

Real World **DBA Best Practices**

Arup Nanda
Long Time Oracle DBA

Who am I

- Oracle DBA for 20 years and counting
- Speaks at conferences, write articles, 6 books
- Brought up the Global Database Group at a major corporation in New York
- Works as a fulltime consultant now

Why this Session

- I have seen and heard too many Rules of Thumb and “Best” Practices.
 - Some of them could be questionable, misleading or even outright wrong!
- Warning: I am questioning everything, leaving nothing to “expert opinions”.
- Sometimes against Oracle recommendations as well!

Guidelines

- Best practices must be justified
 - 👉 *“Use Oracle Flexible Architecture for directory layout”*
 - 👉 *“Ummm ... why?”*
 - 👉 *“Because, it’s the best practice, stupid!”*
 - No Justification → Not Acceptable
- It must apply to all cases or show clearly where it’s applicable
 - 👉 *Best practice is to hot-backup the database*
 - 👉 *Unless, there is a performance gain by running in NOARCHIVELOG mode*
- You must understand. What, How, Why, Why Not – ask these.

Different Oracle Homes


- Traditional Approach:
 - /u01/app/oracle/12.1.0.2
 - Patches are applied to the same OH
- Suggestion:
 - Use a *different* Oracle Home for each upgrade and patching
 - Starter OH: /u01/app/oracle/12.1/db1
 - In next patch, create a different OH
 - /u01/app/oracle/12.1/db2
 - Apply the patch *here*, not on db1

@ArupNanda

© 2018 <http://arup.blogspot.com>

5

New Approach

Step	Traditional	
1	Shutdown DB	
2	Apply Patch	
3		
4		
5	Startup DB	

Database Down upto 3 hours **Could be 5 mins**

100% effective in software only changes

@ArupNanda

© 2018 <http://arup.blogspot.com>

6

New OH Each Time

Current OH: /u01/app/oracle/12.1/db4

New OH: /u01/app/oracle/12.1/db5

1. Reduction in risk of new code.
 - Patch fails -> reset the OH to the old one – db4 and restart the db
 2. Diff is possible, what changed by the patch
 3. Catalog scripts (c*,u*, catproc) preserved
 4. Some binaries are available on older OH
 5. OPatch "bugs" are also mitigated
 - Add to Inventory
- ```
./runInstaller -silent -attachHome -invPtrLoc
./oraInst.loc ORACLE_HOME="/u01.../db5"
ORACLE_HOME_NAME="OraHome121_5"
```

## Dump "OFA"

- Oracle Flexible Architecture (OFA)  
/u01/app/oracle
  - ↳ admin/SID/bdump
  - ↳ admin/SID/udump
  - ↳ oradata/SID/*datafiles*
- Does not allow separation of filesystems for security, performance, physical location, etc.
- Does not allow for passive failovers.

## Non-OFA Layout

- Suggested Layout
  - /oracle
    - ↳ admin/SID/\* - not allowed to users
  - /u01/udump - allowed for users
  - /prddata/SID/mount1/*datafiles* - *high performance*
  - /prddata/SID/mount2/*datafiles* - *low performance*
- This mount point naming convention –  
/prddata/SID/mount1– allows mounting on a different host, for passive failover.
- On QA, use /qadata/SID/mount1/...

## Set Audit Trail

- Set audit\_trail = db
  - Even if you do not need to audit
- True or False: Setting audit\_trail to DB will start auditing and increase I/O?
- FALSE! You need to issue AUDIT statements as well, e.g.
  - SQL> AUDIT SELECT ON EMP;
- This parameter needs a recycle; so set it even if you don't plan on using it.
- 11g already has it by default, unless turned off.

## Audit Session

- Auditing is expensive; we need biggest bang for the buck - *Session Auditing*  
SQL> audit session;  
*[11g has it by default]*
- Purpose:
  - Calculate CPU consumption and profile users
  - Calculate I/O used by users
  - Identify if someone's account was locked after repeated wrong passwords.

## Understand the CPU Usage

```
select username, to_char(logoff_time,'mm/dd') ts,
 count(1) cnt,
 sum(session_cpu) sum_cpu, avg(session_cpu) avg_cpu,
 min(session_cpu) min_cpu, max(session_cpu) max_cpu
from dba_audit_trail
where logoff_time between '&start_date' and '&end_date'
group by username, to_char(logoff_time,'mm/dd')
order by username, to_char(logoff_time,'mm/dd')
```

### Output

| USERNAME | TS    | CNT | SUM_CPU | AVG_CPU | MIN_CPU | MAX_CPU |
|----------|-------|-----|---------|---------|---------|---------|
| USER1    | 04/04 | 3   | 918     | 306     | 17      | 859     |
| USER2    | 04/04 | 36  | 15,286  | 425     | 0       | 4,094   |
| USER3    | 04/04 | 3   | 794     | 265     | 174     | 379     |
| USER4    | 04/04 | 187 | 396,299 | 2,119   | 1       | 124,274 |

## Know Activity by Users

```
select username, to_char(logoff_time,'mm/dd') ts,
 sum(logoff_lread) lread,
 sum(logoff_pread) pread,
 sum(logoff_lwrite) lwrite,
 sum(session_cpu) scpu
from dba_audit_trail
where logoff_time between '&start_date' and '&end_date'
group by username, to_char(logoff_time,'mm/dd')
order by username, to_char(logoff_time,
```

| order by username, to_char(logoff_time, |               |            |                |           |                |
|-----------------------------------------|---------------|------------|----------------|-----------|----------------|
|                                         | Logical Reads |            | Physical Reads |           | Logical Writes |
| Output                                  |               |            |                |           | Session CPU    |
| USERNAME                                | TS            | LREAD      | PREAD          | LWRITE    | SCPU           |
| USER1                                   | 04/04         | 283,271    | 10,858         | 33        | 918            |
| USER2                                   | 04/04         | 4,570,965  | 6,225          | 2,854     | 15,286         |
| USER3                                   | 04/04         | 601,838    | 1,988          | 26        | 794            |
| USER4                                   | 04/04         | 33,639,028 | 4,545,505      | 1,083,473 | 396,299        |

Useful for capacity planning for CPU and I/O, setting appropriate values in profile and so on.

@ArupNanda

© 2018 <http://arup.blogspot.com>

13

## Trace Account Locks

- Identify when someone's account was locked

```
select to_char(timestamp,'mm/dd/yy hh24:mi') ts,
 os_username, userhost, returncode
from dba_audit_trail
where username = 'ARUP'
order by timestamp;
```

Output

| TS             | OS_USERNAME | USERHOST    | RETURNCODE |
|----------------|-------------|-------------|------------|
| 01/10/18 14:12 | arupnan     | CORP\UPNANT | 0          |
| 01/10/18 15:12 | arupnan     | CORP\UPNANT | 0          |
| 01/11/18 04:00 | orandsp     | hndspdb1    | 1017       |
| 01/12/18 04:00 | orandsp     | hndspdb1    | 1017       |
| 01/13/18 04:00 | orandsp     | hndspdb1    | 1017       |
| 01/14/18 04:00 | orandsp     | hndspdb1    | 1017       |
| 01/15/18 04:00 | orandsp     | hndspdb1    | 28000      |

Login  
OK

Wrong  
Password

Account  
Locked

@ArupNanda

© 2018 <http://arup.blogspot.com>

14

## Audit DDL

- Because someone will always complain, what happened to his/her table  
*.... and you are the DBA and you are saying you don't know what happened to it?!!!!!!!!!!!!*
- SQL: AUDIT TABLE BY SESSION;
- stmt\_audit\_option\_map shows the statements
- AUDIT ALL BY SESSION does most DDLs
- *Caveat:* in DW environments, users create and drop a large number of tables; so this may not be advisable.

## No .LOG for Redos

- Common Practice:
  - Redo logs are named <Name>.log
- Problem:
  - Deletion of log files via some cron that deletes “.log” files generated, e.g. sqlnet.log.
  - Remote listener attacks can potentially change the listener log to redo1a.log
- Suggestion:
  - Choose .redo or .rdo for redo log files.



# Listener

- Set Admin Restrictions in LISTENER.ORA
  - Set `ADMIN_RESTRICTIONS_LISTENER=on`
  - This prevents online modification of the listener parameters
  - Modify the `listener.ora` file and use  
`$ lsnrctl reload`
  - Completely online operation.
- Why?
  - This forces you to place comments in the `listener.ora` file to document why a change was made
  - Eliminates the risk of remote listener hijacking threats, a common listener vulnerability

# External Procedures

- Common Practice
  - The `listener.ora` has External Procedures
- Problem
  - Extprocs are external programs running as "oracle"
  - Gateways for many hacker attacks.
  - Most people don't use it; yet it's defined by default
- Recommendation
  - Remove it from `listener.ora`
  - If needed, use a different listener, with only `extproc`
  - In case of any threats, you can shut it down while not affecting normal database traffic

## Create a Controlfile on Trace

- Execute:  
`SQL> alter database backup controlfile to trace as '/path/cr_db.sql' reuse;`
- It creates a CREATE CONTROLFILE script
  - You can use it to recreate controlfile
  - Or, the database itself
  - Self documenting the datafiles and redo logs
- Change Control:
  - Write a separate file for each day
  - Do a diff to find the added files, redo logs, etc.

## Build a Metadata Repository

- Use Data Pump to Create a Repository of Objects, Grants, Tablespaces, etc.:  
`$ expdp u/p content=metadata_only full=y  
diectory=tmp_dir dumpfile=md.dmp`
- Import this to create an SQL File  
`$ impdp u/p diectory=tmp_dir  
dumpfile=md.dmp sqlfile=md.sql`
- See my paper: Datapump: Not Just for Data Movement

## Validate Backups

- Do you know if your backups are any good?
- Use RMAN Validation
- After the backup is taken  
`validate backupset 6;`
- If corruptions detected:  
`validate found one or more corrupt blocks`

## Preview RMAN Restore

- Always preview a restore  
`RMAN> restore tablespace users preview;`
- Does not actually restore but checks the availability of files
- Not the same as VALIDATE
  - Preview checks what files are required
  - Validate assumes you know that
- Not the same as TEST  
`RMAN> restore tablespace users test;`
  - Preview does not actually start the recovery process; so the tablespace need not be offline.
  - Test needs it to be OFFLINE

```
Recovery must be done beyond SCN 14984187 to clear datafile fuzziness
Finished restore at 20-AUG-17
```

```
Starting recover at 20-AUG-17
using channel ORA_DISK_1
```

```
RMAN-00571: =====
```

```
RMAN-00569: ===== ERROR MESSAGE STACK FOLLOWS =====
```

```
RMAN-00571: =====
```

```
RMAN-03002: failure of recover command at 08/20/2017 02:48:42
```

```
RMAN-06555: datafile 13 must be restored from backup created before 20-AUG-17
```

## Save RMAN Log

- You copy to tape:
  - RMAN backup files
  - Init file
  - Archived logs
  - But not RMAN Log files, do you?
- RMAN Logs contain information about the backup pieces, names, location, etc.
- Proves invaluable during recovery

```
input datafile fno=00084 name=/f1.dbf
output filename=/backup/loc3/data_D-CRMPRD_I-
79785763_TS-DWT_ODS8_RES_FN
O-96_43ie2scm.rman tag=FULLBKPFS recid=174298
stamp=618757792
```
- Allows you to look for specific files from backup sets

# DBID

- Important for Recovery
- Note the DBID and keep it in a separate place
- Write DBID to alert log every time backup is taken

```
declare
 l_dbid number;
begin
 select dbid into l_dbid
 from v$database;
 dbms_system.ksdwrt(2, 'DBID=' || l_dbid);
end;
```

# Do Not Use SPFILE

- SPFILE Advantages:
  - Can be on shared filesystem, incld. on ASM
  - Can be backed up by RMAN
  - Can be updated automatically by command line by ALTER SYSTEM SET ... SCOPE = SPFILE;
  - >10g, CREATE SPFILE FROM MEMORY
- SPFILE Disadvantage
  - Older version overwritten
  - Comments possible; but only for the current entry

## PFILE Advantages

- Place comments in the init.ora file

```
AKN 3/20/17 added because ...
RJN 4/10/17 changed from 1M to 2M
JER 10/3/17 changed from 2M to 4M
DFW 12/7/17 changed from 4M to 6M SR# ...
log_buffers = 6M
```
- Has a history of changes, with the names and dates of changes
- Very useful for troubleshooting and for record keeping

## If you must use SPFILE

- Make sure you have a version control system in place to track parameter changes
- Example:
  - SQL> create pfile='/tmp/a' from spfile;
  - Check diff between this and the previous
  - Write the differences to a log file
- In >= 11g, you can create PFILE from memory:

```
SQL> create pfile='...' from memory;
```

## New Oracle User for Clients

- Problem:
  - App running on the DB server, needs SQL\*Plus
  - \$OH/bin/sqlplus is not accessible to world
- Common Solution:
  - Change \$OH permissions to allow all others
  - Make app part of the “dba” group
- Suggestion:
  - Create a separate Oracle user: “appora”
  - Install the Oracle client under that user

## Reliable Archivelog Location

- Rate the most important
  - Datafiles
  - Archivelogs
  - Backup of datafiles
- Most important is *archivelogs*
  - If datafiles are lost, they can be recreated
  - Archived logs are *never* recreatable
  - Missing archived logs = halted recovery
- Separate DB Backup from Archived Logs
- Flash Recovery Area
  - Do NOT use for Archived Logs

## Multiple ASM Disk Groups

- Common Practice
  - Create a single disk group for everything, because it simplifies administration
- Problem
  - If a single disk encounters a problem, you affected everything
- Recommendation
  - Create at least 4 groups for database – Data, Index, Redo1 and Redo2. Arc Log and Backup separate.
  - If index diskgroup has an issue, you can at least rebuild the indexes
  - May not be relevant for a large number of small DBs

## Separate Instance and DB Names

- Common Practice:
  - DB\_NAME same as Instance Name
- Suggestion:
  - Append “1” after DB Name for Instance, e.g.
    - DB Name: PRODB
    - Instance: PRODB1
  - If you ever need to convert the DB to RAC, you will not need to change the Instance Name
  - No need to change Init.ora, PW File, etc.



## Use oraenv

- Oracle supplied tool, in \$OH/bin
- Look up the OH in /etc/oratab or /var/opt/oracle/oratab (in Solaris)
- Use this instead of manually issuing export ORACLE\_HOME=..., ORACLE\_SID=...
- Why this?
  - One place for ORACLE\_HOME info
  - It makes your job easier while changing OH
  - It makes a consistent interface – jobs, commands

## Enable Service Names

- In the instance, check service names present already:  
`SQL> show parameter service_names`
- Create additional service names:  
`SQL> alter system set service_names =  
'SVC1', 'SVC3', 'SVC3';`
- Check is listener is listening for these:  
`$ lsnrctl services`
- In RAC, you should use SRVCTL:  
`$ srvctl add service -d MYDB -s SVC1 ...`

## Service Names

- Oracle database can be accessed via SID or Service Name
- Conventional TNS Entry

```
prodb1 =
(DESCRIPTION =
 (ADDRESS_LIST =
 (ADDRESS = (PROTOCOL = TCP)(HOST = prolin1)
 (PORT = 1521)))
 (CONNECT_DATA = (SID = PRODB1)))
```

- Service Name

```
(CONNECT_DATA = (SERVICE_NAME = PRODB1)))
```

## Why Service Names?

- No change in functionality
- Separates use from user, e.g. SCOTT logging from laptop uses service SVC1; but from app server SVC2.
- Enhances resource manager use
- Metrics on services V\$SERVICEMETRIC
- Allows load balancing and failover in RAC, Streams or Data Guard environments
- Allows fine grained failover capabilities
  - Service SVC1 fails from node1 to node2; but SVC2 fails to node3

## Kill Inactive Sessions

- Problem:
  - Some apps, especially web apps under connection pool, remain inactive draining resources.
- Suggestion:
  - Use resource manager and set the inactive session disconnect timeout
- Why RM, why not Profiles?
  - RM allows you to turn on and off via scheduling and event. Profiles are hard coded.
  - RM allows service name based control

## Using ORADEBUG

- Problem:
  - Database Issue; you want to use oradebug; but SQL\*Plus hangs!
- When SQL\*Plus does not work, use  
`$ sqlplus -prelim`  
It does not establish a connection  
You can run ORADEBUG now

## Parallel Query Groups (RAC)

- In a RAC Database, parallel query slaves can go to any instance, causing inter-instance traffic to go up.
- To control this, put in init.ora (static)  
MYDB1.INSTANCE\_GROUPS='node1','all\_nodes'  
MYDB2.INSTANCE\_GROUPS='node2','all\_nodes'
- When using PQ, use:  
SQL> ALTER SESSION set  
parallel\_instance\_group = 'node1'
- This will force the PQ slaves to go to only node1;  
not to node2.

## Check Listener Log

- Create External Tables on Listener Logs to identify issues, profile users, etc.
- See my blog [arup.blogspot.com](http://arup.blogspot.com)

## Remember

- It's *not* a best practice, if it is not justified
- You have to understand *why*; not just *what*
- Best practice needs to be situation-aware
  - Which goes back to “you have to understand”
- Always question whenever someone tells you it's a best practice
- Always spread your own best practices with correct rationale.

# Thank You!

## Q&A

Download from Blog