

DOAG Regional 2014

Oracle Database 12c
Information Lifecycle Management
Komprimierung und Archivierung
– hört das denn niemals auf?

Mirko Hotzy
Principal Consultant
Discipline Manager Databases
ST-IMS
Mirko.Hotzy@Trivadis.com

30. Januar 2014

BASEL BERN LAUSANNE ZÜRICH DÜSSELDORF FRANKFURT A.M. FREIBURG I.BR. HAMBURG MÜNCHEN STUTTGART WIEN

1

2014 © Trivadis

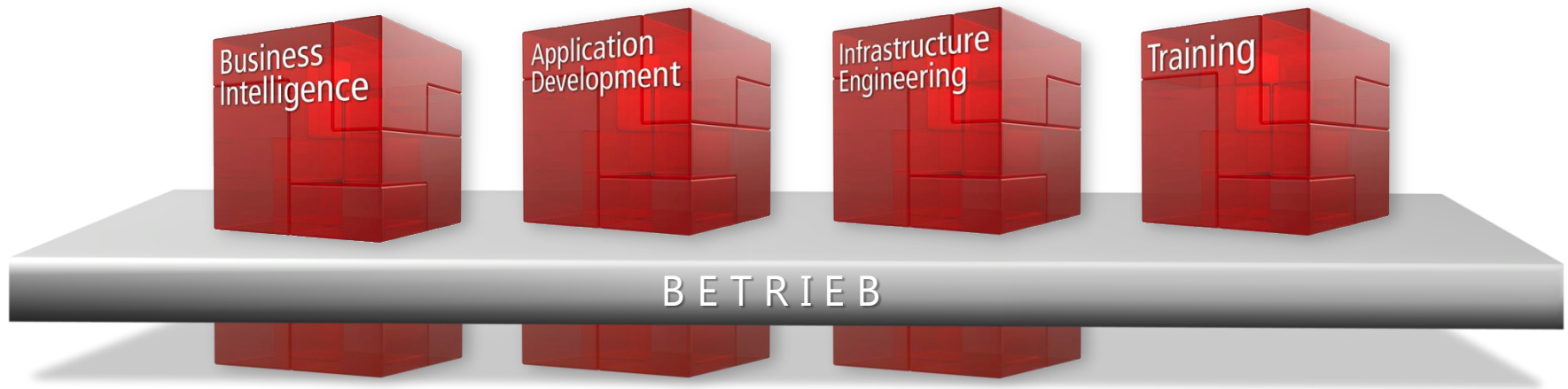
Oracle Database 12c - Information Lifecycle Management
Januar 2014

trivadis
makes IT easier. ■ ■ ■

Unser Unternehmen

Trivadis ist **führend bei der IT-Beratung, der Systemintegration, dem Solution-Engineering** und der Erbringung von **IT-Services** mit Fokussierung auf **ORACLE®** und  **Microsoft** Technologien im D-A-CH-Raum.

Unsere Leistungen erbringen wir aus den strategischen Geschäftsfeldern:



Trivadis Services übernimmt den korrespondierenden Betrieb Ihrer IT Systeme.

Mit über 600 IT- und Fachexperten bei Ihnen vor Ort



12 Trivadis Niederlassungen mit über 600 Mitarbeitenden

200 Service Level Agreements

Mehr als 4'000 Trainingsteilnehmer

Forschungs- und Entwicklungsbudget: CHF 5.0 / EUR 4 Mio.

Finanziell unabhängig und nachhaltig profitabel

Erfahrung aus mehr als 1'900 Projekten pro Jahr bei über 800 Kunden

Technik allein bringt Sie nicht weiter. Man muss wissen, wie man sie richtig nutzt.



BASEL BERN BRUGG LAUSANNE ZÜRICH DÜSSELDORF FRANKFURT A.M. FREIBURG I.BR. HAMBURG MÜNCHEN STUTTGART WIEN



Wer bin ich? – Mirko Hotzy, Trivadis GmbH

Aktuelle Position

- Principal Consultant, Discipline Manager Datenbanken, Partner bei der Trivadis GmbH

Bisherige Tätigkeiten / Know-How

- 18 Jahre fundiertes technisches Oracle Datenbank Know-How
- Technisches und strategisches Oracle Consulting
- Oracle Datenbankschulungen (New Features Datenbanken, Best Practices für Oracle Infrastrukturen, Oracle Storage Management)
- Oracle Lizenzierung – Oracle Software Asset Management Beratung, Technische Vermessungen, Infrastruktur-Optimierungen

Bücher

- Co-Autor: Der Oracle DBA (Hanser Verlag)



„Praxiserfahrung ist durch nichts zu ersetzen! Eine fundierte Beratung zeichnet sich durch den unmittelbaren Praxisbezug aus. Komplexe Theorie ist nur durch einen Experten mit viel Erfahrung lösungsorientiert vermittel- und anwendbar.“



Agenda

1. Einführung
2. Daten-Komprimierungslevel
3. Heat Map
4. Automatic Data Optimization
5. Row Level – Segment Level Compression Tiering
6. Einschränkungen
7. Kernaussagen

Information Lifecycle Management

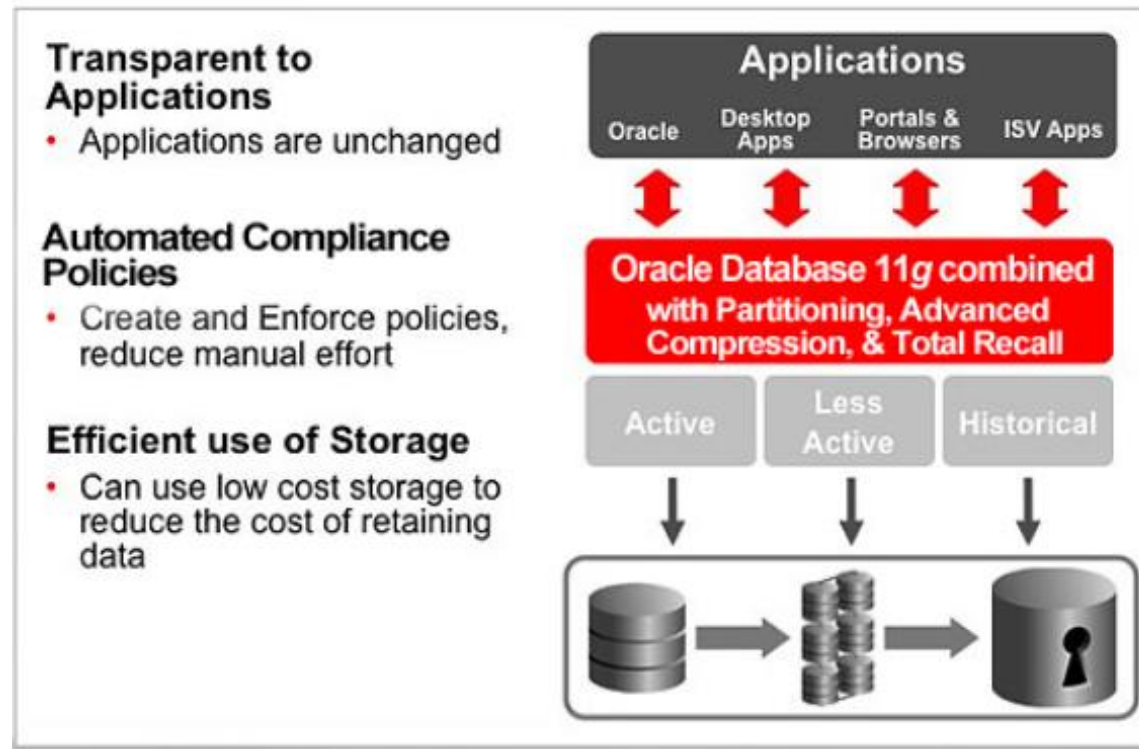


- Information Lifecycle Management (ILM) refers to a wide-ranging set of strategies for administering storage systems on computing devices
- ILM comprises the policies, processes, practices, and tools used to align the business value of information with the most appropriate and cost effective IT infrastructure from the time information is conceived through its final disposition
- In the year 2004, attempts have been made by the Information Technology and Information Storage industries to assign a new broader definition to ILM

Information Lifecycle Management in Oracle Database 11g

- Optimiertes Datenmanagement auf allen Ebenen und natürlich völlige Transparenz gegenüber den Applikationen

<http://www.oracle.com/us/technologies/information-lifecycle-management/overview/index.html>



Source: Oracle Corporation ILM Webpage

Oracle Database 12c – Positionierung und Mehrwert



- Heat Map
 - Automatisches Monitoring und Klassifizierung der Daten (hot/cold)
- Automatic Data Optimization
 - Optimiertes Daten-Management (Nutzung und Datenarchivierung) auf allen Ebenen
 - Völlig Transparent gegenüber den Applikationen
 - Automatisches verschieben der Daten (Komprimierung und Storage Tiering)
 - Neue Komprimierungs-Funktionalitäten
 - Technische Lösung für Compliance- und Archivierungsstrategien
 - Für alle Daten- und Storage-Klassen
- Mehrwert
 - Kostenreduzierung des Speicherplatzes und Datenverwaltung über den gesamten Lebenszyklus
 - Automatische und optimierte Datenverwaltung und Administration direkt im Datenbank-Kernel implementiert
 - Oracle's Antwort gegenüber „Dritthersteller Archivierungssoftware“

Agenda

1. Einführung
2. Daten-Komprimierungslevel
3. Heat Map
4. Automatic Data Optimization
5. Row Level – Segment Level Compression Tiering
6. Einschränkungen
7. Kernaussagen

Oracle Komprimierungs-Features

| Oracle Database 11g | Oracle Database 12c |
|---|--|
| OLTP Compression | Advanced Row Compression |
| Secure Files Compression | Advanced LOB Compression |
| Secure Files Deduplication | Advanced LOB Deduplication |
| Hybrid Columnar Compression (verfügbar for Oracle Exadata, Sun ZFS Storage Appliance, Pillar Axiom) | Columnar Store Compression |
| Neu | Heat Map (Segment und Zeilen-Ebene) |
| Neu | Automatic Data Optimization |
| Neu | Temporal Komponente (In Database-Archiving) |

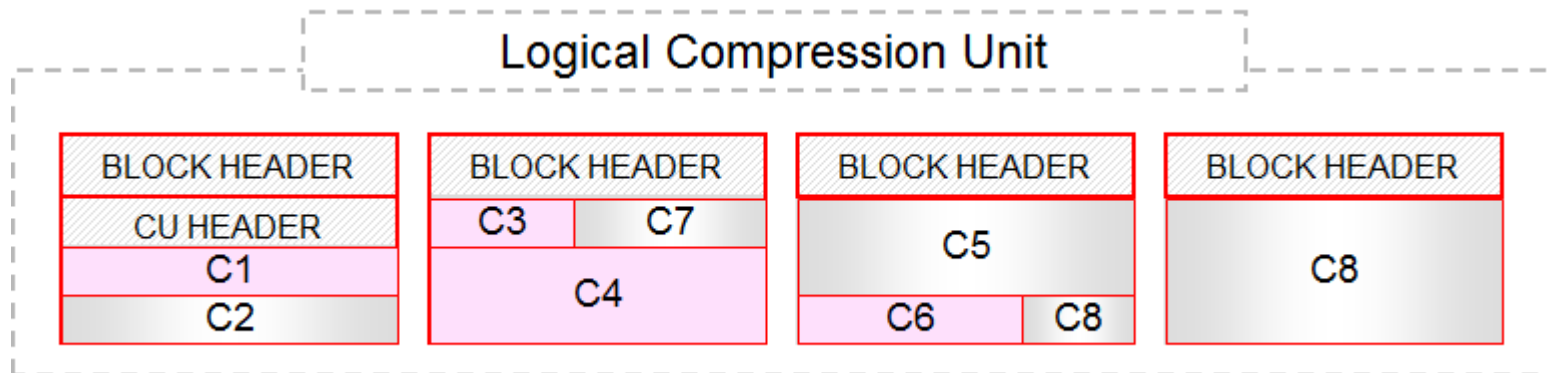
Daten-Komprimierungs-Ebenen – Row Compression

- Advanced Row Compression (ca. Faktor 3 in Abhängigkeit der Daten)
 - COMPRESS BASIC (alias Batch Compression)
 - COMPRESS ADVANCED (alias Oltp Compression)

```
CREATE TABLE sales (prod_id NUMBER NOT NULL,  
                    cust_id NUMBER NOT NULL,  
                    ...)  
PARTITION BY RANGE (time_id) (  
    PARTITION sales_2010 VALUES LESS THAN(TO_DATE(...))  
        ROW STORE COMPRESS BASIC,  
    PARTITION sales_2011 VALUES LESS THAN (MAXVALUE)  
        ROW STORE COMPRESS ADVANCED  
)
```

Daten-Komprimierungs-Ebenen – Columnar Store Compression

- Columnar Query Compression (ca. Faktor 10 in Abhängigkeit der Daten)
- Columnar Archive Compression (ca. Faktor 15 in Abhängigkeit der Daten)
- Nutzung der Hybrid Columnar Komprimierungstechnologie
 - Optimiert für DWH, Decision support, Exadata
 - Komprimierungseinheiten
 - Benutzt ein logisches Konstrukt, die sogenannte “Compression Unit” um ein Zeilenset zu speichern



Agenda

1. Einführung
2. Daten-Komprimierungslevel
3. Heat Map
4. Automatic Data Optimization
5. Row Level – Segment Level Compression Tiering
6. Einschränkungen
7. Kernaussagen

Heat Map – Ziele

- Datenbankebene
 - Zeigt die Nutzungsdaten von Tabellen, Indices und Partitionen
- Zeilen Ebene
 - Zeigt den Datenzugriff (lesend und schreibend) Zeilen basiert an
- Umfassende Darstellung
 - Anzeige der Lese- und Schreibzugriffe
 - Unterscheidet zwischen zeilenbasiertem Zugriff und “Full-Scan” der Daten
 - Zeigt den letzten Zugriff sowie Histogramme
- Einfache Benutzung
 - Einschalten auf Tablespace und Tabellen-Ebene
 - Alle gesammelten Informationen werden im SYSAUX Tablespace gespeichert

Heat Map – Datenklassifizierung



- Hot/Cold Klassifizierung der Daten

```
ALTER SYSTEM SET HEAT_MAP = ON
```

```
ALTER SESSION SET HEAT_MAP = ON
```

- PL/SQL API DBMS_HEAT_MAP verfügbar

```
SELECT tablespace_name, min_writetime, min_ftstime  
FROM table(dbms_heat_map.tablespace_heat_map('EXAMPLE'))
```

- Die Objekte im SYSAUX und SYSTEM Tablespace werden nicht überwacht!

Heat Map – Ebenen der Klassifizierung

- Detaillierte Views auf Tablespace-, Objekt- und Segment-Ebene verfügbar
 - DBA_HEATMAP_TOP_OBJECTS
 - DBA_HEATMAP_TOP_TABLESPACES
 - [DBA|ALL|USER]_HEAT_MAP_SEGMENT,
[DBA|ALL|USER]_HEAT_MAP_SEG_HISTOGRAM
 - V\$HEAT_MAP_SEGMENT, GV\$HEAT_MAP_SEGMENT
- Tiefe Integration in den Datenbank-Kernel
 - Interne Zugriffe für Systemaufgaben werden nicht berücksichtigt
 - Sammeln von Statistiken, DDLs oder Tabellen-Redefinitions-Operationen werden nicht berücksichtigt
- Individuell konfigurierbar für jede Session
 - Manuelles an- und ausschalten der Aktivitätsüberwachung
 - Möglichkeit der Deaktivierung der Aktivitätsprotokollierung während niedriger priorisierten Phasen oder Wartungsaktivitäten

Agenda

1. Einführung
2. Daten-Komprimierungslevel
3. Heat Map
4. Automatic Data Optimization
5. Row Level – Segment Level Compression Tiering
6. Einschränkungen
7. Kernaussagen

Automatic Data Optimization (ADO) – Überblick

- Deklarative Syntax für das Konfigurieren von ILM Policies auf Zeilen, Segment und Tablespace-Ebene
- Implementierung der eigenen ILM Strategie
- Automatische Datenverschiebungen auf
 - Unterschiedlichen Storage Tiers
 - Mit unterschiedlichen Komprimierungsleveln
 - Für alle Datentypen
- Nutzung von DBMS_ILM und DBMS_ILM_ADMIN PL/SQL Packages für erweitertes Policy Management und spezifische Anpassungen

Automatic Data Optimization (ADO) – Hinzufügen einer Policy

- Deklarative Policy Spezifizierung: Condition → Action

```
ALTER TABLE emp  
ILM ADD POLICY ROW STORE COMPRESS ADVANCED ROW AFTER 14 DAYS  
OF NO MODIFICATION
```

- Bedingungen können Zeitperioden nach Erstellung, Zugriffshäufigkeit oder Modifizierung der Daten sein
- Aktionen können sein:
 - Compression Tiering oder Tablespace Tiering
- Policies können von einem Tablespace oder einer Tabelle vererbt werden
 - Neue Tabellen erben die definierten Eigenschaften von einem Tablespace
 - Neue Partitionen erben von einer bestehenden Tabellen (auch bei Interval Partitionierung) inherit form table

Automatic Data Optimization (ADO) – Policy Ausführung

- Sofortige oder auch im Hintergrund ausgeführte Policy-Regeln
 - Policies auf Zeilenebene werden periodisch ausgeführt
 - Benutzer können die Ausführungsfrequenz konfigurieren
 - Policies auf Segment-Ebene werden im Maintenance Window ausgeführt
- Policies können auch um Business Rules erweitert werden
 - Benutzer können selbst definierte Bedingungen für die Datenverschiebungen definieren
 - z.B. 3 Monate nach dem Versanddatum einer Bestellung

Agenda

1. Einführung
2. Daten-Komprimierungslevel
3. Heat Map
4. Automatic Data Optimization
5. Row Level – Segment Level Compression Tiering
6. Einschränkungen
7. Kernaussagen

Row-Level – Segment Level Compression Tiering

- ADO Policies können auf Zeilen, Segment und Tablespace-Ebene definiert werden
- Automatisierung zwischen den Storage-Tiers kann aktiviert werden
- Unterschiedliche Komprimierungslevel können definiert werden
- ADO Policies über CREATE TABLE and ALTER TABLE
 - Anlegen, löschen, aktivieren, deaktivieren
- Default Compression Mapping
 - COMPRESS ADVANCED auf einer Heap-Tabelle ist verbunden mit der Standard Komprimierung für Indices und LOW für LOB-Segmente.
 - COMPRESS FOR QUERY LOW/QUERY HIGH auf einer Heap-Tabelle ist verbunden mit der Standard Komprimierung für Indices und MEDIUM für LOB Segmente
 - COMPRESS FOR ARCHIVE LOW/ARCHIVE HIGH auf einer Heap Tabelle ist verbunden mit der Standard Komprimierung für Indices und HIGH für LOB Segmente

Beispiel Segment-Level Komprimierung

```
SQL> CREATE TABLE sales (prod_id NUMBER NOT NULL,  
2      cust_id NUMBER NOT NULL,  
3      time_id DATE NOT NULL,  
4      channel_id NUMBER NOT NULL,  
5      promo_id NUMBER NOT NULL,  
6      quantity_sold NUMBER(10,2) NOT NULL,  
7      amount_sold NUMBER(10,2) NOT NULL)  
8  ILM ADD POLICY COMPRESS FOR ARCHIVE HIGH SEGMENT  
9  AFTER 6 MONTHS OF NO ACCESS;
```

```
SQL> SELECT substr(policy_name,1,24) AS policy_name,  
2      policy_type, enabled  
3  FROM user_ilmpolicies;
```

| POLICY_NAME | POLICY_TYPE | ENABLE |
|-------------|---------------|--------|
| ----- | ----- | ----- |
| P1 | DATA MOVEMENT | YES |

Beispiel Segment-Level Komprimierung und Storage Tiering

- Hinzufügen einer Segment-Level Komprimierung Policy "nach 30 Tagen ohne modifizierung

```
ALTER TABLE sales  
ILM ADD POLICY ROW STORE COMPRESS ADVANCED SEGMENT AFTER 30  
DAYS OF NO MODIFICATION
```

- Hinzufügen einer Storage Tier Policy

```
ALTER TABLE sales  
ILM ADD POLICY TIER TO my_low_cost_sales_tablespace
```

Beispiel Row-Level Compression Tiering

- Syntaxbeispiel für das Anlegen einer ADO-Policy “Row-Level Compression Tiering”
- Row-Level Compression Policies arbeiten nur auf Datenbankblöcken, die vollständig die definierte Policy erfüllen!

```
ALTER TABLE sales  
ILM ADD POLICY ROW STORE COMPRESS ADVANCED ROW AFTER 30 DAYS  
OF NO MODIFICATION
```

- OLTP Komprimierung für alle Datenbankblöcke, die > 30 Tage keine Änderungsaktivität mehr hatten
- PL/SQL API verfügbar
 - DBMS_ILM (z.B. Jobs anlegen für automatisiertes Storage Tiering)
 - DBMS_ILM_ADMIN (zur Implementierung von Business Rules)

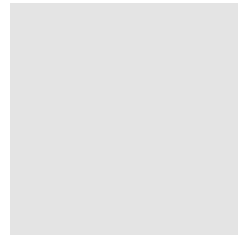
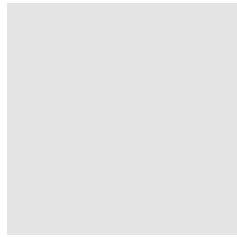
Agenda

1. Einführung
2. Daten-Komprimierungslevel
3. Heat Map
4. Automatic Data Optimization
5. Row Level – Segment Level Compression Tiering
6. Einschränkungen
7. Kernaussagen

Einschränkungen von ADO und Heat Map

- ADO und Heat Map sind nicht in Verbindung mit Container Datenbanken (CDB) unterstützt.
- ADO prüft nicht bei der Nutzung von Storage-Tiering den freien Platz im Ziel-Tablespace!
- Wenn ein Policy-Job (Scheduler Job) mehr als 2x fehlschlägt, dann muss dieser von Hand gestartet werden!
- ADO-Policies werden automatisch während der Maintenance Windows ausgeführt!
 - Außerhalb der Maintenance Windows werden die Policies nicht beachtet
 - Die einzige Ausnahme sind Jobs, die im Offline-Modus der ADO-Policies Indices rebuilden
- Supplemental logging kann nicht parallel zu ADO aktiviert werden

Information Lifecycle Management – Kernaussagen



- 😊 Völlig transparent gegenüber den Applikationen
- 😊 Verschiedene Datenkomprimierungs-Level verfügbar
- 😊 Definition von ILM-Policies für Automatische Datenverschiebung
- 😞 Man muss die Datenklassifikation basierend auf den Heat Map Statistiken manuell durchführen
- 😞 Einschränkungen beachten
- 😞 Advanced Compression Option
- ! Passendes Konzept ist ein Muss!