

---

Sind wir eigentlich  
ganz dicht?



Software

Eero Mattila  
Principal Systems Consultant

[eero.mattila@quest.com](mailto:eero.mattila@quest.com)

---

# Quest Software, now a part of Dell



# Agenda

- Warum überhaupt komprimieren?
- Überblick der Möglichkeiten
- Indexkomprimierung
- Tabellenkomprimierung
- Komprimierung von unstrukturierten Daten
- Exadata: Hybrid Columnar Compression
- Backup-Komprimierung
- Fazit



# Warum überhaupt komprimieren?

- Datenexplosion
  - Wir wissen immer mehr...
  - Gesetzliche Bestimmungen
- Speicherplatz kostet Geld
- Performance muss stimmen



# Überblick

- Index Key Compression seit 8i
- Tabellenkomprimierung seit 9g R2
- Unstrukturierte Daten seit 10g
- Advanced Compression Option seit 11g
- Hybrid Columnar Compression in Exadata V2



# Indexkomprimierung

- Index Key Compression seit 8.1.5
- Kostenlos verfügbar in allen Editionen
- Non-unique single column Indizes
- Unique und non-unique multi column Indizes
- B\*Tree- Indizes und Index Organized Tables (IOT)
- Transparent für Anwendungen



# Indexkomprimierung (2)

- Eliminierung redundanter Schlüsselwerte
- Kann Platzbedarf und I/O erheblich reduzieren

## Nicht komprimierter Index

```
0:Hund:ROWID
1:Katze:ROWID
2:Maus:ROWID
3:Hund:ROWID
4:Hund:ROWID
5:Maus:ROWID
6:Katze:ROWID
7:Maus:ROWID
8:Hund:ROWID
```

## Komprimierter Index

```
Präfix  0:0:ROWID
0:Hund  1:1:ROWID
1:Katze  2:2:ROWID
2:Maus   3:0:ROWID
         4:0:ROWID
         5:2:ROWID
         6:1:ROWID
         7:2:ROWID
         8:0:ROWID
```



# Indexkomprimierung (3)

- Definition beim Erstellen oder auch nachträglich:

```
CREATE INDEX index ON tabelle (spalte1, ...) COMPRESS x;
```

```
ALTER INDEX index REBUILD [ONLINE] COMPRESS x;
```

x = Anzahl der zu komprimierenden führenden Spalten



# Indexkomprimierung (4)

- Ermitteln der optimalen Komprimierungsrate

```
ANALYZE INDEX comp_test VALIDATE STRUCTURE;
```

```
SELECT blocks, opt_cmpr_count, opt_cmpr_pctsave  
FROM index_stats;
```

```
BLOCKS OPT_CMPR_COUNT OPT_CMPR_PCTSAVE  
-----  
988 2 37
```

OPT\_CMPR\_COUNT gibt die Anzahl der zu komprimierenden Spalten an,  
OPT\_CMPR\_PCTSAVE die zu erwartende Platzersparnis in %



# Indexkomprimierung (5)

- Probe aufs Exempel

```
ALTER INDEX comp_test REBUILD COMPRESS 2;  
ANALYZE INDEX comp_test VALIDATE STRUCTURE;
```

```
SELECT blocks, opt_cmpr_count, opt_cmpr_pctsave  
FROM index_stats;
```

```
BLOCKS OPT_CMPR_COUNT OPT_CMPR_PCTSAVE  
-----  
621 2 0
```

- Vorsicht mit ANALYZE INDEX ... VALIDATE STRUCTURE
  - DML-Lock!



# Tabellenkomprimierung

- Eingeführt mit Oracle9i R2 für Bulk-Load-Operationen
  - Direct Path Load, CTAS, paralleles DML, Insert APPEND
  - Nur interessant für Datawarehouse-Umgebungen
- Oracle 11g: Advanced Compression Option
  - Separate Lizenz
  - Komprimierung für OLTP
  - Basic Compression (Bulk-Operationen) weiterhin kostenlos
- Komprimierung auf Blockebene
  - Wiederkehrende Spaltenwerte in Symboltabelle im Blockheader
  - Ersparnis bei Speicherplatz sowie logischem und physikalischem I/O



# Tabellenkomprimierung (2)

- Syntax:

```
CREATE TABLE [...] COMPRESS FOR [OLTP|BASIC];
```

```
ALTER TABLE tabelle COMPRESS FOR [OLTP|BASIC];
```

- Vorhandene volle Blöcke werden nicht komprimiert

- Komprimierung greift erst, wenn Sätze gelöscht und eingefügt werden
- Wenn die Tabelle gleichzeitig reorganisiert wird, greift Komprimierung sofort:

```
ALTER TABLE tabelle MOVE COMPRESS FOR [OLTP|BASIC];
```

```
Oder DBMS_REDEFINITION (online)
```

- Standardeinstellung für Tablespace

```
CREATE|ALTER TABLESPACE [...] DEFAULT COMPRESS FOR [OLTP|BASIC];
```



# Tabellenkomprimierung (3)

- Beispiel:

```
CREATE TABLE doag_normal  
AS SELECT * FROM dba_objects;
```

```
CREATE TABLE doag_compr COMPRESS FOR OLTP  
AS SELECT * FROM dba_objects;
```

```
SELECT segment_name, blocks  
FROM user_segments  
WHERE segment_name LIKE 'DOAG%';
```

SEGMENT_NAME	BLOCKS
DOAG_COMPR	<b>384</b>
DOAG_NORMAL	<b>1152</b>



# Tabellenkomprimierung (4)

- Welche Tabellen komprimieren?
- 11g R2: DBMS\_COMPRESSION.GET\_COMPRESSION\_RATIO
- Früher: DBMS\_COMP\_ADVISOR (auf OTN verfügbar)
- Nur BASIC, OLTP, HCC
  - Keine Unterstützung für Indizes, unstrukturierte Daten
- Automatic Segment Advisor
  - Mit 11g R2 Empfehlungen für Komprimierung
  - Diagnostic und Tuning Packs zusätzlich erforderlich



# Was kostet die Komprimierung?

- Advanced Compression Option ist lizenzpflichtig
- Keine Zusatzinstallation – immer da in Enterprise Edition
  - Zauberwort COMPRESS genügt...
- Hidden Parameter `_OLTP_COMPRESSION=FALSE`
- CPU statt I/O und Speicherplatz?
  - Kein messbarer Overhead beim Lesen
  - Performance-Steigerung durch geringeren I/O (weniger Blöcke)
  - Meist geringfügiger Overhead bei DML -> Testen!
  - Asynchrone Komprimierung, wenn bestimmter Blockfüllgrad erreicht
- Transparent für Anwendungen



# Komprimierung unstrukturierter Daten

- Oft ideale Kandidaten, wenn nicht per se komprimiert
- Oracle10g: UTL\_COMPRESS für RAW, BLOB, BFILE
- SecureFiles in Oracle11g
  - Setzt ACO voraus
  - Komprimierung in 3 Stufen: LOW, MEDIUM, HIGH
  - CPU-Overhead vs. Platzersparnis
  - Deduplizierung





# Exadata Hybrid Columnar Compression

- Logische Komprimierungsgruppen (Compression Unit)
  - Umfasst mehrere Blöcke
  - Jede Spalte wird einzeln komprimiert
  - Wiederkehrende Daten in gleichen Blöcken
- Optimiert für Warehouse- oder Archiv-Systeme
  - `COMPRESS FOR QUERY` – optimiert für Abfrage-Performance
  - `COMPRESS FOR ARCHIVE` – optimiert für Platzersparnis
- Sehr hohe Komprimierung bei optimaler Performance
  - Faktor 15: aus 1 TB werden ~65 GB



# Backup-Komprimierung

- RMAN

- Blockweise, physikalische Datensicherung

```
CONFIGURE DEVICE TYPE [DISK|SBT] BACKUP TYPE TO COMPRESSED  
BACKUPSET;
```

```
CONFIGURE COMPRESSION ALGORITHM '[BASIC|LOW|MEDIUM|HIGH]'
```

- BASIC kostenfrei, sonst Advanced Compression Option

- Data Pump Export

- „Logisches“ Backup
- Parameter `COMPRESSION [ALL|METADATA_ONLY|DATA_ONLY]`



# Backup-Komprimierung (2)

- Dekomprimierung bei Restore bzw. Import transparent
- Performance abhängig von vielen Faktoren
  - Charakter der Daten, Netzwerk, CPU-Ressourcen usw.
- Gründlich testen, vor allem Restore!
  - Was nützt die kleinste Sicherung, wenn die Wiederherstellung ewig dauert...



# Fazit

- Komprimierung kann gute Dienste leisten
- Der Möglichkeiten sind viele
- Vorsicht – oft Extrakosten
- Performance-Nachteile möglich, wenn auch selten
- Gründlich testen



Herzlichen Dank für Ihre Aufmerksamkeit!

Welche Fragen haben Sie?

[eero.mattila@quest.com](mailto:eero.mattila@quest.com)

