

Die Top-10-Monitoring-SQL-Befehle

Marco Patzwahl, MuniQSoft GmbH

Viele Kunden haben mehr als hundert Datenbanken zu betreuen. Da kommt man ohne automatisierte Überwachungs-Skripte nicht sehr weit. Deswegen soll diese kleine Sammlung an SQL-Befehlen helfen, das Monitoring der Datenbanken zu vereinfachen. Der Fokus liegt dabei auf der Datenbank-Version 12.1, jedoch sind Anmerkungen bei den Befehlen, wie man diese auch in älteren Datenbank-Versionen verwenden kann. Die Skripte sind zum Download verfügbar.

1. Alert-Log überwachen

Ab Version 11.2.0.2 kann die Alert-Datei (XML-Variante) mittels „SELECT“ ausgewertet werden (siehe Listing 1).

Die Überwachung findet alle fünf Minuten statt. Wenn die Abfrage sehr lange läuft, ist die XML-Datei wahrscheinlich schon sehr groß geworden. Man kann sie entweder manuell löschen (nachdem sie eventuell vorher gesichert wurde). Alternativ kann auch Oracle die Datei löschen, wenn man Instance- und Datenbank-Name entsprechend ersetzt (siehe Listing 2).

2. Listener.log überwachen

Ab Version 11.2.0.2 lässt sich die „listener.log“-Datei (XML-Variante) mittels

„SELECT“ auswerten (siehe Listing 3).

Die Überwachung erfolgt alle fünf Minuten. Dieser Ansatz funktioniert nur, wenn der Listener seine „listener.log“ und „log.xml“ in die „diag“-Struktur schreibt. Zuständig dafür ist der „listener.ora“-Parameter: „DIAG_ADR_ENABLED_<listener_name>=ON|OFF“. Wenn man ein anderes Verzeichnis angibt, landen die Daten nicht mehr in der internen „diag“-Struktur: „ADR_BASE_< listener_name > = <pfad >“.

3. Backups überwachen

Man sollte täglich prüfen, ob die letzten Backups gelaufen sind. Wenn nicht, ist das Backup zu wiederholen

(siehe Listing 4). Die Überwachung geschieht einmal am Tag.

4. Welche Datenbank-Objekte invalide sind

Man bekommt für jedes Schema maximal eine Zeile nur mit der Anzahl der ungültigen Objekte angezeigt. Wenn keine Zeile zurückkommt, ist auch kein Objekt defekt (siehe Listing 5). Das Zeit-Intervall für die Überwachung ist einmal am Tag.

5. Tabellen/Dateien mit Block-Korruption

Bei dieser Abfrage sollte dreimal „0“ zurückkommen (siehe Listing 6). Ansonsten sollte man die Views jeweils

```
SELECT
to_char(originating_timestamp,'DD.MM.YYYY Hh24:MI:SS') as log_date, message_text
FROM v$diag_alert_ext
WHERE trim(component_id) = 'rdbms'
AND (message_text like '%ORA-%'
OR message_text like '%TNS-%'
OR message_text like '%Checkpoint not%')
AND originating_timestamp > sysdate-interval '1' hour;
```

Listing 1

```
dos/unix> adrci exec= „set home diag/rdbms/o12c/o12c; purge -age 0 -type alert“
```

Listing 2

```
SELECT
to_char(originating_timestamp,'DD.MM.YYYY Hh24:MI:SS') as log_date,message_text
FROM v$diag_alert_ext
WHERE trim(component_id) = 'tnslsnr'
AND message_text like '%TNS-%'
AND originating_timestamp>sysdate-interval '1' hour;
```

Listing 3

```

SELECT start_time, to_char(end_time,'HH24:MI:SS') end_time, elapsed_seconds as elap_sec,input_
type, status,output_device_type as out_device,
round(input_bytes/1024/1024/1024) INPUT_GB, round(output_bytes/1024/1024/1024) OUTPUT_GB
FROM v$rman_backup_job_details r
ORDER BY 1;

```

Listing 4

```

SELECT o.*,i.ind_i as "Ind (I)" FROM (
SELECT o.owner,
count( CASE WHEN o.object_type= 'TABLE' AND o.status<>'VALID' THEN 'x' END ) as "TAB (I)",
count( CASE WHEN o.object_type= 'VIEW' AND o.status<>'VALID' THEN 'x' END ) as "Views (I)",
count( CASE WHEN o.object_type= 'CLUSTER' AND o.status<>'VALID' THEN 'x' END ) as "Clus
(I)",
count( CASE WHEN o.object_type= 'TYPE' AND o.status<>'VALID' THEN 'x' END ) as "Type (I)",
count( CASE WHEN o.object_type= 'SYNONYM' AND o.status<>'VALID' THEN 'x' END ) as "Syn (I)",
count( CASE WHEN o.object_type= 'PACKAGE BODY' AND o.status<>'VALID' THEN 'x' END ) as
"PackB (I)",
count( CASE WHEN o.object_type= 'PROCEDURE' AND o.status<>'VALID' THEN 'x' END ) as "Proc
(I)",
count( CASE WHEN o.object_type= 'TRIGGER' AND o.status<>'VALID' THEN 'x' END ) as "Trig
(I)",
count( CASE WHEN o.object_type= 'FUNCTION' AND o.status<>'VALID' THEN 'x' END ) as "Func
(I)"
FROM all_objects o
GROUP BY o.owner) o,
(SELECT owner,
count( CASE WHEN status NOT IN ('VALID','N/A') THEN 'x' END ) as ind_i
FROM all_indexes i GROUP BY owner) I
WHERE o.owner=i.owner
AND "TAB (I)" + "Views (I)" + "Clus (I)" + "Type (I)" + "Syn (I)" + "PackB (I)" + "Proc (I)"
+ "Trig (I)" + „Func (I)" + i.ind_i>0;

```

Listing 5

```

SELECT
(SELECT count(*) FROM gv$backup_corruption) as BAK_CORRUPT,
(SELECT count(*) FROM gv$copy_corruption) AS COPY_CORRUPT,
(SELECT count(*) FROM gv$database_block_corruption) AS DB_BLOCK_CORRUPT
FROM dual;

```

Listing 6

```

SELECT instance_name, host_name, startup_time, status, logins
FROM gv$instance
ORDER BY 1;

```

Listing 7

```
SELECT con_id, name, open_mode
FROM v$pdb
ORDER BY 1;
```

Listing 8

```
SELECT
round((space_limit)/1024/1024/1024,2) "Max Space (GB)",round((space_limit-space_
used)/1024/1024/1024,2) "Free Space (GB)"
FROM v$recovery_file_dest;
```

Listing 9

```
SELECT username, account_status, lock_date, expiry_date
FROM dba_users
WHERE username NOT IN (SELECT distinct schema_name FROM v$sysaux_occupants)
AND username NOT IN ('APPOSSYS','FLOWS_FILESv','XS$NULL','ORACLE_OCM',
'OUTLN');
```

Listing 10

```
SELECT username, account_status, lock_date, expiry_date
FROM dba_users
WHERE (oracle_maintained = 'N'
OR username IN ('SYS','SYSTEM'))
AND account_status <> 'OPEN';
```

Listing 11

```
SELECT min(first_time) as start_time, sum("<1min"), sum("1min<x<5min"), sum("5min<x<10min"),su
m("10min<x<60min"),sum(">60min") FROM (SELECT
first_time,
case when (round((first_time-lag(first_time,1)over ( order by sequence#))*24*60))<=1 then 1
end as "<1min",
case when (round((first_time-lag(first_time,1)over ( order by sequence#))*24*60))>1 AND
(round((first_time-lag(first_time,1)over ( order by sequence#))*24*60))<=5 then 1 end as
"1min<x<5min",
case when (round((first_time-lag(first_time,1)over ( order by sequence#))*24*60))> 5 AND
(round((first_time-lag(first_time,1)over ( order by sequence#))*24*60))<=10 then 1 end as
"5min<x<10min",
case when (round((first_time-lag(first_time,1)over ( order by sequence#))*24*60))> 10 AND
(round((first_time-lag(first_time,1)over ( order by sequence#))*24*60))<=60 then 1 end as
"10min<x<60min",
case when (round((first_time-lag(first_time,1)over ( order by sequence#))*24*60))>60 then 1
end as ">60min"
FROM v$loghist
WHERE first_time > (sysdate - 1));
```

Listing 12

```
SELECT tum.tablespace_name,round(tum.used_space*t.block_size/1024/1024/1024,2) used_GB,
round(tum.tablespace_size*t.block_size/1024/1024/1024,2) max_possible_size_gb,round(used_per-
cent,2) as used_percent
FROM dba_tablespace_usage_metrics tum, dba_tablespaces t
WHERE tum.tablespace_name = t.tablespace_name;
```

Listing 13

nach den Details durchforsten. Die Überwachung erfolgt einmal am Tag.

6. Zustand der Datenbank allgemein

Listing 7 zeigt, ob die Instanz hochgefahren wurde und Anmeldungen der Benutzer möglich sind (normale Datenbank/RAC). Für die Pluggable Database (ab 12.1) gilt Listing 8. Alle fünf Minuten findet diese Überwachung statt.

7. Freier Speicherplatz in der Fast Recovery Area

Falls die Fast Recovery Area für die Instanz zum Einsatz kommt, sollte man regelmäßig prüfen, ob dort noch genug freier Speicherplatz zur Verfügung steht (siehe Listing 9). Falls nicht, sind nur mit RMAN alte Dateien zu löschen oder alternativ die Fast Recovery Area zu vergrößern. Dies wird alle fünf Minuten überwacht.

8. Gesperrte und abgelaufene Accounts

Listing 10 zeigt (bis Version 12.1) alle Accounts, die derzeit gesperrt sind. Damit nicht die Oracle-eigenen, be-

reits vorgesperrten Accounts mit angezeigt werden, sollte man diese herausfiltern. Ab Version 12.1 lässt sich die Ausschluss-Liste abhängig von der Datenbank-Version erweitern (siehe Listing 11). Die Überwachung erfolgt jede Stunde.

9. Redo-Log-Switches

Dieser Select zeigt für die letzten vierundzwanzig Stunden an, wie oft Log-Switches unter einer Minute, zwischen einer und fünf Minuten, zwischen fünf und zehn Minuten, zwischen zehn und sechzig Minuten und über sechzig Minuten aufgetreten sind (siehe Listing 12). Ideal wäre ein Bereich zwischen zehn und sechzig Minuten. Wenn die Datenbank darunter liegt, kann man sich überlegen, die Größe der Redo-Log-Datei zu verändern. Das Zeit-Intervall für die Überwachung ist jede Stunde.

10. Wachstumsmöglichkeiten der Tablespaces

Der SELECT zeigt den aktuellen Füllpegel des Tablespace und die maximale

Größe, die er laut seiner Einstellung erreichen könnte (siehe Listing 13). Ob die Platten diesen Platz wirklich zur Verfügung stellen könnten, wird jedoch nicht erkannt. Die Überwachung erfolgt alle fünf Minuten.

Hinweis: Eine digitale Version dieses Artikels lässt sich unter <http://www.muniqsoft.de/tipps/publikationen.htm> herunterladen.

Marco Patzwahl
m.patzwahl@muniqsoft.de



„Lizenz-Compliance wurde bisher in vielen Unternehmen nicht mit der Priorität behandelt, die notwendig wäre ...“

Es ist nicht immer leicht, die Lizenzbedingungen von Oracle zu verstehen. Deshalb hat sich der Arbeitskreis Lizenzierung unter der Leitung von Michael Paege, stellv. Vorstandsvorsitzender und Leiter Competence Center Lizenzierung der DOAG, dem Thema angenommen. Entstanden ist ein kompakter, digitaler Lizenzguide, in dem die wichtigsten Aspekte der Lizenzierung von Oracle-Software zusammengefasst sind. Diesen können DOAG-Mitglieder unter www.doag.org/go/lizenzguide kostenfrei bestellen.

Für Michael Paege ist das Hauptthema nach wie vor die Lizenzierung bei Virtualisierung mit VMware. Vor drei Jahren hat die DOAG ein Presse-Roundtable zu diesem Thema veranstaltet. Danach fanden auch Gespräche zwischen der DOAG und dem Chief Customer Relation Officer des Oracle Headquarters statt. Leider ist es nicht gelungen, in dem gewünschten Maß bei Oracle nachhaltige Verbesserungen zu erreichen.

Im Hinblick auf den Themenkomplex „Virtualisierung“ sowie die Metrik „Named User Plus“ in Verbindung mit Datentransfers wird die Lizenz-Compliance leider in vielen Unternehmen nicht mit der Priorität behandelt, die notwendig wäre. Bei Audits werden dann viele Firmen auf dem falschen Fuß erwischt. Viele Unternehmen – vor allem, wenn sie sich ursprünglich für die Metrik Named-User-Plus (NUP) entschieden hatten – sind heutzutage nicht mehr korrekt lizenziert. Mit der Zeit haben sie Architektur-Änderungen vorgenommen, neue Lösungen eingeführt, und auch angefangen, Daten mit anderen Systemen und der Außenwelt auszutauschen. Die Idee hinter dem Lizenzguide ist es, Transparenz zu schaffen. Die Informationen zur Lizenzierung musste man sich bisher mühsam aus den jeweiligen Verträgen und auf den Oracle-Webseiten zusammensuchen. Jetzt gibt es zum ersten Mal ein kompaktes, übersichtliches Dokument, in dem die wichtigsten Informationen zu finden sind.