**Oracle Database Administration- Oracle 12c Upgrade Document**

**Description of the Organization**

Mountain consulting law firm is a small law firm comprised of five lawyers, six paralegals, three administrative staff and three IT analysts (two system analysts, one DBA) serving the Washington DC area. The firms’ five lawyers represent and assist clients in various cases within the jurisdiction of Virginia and District of Columbia. Each lawyer in the firm has a docket. Each docket can contain one or more clients. Each client can have one or more cases. Providing support to the lawyers are a group of paralegals. At any given time one or more paralegals can manage one or more cases.

Owing to a partnership with Oracle Corporation until recently the firm was utilizing Oracle 11g database to persist employee assignments, client relationships and case data. The firm has now decided to upgrade to the latest Oracle 12c databases. By having a detailed record of client cases and the personnel assigned to those work items the firm wants to generate productivity metrics in order to make recommendations to the CEO for a possible expansion of the firm. The firm hopes to evolve its data model in a secure, reliable and consistent way for the coming years.

**Description of the server environment (servers, disk sizes, networks, OS types)**

The firm has 15 Windows Server 2008 R2 boxes in place for its employees to work on. Each server is named mountainviewlawbox##legal or moutnainviewlawbox##admin accordingly to distinguish machines with access to legal data and resources and those with admin access. The OS type on all boxes is Windows. The disk sizes for individual servers are 400 GB. The office is setup to have a local area network. Each machine is installed with Oracle 11g database.

**Description of the current database software being utilized by the organization**

The firm currently uses Oracle 11 g software. The current software allows for the maintenance of specific tablespaces and associated tables for the LAWYER, DOCKET, CLIENT, PARALEGAL and CASE entities. The current architecture is not multitenant and this is what the law firm would like to eventually change to in the coming time in order to utilize pluggable databases. The first part of this upgrade will enable to firm to use 12c database after the initial upgrade is done the firm will hire a DBA consulting firm to setup pluggable databases. The current database is maintained by the DBA and is accessible to the administrative staff data entry clerk to enter and update latest details. The database is exposed to the administrative staff and legal team via a front end portal that has each user authenticate into the system and into their particular dashboard screen which displays docket view and personnel assignments.

**Entity Relationship Diagram of the Mountain Consulting Law Firm**

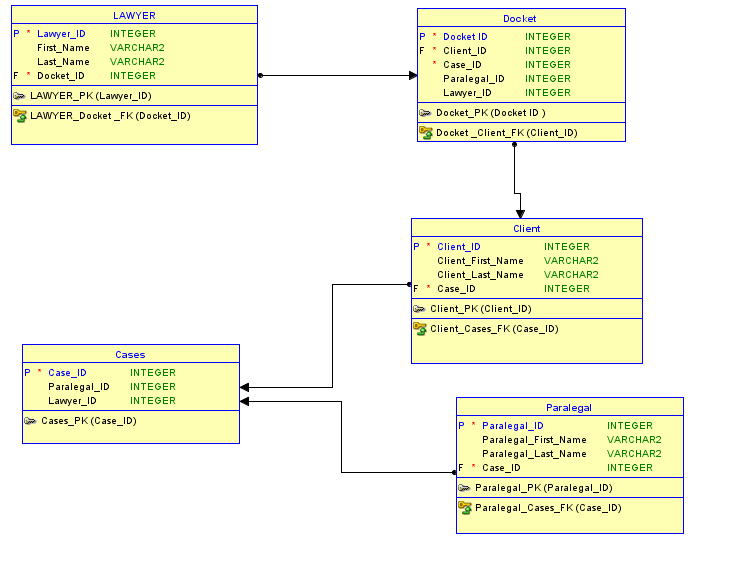
**(0,1) Lawyers has a Docket(0,1)**

**(0,1) Docket has Clients(1,N)**

**(0,1) Client have Cases(1,N)**

**(1,N) Paralegals assist Lawyer(0,1)**

**(1,N) Paralegals manage Cases(0,1)**



**Description of the new server environment**

The firm will not be requiring new servers as they plan to patch their existing boxes with the Oracle 12 c upgrade. The firm will therefore be maintaining the 15 Windows Server 2008 R2 boxes in place for its employees to work on. Each server is named mountainviewlawbox##legal or moutnainviewlawbox##admin accordingly to distinguish machines with access to legal data and resources and those with admin access. The OS type on all boxes is Windows. The disk sizes for individual servers are 400 GB.

**Oracle Software file structure**

Mountain View Law Firm Database structure follows Oracle Flexible Architecture for each of its servers.

Oracle Inventory Directory - **/home/oracle/oraInventory**

Oracle base Directory - **/u##/app/oracle**

Oracle home directory- **/ORACLE\_BASE/product/11.2.0.2/mountainlaw\_oracle**

Automatic Diagnostic Repository- **/u##/app/oracle/diag/rdbms/11g/011G**

As this is a windows system, the system administrator creates **OS group (dba)** AND associated **OS user (oracle) on the servers.**

Set of directory where oracle is installed- **/home/oracle/orainst/11.2.0.2**

Control Files are located at- **/u##/oradata/mountainlaw\_controlfile/control01.ctl**

Redo Logs are located at- /**u##/oradata/mountainlaw\_redologs/redo.log**

Tablespaces are located at - **/u##/dbfile/mountainlaw\_tblspace/lawyers01.dbf**

**Creating necessary Tablespaces**

The first tablespace is lawyers tablespace. It is meant to hold tables for primary, long term employees such as firm attorneys and partners.

**create tablespace lawyers**

**datafile '/ u01/dbfile/mountainlaw\_tblspace/ lawyers01.dbf'**

**size 100m**

**autoextend on maxsize 1000m**

**extent management local**

**uniform size 128k**

**segment space management auto;**

The second tablespace is paralegals tablespace. It is meant to hold tables for short term employees such as interns and paralegals.

**create tablespace paralegals**

**datafile '/ u01/dbfile/mountainlaw\_tblspace /paralegals01.dbf'**

**size 100m**

**autoextend on maxsize 1000m**

**extent management local**

**uniform size 128k**

**segment space management auto;**

The third tablespace is the docket tablespace. It is meant to hold tables for clients and cases for the firms attorneys.

**create tablespace docket**

**datafile '/ u01/dbfile/mountainlaw\_tblspace /docket01.dbf'**

**size 100m**

**autoextend on maxsize 1000m**

**extent management local**

**uniform size 128k**

**segment space management auto;**

**Manual Upgrade using Command Line Interface**

The manual upgrade utilizes the command line interface to perform the database upgrade to a new version. This allows for pinpoint control as to where components are moved and how they are updated. In summary there are the following steps required to carry out the manual upgrade. The first part is the preparation step and includes taking a backup of the database in case that the upgrade process is unable to complete. In this case we take a hot backup of the database. The second part is upgrading the database using scripts.

**Performing a hot backup**

In this step we backup the database and create a restore point. In case anything goes wrong we can restore to a previous healthy state.

First we need to validate the location of the database, it is usually under the following location (**app/oradata/mountainlaw11g**).

Second, we need to make sure we have all the control files, redo log files and data files in the right location. We should locate the pfile and sp files of the database instance and back them up. Then we can connect to the database instance using (**SQL: sqlplus / as sysdba**) We should verify that we are connected to the right database using (**SQL: select name from v$database;)** We should next make sure that the database is in archive log mode using (**SQL: select log\_mode from v$database**;) If the database is not in archive log mode then put the database in the archive log mode by looking at the location where the archive logs would be written using (**SQL:  show parameter log\_archive\_dest\_1**;). We should then set the destination of the log location as follows.( **SQL:  alter system set log\_archive\_dest\_1=’LOCATION=<path to desired directory>’ scope=spfile;**). To make sure our changes translate we should shut down the database using (**SQL: shutdown immediate**;). Then we should start the database in mount mode using (**SQL:  startup mount**;). We should issue the alter command to start archive logging using (**SQL : Alter database archivelog**;). Once the database has been enabled to be in archive log mode we should open it up for transactions using (**SQL: Alter database open**;)

In order to complete the process of taking a hot backup, we need to find out the number of tablespaces associated with the database using (**SQL: select tablespace\_name from dba\_tablespaces**;). Run the command to verify if the system is ready to backup tablespaces using (SQL: **select \* from v$backup**;) .Put the tablespaces in hot backup mode using (**SQL: Alter database begin backup**;)( **SQL: select \* from v$backup**;) Next, you should copy the tablespace files on the hard drive to the backup location. Next you should backup the control file using (**SQL: Alter database backup controlfile to ‘<path>\ backup filename’**). And finally you should backup the archive logs to the backup location. This completes the hot backup of the database.

**Restoring the oracle database from a hot backup**

**In case that you need to restore the oracle database from a hot backup you need to copy the tablespace files and control files from backup location to installation directory of database instance. You should rename all files to their original version. You should not copy the redo logs.**  Start the database in mount mod**e (SQL: startup mount**;) and recover the database using the following command (**SQL: Recover database until cancel using backup controlfile**;) Once you have all the logs restored you should run the following query to open the database for transactions.(**SQL: Alter database open resetlogs**;)

**Upgrading the database via Command Line Interface**

The upgrade process starts with the preupgrade steps. First we need to stop and remove DBConsole using **([bash ~]$ emctl stop dbconsole).** Then we need to switch to /rdbms/admin and execute emremove.sql using the following commands.

**[bash ~]$ cd /opt/oracle/app/product/12.1.0/dbhome\_1/rdbms/admin/**

**[bash admin]$ sqlplus / as sysdba**

**(SQL> @emremove.sql)**

Secondly, we should switch to /rdbms/admin and execute the preupgrd.sql script using the following commands.

**[bash]$ sqlplus / as sysdba**

**(SQL> @preupgrd.sql)**

We should then check the output by looking at log files that are generated at /cfgtoollogs//preupgrade. If there are inconsistencies then you should use the preupgrade\_fixups.sql script.

Next you should shutdown database, copy needed files (pfile/spfile and password file) to new oracle home

**(SQL> shutdown immediate)**

**(SQL> exit)**

**[bash ~]$ cd $ORACLE\_HOME/dbs**

**[bash dbs]$ cp spfileDBUPGRD.ora orapwDBUPGRD /opt/oracle/app/product/12.1.0/dbhome\_1/dbs**

Next you need to prepare for the upgrade process by changing the environment as follows.

Notice: Under Windows you have to create a windows service with oradim

**[bash dbs]$ export ORACLE\_HOME=/opt/oracle/app/product/12.1.0/dbhome\_1**

**[bash dbs]$ export PATH=$ORACLE\_HOME/bin:$PATH**

**[bash dbs]$ sqlplus / as sysdba**

**(SQL> startup nomount;)**

**(SQL> alter database mount;)**

**(SQL> alter database open upgrade;)**

To perform the upgrade you should enter the following commands.Specify -n or -n 0 to disable no parallel.

**[bash dbs]$ cd $ORACLE\_HOME/rdbms/admin**

**[bash admin]$ pwd**

**/opt/oracle/app/product/12.1.0/dbhome\_1/rdbms/admin**

**[bash admin]$ $ORACLE\_HOME/perl/bin/perl catctl.pl -n 4 catupgrd.sql**

You should keep monitoring the alert log below to see any inconsistencies in the upgrade process.

**grep SERVER /opt/oracle/app/diag/rdbms/dbupgrd/DBUPGRD/trace/alert\_DBUPGRD.log**

Once the upgrade is complete you need to perform the post upgrade steps.

**[bash admin]$ sqlplus / as sysdba**

**(SQL> startup)**

**ORACLE instance started.**

**(SQL> @?/rdbms/admin/utlu121s.sql)**

You should next run the mandatory catuppst.sql script by issuing the following command.

**(SQL> @?/rdbms/admin/catuppst.sql**

To check if the upgrade progressed without any errors you can compile statistics and do a count on the invalid status.

**(SQL> execute dbms\_stats.gather\_fixed\_objects\_stats;)**

**(SQL> @?/rdbms/admin/utlrp.sql)**

**(SQL> select count(1) from dba\_objects where status='INVALID';)**

**COUNT(1)**

**----------**

**0**

If you do see any errors then you can execute tasks to fix components via the “postupgrade\_fixup.sql” script.

**[bash]$ sqlplus / as sysdba @/opt/oracle/app/product/11.2.0.3/dbhome\_2/cfgtoollogs/DBUPGRD/preupgrade/postupgrade\_fixups.sql**

**This completes the database upgrade to 12c from 11g.**

**High Level Flowchart of the Upgrade Process**

Conclude Upgrade to 12 c

If Upgrade fails revert to Backup.

Check if DB is in Archive mode then Backup DB.

Run catuppst.sql

Copy needed files (pfile, spfile, ctrl files ,tablespaces) to new home

Hot Backup

Run postupgrd\_fixups.sql

Check Invalid Objects

Monitor log

Run catupgrd.sql

Alter DB open Upgrade

Mount DB

Change Env

Shutdown DB

Run preupgrd.sql

Pre Upgrade Steps

Works Cited

Kuhn D. (2013) Pro Oracle Database 12c Administration: Second Edition: New York, NY.Apress

Oracle Docs. (2012). [Oracle Database Online Documentation 12c Release 1 (12.1)](https://docs.oracle.com/database/121/index.html) Retrieved from: http://docs.oracle.com/database/121/UPGRD/intro.htm#UPGRD52616

Journey with Database.(2015). Oracle Database Online Documentation for Upgrade Process Retrieved from: <http://tutorial-sql-databases.blogspot.com/2014/09/manual-upgrades-to-oracle-database-12c.html?_sm_au_=iVV1j17D6SBtv7Mm>

Gumpx: DBA’s online Diary.(2015). Upgrading Manually to Oracle 12c database. Retrieved from: https://gumpx.wordpress.com/2013/07/13/upgrade-to-oracle-12c-database-using-manual-method/