

# Oracle Statistik Reports Leitfaden zur Auswertung

**Angelika Gallwitz**  
**freiberufliche Beraterin**  
**Bad Homburg**

## Schlüsselworte

Statspack Reports, AWR Reports, Top 5 Timed Events, Baseline, Tuning, Thresholds, Wait Class

## Einleitung

Oracle Statspack/AWR Reports werden herangezogen, um Performance Engpässe zu erkennen und Maßnahmen zu ergreifen. Wie erzeuge und interpretiere ich einen Statistik Report. Wie verändere ich Snap Level und Threshold.. Anhand der Wait Events und Wait Classes wird eine Zuordnung zu den Problembereichen vorgenommen und konkrete Empfehlungen zu Tuning Maßnahmen gegeben.

## Agenda

Statspack , AWR Überblick  
System Voraussetzungen für Statistiken  
Demo Perfstat User anlegen Reports erzeugen  
Statspack Statistiklevel  
AWR Architekturübersicht  
Demo Statistiküberblick  
AWR Statistiklevel  
Welche Statistiken werden gesammelt  
Performance Tuning Abfragen  
ADDM  
Demo AWR Snapshots  
Demo Reports  
Baseline erzeugen und übertragen  
Vorbereitung der Analyse  
Auswertung von Wait Events per Oracle AWR Report  
Top 5 Timed Events In OLTP Umgebung  
Top 5 Timed Events In Decision Support/Datawarehouse Umgebung  
Leitfaden zur Vorgehensweise

## Statspack , AWR Überblick

Mit historischen Performance Statistiken können nachträglich Performance Probleme analysiert werden. Oracle stellt dafür den User Perfstat mit seinem Statspack Repository und das AWR Automatic Workload Repository zur Verfügung.

Für die Erzeugung des AWR wird die Option „Diagnostik Pack“ benötigt, für Statspack Auswertungen nicht.

AWR ist mächtiger und kann auch im OEM Oracle Enterprise Manager angezeigt werden, ebenso ADDM Automatic Database Diagnostic Monitor.

### AWR Architekturübersicht

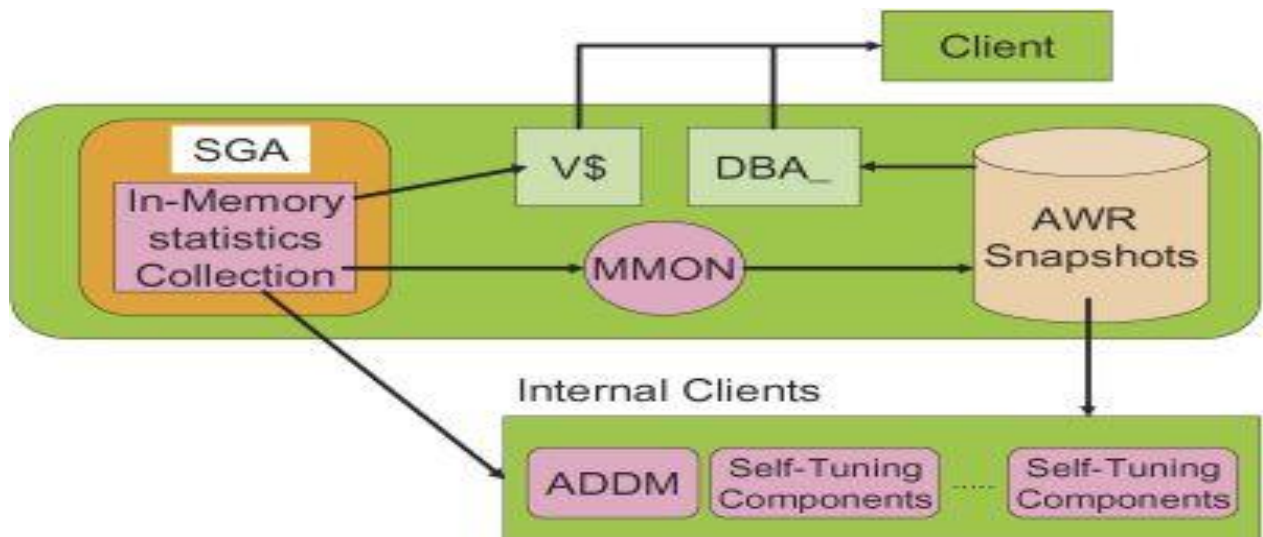


Abb. 1: Ablauf AWR Reports aus Ordix News 2005

Welche Statistiken werden gesammelt?

Das wird mit folgendem Statement angezeigt:

```
SELECT
  statistics_name,
  activation_level,
  Session_settable,
  session_status,
  statistics_view_name
FROM v$statistics_level order by 1;
```

Auswertung von Wait Events per Oracle AWR Report

Der Einstieg in die Analyse erfolgt über die DB Time und die Top 5 Timed Wait Event.

Event	Waits	Time(s)	Avg wait (ms)	% DB time	Wait Class
DB CPU		137,728		15.93	
direct path read	1,467,192	104,098	71	12.04	User I/O
library cache lock	859,232	54,991	64	6.36	Concurrency
db file scattered read	2,482,750	47,306	19	5.47	User I/O
enq: SQ - contention	2,106,732	41,380	20	4.79	Configuration

Abb. 2: TOP 5 Timed Events in DWH Umgebung

In diesem Beispiel beträgt die CPU Time 15,9 % der DB Time Kern Kapitel der Analyse von Oracle Statistik Reports sind seit Oracle Version 10 die Wait Events.

Viel interessanter ist die **DB Time**. Sie ist die Gesamtzeit aller User Prozesse, die während ihrer Datenbankabfrage entweder aktiv arbeiten oder aktiv warten.

Durchschnittliche Active Sessions = Total DB time / Elapsed Time

Die "DB Time" beinhaltet nur Foreground Sessions Incl. CPU, IO time und Wait time  
Die Idle Time fließt hier nicht mit ein

In dem Beispiel Abb 2 Top 5 Timed Events Treten unterschiedliche Waits auf.  
Sie gehören zu unterschiedlichen Wait Classes

Folgende Wait Classes gibt es:

```
SELECT name,wait_class  
FROM v$event_name;
```

Direct Path Read gehört zur Wait Class I/O und ist ein Lesezugriff von Oracle direkt in das PGA Memory. Tritt vorwiegend bei Sortiervorgängen auf und Parallel Query. Durch Einschalten des Async I/O kann die elapsed time und wait time reduziert werden

Bei Asynch I/O wird der I/O Request submitted und der Prozess geht weiter . Das Ergebnis des I/O Requests wird später angezeigt. Bis dahin habe ich ein Direct Path Read erzeugt.Jedoch der Wait wird sofort aufgelöst.

Auch an dieser Stelle muss darauf hingewiesen werden, dass der AWR nicht immer die wirklichen Wartezeiten anzeigt.

Entsprechend der Wait Class USER I/O, Concurrency, System I/O , Configuration werden die dazugehörigen Kapitel des Reports als nächstes untersucht. Performance Tuning Abfragen

Die DB Time wird vorwiegend von SQL Abfragen an die Datenbank verbraucht.

Welche aktuellen Wait Events auftreten und was die Top SQL Statements sind, kann mit folgenden Abfragen weiter untersucht werden. Die SQL\_ID lese ich aus dem Statistik Report

```
SELECT inst_id,event,pi,p2,p3
FROM gv$session_wait
ORDER BY inst_id ,event;
```

Die Sql Statements aus meinem Statspack Report, kann ich mit dem @SPREPSQL Report weiter untersuchen.

Über die „Snap\_id“- Intervalle und dem „old\_hash\_value“ komme ich an die SQL\_ID.

```
SELECT snap_id, old_hash_value,
       sql_id, elapsed_time, address
FROM stats$sql_summary
WHERE old_hash_value=2689373535 and snap_id>=1 and snap_id <=3
ORDER BY snap_id;
```

SNAP_ID	OLD_HASH_VALUE	SQL_ID	ELAPSED_TIME	ADDRESS
1	2689373535	6gvch1xu9ca3g	806536853	00000000DE64AD38
2	2689373535	6gvch1xu9ca3g	806660740	00000000DE64AD38
3	2689373535	6gvch1xu9ca3g	1020297253	00000000DE64AD38

```
SELECT active_session_history_user.user_id,
       dba_users.username,
       sqlarea.sql_text,
       SUM(active_session_history.wait_time +
          active_session_history.time_waited) waited
FROM v$active_session_history active_session_history,
     v$sqlarea sqlarea,
     dba_users
WHERE active_session_history.sql_id=sqlarea.sql_id
AND sql_id='6gvch1xu9ca3g'
AND active_session_history.user_id = dba_users.user_id
GROUP BY active_session_history.user_id,
         sqlarea.sql_text,dba_users.username;
```

Kontaktadresse:

Angelika Gallwitz  
Freiberufliche Beraterin  
In den Hessengärten 48  
D – 61352 Bad Homburg

Telefon: +49(0)6172-488602  
Fax +49(0)6172-944955  
E-Mail [angelika.gallwitz@t-online.de](mailto:angelika.gallwitz@t-online.de)

Internet Adresse: