

Seit dem ersten Release von HTML DB zusammen mit Oracle10g im Jahr 2003 ist eine Menge passiert. HTML DB heißt schon seit einiger Zeit Apex; die Version 4.1 ist verfügbar und es gibt eine Fülle neuer Features und Möglichkeiten. Wie bei jedem Software-Produkt gibt es inzwischen auch in Apex Features, die zwar sehr nützlich sind, jedoch in Artikeln, Blogs oder Howto-Dokumenten kaum erwähnt werden. Dieser Artikel stellt einige selten genutzte, aber sehr nützliche Apex-Features vor.

# Versteckte Juwelen: kaum bekannte Apex-Features

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

## Format-Masken für Berichte

Es ist allgemein bekannt, dass man eine numerische oder Datumsspalte in einem Apex-Bericht formatieren kann. Dabei werden die aus SQL bekannten Format-Masken wie „9G999D99“ oder „DD-MM-YYYY“ verwendet. Apex bietet jedoch zusätzliche Format-Masken an, welche die Lesbarkeit eines Berichts massiv verbessern können:

- „PCT\_GRAPH“ stellt einen Prozentwert als Balkengrafik dar

- „SINCE“ übersetzt ein Datum in einen Text
- „FILESIZE“ übersetzt eine Zahl in eine lesbare Dateigröße (KB, MB, GB, TB)

Der Bekanntheitsgrad von „PCT\_GRAPH“ hat sich seit der Version 4.0 verbessert, da es als Schnellauswahl direkt unterhalb des Eingabefelds für die Format-Maske bereitsteht (siehe Abbildung 1). Es ist geeignet für Zahlen zwischen 0 und 100. Durch Doppelpunkte getrennt werden zunächst die Vordergrundfarbe, danach die Hinter-

grundfarbe und schließlich die Breite des Balkens in Pixel angegeben. Der Balken wird von rechts her gefüllt – möchte man es von links her haben, kann man in der Berichtsabfrage den Wert der Tabellenspalte von 100 abziehen. Der Prozentbalken wird mit HTML-Code erzeugt; da dieser ab Apex 4.0 standardmäßig maskiert wird, ist es wichtig, die Darstellung der Spalte auf „Standardberichtsspalte“ umzustellen.

Die Format-Masken „SINCE“ und „FILESIZE“ sind noch einfacher zu nutzen: Das Schlüsselwort wird einfach bei „Number / Date Format“ eingetragen und es muss nichts mehr angehängt werden. „SINCE“ ist sogar multilingual und unterstützt alle Sprachen, für die eine Übersetzung der Apex-Entwicklungsumgebung installiert ist. Ist die Apex-Anwendung beispielsweise auf „deutsch“ eingestellt und die deutsche Übersetzung für den Application Builder installiert, so werden die Texte auch auf Deutsch dargestellt (siehe Abbildung 2).

Die Vorteile der eingebauten Format-Masken liegen klar auf der Hand: Erstens muss man sie nicht selbst entwickeln und zweitens wirken sie sich nicht negativ auf Sortier- und Suchfunktionen aus. Alle drei Format-Masken stehen auch auf PL/SQL-Ebene als Funktionen im Package „Apex\_UTIL“ bereit:

- Apex\_UTIL.GET\_SINCE
- Apex\_UTIL.FILESIZE\_MASK
- Apex\_UTIL.HTML\_PCT\_GRAPH\_MASK

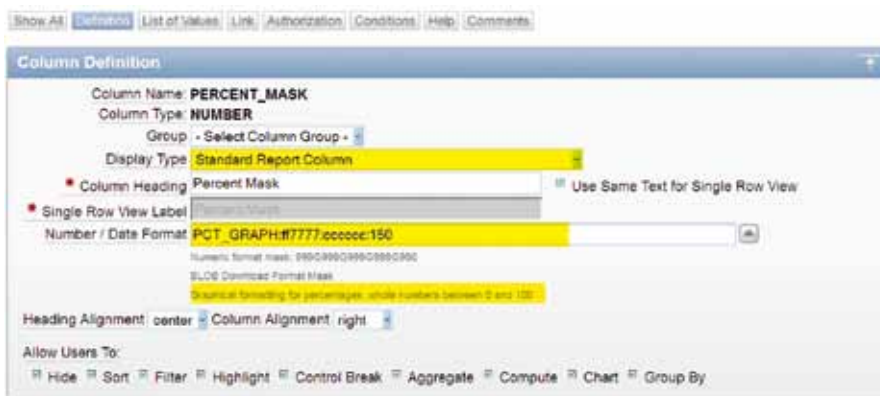


Abbildung 1: Format-Maske „PCT\_GRAPH“

Percent	Percent Mask	Datum	Datum Mask	Filesize	Filesize Mask
15	<div style="width: 15%; background-color: #ff9900; border: 1px solid #ccc;"></div>	27.01.81	vor 30,8 Jahren	1762376761762	2TB
85	<div style="width: 85%; background-color: #ff9900; border: 1px solid #ccc;"></div>	06.11.11	vor 2 Wochen	176237	172KB

Abbildung 2: Format-Masken in Aktion

### Schnell-Auswahlen nutzen

Schnell-Auswahlen sind Links unterhalb von Eingabefeldern in Formularen – beim Klick darauf wird das Eingabefeld automatisch gefüllt. Im Application Builder werden die Schnell-Auswahlen sehr intensiv genutzt, zum Beispiel bei den Format-Masken (siehe auch Abbildung 1). Apex erlaubt Schnell-Auswahlen auch in eigenen Anwendungen, und zwar für alle Formular-Elemente. Navigiert man zu den Einstellungen eines Elements, so gibt es den Abschnitt „Schnell-Auswahl“ (Quick Picks); die Abbildungen 3 und 4 zeigen das anhand eines Beispiels. Das Formular ist sofort wesentlich leichter zu bedienen.

### Management der Apex-Komponenten

Viele der weniger bekannten Features beschäftigen sich mit dem Management der Apex-Komponenten. Auch Features, die nicht direkt zur Benutzeroberfläche einer Anwendung beitragen, gehören nahezu immer zu den „kaum bekannten“ Funktionen.

### Erstell-Optionen

Erstell-Optionen (Build Options) erlauben es, mehrere Komponenten einer Apex-Anwendung auf einmal zu aktivieren oder zu deaktivieren, ohne dass dazu eigene Bedingungen erforderlich sind. Dazu ist zuerst eine neue Erstel-Option zu erzeugen, was in den „gemeinsamen Komponenten“ und dort unter „Erstell-Optionen“ geschieht. Das Beispiel in Abbildung 5 erzeugt eine neue Erstel-Option mit dem Status „Exclude“. Mit dieser Erstel-Option können nun alle Anwendungskomponenten versehen werden (siehe Abbildung 6). Solange der Status auf „Exclude“ steht, werden sie zur Laufzeit nicht ausgeführt. Erstel-Optionen werden auch beim Export berücksichtigt; es ist möglich, eine Anwendung mit oder ohne Komponenten einer bestimmten Erstel-Option zu exportieren.

Erstell-Optionen sind sehr zu empfehlen, wenn innerhalb der Anwen-

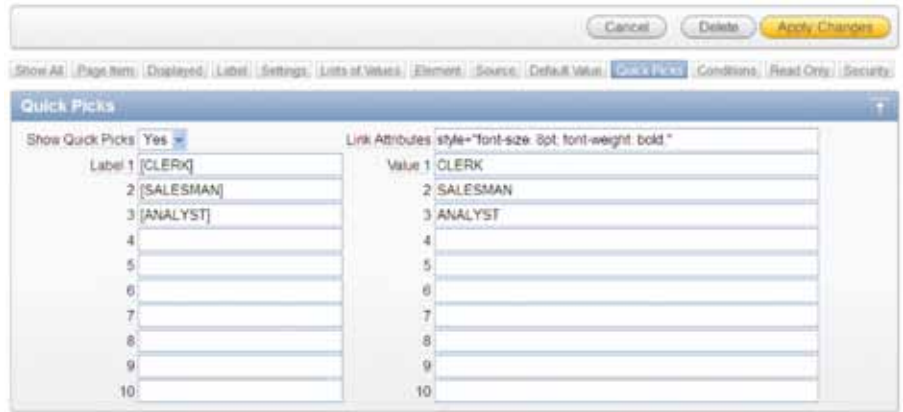


Abbildung 3: Einrichten einer Schnell-Auswahl



Abbildung 4: Schnell-Auswahlen in Aktion



Abbildung 5: Eine neue Erstel-Option mit dem Status „EXCLUDE“

nung eine neu eingebaute Funktionalität aus vielen Komponenten auf mehreren Seiten besteht – das Aktivieren oder Deaktivieren wird mit einfachem Setzen der Erstel-Option auf „Include“ oder „Exclude“ realisiert. Anwendungskomponenten dokumentieren, zu welchem Zweck und gebe-

nenfalls zu welcher Zeit eine Komponente eingeführt wurde.

### Kommandozeilen-Export und Apex\_APPLICATION\_INSTALL

Wenn es darum geht, Anwendungskomponenten auf einen anderen Ser-

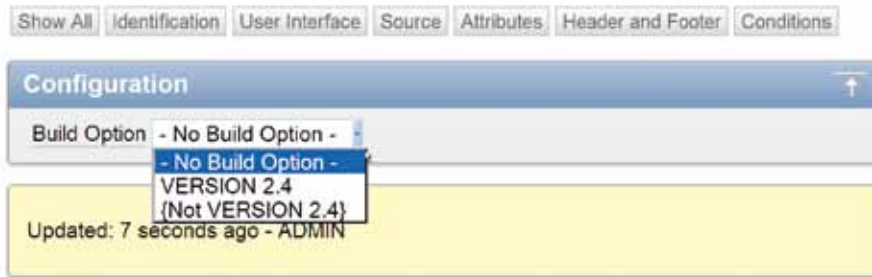


Abbildung 6: Erstell-Option bei einer Apex-Komponente einstellen

```
$ apxexp -db localhost:1521:orcl \
        -user scott -password **** \
        -instance

Exporting Application 100:'Bildbearbeitung'
  Completed at Tue Nov 22 11:25:05 CET 2011
Exporting Application 102:'Sample Application'
  Completed at Tue Nov 22 11:25:07 CET 2011
Exporting Application 109:'ODD Editions'
  Completed at Tue Nov 22 11:25:08 CET 2011
:
$
```

Listing 1: Export einer Apex-Anwendung von der Kommandozeile

ver zu übertragen oder in ein Versions-Kontrollsystem wie CVS oder Subversion einzustellen, so ist der Anwendungsexport das Mittel der Wahl. Aus dem Application Builder heraus hat wahrscheinlich jeder schon mal eine Anwendung exportiert. Weniger bekannt ist, dass Apex auch ein Kommandozeilen-Werkzeug mitliefert, das nicht nur den Export einzelner Anwendungen, sondern auch den aller Anwendungen eines Workspace oder gar der Apex-Instanz erlaubt (siehe Listing 1). Das Aufsetzen eines Jobs, der täglich alle Anwendungen exportiert und ins Versions-Kontrollsystem eincheckt, ist dann nur noch eine Kleinigkeit. Das Kommandozeilen-Werkzeug muss allerdings erst eingerichtet werden. Wie das geht, ist in einem Tipp der deutschsprachigen Apex-Community unter <http://tinyurl.com/apxcm-dexp> beschrieben.

Ein weiteres Werkzeug ist der „Apex Export Splitter“, der eine Exportdatei in ihre Komponenten zerlegt – es entsteht eine Verzeichnisstruktur, in der eine ei-

## KeepTool mit neuer Version 10

Das handliche Werkzeug für Oracle™-Datenbanken



Zahlreiche neue Funktionen, z.B.

- Darstellung Ihrer Daten als Pivottabelle, ggf. mehrstufig.
- Praktische Hinweistexte bei der Datenerfassung.
- Überwachung und Steuerung der Optimizer-Statistiken.
- Data Pump-Schnittstelle.
- Jumplist für den Windows 7™ Taskbar.

Laden Sie die kostenlose Testversion unter [www.keeptool.com](http://www.keeptool.com) herunter.



# keeptool

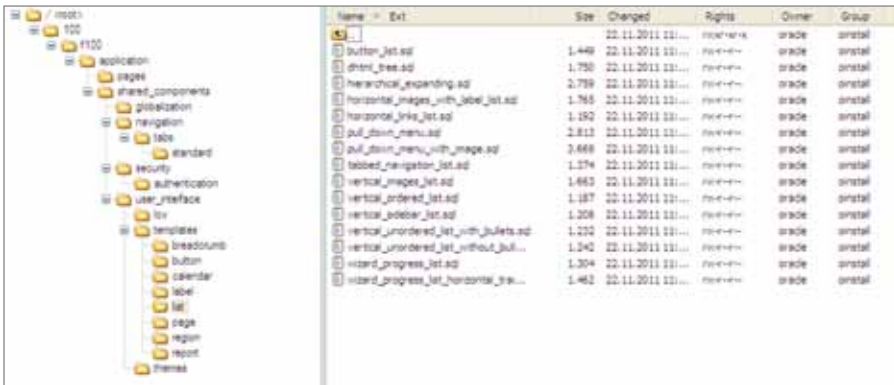


Abbildung 7: Der Apex Export Splitter zerlegt eine Export-Datei

```
SQL> begin
  2 apex_application_install.set_workspace_id(3138423103500405);
  3 apex_application_install.set_application_id(999999);
  4 apex_application_install.set_schema('SCOTT');
  5 -- Offset auf eine zufällige große Zahl setzen ...
  6 apex_application_install.set_offset(2763898718743);
  7 end;
  8 /
```

PL/SQL procedure successfully completed.

```
SQL> @app_17378/pages/page_00021.sql
...PAGE 21: Interaktive Berichte
```

```
SQL> commit;
```

Commit complete.

Listing 2: Eine Apex-Seite wird mit SQL\*Plus erfolgreich in eine andere Anwendung importiert

gene Exportdatei für jede Apex-Komponente enthalten ist (siehe Abbildung 7). Auch dies ist im Tipp der deutschsprachigen Apex-Community näher erklärt.

Kombiniert man dies mit dem erwähnten täglichen Exportieren und Einchecken ins Versions-Management, so bekommen Apex-Entwickler Zugriff auf tägliche Versionsstände für einzelne Apex-Komponenten. Das Wiederherstellen einer versehentlich gelöschten Apex-Komponente ist damit überhaupt kein Problem mehr.

In diesem Zusammenhang ist das in Apex 4.0 eingeführte PL/SQL-Paket „Apex\_APPLICATION\_INSTALL“ zu erwähnen. Wie die meisten Entwickler wissen, ist eine Apex-Exportdatei ein SQL-Skript, das prinzipiell auch mit SQL\*Plus eingespielt werden kann. Sobald man jedoch versucht, eine Anwendung in einen anderen Workspace

oder eine Anwendungskomponente in eine andere Anwendung zu importieren, schlägt ein solcher Import mit einer Fehlermeldung fehl.

„Apex\_APPLICATION\_INSTALL“ erlaubt es, für einen Import mit SQL\*Plus die Workspace-ID, die Applikations-ID und andere Einstellungen vor dem Import zu setzen. Diese Einstellungen gehen den in der Exportdatei enthaltenen Einstellungen vor. Listing 2 zeigt, wie eine Anwendungsseite aus einer anderen Anwendung erfolgreich mit SQL\*Plus in die Anwendung „999999“ importiert werden kann.

Voraussetzung für die Nutzung von „Apex\_APPLICATION\_INSTALL“ ist allerdings, dass die Apex-Exportdatei mit der Version 4.0 oder höher erstellt wurde. Ist sie älter, sollte sie zunächst in Apex 4.0 oder höher importiert und daraus wieder exportiert werden.

## URL-Syntax

Als Apex-Entwickler baut man fast ständig Apex-URLs nach dem Schema „f?p=“ zusammen. Und auch hier hat man mehr Möglichkeiten, als den meisten Entwicklern bewusst ist. So stehen viele Entwickler vor dem Problem, den Inhalt eines Elements per URL setzen zu wollen, was unproblematisch ist, jedoch soll im Inhalt ein Doppelpunkt oder ein Komma enthalten sein. Beides sind allerdings spezielle Zeichen – so ist der Doppelpunkt das Trennzeichen für die einzelnen URL-Bestandteile.

Mit dem zusätzlichen URL-Parameter „p\_sep“ kann man das Trennzeichen selbst definieren. Die Übergabe des Inhalts „Name: Apex“ per URL für das Element „P1\_PRODUKT“ funktioniert dann wie in Listing 3.

```
f?p=100!1!&SESSION.!!!P1_
PRODUKT!Produkt:Apex&p_sep=!
```

Listing 3: Apex-URL mit einem anderen Trennzeichen

Dem Komma, mit dem die Werte für mehrere Elemente voneinander getrennt sind, kann ein Backslash (\) vorangestellt werden. Es gibt also kaum Einschränkungen für die Zeichen, die per URL übergeben werden können.

Wenn man den URL-Parameter „p\_trace“ auf „YES“ stellt, erzeugt Apex für das Seiten-Rendering eine SQL-Trace-Datei. Diese wird in der „user\_dump\_destination“ des Datenbank-Servers abgelegt und enthält sehr detaillierte Informationen zu den einzelnen SQL- und PL/SQL-Anweisungen, die von Apex ausgeführt wurden. Typischerweise wird „p\_trace“ bei der Diagnose von Performance-Problemen eingesetzt.

Ab Apex 4.0 kann man die Sprach-einstellung mit dem URL-Parameter „p\_lang“ einstellen – das ist wesentlich einfacher als das umständliche Einstellen der „Language Preference“ im Browser. Mit „p\_lang“ lassen sich verschiedene URLs zur Anwendung direkt so konstruieren, dass fremdsprachige Nutzer gleich in der richtigen Sprache landen.





Abbildung 8: Die Powersuche

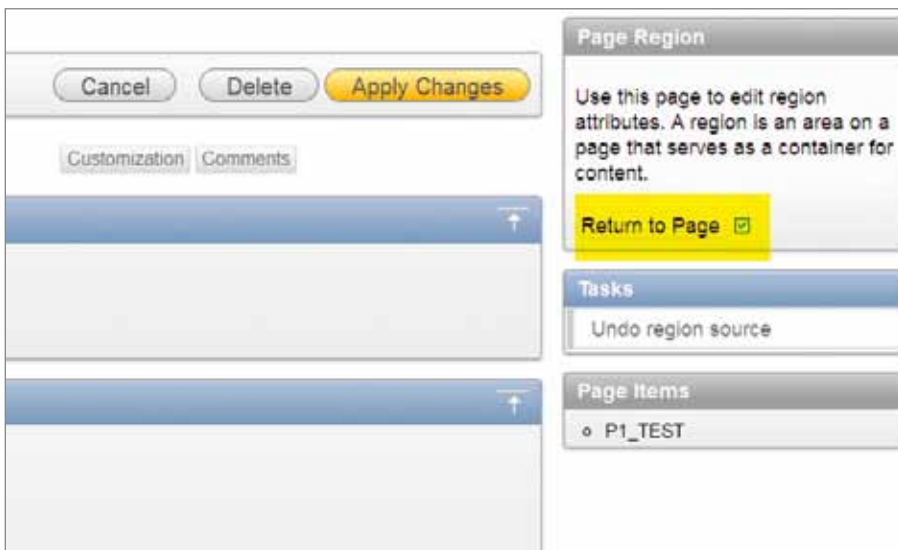


Abbildung 9: Return to Page: Beim Speichern der Änderungen landet man wieder auf der gleichen Seite



Abbildung 10: Exportieren aus der Vergangenheit

## Wertelisten

Generell sollten Wertelisten möglichst dynamisch definiert sein, denn statische Wertelisten existieren nur in Apex und können nur im Application Builder angepasst werden. Ein Fremdschlüssel, der die Werteliste auch im Datenmodell durchsetzt, ist logischerweise ebenfalls nicht möglich. Standardbeispiele für statische Wertelisten sind „Ja/Nein“ oder „Frau/Herr“, wobei sich über letzteres Beispiel sicherlich streiten lässt, denn es gibt ja noch weitere Anredeformen.

Bei der Definition einer statischen Werteliste werden die einzelnen Einträge durch ein Komma getrennt, Anzeige- und Rückgabewert innerhalb eines Eintrags durch ein Semikolon. Beim Schlüsselwort „STATIC“ sortiert Apex die Einträge automatisch, bei „STATIC2“ sind die Einträge in der definierten Reihenfolge dargestellt.

Die einzelnen Trennzeichen können jedoch selbst definiert werden: Sollen Komma oder Semikolon in den Einträgen zum Tragen kommen, braucht man andere Trennzeichen. Listing 4 zeigt Beispiele zur Definition statischer Wertelisten.

```

STATIC:Ja;Y,Nein;N
STATIC2:Ja;Y,Nein;N
STATIC(,#,%):Ja,
wirklich!%Y#Nein; ich will
nicht
STATIC2(,#,%):Ja,
wirklich!%Y#Nein; ich will
nicht

```

Listing 4: Syntax für statische Wertelisten

## Der Application Builder

Mit Apex 4.0 wurde eine wertvolle Navigationshilfe in den Application Builder eingeführt. Oben rechts liegt das Eingabefeld für „Powersuche“ (siehe Abbildung 8). Mit dem hier eingegebenen Begriff wird die gesamte Anwendungsdefinition durchsucht. Das ist sehr nützlich, wenn man zum Beispiel wissen möchte, wo die Werteliste „LOV\_KUNDEN“ verwendet wird. Die Powersuche kann aber noch mehr: Gibt man zum Beispiel die Apex-Syn-

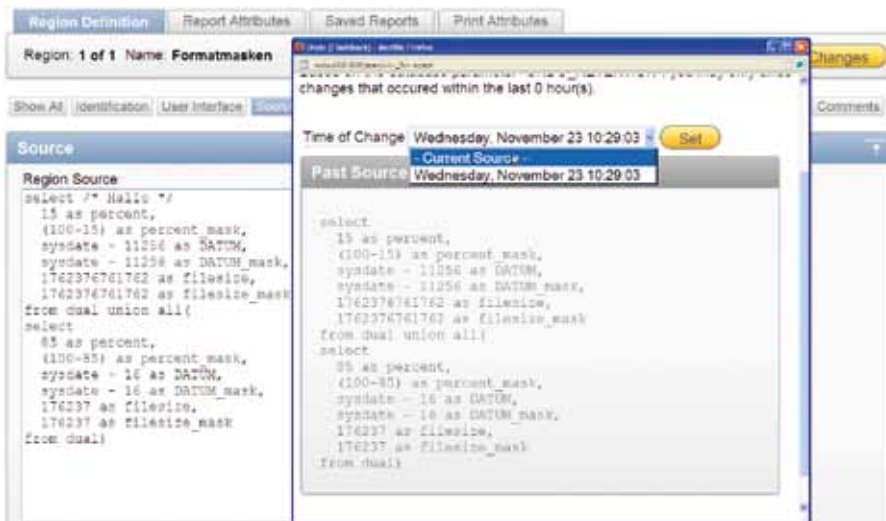


Abbildung 11: Änderungen an der Regionsquelle rückgängig machen

tax „[Anwendungs-ID]:[Page-ID]“ ein, so navigiert der Application Builder direkt dorthin. Das klappt natürlich nur dann, wenn sich die Anwendung im gleichen Workspace befindet, andernfalls landet man auf der Login-Seite.

Vielfach ist man in der Situation, das Aussehen eines Templates oder einer Region ändern zu wollen. Dabei ist es üblich, dass man sich die durchgeführten Änderungen immer wieder ansieht. Typischerweise ist die Anwendung dabei in einem zweiten Browser-Fenster geöffnet.

Das in Abbildung 9 dargestellte „Return to Page“ ist hier extrem nützlich – bei Anklicken wird man nach dem Speichern der Änderungen nicht mehr zurück zur Seitenübersicht, sondern nochmals zur gleichen Seite geführt. Man kann also direkt die nächste Änderung machen. Das spart eine Menge Klicks bei der Arbeit.

### Blick in die Vergangenheit

Apex nutzt sehr intensiv das Flashback Query. Damit kann man eine Anwendung so exportieren, wie sie beispielsweise vor fünf Minuten aussah (siehe Abbildung 10). Wie weit man in die Vergangenheit zurückgehen kann, hängt von der Konfiguration der Datenbank (des UNDO-Tablespace) ab, Zeiträume bis circa zehn Minuten sollten aber auf nahezu allen Systemen möglich sein. Mit einem solchen Ex-

port lassen sich versehentlich gelöschte Komponenten einfach wiederherstellen – vorausgesetzt, man bemerkt den Fehler schnell genug.

Undo-Funktionen mithilfe von Flashback sind auch in den Dialogen zur Regionsbearbeitung vorhanden. Hat man sein Berichts-SQL gerade umfangreich geändert und es funktioniert nicht mehr, kommt man sehr einfach zur alten Version zurück. Klickt man auf der rechten Seite „Undo Region Source“ an, so erscheint der in Abbildung 11 dargestellte Dialog.

Flashback Query ist auch bei interaktiven Berichten standardmäßig aktiviert. Endanwender können sich die Daten also ansehen, so wie sie vor wenigen Minuten aussahen.

### Fazit

Apex bietet inzwischen eine Fülle von Funktionen und Möglichkeiten an. Wie immer gibt es dabei Funktionen, die in Vorträgen, Konferenzen und Howto-Dokumenten im Mittelpunkt stehen, und solche, die kaum bekannt sind. Gerade diese sind aber oft sehr nützlich und deren Kenntnis kann im Bedarfsfall einiges an Entwicklungsaufwand einsparen.

### Weitere Informationen

Auf den Webseiten von Oracle und im Internet ist umfangreiches Materi-

al auch in deutscher Sprache vorhanden:

1. Deutschsprachige Apex-Community: <http://tinyurl.com/apexcommunity>
2. Apex im OTN: <http://otn.oracle.com/apex>
3. Oracle-Datenbank auf Twitter: <http://twitter.com/OracleBUDB>

Carsten Czarski  
 ORACLE Deutschland B.V. & Co. KG  
 carsten.czarski@oracle.com  
<http://twitter.com/cczarski>  
<http://sql-plsql-de.blogspot.com>



Unsere Inserenten	
BERENBERG BANK www.berenberg.de	S. 15
Hunkler GmbH & Co. KG www.hunkler.de	S. 3
WIN-Verlag GmbH & Co. KG win-verlag.de	U 2
imining gmbh www.imining.de	S.13
KeepTool GmbH www.keeptool.com	S. 31
Libelle AG www.libelle.com	S. 7
MuniQsoft GmbH www.muniqsoft.de	S. 55
ORACLE Deutschland B.V. & Co. KG www.oracle.com	U 3
PROMATIS software GmbH www.promatis.de	S. 19
Trivadis GmbH www.trivadis.de	U 4
usb GmbH www.usb-muc.com	S. 23