Oracle DBA Best Practices

# Performance Diagnosis

## Key Concepts

**DB Time**

* total time in DB calls by *foreground sessions*
* includes CPU, IO and non-idle wait time
* sum of DB time for all active sessions

**Average Active Sessions**

* = (DB Time / wall clock time)
* new metric for measuring DB load

1. Use ADDM for DB-wide performance diagnostic
2. Use ASH for targeted performance analysis
   1. first dimension by a time, then by
   2. either SQL ID or session ID or wait class or …
3. Use AWR baseline for comparative analysis

# SQL Optimization

1. Use real-time SQL monitoring to understand execution
2. Use SQL Tuning Advisor
   1. Automatic SQL Tuning
      1. captures high-load SQL
      2. tunes SQL w/o changing apps by creating profiles
      3. validates profiles by test executing them
      4. implements (optional) greatly improved SQL plans
      5. reports analysis
      6. runs during maintenance window
3. Use automatic stats collection to manage Optimizer Statistics
   1. out-of-the-box, runs in maintenance window
   2. restartable
   3. gathers stats on user and dictionary objects
   4. parameters chosen automatically based on DML monitoring, column usage & iterative sampling
   5. more best practices
      1. gather stats for all objects (dictionary and user)
      2. restoring old stats
         1. used when new stats results in poor execution plans
         2. maintained for 30 days
         3. DBMS\_STATS.RESTORE\_TABLE\_STATS
      3. don’t use ANALYZE command
         1. officially obsolete for optimizer stats
         2. can’t gather GLOBAL stats for partitioned objects
         3. invalidates all dependent cursors at once
         4. DBMS\_STATS marks cursors as unusable and recompiles gradually
4. Use SQL Performance Analyzer (SPA) to validate statistics refresh (part of RAT?)
   1. capture workload in STS using automatic cursor cache
   2. execute SPA pre-change trial
   3. refresh stats using PENDING option
   4. execute SPA post-change trial
   5. run SPA report to compare

# Space Management

**Goals**

* eliminate/minimize fragmentation
* optimize data access and transaction performance

1. Use locally management tablespaces for extent management; use Automatic Segment Space Management (ASSM)
   1. online segment shrink remedies fragmentation
      1. ROW MOVEMENT must be enabled for heap organized segments
      2. segment must be in ASSM
   2. Automatic Segment Advisor evaluates segments for fragmentation and makes appropriate recommendations



1. Use locally managed temporary tablespace
   1. allows high concurrency space management
   2. can be shrunk using SHRINK SPACE and/or SHRINK TEMPFILE

**Choosing temp extent size**

* 5M-10M
  + DSS, OLAP apps involving huge sorts, hash joins
  + large temp lobs are predominant
* 64K or multiple
  + global temp tables are predominant and amount of data loaded is small
  + app is predominantly OLTP