manuelle Prüfung vor.

Workspace	Page Views	Percentage	Page View Percentage	Elapsed Time	IP Addresses	Report Rows	Content
TECHNOLOGY_DE	33,353	92.7787		14,557.72	340	66,600	0
SCMAPPING	1,947	5.4160		94.49	95	38,340	0
SCCLOUD	579	1.6106		183.18	17	3,223	0
INTERNAL	56	0.1558		9.00	21	17	0
KALENDER	8	0.0223		0.57	4	0	0
FUNC SPECS	6	0.0167		8.59	1	16	0
							0

1 - 6

Abbildung 5: APEX-Aktivität, gegliedert nach Anwendung

Die Grundlage für jede Form der Kostenverrechnung sind Nutzungsdaten - und diese werden von APEX gesammelt. So werden alle Seitenabrufe im *APEX Activity Log* mitprotokolliert - nebst verbrauchter Zeit. Es lassen sich also sowohl einfache als auch komplexere Abrechnungsmodelle realisieren. Beispiele für sehr einfache Modelle wären die pauschale Verteilung der Kosten ...

- Anhand der Anzahl Workspaces
- Anhand der Anzahl Anwendungen
- Anhand der Anzahl Benutzer
- Anhand des belegten Plattenplatzes im Tablespace

Ein Beispiel für ein komplexeres Modell wäre die Verteilung anhand der Anzahl von Klicks, die auf die Anwendungen eines Workspace stattfinden oder die verbrauchte CPU-Zeit pro Anwendung. Welche Variante die Richtige ist, hängt stark von den tatsächlichen Kostentreibern des APEX-Servers ab. Zusätzlich können natürlich auch andere, von APEX unabhängige Metriken die Grundlage für eine Abrechnung sein.

## Fazit

Application Express setzt einige der Konzepte und Ideen, die unter dem Begriff Cloud Computing diskutiert werden, bereits seit Jahren um. Wie der öffentliche Demoserver **apex.oracle.com** zeigt, können Entwickler einen Workspace im Selbstbedienungsverfahren beantragen und selbstständig verwalten und nutzen. Dem Entwickler steht die volle Bandbreite der Oracle-Datenbank zur Verfügung, egal

ob diese als *Single Instance* auf einer CPU oder als RAC-Datenbank auf mehreren Rechnerknoten läuft.

Auf [apex.oracle.com](http://apex.oracle.com) läuft allerdings genau das gleiche APEX, das auch heruntergeladen werden kann - es spricht also überhaupt nichts dagegen, ein eigenes "**apex.meinefirma.de**" im Unternehmen auszubauen - und mit dem Cloud Computing für Anwendungsentwickler sofort zu beginnen.

## Weitere Informationen

Auf den Webseiten von Oracle und im Internet ist umfangreiches Material auch in deutscher Sprache vorhanden:

[1] Deutschsprachige APEX Community

[http://blogs.oracle.com/apexcommunity\\_deutsch](http://blogs.oracle.com/apexcommunity_deutsch)

- a. Vorfahrtsregeln in APEX mit dem Ressourcenmanager
- b. Eigene Bereitstellungsverfahren für APEX Workspaces einrichten

**Kontakt:**

*Carsten Czarski*

[Carsten.Czarski@oracle.com](mailto:Carsten.Czarski@oracle.com)

<http://sql-plsql-de.blogspot.com>

*Twitter: @cczarski*