

Compression

DOAG Webinar
10.06.2016
Klaus Reimers

info@ordix.de

www.ordix.de

ORDIX[®] best practice
einfach. gut. beraten.



- Überblick
- Table Compression
 - Basic Compression
 - Advanced Compression
- Index Compression
 - Basic Compression
 - Advanced Compression
- LOB Compression und Deduplication
- rman Compression
- DataPump Compression
- Sonstiges

Überblick

Funktionalität	seit Version	SE2	EE	Compression Option	In-Memory Option
Basic Table Compression	9i		X		
Advanced Table Compression	11g		X	X	
Basic Index Compression	8i	X			
Advanced Index Compression	12c		X	X	
LOB Compression / Deduplication	11g		X	X	
rman Basic Compression	8i	X			
rman Advanced Compression	11g		X	X	
DataPump Compression	11g		X	X	
Network Compression	11g/12c		X	X	
Information Lifecycle Management	12c		X	X	
In-Memory Compression	12c		X		X

Agenda

- Überblick
- Table Compression
 - Basic Compression
 - Advanced Compression
- Index Compression
 - Basic Compression
 - Advanced Compression
- LOB Compression und Deduplication
- rman Compression
- DataPump Compression
- Sonstiges

Table Compression

Technik

Prinzip der Speicherung von Daten in einer *normalen* Tabelle:

2005	Jan ULLRICH	3	T-MOBILE TEAM
2005	Ivan Basso	2	TEAM CSC
2005	Lance ARMSTRONG	1	DISCOVERY CHANNEL TEAM
2004	Ivan Basso	3	TEAM CSC
2004	Andreas KLÖDEN	2	T-MOBILE TEAM
2004	Lance ARMSTRONG	1	US POSTAL-BERRY FLOOR
2003	Alexandre VINOKOUROV	3	TEAM TELEKOM
2003	Jan ULLRICH	2	TEAM BIANCHI
2003	Lance ARMSTRONG	1	US POSTAL-BERRY FLOOR

Prinzip der Speicherung von Daten in einer *komprimierten* Tabelle:

02	04	08	0A	
02	05	07	0B	
02	03	06		DISCOVERY CHANNEL TEAM
01	05	08	0B	
01		Andreas KLÖDEN	07	0A
01	03	06	09	
00		Alexandre VINOKOUROV	08	TEAM TELEKOM
00	04	07		TEAM BIANCHI
00	03	06	09	

S0B				TEAM CSC
S0A				T-MOBILE TEAM
S09				US POSTAL-BERRY FLOOR
S08				3
S07				2
S06				1
S05				Ivan Basso
S04				Jan ULLRICH
S03				Lance ARMSTRONG
S02				2005
S01				2004
S00				2003

Daten-Tabelle

Symbol-Tabelle

Table Compression (Basic)

Syntax

```
-- Anlegen einer komprimierten Tabelle
-- Oracle 9i + 10g
CREATE TABLE COMP1(Sp1 number) COMPRESS;
-- Oracle 11g
CREATE TABLE COMP2(Sp1 number) COMPRESS BASIC;
CREATE TABLE COMP2(Sp1 number) COMPRESS FOR DIRECT_LOAD OPERATIONS;
-- Oracle 12c
CREATE TABLE COMP3(Sp1 number) ROW STORE COMPRESS BASIC;
```

```
-- Umwandeln einer bestehenden Tabelle
ALTER TABLE COMP1 MOVE COMPRESS;
ALTER TABLE COMP1 MOVE COMPRESS BASIC;
ALTER TABLE COMP1 MOVE COMPRESS FOR DIRECT_LOAD OPERATIONS;
ALTER TABLE COMP1 MOVE ROW STORE COMPRESS BASIC;
```

Agenda

- Überblick
- Table Compression
 - Basic Compression
 - Advanced Compression
- Index Compression
 - Basic Compression
 - Advanced Compression
- LOB Compression und Deduplication
- rman Compression
- DataPump Compression
- Sonstiges

Table Compression (OLTP)

Technik

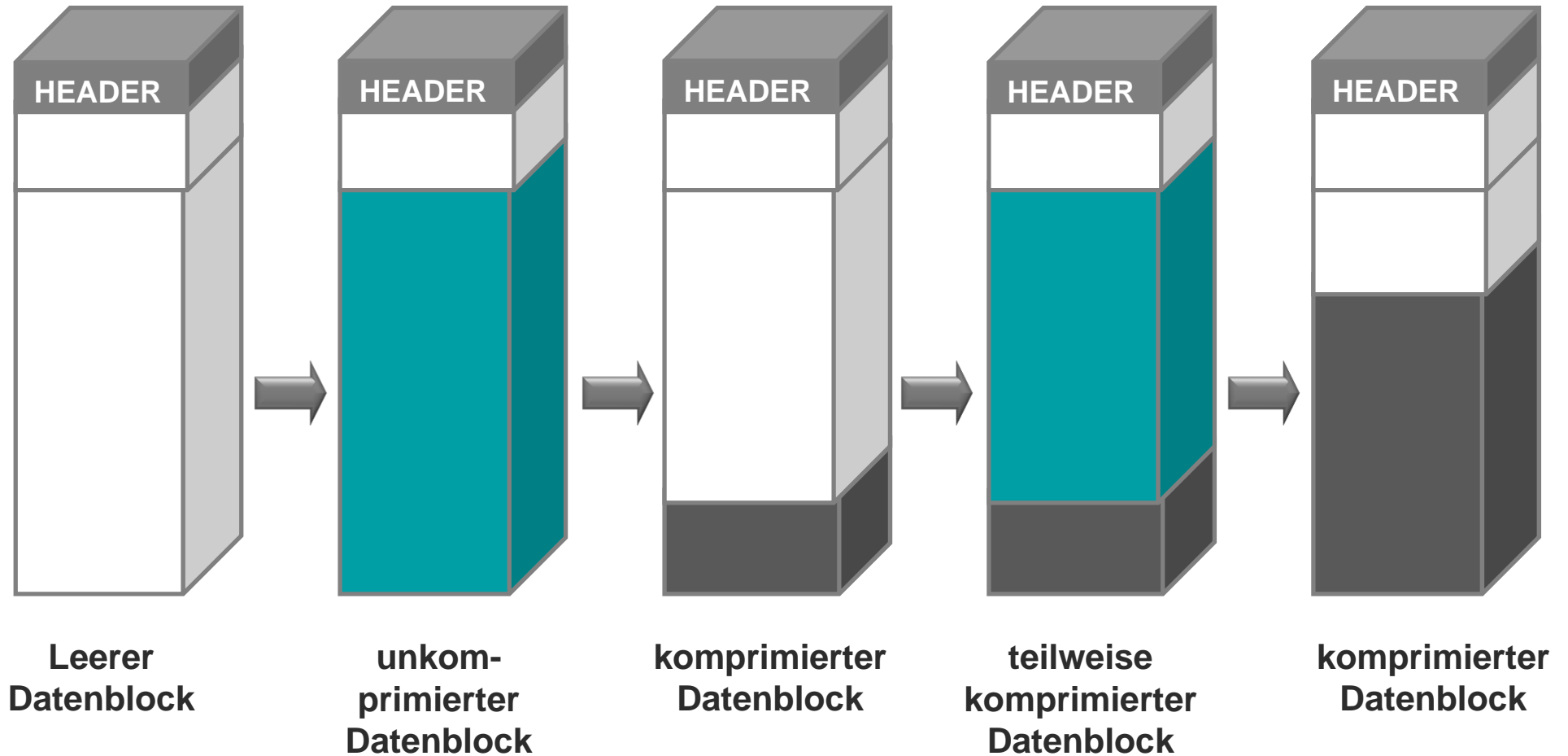


Table Compression (Advanced)

Syntax

```
-- Anlegen einer komprimierten Tabelle
-- Oracle 11g
CREATE TABLE COMP1(Sp1 number) COMPRESS FOR OLTP;
CREATE TABLE COMP2(Sp1 number) COMPRESS FOR ALL OPERATIONS;
-- Oracle 12c
CREATE TABLE COMP3(Sp1 number) ROW STORE COMPRESS ADVANCED;
```

```
-- Umwandeln einer bestehenden Tabelle
ALTER TABLE COMP1 MOVE COMPRESS FOR OLTP;
ALTER TABLE COMP1 MOVE COMPRESS FOR ALL OPERATIONS;
ALTER TABLE COMP1 MOVE ROW STORE COMPRESS ADVANCED;
```

Realer Test beim Kunden

Aktion	Details	unkomprimiert	komprimiert
Größe der Tabelle		2910 MB	749 MB
Lesen mit count(*)	Laufzeit	67 sec	30 sec
	Gelesene Blöcke	3719545	952518
Massen Update (200000 rows)	Laufzeit	13:11 min	3:05 min
	Gelesene Blöcke	7382951	952543
Massen Insert (1 Mio rows)	Laufzeit	2:28 min	5:16 min
Massen Delete (200000 rows)	Laufzeit	4:48 min	2:09 min
	Gelesene Blöcke	75875428	57096136

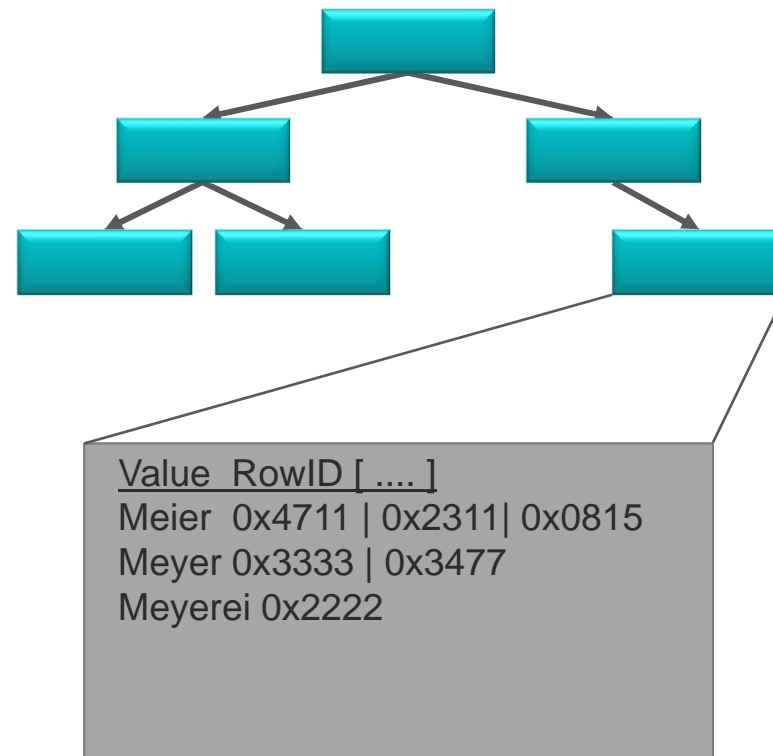
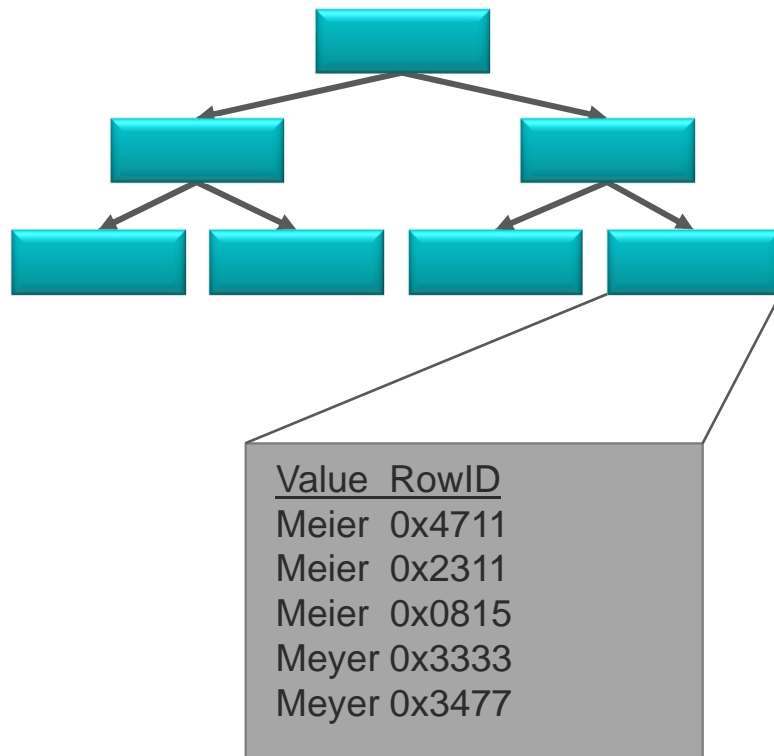
Agenda

- Überblick
- Table Compression
 - Basic Compression
 - Advanced Compression
- Index Compression
 - Basic Compression
 - Advanced Compression
- LOB Compression und Deduplication
- rman Compression
- DataPump Compression
- Sonstiges

Index Compression

Technik

- Doppelte Indexwerte werden nur einmal gespeichert
- Bei vielen doppelten Werten Reduzierung der Anzahl an Blättern



Index Basic Compression

Syntax

- Analyse:
 - Mit eigenem SQL:
 - `num_rows / distinct_keys > 1000`
 - `leaf_blocks > 500`
 - Advisor

```
analyze index index_no_comp  
  validate structure;
```

- Durchführung:

```
alter index index_no_comp  
  rebuild compress online;
```

Agenda

- Überblick
- Table Compression
 - Basic Compression
 - Advanced Compression
- Index Compression
 - Basic Compression
 - Advanced Compression
- LOB Compression und Deduplication
- rman Compression
- DataPump Compression
- Sonstiges

Index Advanced Compression

- Einsetzbar für alle Indizes
- Benötigt die Compression-Option
- Kandidaten:
 - Wenig Duplikate bei der führenden Spalte
 - Ungleichmäßige Werteverteilung

```
CREATE INDEX ind1 ON tabl (sp)

    compress advanced low;

ALTER INDEX ind1

    REBUILD compress advanced low;
```

Agenda

- Überblick
- Table Compression
 - Basic Compression
 - Advanced Compression
- Index Compression
 - Basic Compression
 - Advanced Compression
- LOB Compression und Deduplication
- rman Compression
- DataPump Compression
- Sonstiges

LOBs

- Unterscheidung
 - BASICFILE
 - SECUREFILE

- Securefile
 - Compression
 - Deduplication
 - Encryption

- Securefile
 - Seit Oracle 11g verfügbar
 - Ab Oracle 12c Default bei der Anlage von LOBs

- Minimierung des Speicherplatzes und des IO
- Eliminierung bei redundanter Datenhaltung im LOB als SECUREFILE

```
CREATE TABLE T_LOB ( mitarbeitername VARCHAR2(30),  
                    lob_val          CLOB)  
  LOB(lob_val )  
  STORE AS SECUREFILE  
  (COMPRESS HIGH DEDUPLICATE );
```

Agenda

- Überblick
- Table Compression
 - Basic Compression
 - Advanced Compression
- Index Compression
 - Basic Compression
 - Advanced Compression
- LOB Compression und Deduplication
- rman Compression
- DataPump Compression
- Sonstiges

rman Messreihe – ohne Parallelisierung



Backup	Prelife Volumen	Prelife Dauer	Hochrechnung Volumen	Hochrechnung Dauer
Full uncompressed	29 GB	5:25 min	9,1 TB	28,5 h
Full compressed basics	7,5 GB	12:45 min	2,3 TB	67 h
Full compressed low (\$)	10 GB	3:15 min	3,2 TB	17 h
Full compressed medium (\$)	8 GB	7:05 min	2,5 TB	37 h
Full compressed high (\$)	6 GB	61:05 min	1,9 TB	320 h
Incremental uncompressed	8 MB	3:05 min	2,3 GB	16 h
Incremental compressed	1 MB	2:55 min	315 MB	15 h
Incremental compressed mit block change tracking	1 MB	0:10 min	315 MB	52 min
Archivelogs uncompressed	12 GB	2:10 min	62 GB	11 min
Archivelogs compressed	2 GB	4:20 min	11 GB	22 min

Agenda

- Überblick
- Table Compression
 - Basic Compression
 - Advanced Compression
- Index Compression
 - Basic Compression
 - Advanced Compression
- LOB Compression und Deduplication
- rman Compression
- DataPump Compression
- Sonstiges

- **COMPRESSION**
 - ALL
 - DATA_ONLY
 - METADATA_ONLY
 - NONE

- **COMPRESSION_ALGORITHM**
 - BASIC
 - LOW
 - MEDIUM
 - HIGH

- Auch BASIC nur über die Compression-Option nutzbar (anders als bei rman)

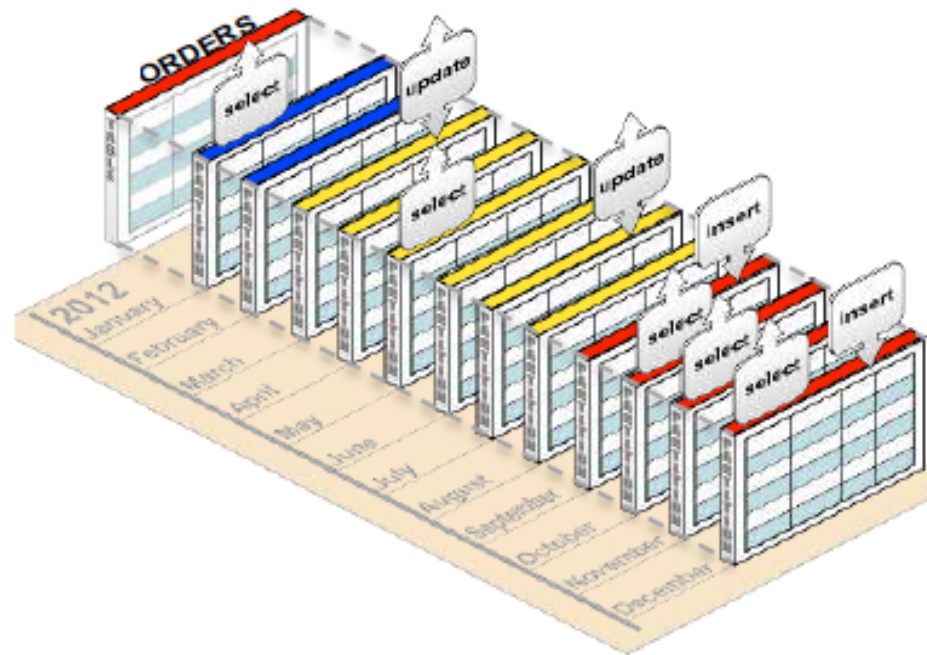
Agenda

- Überblick
- Table Compression
 - Basic Compression
 - Advanced Compression
- Index Compression
 - Basic Compression
 - Advanced Compression
- LOB Compression und Deduplication
- rman Compression
- DataPump Compression
- Sonstiges

Information Lifecycle Management (ILM)

Heat Map

- Voraussetzung für automatische Datenoptimierung
- Einschalten über
 - `ALTER SYSTEM SET HEAT_MAP = ON;`



Quelle:
Oracle Präsentation

Information Lifecycle Management (ILM)

Beispiele

```
ALTER TABLE TAB
```

```
    ILM ADD POLICY COMPRESS FOR ARCHIVE HIGH SEGMENT  
        AFTER 6 MONTHS OF NO ACCESS;
```

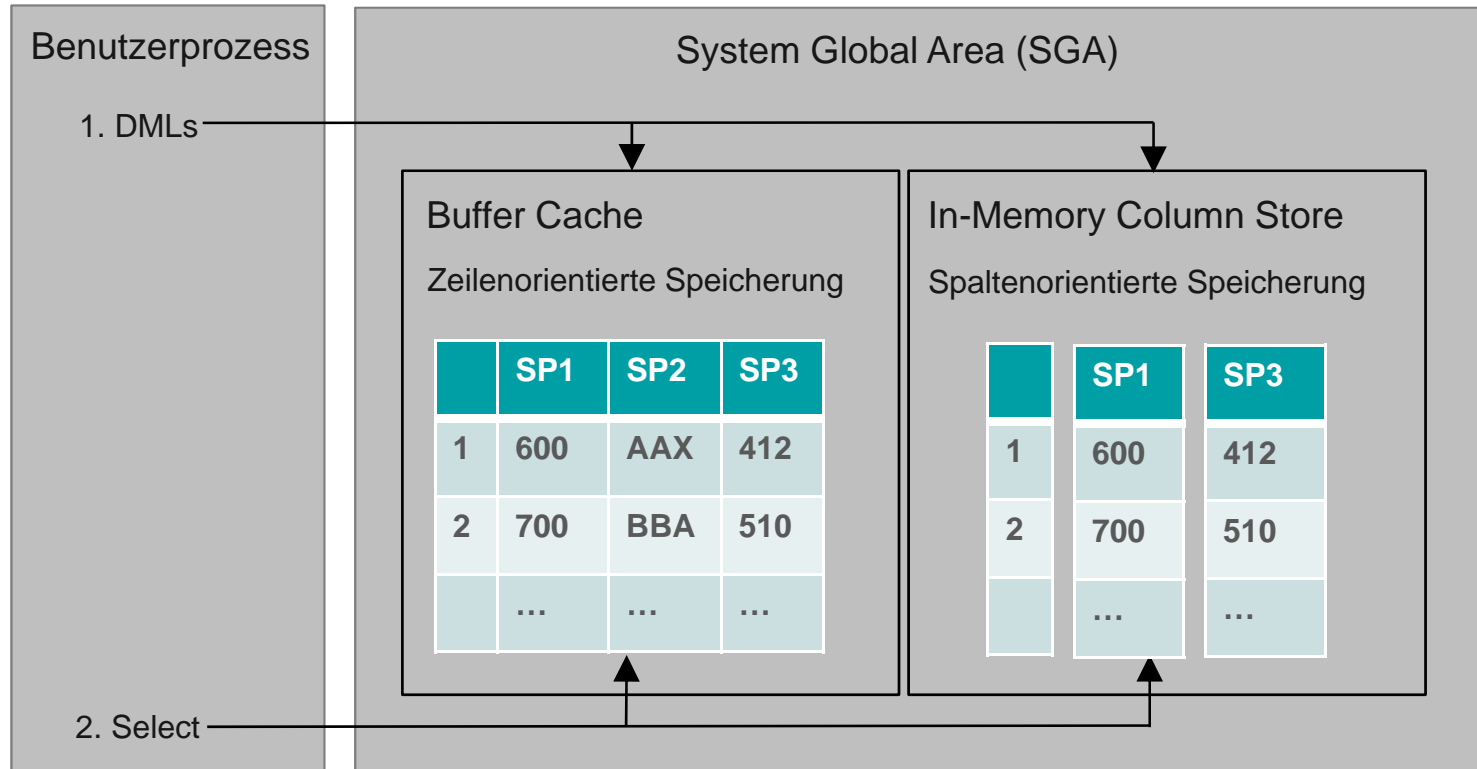
```
ALTER TABLE TAB
```

```
    ILM ADD POLICY COMPRESS FOR ARCHIVE HIGH SEGMENT  
        AFTER 6 MONTHS OF NO MODIFICATION;
```

```
ALTER TABLE TAB
```

```
    ILM ADD POLICY ROW STORE COMPRESS ADVANCED ROW  
        AFTER 30 DAYS OF NO MODIFICATION;
```

In-Memory Column Store (IMCS)



- Der In-Memory Column Store ist ein neuer Pool in der SGA.
 - transaktionskonsistent

In-Memory Column Store

Compression

▪ Kompressionslevel

- NO MEMCOMPRESS
- MEMCOMPRESS FOR QUERY [LOW|HIGH]
- MEMCOMPRESS FOR CAPACITY [LOW|HIGH]
- MEMCOMPRESS FOR DML

```
CREATE TABLE BIG_TABLE (..) INMEMORY MEMCOMPRESS FOR QUERY;  
ALTER TABLE BIG_TABLE INMEMORY MEMCOMPRESS FOR CAPACITY HIGH;
```

Fazit

- In vielen Projekten mit sehr viel Erfolg eingesetzt
 - IO fundamental minimiert
 - Mehraufwand im Cache häufig vernachlässigbar
 - vor allem Index Compression
 - häufig Table Compression bei statischen Tabellen
- Bewertung notwendig, ob Compression-Option wirklich notwendig ist
 - häufig reichen die Basis-Funktionalitäten
- Der Einsatz ist ein Projekt
 - man muss wissen, was man erreichen will
 - Anforderungen müssen spezifiziert werden
 - Erfolg muss verifiziert werden



**Vielen Dank für
Ihre Aufmerksamkeit!**

ORDIX AG

Zentrale Paderborn
Westernmauer 12 - 16
33098 Paderborn
Tel.: 05251 1063-0
Fax: 0180 1 67349 0

Seminarzentrum Wiesbaden
Kreuzberger Ring 13
65205 Wiesbaden
Tel.: 0611 77840-00

Weitere Geschäftsstellen
in Essen, Gersthofen,
Köln und Münster

info@ordix.de
www.ordix.de