

# Forms kann das, wieso geht das in APEX nicht?

**Peter Raganitsch**  
**click-click IT Solutions e.U.**  
**Wien, Österreich**

## **Schlüsselworte**

Erstellen von APEX Applikationen mit Client-Server Funktionalität, ähnlich wie in Oracle Forms.

## **Einleitung**

Oracle APEX steht für schnell und einfach entwickelte leichtgewichtige Web-Applikationen. Entwickler und Anwender die zuvor mit Oracle Forms (oder anderen Client/Server Technologien) gearbeitet haben vermissen oft den Komfort und die Möglichkeiten einer Client/Server Anwendung. Komplexe Masken mit mehreren Datenblöcken, Karteireiter, Master-Detail Relations, Multi-Record-Blöcke usw. sind alle nicht – oder nur schwierig und mit starken Einschränkungen – auf Standard APEX Funktionen abzubilden.

## **Unterschiede APEX und Forms**

Als erster und größter Unterschied zwischen diesen beiden Produkten lässt sich deren Architektur nennen. So ist APEX eine klassische Webanwendung, Forms eine Client-Server Anwendung.

Während Forms als echte Client-Server Anwendung (Version 6i) bzw. in einem Java-Applet (Version 9-11gR2) läuft und dort die volle Kontrolle über sämtliche Komponenten hat, erzeugt APEX lediglich HTML (und ggf. JavaScript) Code, welcher von einem beliebigen Browser interpretiert wird.

Das bedeutet soviel wie Forms kann beliebige Komponenten (Felder, Multi-Record-Block) und Verhalten (sofortige Feldvalidierung, Tastaturnavigation, ...) implementieren, in APEX ist man auf den HTML-Standard und die jeweilige Browser-Unterstützung angewiesen.

Oberflächlich betrachtet bedeutet das, dass man in Oracle APEX lediglich sehr einfache und statische Web-Anwendungen erstellen kann. Zwar mit jeglicher Webdesign-Unterstützung – also gut aussehend und modern – doch limitiert aufgrund des HTML-Standards.

Diese Unterschiede bringen mit sich, dass darin entwickelte Anwendungen nicht die gleichen Möglichkeiten (Komponenten) haben können und deswegen auch anders designt werden müssen. Es ist damit nicht möglich 1:1 die selbe Anwendung in Oracle APEX zu entwickeln, wie man sie bereits in Oracle Forms entwickelt hatte.

## **Webanwendung und deren Möglichkeiten**

Das Internet ist groß und vielfältig, aber HTML ist statisch. So die landläufige Meinung.

Bei genauerer Betrachtung der Webentwicklung kommt JavaScript ins Spiel. Die Programmiersprache für den Browser. Damit kann die Client (Browser) Programmierung übernommen werden und auch Oracle APEX macht sich dies mit den Dynamic Actions zu Nutze.

Doch ist JavaScript viel mächtiger als uns die von Oracle APEX mitgelieferten Dynamic Actions suggerieren, denn in den Weiten des Internets haben sich schon längst einige JavaScript-Frameworks gebildet. Eines – und lt. Meinung des Autors das erwachsenste – davon ist Ext JS von Sencha.

Dabei handelt es sich um eine JavaScript-UI Bibliothek, die uns wieder sehr stark an Client/Server Anwendungen erinnert.

### **Erweiterungsmöglichkeit von Oracle APEX**

Oracle APEX verfügt seit Version 4.0 über eine Plugin-Schnittstelle. Damit lassen sich eigene Komponenten (Felder, Regionen, Prozesse, Dynamic Actions, ...) entwickeln bzw. von Drittanbietern zukaufen.

Auf der Internetseite [www.apex-plugin.com](http://www.apex-plugin.com) findet man über 100 Plugins für Oracle APEX, die jeweils ein kleines Stück Funktionalität bzw. eine neue Komponente liefern. Als Beispiel dafür lassen sich z.B. Regions-Plugins für Oracle Maps oder Open Street Maps nennen, oder Modale Fenster, Multi-Select-Liste, Super-LOV uva.

All diese Plugins widmen sich aber jeweils nur einem einzigen Zweck, keines dieser Plugins zielt auf mehrere Funktionen oder ganze Lösungen ab.

Will man sich eine Bibliothek wie Ext JS von Sencha zu Nutze zu machen, dann ist es mit einem einzelnen Plugin nicht mehr getan. Hier müssen viele Plugins zusammenspielen, um dieses neue Level an Funktionalität zu gewährleisten.

... mehr dazu im Vortrag „Forms kann das, wieso geht das in APEX nicht?“ am 21.11.2012 von 12:00 bis 12:45

**Kontaktadresse:**

Peter Raganitsch

click-click IT Solutions e.U.

Welschgasse 8

A-1230 Wien

Telefon: +43 (0) 1-3119425-0

Fax: +43 (0) 1-3119425-99

E-Mail [peter.raganitsch@click-click.at](mailto:peter.raganitsch@click-click.at)

Internet: [www.click-click.at](http://www.click-click.at)