



Wie lege ich als Entwickler die Datenbank definitiv lahm?

SQL-Performance-Fallstricke

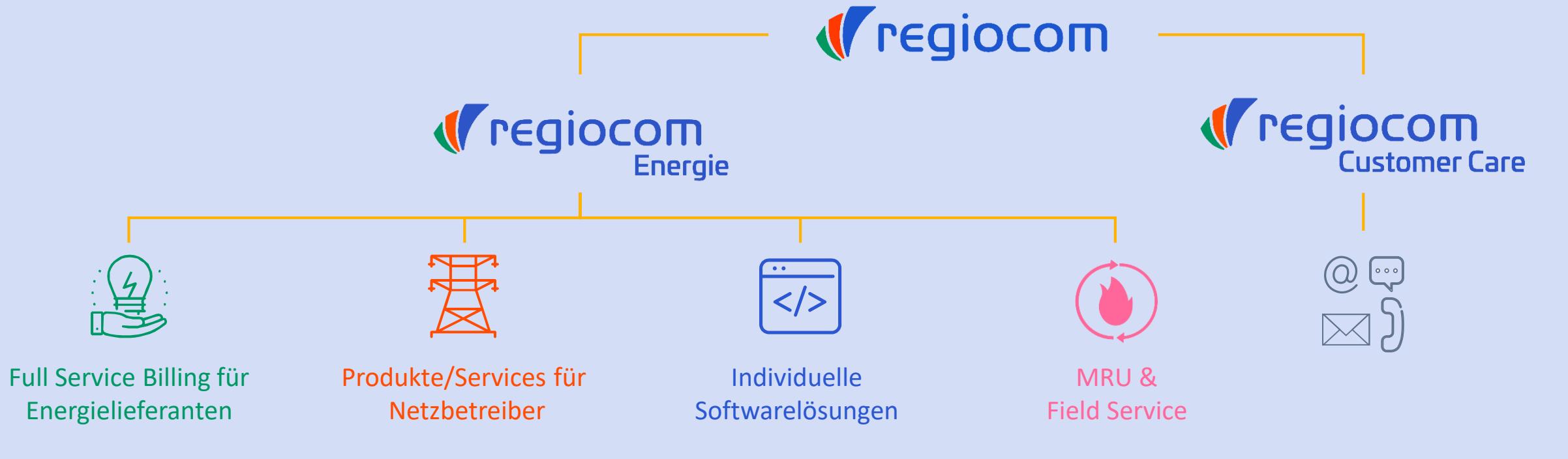
Über mich

Even Langer

- PO und Oracle Certified Professional
- Arbeitsschwerpunkte:
 - Performancetuning
 - Verarbeitung von Massendaten
 - Applikationsdesign/Systemarchitektur
 - Moderation TechTalk-Vortragreihe
- Hobbies:
 - People-Fotografie www.evenlanger.de
 - Reisen



Die Geschäftsfelder und Partnerunternehmen



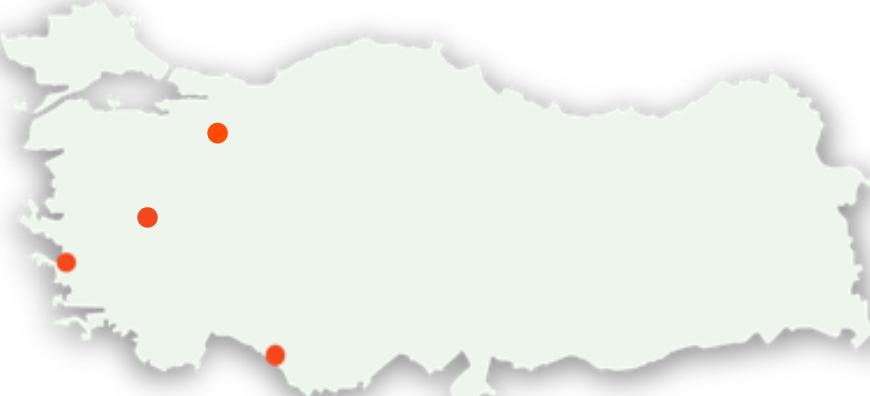
> 6500 Mitarbeiter



Deutschland



Bulgarien



Türkei



Österreich

Griechenland

Motivation

Beispielabfrage auf Prod-DB

```
select *  
from vet_lieferung,  
     net_geschaeftspartner,  
     lok_marktlokation,  
     crm_bestellung,  
     vet_netznutzung;
```

Anzahl Datensätze

897158
968618
1550372
921359
1659934

Ergebnis $2*10^{30}$ Datensätze



Vorbemerkung

Oracle-DB liest keine Datensätze,
sondern Blöcke (Standard 8KB)

Oracle-DB entscheidet sich meist für einen Full Table Scan,
wenn mehr als 6-8% der Daten gelesen werden sollen.

Vorbemerkung

Wichtigste Optimierungsregeln:

Passendes Datenmodell

Nicht benötigte Abfragen entfernen

Selektivität, Selektivität, Selektivität!

Optimierungsmöglichkeiten

ja

Hohe Selektivität erwartet

nein

Indexe
Statistiken
Constraints
Hints

Komprimierung Tabellen/Indexe
Partitionierung
Alle relevanten Attribute im Index
Aufsplittung des Statements
Bitmap-Indexe (DWH)

Demo Indexe



Indexe

- Grundfragen:
 - Nach welchen Attributen wird gefiltert?
 - Wie ist die Selektivität dieser Attribute?
 - Gibt es bereits passende Indexe?
 - Wird „like“ verwendet?
- Kombinierte Indexe
- Function-based-Indexe
- Foreign-Key-Indexe auf referenzierenden Attributen
- Bitmap-Indexe (in der Regel für DWH-Einsatz)

Indexe

- Implizite Typ-Konvertierungen vermeiden!
- Reorganisation von Indexen, wenn sie entarten!
- Wo sind Indexe meist nicht notwendig:
 - Migrationen
 - DWH-Beladung
 - Berichte über alle Daten
 - einmalige Ausführungen/Analysen
 - Redundante Indexe

Demo Constraints



Constraints

- Helfen dem Optimierer die Datenkonstellation besser einzuschätzen
- Unique-Constraints:
 - Index Unique Scan statt Index Range Scan
 - Optimierer weiß, dass es nur einen Treffer geben kann
- Foreign-Key-Constraints:
 - Optimierer weiß, dass Parent existiert
 - Verhindern Deadlocks ORA-00060
- Check-Constraints:
 - NOT NULL Constraints können Table Full Scan verhindern
 - Checks wie `ort in ('BERLIN', 'MÜNCHEN', 'HAMBURG')` helfen Optimierer bei Abfragen der Art `where ort = 'MAGDEBURG'`



Demo Statistiken

Statistiken

- Entscheidend bei großen Datenmengen/vielen Joins!
- Werden durch Oracle prinzipiell in der Nacht erstellt
- Manuelle Berechnung:
 - bei Migrationen
 - nach größeren Importen/Datenänderungen
 - wenn nicht alle relevanten Attribute Histogramme erhalten
- siehe Oracle-Package `dbms_stats`
- siehe `user_tab_modifications`

TABLE_NAME	INSERTS	UPDATES	DELETES	TIMESTAMP
ADM_AUDIT_DDL	133	0	0	24.05.2023 15:42:41
ADM_CODE_STATISTIK	0	0	0	24.05.2023 16:58:26
CTRL_GRANTS_RULES	1	0	0	19.01.2023 11:11:36
CTRL_LOG	138	0	0	24.05.2023 14:27:32
CTRL_SESSIONINFO_SYSTEMVARS	0	0	0	24.05.2023 16:58:26
EFC_BZR_DETAIL	517771	0	28489	24.05.2023 08:12:02

Demo Komprimierung



Komprimierung

- Tabellen:
 - Kann global für einen Tablespace eingestellt werden
 - Advanced Compression: Extra Cost Option
- Indexe:
 - Kombinierte Indexe -> Index Key Compression
 - Advanced Index Compression: Extra Cost Option
- LOBs:
 - Ist unabhängig von der Tabellenkomprimierung
 - Muss manuell eingestellt werden
 - Ersparnis bis 50%

Joins

- ANSI-Schreibweise wechseln
- Outer Joins verhindern Index-Zugriff auf die optionalen Tabellen
- Ergebnismenge bei Cross Joins beachten
- analytische Funktionen verwenden
- kippende Ausführungspläne bei Statements mit vielen Joins:
 - Fehler in der Abschätzung der Ergebnismengen potenzieren sich
 - Statistiken kontrollieren
 - Ggf. spezielle Statistiken für Attributkombinationen
 - Verwendung von with-Klauseln mit Hint materialize prüfen
 - Aufsplitten großer Statements

Weitere Möglichkeiten

- Hints
- Partitionierung:
 - Verschiedene Methoden verfügbar (List, Range, Hash)
 - Passendes Attribut finden
 - Extra Cost Option
- Materialized Views
- Trigger vermeiden
- PLSQL:
 - mengenbasiert arbeiten!
 - BULK-Operationen verwenden
- Bind-Variablen verwenden/erzwingen

Oracle-Werkzeugkasten

- Ermittlung der Problemfälle:
 - Enterprise Manager
 - Dictionary-Views: v\$sql, v\$active_session_history, v\$sql_plan, v\$sql_bind_capture, ...
 - Automatic Workload Repository
 - AWR-Views: dba_hist_sqlstat, dba_hist_sqltext, dba_hist_active_sess_history, ...
 - Package DBMS_HPROF – Profiling von PLSQL-Ausführungen
 - Tracefiles
- Lösungsvorschläge:
 - SQL Tuning Advisor
 - SQL Access Advisor
 - Segment Advisor
 - Compression Advisor
 - Plan Baseline Management



Vielen Dank für eure
Aufmerksamkeit

Even Langer
Product Owner
Oracle Certified Professional
even.langer@regiocom.com

Fragen