

# Locks, Fragmentierung und Index-Komprimierung – Nützliche SQL-Kommandos

Martin Frauendorfer  
SAP AG  
Walldorf

## Schlüsselworte:

SQL, DBACOCKPIT, SQL Command Editor, Locks, Sperren, Enqueue, Fragmentierung, Komprimierung

## Einleitung

SAP-Standardtransaktionen wie ST04, DB02 oder DBACOCKPIT halten bereits viele nützliche Funktionalitäten bereit. Für manche Bereiche gibt es aber noch Lücken oder mangelnde Flexibilität. Zu diesem Zweck stellt SAP im Rahmen von SAP Hinweis 1438410 zahlreiche SQL-Kommandos zur Verfügung, die unter anderem über den „SQL Command Editor“ in Transaktion DBACOCKPIT oder in anderen Client-Tools wie SQLPLUS ausgeführt werden können. Teilweise finden diese Kommandos auch Eingang in die „Loadable SAP Support Monitors“ in Transaktion DBACOCKPIT (SAP Hinweis 1299493). Bei geeigneter Nutzung ergeben sich aus diesen Skripten umfangreiche Möglichkeiten, allerdings wird aufgrund der Dynamik der Skripten momentan keine Dokumentation mitgeliefert. Auch wenn Zweck und Konfigurierung an vielen Stellen selbsterklärend ist, stellt dies im Einzelfall eine gewisse Einschränkung der Nutzbarkeit dar. Dieser Vortrag greift exemplarisch einige der Skripten heraus und beschreibt, welchem Zweck sie dienen und wie sie sinnvoll konfiguriert werden können.

Aufgrund der technischen Restriktionen des „SQL Command Editors“ in Transaktion DBACOCKPIT handelt es sich bei allen Skripten um ein einziges SQL-Statement – der Begriff „Skript“ ist daher etwas irreführend. Durch Vormaterialisierung von Unterabfragen mittels WITH-Klausel ergibt sich aber eine Skript-ähnliche Schritt-für-Schritt-Abarbeitung.

Die meisten Kommandos beinhalten zu Beginn eine BASIS\_INFO-Sektion („WITH BASIS\_INFO AS ...“), in der bei Bedarf Einstellungen vorgenommen werden können, z.B.:

```
WITH BASIS_INFO AS
...
-1 DBID,
-1 INSTANCE_NUMBER,
TO_DATE('01.01.1000 00:01:30', 'dd.mm.yyyy hh24:mi:ss') BEGIN_DATE,
TO_DATE('31.12.9999 00:01:40', 'dd.mm.yyyy hh24:mi:ss') END_DATE,
'0jy9yy9g2y4dw' SQL_ID,
'enq: TX - row lock contention' EVENT
```

Beispielsweise kann bei Bedarf – z.B. in Real Application Clusters-Umgebungen – eine explizite Instanz-Nummer angegeben werden, für die die Auswertung durchgeführt werden soll. Der Wert „-1“ steht bei „INSTANCE\_NUMBER“ und „DBID“ für die aktuelle Instanz, ansonsten für keine Einschränkung.

## Lock Monitoring

Datenbanksperren im allgemeinen und Datenbank-Enqueues im speziellen können große Auswirkung auf den Systembetrieb haben. SAP-Transaktionen wie DB01 können zur Analyse genutzt werden, allerdings stehen einige Informationen nicht zur Verfügung und es herrscht oft eine gewisse Ratlosigkeit: Welche Session ist für eine Sperre verantwortlich? Was hat diese Session gemacht, während sie eine Sperre gehalten hat? Welcher Datensatz ist für die Sperre verantwortlich? Wieso bricht meine DDL-Operation immer wieder mit ORA-00054 ab?

Etwas Licht ins Dunkel bringen hier verschiedene Lock-Skripten aus Hinweis 1438410:

*Locks\_CurrentBlockingLocks.txt*

Zweck: Anzeige aller aktuell existierenden Sperren, durch die Sessions blockiert werden

Eingabeparameter: Keine

Ausgabespalten:

- SID
- INST\_ID
- PROCESS
- MACHINE
- ACTION
- COMMAND\_TYPE
- SQL\_ID
- PREV\_SQL\_ID
- ENQ\_TYPE
- ENQ\_REQUEST
- OBJECT\_NAME
- ROW\_WAIT\_FILE#
- ROW\_WAIT\_BLOCK#
- ROW\_WAIT\_ROW#
- SECONDS\_IN\_WAIT
- ROW\_EXTRACTION\_COMMAND

Besondere Features:

- Durch Einrücken werden Sperr-Abhängigkeiten anschaulich dargestellt. So läßt sich schnell erkennen, welche Session der eigentliche Auslöser eines komplexen Sperr szenarios ist.
- Es werden nicht nur klassische Enqueues, sondern auch andere Abhängigkeiten wie Locks, Latches oder Redo Log-Waits dargestellt.
- Im Falle eines Row Lock Waits wird ein Kommando generiert, mit dessen Hilfe man den sperrenden Datensatz auslesen kann.

*Locks\_LockedObjects.txt*

Zweck: Anzeige Tabellen, die momentan gesperrt sind (shared oder exclusive)

Eingabeparameter:

- INSTANCE\_NUMBER
- OBJECT\_NAME
- SESSION\_ID

Ausgabeparameter:

- OBJECT\_NAME
- SUBOBJECT\_NAME
- INST\_ID
- SID
- CTIME\_S
- LMODE
- REQUEST
- BLOCK

Besondere Features:

- Shared-Sperren werden ebenso wie Exclusive-Sperren angezeigt, da beide Sperrtypen für ORA-00054-Abbrüche bei DDL-Operationen verantwortlich sein können
- Haltezeit der Sperre wird angezeigt, so dass festgestellt werden kann, wie lange die Sperre auf einer Tabelle typischerweise gehalten wird
- Blockierende Sperren (auf die andere Sessions warten) werden kenntlich gemacht.

*Locks\_LockHolderActivities.txt*

Zweck: Anzeige der (Datenbank-)Aktivitäten der Sessions(s), die eine bestimmte Sperre halten

Eingabeparameter:

- DBID
- INSTANCE\_NUMBER
- BEGIN\_DATE
- END\_DATE
- SQL\_ID
- EVENT

Ausgabeparameter:

- OCCURRENCES
- ACTION

- SQL\_ID
- PERCENT
- TOTAL\_WAITERS
- AVG\_WAITERS

Besondere Features:

- Auswertung der Lock-Holder-Aktivitäten beruht auf den Active Session History-Samples und kann auf beliebige Zeiträume innerhalb der Automatic Workload Repository-Retention eingeschränkt werden.
- Ist der Lock-Holder nicht in der Datenbank aktiv (sondern z.B. auf ABAP-Seite), wird dies mit INACTIVE vermerkt.

*Locks\_TopBlocks.txt / Locks\_TopRows\_11g.txt*

Zweck: Anzeige der historischen Blöcke (Oracle 10g) bzw. Datensätze (11g), die für die meisten blockierenden Sperren verantwortlich waren

Eingabeparameter:

- DBID
- INSTANCE\_NUMBER
- BEGIN\_DATE
- END\_DATE
- SQL\_ID
- EVENT
- NUM\_RECORDS

Ausgabeparameter:

- OBJECT\_ID,
- FILE
- BLOCK
- ROW (nur 11g)
- OCCURRENCES
- PERCENT

Besondere Features:

- Auswertung der Lock-Holder-Aktivitäten beruht auf den Active Session History-Samples und kann auf beliebige Zeiträume innerhalb der Automatic Workload Repository-Retention eingeschränkt werden.
- Ab Oracle 11g ist eine Auswertung bis auf Datensatzebene möglich. Dadurch läßt sich beispielsweise im nachhinein feststellen, welches SAP-Nummernkreisobjekt für eine NRIV-Sperre verantwortlich war.

## Fragmentierungs-Analyse

Segment-Fragmentierung, d.h. die nicht optimale Platzausnutzung innerhalb von Tabellen und Indizes läßt sich auf verschiedene Weise bestimmen. Oft handelt es sich dabei um zeit- und ressourcenintensive Operationen. Im Rahmen von Hinweis 1438410 werden Kommandos zur Verfügung gestellt, die eine Fragmentierungsanalyse relativ schnell auf Basis der Ergebnisse des Oracle Segment Advisors oder auf Basis der CBO-Statistiken durchführen.

*Space\_SegmentFragmentation\_SegmentAdvisor.txt*

Zweck: Bestimmung von Segment-Fragmentierung auf Basis der Ergebnisse des Oracle Automatic Segment Advisor; Automatic Segment Advisor muss dafür regelmäßig (z.B. wöchentlich) eingeplant sein.

Eingabeparameter:

- OWNER
- SEGMENT\_NAME
- SEGMENT\_TYPE
- MIN\_SIZE\_MB
- NUM\_RECORDS
- NUM\_LARGEST\_SEGMENTS
- NUM\_IO\_READ\_SEGMENTS
- QUALITY\_THRESHOLD
- UNUSED\_MB\_THRESHOLD
- SORT\_BY

Ausgabeparameter:

- ANALYSIS\_TIME
- SEGMENT\_NAME
- PART\_NAME
- SIZE\_POS
- IO\_POS
- UNUSED\_MB
- QUALITY\_%
- GROSS\_MB
- NET\_MB

Besondere Features:

- Es besteht die Möglichkeit, die Analyse auf die platzrelevanten (NUM\_LARGEST\_SEGMENTS) bzw. performancerelevanten (NUM\_IO\_READ\_SEGMENTS) Segmente zu beschränken.
- Es können sowohl für den ungenutzten Platz als auch für die Speicherqualität Schwellwerte eingestellt werden.

### *Space\_SegmentFragmentation\_CBOStatistics.txt*

Zweck: Bestimmung von Segment-Fragmentierung auf Basis der existierenden CBO-Statistiken

Eingabeparameter:

- TABLE\_OWNER
- TABLE\_NAME
- TABLESPACE\_NAME
- MIN\_SIZE\_MB
- NUM\_RECORDS
- NUM\_LARGEST\_SEGMENTS
- NUM\_IO\_READ\_SEGMENTS
- QUALITY\_THRESHOLD
- UNUSED\_MB\_THRESHOLD
- QUALITY\_THRESHOLD\_2
- UNUSED\_MB\_THRESHOLD\_2
- SORT\_BY
- INCLUDE\_TAB\_MODIFICATIONS
- INCLUDE\_PARTITIONS
- EXCLUDE\_LONG\_RAW\_TABLES
- EXCLUDE\_MODIFIED\_STATS
- RESTRICTION\_INFO

Ausgabeparameter:

- SEGMENT\_NAME
- PART\_NAME
- SIZE\_POS
- IO\_POS
- UNUSED\_MB
- QUALITY\_%
- SEG\_GROSS\_MB
- SEG\_NET\_MB
- DATA\_GROSS\_MB
- DATA\_NET\_MB

Besondere Features:

- Es besteht die Möglichkeit, die Analyse auf die platzrelevanten (NUM\_LARGEST\_SEGMENTS) bzw. performancerelevanten (NUM\_IO\_READ\_SEGMENTS) Segmente zu beschränken.
- Es können sowohl für den ungenutzten Platz als auch für die Speicherqualität Schwellwerte eingestellt werden.

- Bestehen Einschränkungen bezüglich CBO-Statistiken (gesperrte Statistiken, veraltete Statistiken, LONG RAW-Spalten, ...), kann dies mittels RESTRICTION\_INFO = 'X' angezeigt werden.

## **Index-Komprimierung**

Index-Komprimierung ist ein im SAP-Umfeld ab Oracle 10g verfügbares Feature, das fast nur Vorteile mit sich bringt (geringerer Platzbedarf, weniger I/O, bessere Buffer-Qualität, weniger CPU-Konsum). Für eine optimale Komprimierung muss aber eine geeignete Anzahl von führenden Indexspalten ermittelt werden (→ Präfix), über die komprimiert werden soll. Diese Aufgabe ist nicht trivial und erfordert in aller Regel ressourcenintensive Aktivitäten wie ANALYZE INDEX ... VALIDATE STRUCTURE oder dedizierte SQL-Analyseskripten (Hinweis 1109743). Das folgende Skript stellt einen zeit- und ressourcensparenden konservativen Ansatz auf Basis der CBO-Statistiken dar, der in ca. 80 % der Fälle ebenfalls zur optimalen Empfehlung kommt.

*Space\_IndexCompression.txt*

Zweck: Bestimmung von optimalen Präfixlängen für Index-Komprimierung

Eingabeparameter:

- TABLE\_OWNER
- TABLE\_NAME
- INDEX\_NAME
- MIN\_SIZE\_MB
- NUM\_RECORDS
- MIN\_SAVE\_PCT
- NUM\_LARGEST\_INDEXES
- NUM\_IO\_READ\_INDEXES
- SORT\_BY
- REBUILD\_COMMAND
- ONLINE\_SETTING
- NOLOGGING\_SETTING
- PCTFREE\_SETTING
- PARALLEL\_SETTING
- TABLESPACE\_SETTING
- INITIAL\_SETTING
- FORCE\_SETTING

Ausgabeparameter:

- OWNER
- INDEX\_NAME
- SIZE\_POS
- IO\_POS

- COL
- COMP
- NOCOMP\_MB
- COMP\_MB
- SAVING\_MB
- SAVING\_%
- COMP\_REC

**Besondere Features:**

- Es besteht die Möglichkeit, die Analyse auf die platzrelevanten (NUM\_LARGEST\_SEGMENTS) bzw. performancerelevanten (NUM\_IO\_READ\_SEGMENTS) Segmente zu beschränken.
- Mittels REBUILD\_COMMAND = 'X' kann ein ALTER INDEX REBUILD-Kommando erzeugt werden, mit dem die Komprimierung aktiviert wird.
- Mittels REBUILD\_COMMAND = 'B' kann ein BRSPACE-Aufruf generiert werden, mit dem die Komprimierung aktiviert wird.

**Kontaktadresse:**

**Martin Frauendorfer**  
SAP AG  
Dietmar-Hopp-Allee 16  
D-69190 Walldorf

Telefon: +49 (0) 6227 – 60 50 70  
E-Mail: martin.frauendorfer@sap.com  
Internet: www.sap.de