

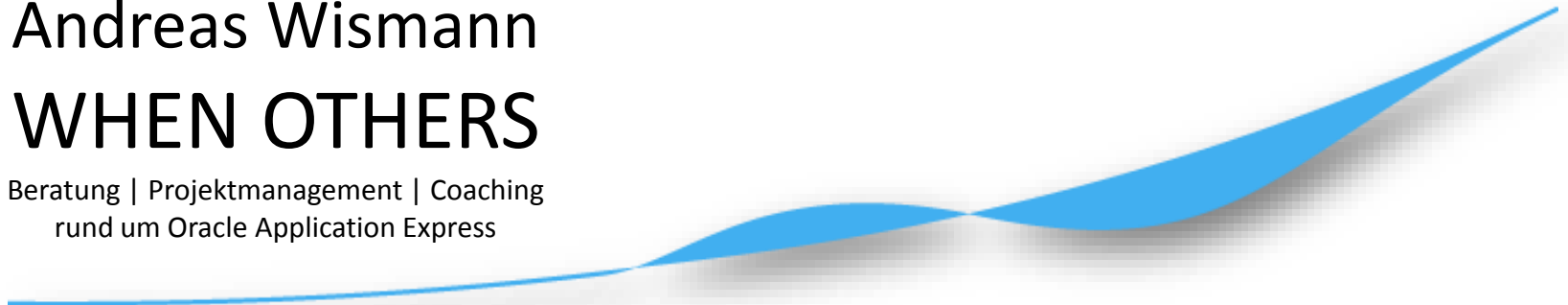
# "Jedes Byte zählt"

## Tuning-Rezepte für APEX-Anwendungen in der Cloud

Andreas Wismann

**WHEN OTHERS**

Beratung | Projektmanagement | Coaching  
rund um Oracle Application Express



# »Wo« kann es langsam sein?

- Webserver
- APEX-Konfiguration
- Datenübertragung
- Menge an HTML-Code
- Menge an Daten
- Geschwindigkeit der Skripte (SQL, PL/SQL, HTML, CSS, JavaScript)



# "Cloud"?

- kein Einfluss auf die Parameter des Webservers und der APEX-Instanz
- "nur" Einfluss auf die Programmierung

# messen...

- wie "groß" ist meine Seite?
- wie "schnell" ist meine Seite?
- Responsezeiten mit dem Kunden festlegen



# Welche Region ist langsam?

- Session Debug View zeigt die Prozessdauer an, aber nicht "auf einen Blick"

Rezept Nr. 1

# Welche Region ist langsam?

- Substitutionsvariable #TIMING#  
in jeden Region Footer einbauen

Kosten

10 Sekunden tippen

- Als Benutzerinfo:

```
<p class="timing">#ROWS_FETCHED#  
von #TOTAL_ROWS# Zeile(n) abgerufen  
in #TIMING# Sekunden</p>
```

# SQL des Schreckens

```
declare
  v_count number;
begin

  select count(*)
    into v_count
    from tabelle
   where gueltig_bis < sysdate;

  if v_count > 0 then
    delete from tabelle
      where gueltig_bis < sysdate;
  end if;

end;
```

Rezept Nr. 2

# Langläufer-SQL

- schon auf Seite 100 (während LOGIN) ausführen
- kritisch hinterfragen
- optimieren

Kosten

Regelmäßiger Code-Review



# Langsame Interaktive Reports

- Report erscheint nicht sofort beim Öffnen der Seite
  - Browser "tut nichts"
  - subjektives Empfinden: »APEX ist schuld«
  - besser wäre: »der Report ist schuld«
- 
- Verhindern, dass Benutzer beim Anklicken einer Seite »warten« müssen

# Langsame Interaktive Reports

- Hidden Item in die WHERE-Klausel einbauen:

```
... where :P1000_SHOW_SQL > 0  
and ...
```

- Default-Anfangswert: 0 oder NULL
- Durch Interaktion auf 1 setzen

Kosten

30 Minuten Programmierung

# Lang laufende JOINS und Sub-Selects

- Subselect:  
where ID in (SELECT ID FROM ... where ... IN (...))
- Subselect ersetzen durch initial zu füllende Collection:  
where ID in (select N001 from apex\_collections  
where collection\_name = '<SUBSELECT\_1')

# Lang laufende JOINS und Sub-Selects

- SQL-Zwischenergebnisse in APEX\_COLLECTIONS parken
- Individuelle Zwischenergebnisse pro Benutzer sind dadurch auch möglich

## Kosten

- ins Thema einlesen
- 15 Minuten Programmierung

# Viele / große Web-Dateien

- JavaScript- und CSS-Dateien verzögern das Laden der ersten Anwendungsseite
- Bilder erscheinen verzögert

## Rezept Nr. 5

# Viele / große Web-Dateien

- Sämtliche Dateien auf der LOGIN-Seite vorausladen
- CDN (Content Distribution Network) verwenden

## Kosten

- 5 Minuten Programmierung
  - 1 Klick

# Dashboard lädt langsam

- ausgerechnet die Haupt-Anwendungsseite...
- viele Benutzer
- Übersicht über sehr große Datenmengen

## Rezept Nr. 6

# Dashboard lädt langsam

- Schalten Sie das APEX Regions-Caching ein
- Invalidieren Sie den Cache beizeiten
- Insbesondere der Kurzzeit-Cache beschleunigt Seiten für "herumklickende" Benutzer

Kosten

2 Klicks



# Page 0

- Region, die alle Table Statistics anzeigt
- definieren auf Page 0 ("Global Page")
- einblenden nur für Entwickler (per Authorization Scheme)

## Kosten

- Report: 5 Minuten
- Authorization Scheme: 2 Minuten

# Viele Regionen oder Items mit gleichen Conditions

- zwei oder mehr Regionen haben dieselbe Condition
- Conditions sind komplex

# Viele Regionen oder Items gleichen Conditions

- Parent-Region mit dieser Condition erstellen (logischer Container)
- andere Regionen und Items unterordnen
- Parent-Region hat selbst kein "sichtbares" Template

Kosten

"Aufräumen": 15 Minuten pro Seite

# Bedingungen für Prozesse

```
if ist_vertrag_gueltig
  (:P500_VERTRAGSNUMMER) = TRUE
and :P500_KUNDENNUMER IS NOT NULL
then
  -- viel Code, viele Aktionen ...
  select ... ;
  select ... ;
end if;
```

**3  
Fehler**

# Bedingungen für Prozesse

```
if ist_vertrag_gueltig
  (:P500_VERTRAGSNUMMER)
and :P500_KUNDENNUMER IS NOT NULL
then
  -- viel Code, viele Aktionen ...
  select ... ;
  select ... ;
end if;
```

**2**  
**Fehler**

# Bedingungen für Prozesse

```
if :P500_KUNDENNUMER IS NOT NULL
and ist_vertrag_gueltig
    (:P500_VERTRAGSNUMMER)
then
    -- viel Code, viele Aktionen ...
    select ... ;
    select ... ;
end if;
```

**1  
Fehler**

# Bedingungen für Prozesse

```
if ist_gueltige_kundennummer(:P500_KUNDENUMMER)
then
-- viel Code, viele Aktionen ...
  select ... ;
  select ... ;
end if;
```

## Conditions

When Button Pressed

Condition Type

Value of Item / Column in Expression 1 Is NOT NULL

[PL/SQL] [item / column=value] [item / column not null] [item / column null] [request=e1] [page in] [page not in] [e:

Expression 1

P500\_KUNDENUMMER



# Bedingungen für Prozesse

- Äußere Bedingungen?  
IMMER als Condition definieren
- Erspart unnötiges Parsen des Programmcodes
- APEX wertet Conditions sehr schnell aus
- Shortcut-Bedingungen im Programmcode an erste Stelle platzieren

Kosten

- zusätzlich: keine



# Schnelle HTML-Tabellen

- SQL-Abfrage in einem Package ausführen
- Alle Zeilen per PL/SQL-Loop durchlaufen
- Jede Zeile als HTML formulieren
- gesamtes Tabellen-HTML zum Browser senden

# Schnelle HTML-Tabellen

```
htp.p ('<table>');

for c in (SELECT ... ) loop
    htp.p ('<tr>'
        || '<td>' || c.spalte1 || '</td>'
        || '<td>' || c.spalte2 || '</td>'
        || '</tr>');
end loop;

htp.p ('</table>');
```

Rezept Nr. 10

# Schnelle HTML-Tabellen

- Reports,
  - die der Benutzer nicht konfigurieren muss
  - die nicht auf APEX Report-Features angewiesen sind
- ... direkt per HTML aus der Datenbank ausgeben

Kosten

30 Minuten pro Report

... weitere Rezepte und Beispiele im Vortrag ...

Andreas Wismann

**WHEN OTHERS**

Beratung | Projektmanagement | Coaching  
rund um Oracle Application Express

