

# apex.meinefirma.de: APEX-Hosting im eigenen Unternehmen

Carsten Czarski  
ORACLE Deutschland B.V. & Co. KG  
München

## Schlüsselworte

APEX, Cloud, Hosting, Entwicklung

## Einleitung

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. Wie der öffentliche Demoserver **apex.oracle.com** zeigt, kommt Oracle Application Express (APEX) kommt diesem Bedürfnis in geradezu idealer Weise entgegen. Dieser Artikel und der **DOAG2011**-Konferenzvortrag stellen vor, wie eine Entwicklungsplattform mit APEX nach den Konzepten des Cloud Computing bereitgestellt und flexibel genutzt werden kann.

## 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 und Bereitstellung neuer Anwendungen ganz erheblich. Wiederum lohnt ein Blick auf den Demoserver **apex.oracle.com**. Wer schon damit gearbeitet hat, kennt den schnellen Prozess, mit dem ein neuer APEX-Workspace bereitgestellt wird. Normalerweise hat man seine Login-Daten innerhalb von 15 Minuten. Anschließend kann man in seinem Workspace schalten und walten, wie man möchte. Ob neue Anwendungen, zusätzliche Benutzer oder neue Tabellen: Solange man sich innerhalb der gesetzten Grenzen (z.B. Tablespace-Quota) bewegt, kann man völlig frei arbeiten.

Verglichen damit ist der vielfach übliche "traditionelle" Prozess mit Beantragung, Meetings, Diskussionen, Bewilligung und meist manueller Bereitstellung der Ressourcen (und sei es nur ein Datenbankschema) schwerfällig, kompliziert und langwierig. Nun sind komplizierte und langwierige Prozesse 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. Da automatisierte Vorgänge stets kostengünstiger sind, als manuelle, 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 dieser Systematik eine "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 erst 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 sollen. 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.



Abbildung 1: Anfrage eines Workspace nach mehr Speicherplatz

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

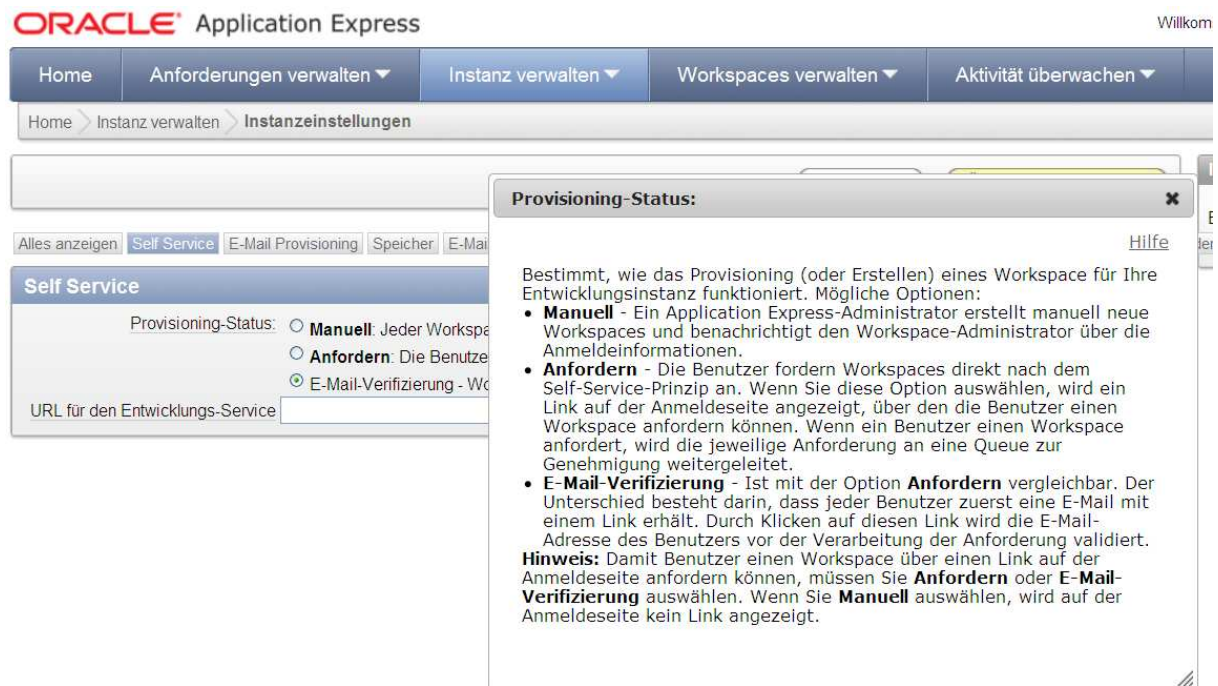


Abbildung 2: Einstellen des Bereitstellungsmodus für neue Workspaces

Dieser Prozess zur Bereitstellung von Workspaces ist sehr bequem - und vor allem out-of-the-box vorhanden. Man muss ihn nur einschalten und schon können alle Mitarbeiter im Unternehmen APEX-Workspaces beantragen. Dennoch dürfte diese "out-of-the-box-Variante" in vielen Fällen nicht zum praktischen Leben im Unternehmen passen: So muss der APEX-Administrator (der DBA) die Anfragen genehmigen. Bei einem unternehmensweit ausgerollten APEX Hosting dürfte dies nicht passend sein - sinnvoller wäre der direkte Vorgesetzte, der auch die Kostenstelle verantwortet. Außerdem möchte man die Beantragungssseite im Unternehmen meist durch einen Login schützen. Nach dem Login ist die Identität des Nutzers bekannt - die Eingabe einer Mailadresse, des Vor- und Nachnamens dürfte sich dann erübrigen. Aus technischer Sicht dürfte die Tatsache, dass der Nutzer

den Namen für das Datenbankschema bei Beantragung des Workspace selbst aussuchen kann, kaum einem DBA gefallen, denn hierfür kommen meistens unternehmensspezifische Namenskonventionen zur Anwendung.

Alles in allem wird ein eigener, unternehmensspezifischer Bereitstellungsprozess für APEX-Workspaces benötigt. Und dies kann mit den APEX Bordmitteln auch sehr einfach implementiert werden.

- Die Anträge werden über eine normale APEX-Anwendung gestellt.
- Die "Antragsdaten" werden in eine eigene Tabelle gespeichert
- Datenbankschema und Tablespace werden per PL/SQL Prozedur erstellt (EXECUTE IMMEDIATE)
- Mit dem PL/SQL-Paket APEX\_INSTANCE\_ADMIN wird der APEX Workspace erstellt

The screenshot shows a web interface for requesting an APEX workspace. At the top, there is a navigation bar with a red background. On the left, a button labeled 'Workspace beantragen' is visible. On the right, a button labeled 'Meine Anfragen' is highlighted in red. Below this bar is a form titled 'Workspace beantragen'. The form contains two text input fields: 'Vorname' with the value 'Carsten' and 'Nachname' with the value 'Czarski'. Below these fields, the text 'APEX-User-ID ADMIN' and 'E-Mail ADMIN@meinefirma.de' is displayed. At the bottom of the form, there is a button labeled 'Workspace beantragen'.

Abbildung 3: Eigene Anwendung zum Beantragen neuer APEX Workspaces

In einem der Tipps der deutschsprachigen APEX-Community (siehe "Weitere Informationen") ist die Vorgehensweise zur Einrichtung eines solchen "Workspace-Provisioning" beschrieben.

Wichtig bei einer APEX-Cloud-Umgebung ist auch 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 benötigt 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 (Abbildung 4). Den Rest macht Application Express alleine - der DBA muss nichts mehr tun: Die Workspaces der Nutzer, die nicht auf die Warnung reagieren, werden nach einer bestimmten Zeit gelöscht. Reagiert ein Nutzer dagegen auf die Warnung, indem er auf den in der Email enthaltenen Link klickt, bleibt der Workspace erhalten.

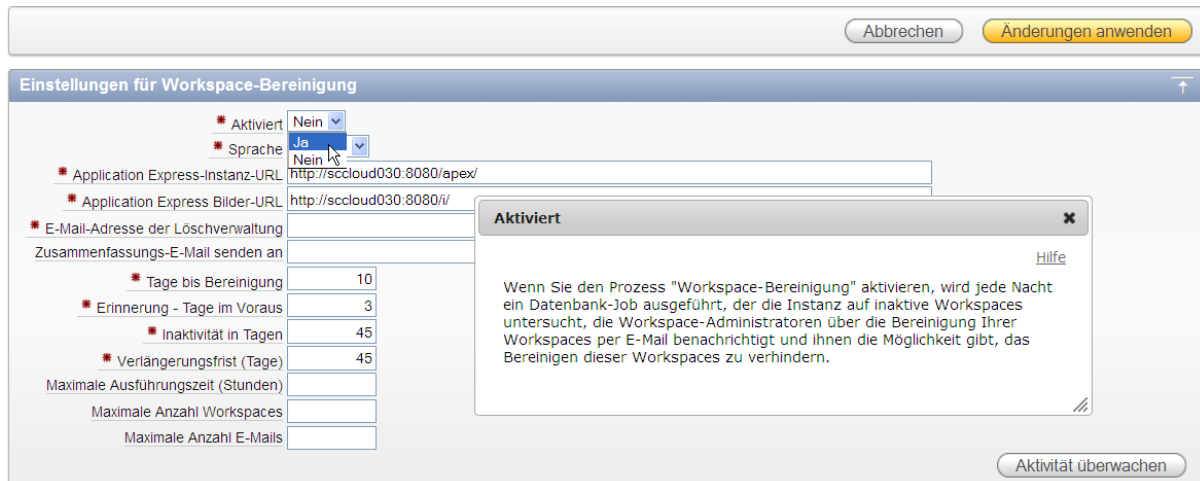


Abbildung 4: Einrichtung der automatischen Bereinigung inaktiver APEX Workspaces

### 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.

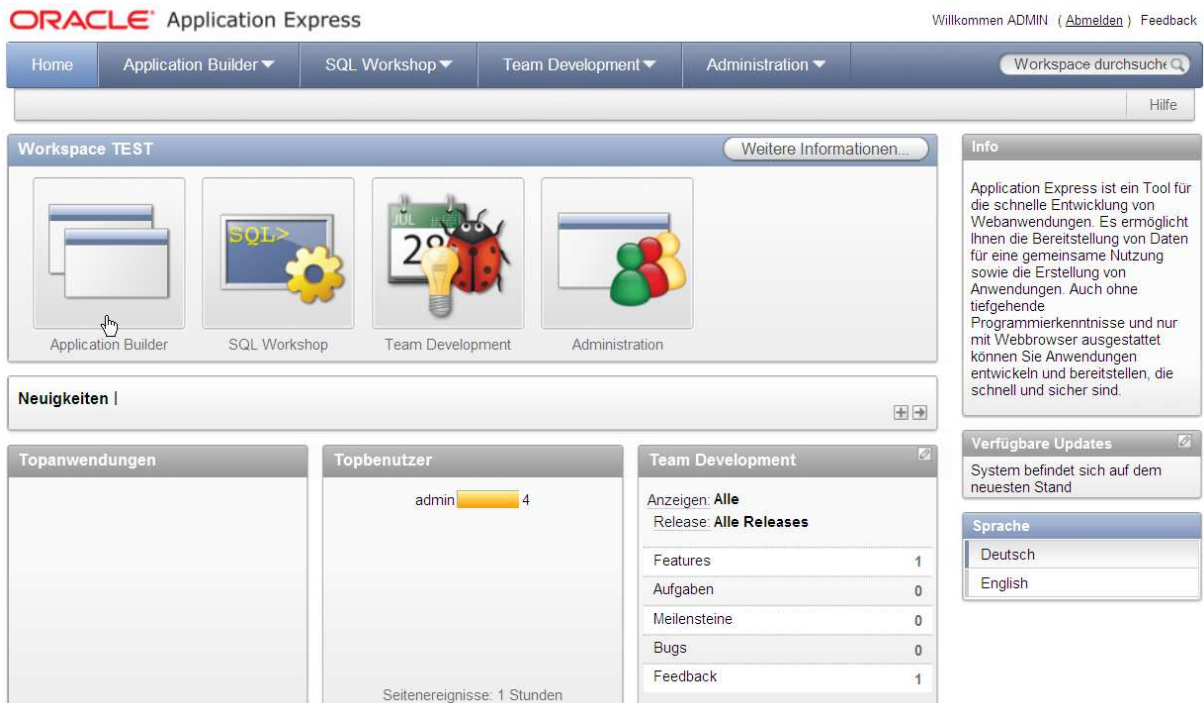


Abbildung 4: APEX ist vollständig webbasiert - der Entwickler arbeitet allein mit dem Browser

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 mehreren Datenbankschemas verknüpft - in den Schemas liegen Datenbankobjekte wie Tabellen, Views oder PL/SQL-Funktionen, mit denen die Anwendung arbeitet. Dem Datenbankschema kann nun (wie immer) eine Speicherplatz-Quota auf einem oder mehreren Tablespace zugeordnet werden: Solange diese Speicherplatz-Quota eingehalten wird, kann der Entwickler also beliebig neue Objekte anlegen oder bestehende erweitern.

Wenn viele Entwickler im Unternehmen auf dem APEX-Server arbeiten, kann noch ein anderes Problem auftauchen, das man in "klassischen" Systemen mit wenigen Entwicklern nicht kennt: Unerfahrene Entwickler können die Ressourcen durch Endlosschleifen oder entsprechende SQL-Abfragen ("Kartesische Produkte") schnell blockieren. Je mehr (unbekannte) Entwickler auf dem System arbeiten, desto höher ist die Wahrscheinlichkeit, dass dies tatsächlich passiert.

Vor der gleichen Herausforderung stand auch apex.oracle.com. Doch mit dem *Ressourcen Manager*, der Teil der Enterprise Edition der Oracle-Datenbank ist, lässt sich dem leicht begegnen. Der auf **apex.oracle.com** aktive Ressourcenmanager-Plan kann sogar auf Joel Kallmans Blog ("Weitere Informationen") eingesehen werden: Alle Aktivitäten (SQL-Abfragen oder PL/SQL-Code) laufen 10 Sekunden lang in der höchsten Prioritätsklasse - dieser Stufe steht 70% der CPU-Leistung zur Verfügung. Länger laufende Abfragen werden für die nächsten zwei Minuten in die mittlere Klasse eingestuft - diese hat Zugriff auf nur noch 8% der CPU-Leistung. Die dritte Stufe kann auf 2% der CPU-Leistung zugreifen - dort sind alle Aktivitäten eingestuft, die länger als 2, aber weniger als 30 Minuten laufen. Nach 30 Minuten wird das SQL-Kommando abgebrochen (CANCEL\_SQL).

Die konkreten Einstellungen wird man sicherlich in jedem Einzelfall anpassen wollen - aber es steht schon fest, dass viele parallel arbeitende Entwickler mit unterschiedlichem Kenntnisstand kein Hindernis (eher ein Grund) für eine "APEX Cloud" sind. Die Datenbank bietet die nötigen Werkzeuge an - man muss sie nur nutzen. Auch das Priorisieren von APEX-Anwendungen untereinander kann der Ressourcenmanager leisten - wie das geht, ist in einem Tipp der deutschen APEX Community beschrieben.

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

APEX selbst bringt kein spezielles Modul zum Abrechnen von Kosten mit. Gerade hier hat 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	Application	Application Name	Page Views	Percentage	Graph
TECHNOLOGY_DE	178	technology.de 3.0	188	30.03	
SCCLOUD	176	SC-Cloud	185	29.55	
INTERNAL	4550	Oracle APEX Login	144	23.00	
SCMAPPING	179	DBMProfiler	29	4.63	
KALENDER	109	VERANSTALTUNGEN_V3	18	2.88	
TECHNOLOGY_DE	149	Konferenz Folien	18	2.88	
COMMUNITY	114	Community-Mitglieder	15	2.40	
INTERNAL	4050	Oracle APEX Internal Administration	14	2.24	
SCMAPPING	133	CRT	14	2.24	
KIP	142	GCM_ACTIVITIES	1	0.16	

1 - 10

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. Daneben können auch andere, von APEX unabhängige Metriken, zum Einsatz kommen - alle Informationen, die Oracle im Data Dictionary und im Automated Workload Repository sammelt, können hierfür herangezogen werden.

## Fazit

Application Express setzt die Konzepte und Ideen, die unter dem Begriff *Cloud Computing* diskutiert werden, bereits seit Jahren um. Wie der öffentliche Demoserver [apex.oracle.com](http://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 *Real Application Cluster* auf mehreren Rechnerknoten läuft.

Auf [apex.oracle.com](http://apex.oracle.com) läuft genau die gleiche APEX-Software, die 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://tinyurl.com/apexcommunity>

[2] Joel Kallmans Blog

<http://joelkallman.blogspot.com>

<http://joelkallman.blogspot.com/2009/08/oracle-database-resource-manager-and.html>

[3] APEX Demoserver

<http://apex.oracle.com>

## **Kontaktadresse:**

Carsten Czarski

ORACLE Deutschland B.V. & Co KG

Riesstr. 25

80992 München

Telefon: +49 (0) 89 1430 2116

E-Mail [carsten.czarski@oracle.com](mailto:carsten.czarski@oracle.com)

Internet: [www.oracle.de](http://www.oracle.de)

Blog des Autors: [sql-plsql-de.blogspot.com](http://sql-plsql-de.blogspot.com)

Twitter; [@cczarski](https://twitter.com/cczarski) [@oraclebudb](https://twitter.com/oraclebudb)