

# New York Oracle Users Group

## Migrating To 10g Database On Linux



# Company Facts

- Established 1993 as Digital Telemedia
- Two live data centers (NY, NJ)
- One backup facility (NY)
- All facilities interconnected via private Gig-E fiber
- 4 Gbps aggregate outbound bandwidth (2 Gbps per “live” facility)
- Bandwidth scalability available from Qwest and AboveNet
- 99.999% Uptime SLA
- “Managed Database Hosting” practice with 100% hardware SLA to support ASPs and enterprise clientele
- Most hosted ASPs in Financial Services and Medical/Pharmaceutical vertical markets

# Metro Area Network

## AboveNet, 111 8<sup>th</sup> Avenue, NYC

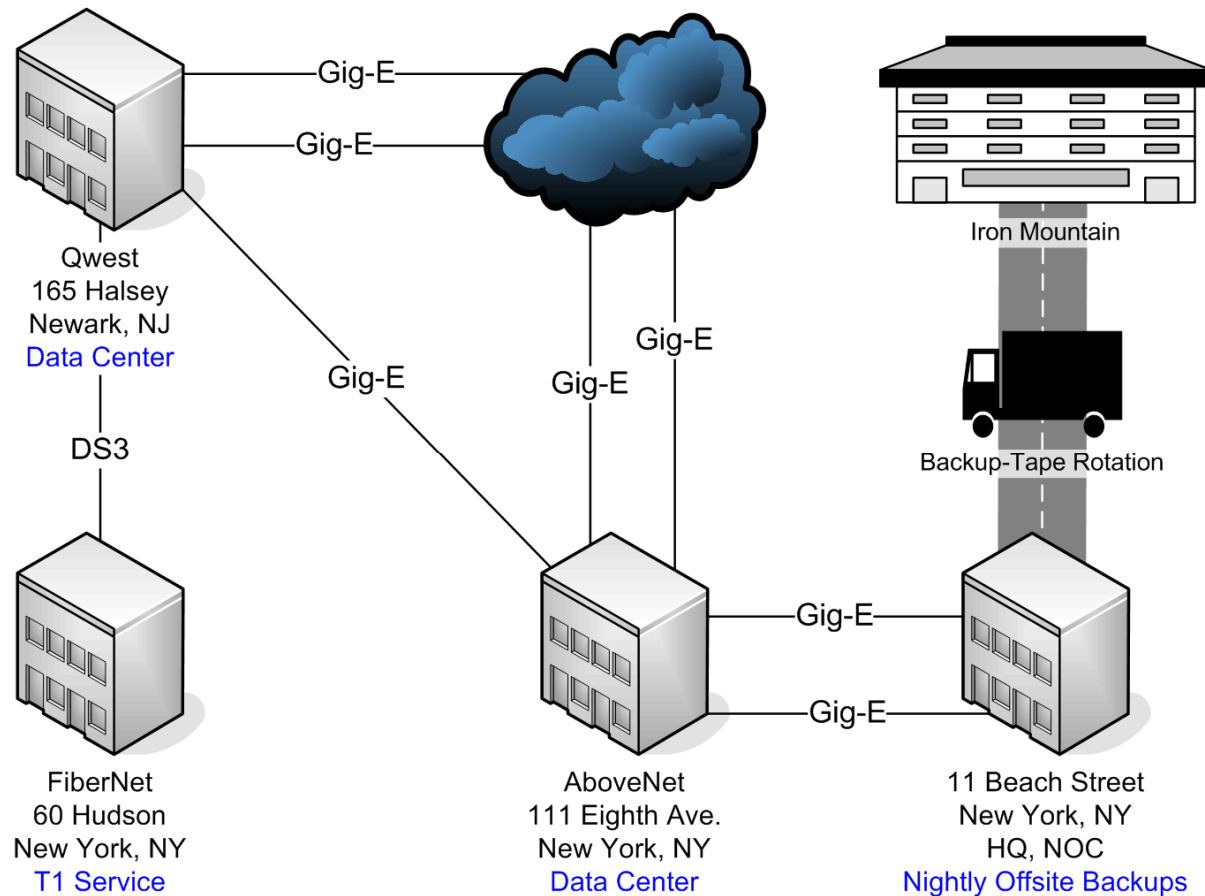
- Internally wired via multiple OC48s, to which Logicworks is connected via dual Gig-E fiber on Cisco 6509s.
- Backbone routers are Juniper M160s, and Edge Routers are GSR 12000s.
- The site has multiple outbound OC48 and OC192

## Qwest, 165 Halsey, Newark, NJ

- The data center itself is SAS70 Type II compliant, and connected to transport via a pair of OC48s on SONET.
- OC48s connect to a pair of Cisco 12016 routers, and from there go to each individual co-location switch, which are Cisco 6509s exclusively.
- Logicworks is connected via dual Gig-E.
- The site has access to multiple outbound OC192s.

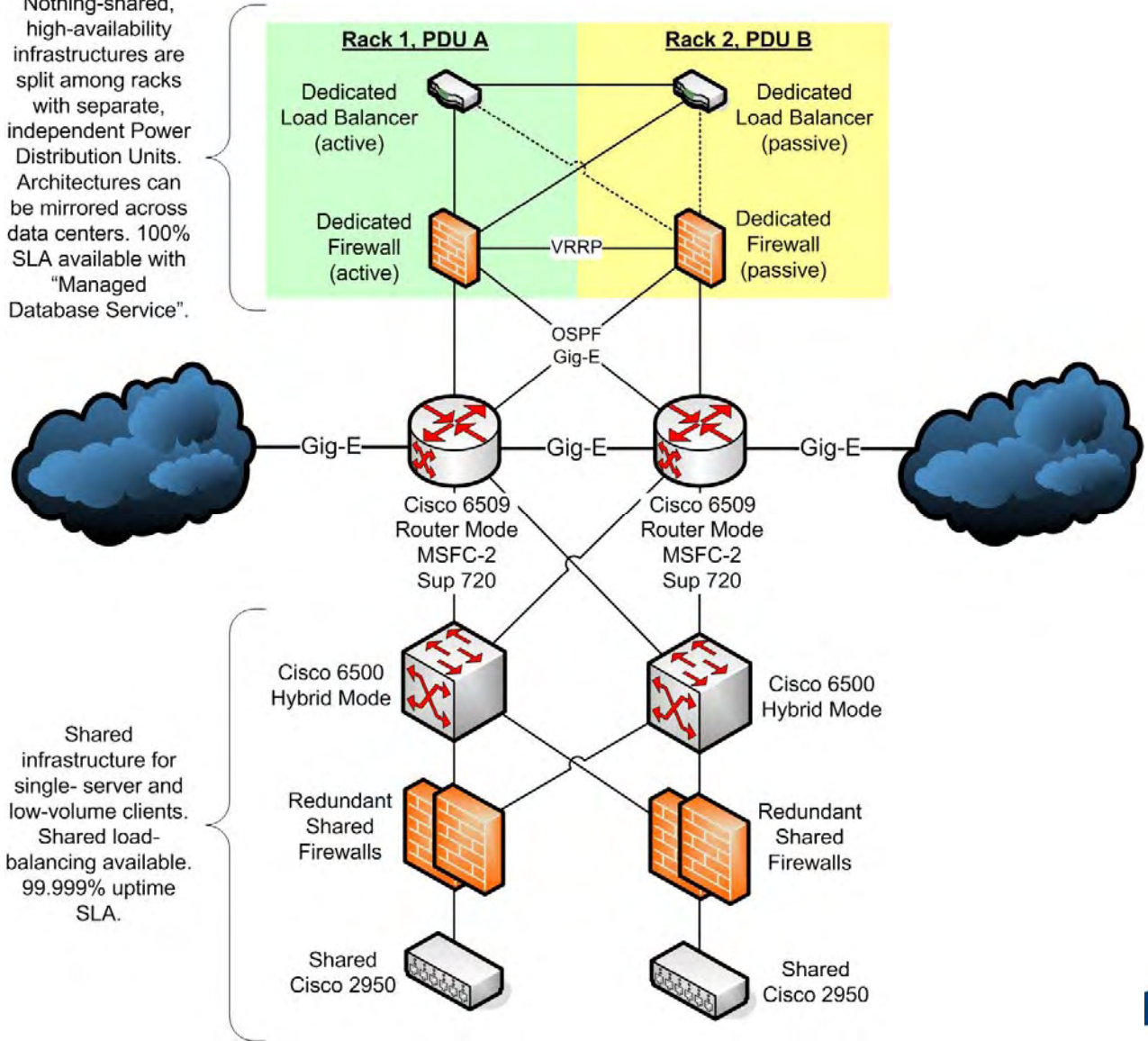
## Qwest and AboveNet

- Neither facility was adversely affected during the "Blackout" of 2003, having automatically and seamlessly switched to diesel generator backup power.
- The facilities are located on independent electrical power grids.



# Representative High-Availability Architecture

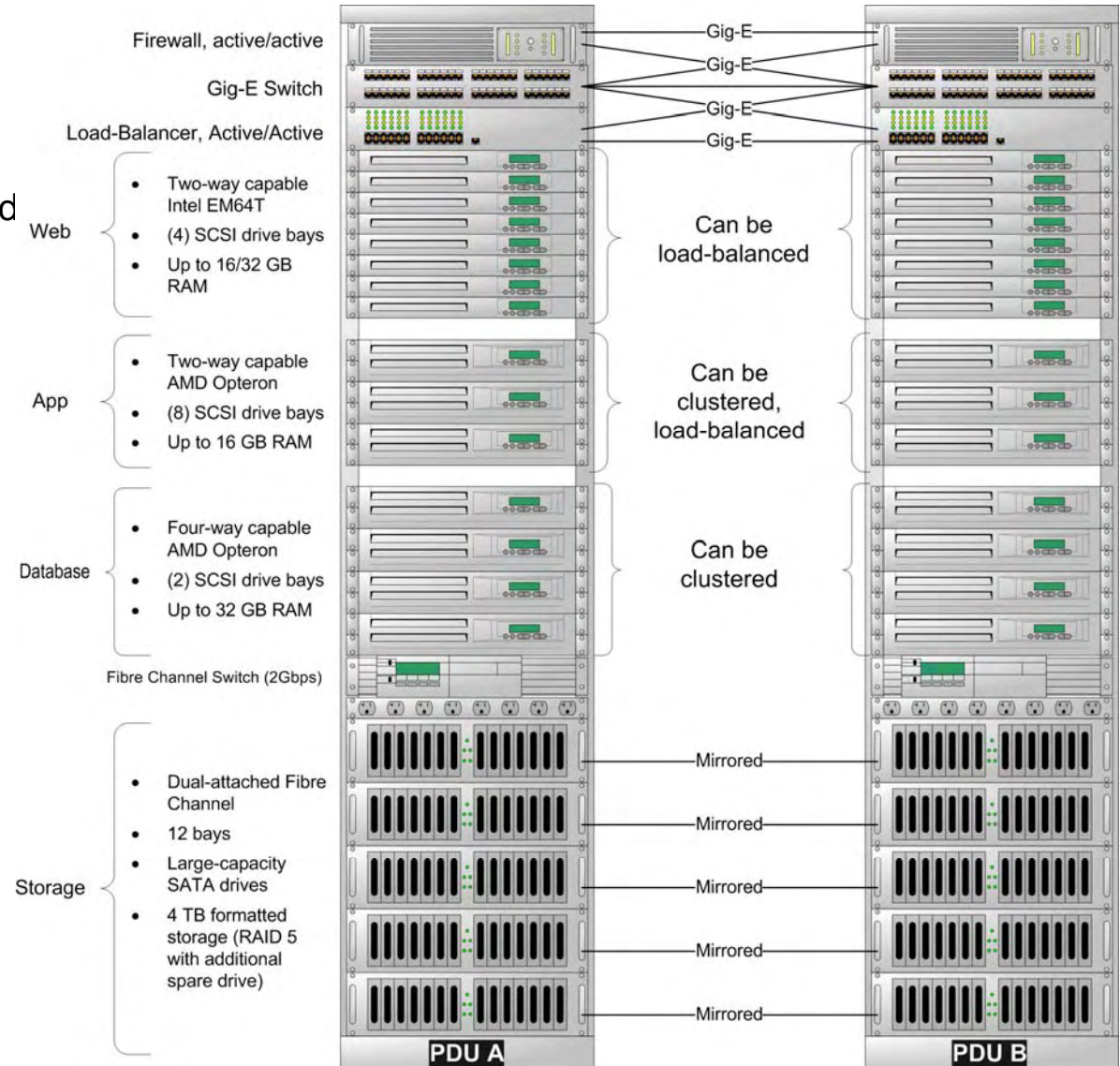
Nothing-shared, high-availability infrastructures are split among racks with separate, independent Power Distribution Units. Architectures can be mirrored across data centers. 100% SLA available with "Managed Database Service".



Shared infrastructure for single-server and low-volume clients. Shared load-balancing available. 99.999% uptime SLA.

# Representative Hosting "Island"

- Illustrated is an "Island" with two racks.
- Full redundancy at all levels.
- Additional racks can be added as needed.
- Racks can be mirrored within Islands.
- Islands can be mirrored across data centers.
- Racks fed by independent Power Distribution Units.
- Racks are interconnected via meshed Gig-E.
- Load-balancers and firewalls can perform functions for either or both racks.
- Homogeneous hardware inventory maintained by Logicworks.
- Hardware and server specifications to be co-developed with client.



# Managed Database Hosting Service

- 100% Hardware SLA
- Install and configure the database application;
- Install and/or upgrade any server software and security patches;
- Generate regular status reports (configure built-in reporting tools);
- Manage backups and data restoration;
- Configure, implement, and test replication and dynamic failover architectures;
- Co-develop failover procedures with customer;
- Cooperatively manage failover and fail-back scenarios;
- Alerting through comprehensive health-tests;
- Application viability monitoring (read/write tests);
- Advisory services or identification root-cause of problems (when applicable).

# Logicworks Enterprise Hosting Advantages

- Hardware and services are implemented successfully and in a significantly shorter period of time;
- Successful implementations result in quicker ROI;
- Homogeneous, streamlined infrastructure and server environment for optimum performance, reliability, and maintenance;
- Client staff can concentrate on core skills, business objectives, and revenue streams;
- Clients no longer need to expend limited and valuable resources on provisioning and vendor management;
- Client IT staff can focus applications, data, and data integrity instead having to spend time on hardware and maintenance issues;
- Deployed solutions are monitored and serviced 24/7 for maximum uptime and reliability;
- Professional, proactive network maintenance;
- Problems can be identified and resolved more quickly;
- Adverse events are minimized or avoided entirely;
- Business continuity can be easily implemented and activated through fast, secure data backup and restoration protocols;
- Easy migration to the newest hardware;
- Easy and quick scalability, upward and outward;
- Clustered and high-availability solutions – also available for retrofitting;
- Continuous improvements through reinvestments in infrastructure;
- Cutting-edge technologies are available cost-effectively and hassle-free;
- Redundant outbound connections from independent providers;
- Hosting solutions are located in battle-proof data centers (neither facility was adversely affected by the 2003-Blackout);
- Logicworks' data centers are connected by our own Gig-E fiber facilitating true geographically diverse high-availability hosting architectures.
- Spare parts for absolutely everything;
- Diesel generator backup with enough fuel for prolonged outages;

# Activities & Capabilities

- Exclusively providing managed hosting services since 1998.
- Offering multi-tier enterprise-level solutions since 2000.
- 24/7 on-site expert support.
- 2-hour hardware replacement guarantee.
- All-inclusive pricing for managed services.
- Partnered with Oracle, Microsoft, VERITAS, and MySQL for vendor support.
- Support agreements with Red Hat, AboveNet, Qwest, Time Warner Telecom, Cisco, QLogic, Foundry Networks, and server hardware and component vendors and suppliers.



# Contacts

Logicworks, 11 Beach Street, New York, NY 10013

– (212) 625-5300, [www.logicworks.net](http://www.logicworks.net)

- Shirish Ojha, Senior Oracle Engineer

– (212) 625-5363, [sojha@logicworks.net](mailto:sojha@logicworks.net)

- Nirmal Mehta, Senior Oracle Engineer

– (212) 625-5365, [nmehta@logicworks.net](mailto:nmehta@logicworks.net)

- Marc Simony, Director of Business Development

– (212) 625-5369, [msimony@logicworks.net](mailto:msimony@logicworks.net)

# Step 1 – Prepare To Upgrade

- 1.1 Become Familiar with Features of 10g
- 1.2 Determine Upgrade Path to 10g
- 1.3 Choose Upgrade Method
- 1.4 Choose Oracle Home Directory for 10g
- 1.5 Prepare a Backup Strategy
- 1.6 Develop a Testing Plan

# 1.1 – Become Familiar with Features

- Oracle 10g Database New Features Guide .
- Oracle 10g Database Administrator's Guide .
- Oracle 10g Database Reference Guide for new/changed/deprecated initialization parameters.
- Other Documents at <http://otn.oracle.com/>

## 1.2 – Determine Upgrade Path

- Direct upgrade from 9iR2 (9.2.0), 9i (9.0.1), 8i (8.1.7), 8.0.6
- For others only a “double-hop” upgrade is supported
  - 8.1.5 -> 8.1.7 -> 10.1.0
  - 7.3.4 -> 9.2.0 -> 10.1.0

*What is difference between Upgrade and Migrate ?*

## 1.3 – Choose Upgrade Method

- Database Upgrade Assistant interactively steps you through the upgrade process and configures the database for 10g.
- Manual Upgrade consists of running SQL scripts and utilities to upgrade a database to 10g.
- Export/Import utilities physically copy data from your current database to the new 10g database.
- Data Copying from one Oracle database to another using database links.

# Upgrade Methods

- Database Upgrade Assistant (DBUA)
  - Adjusts configuration automatically
  - Initialization parameters
  - Tablespace sizes
- Creates files in new ORACLE\_HOME
- Interacts with DBA for decisions
- Displays upgrade progress
- Reports errors
- Review upgrade results

# Manual Upgrade

- Run Pre-Upgrade Information Utility
  - utlu101i.sql
- Make some adjustments in existing environment, others in new environment
- Run upgrade script (e.g., u0902000.sql)
- Perform post upgrade operations

# Export/Import

- Create new 10.1.0 database
- Precreate tablespaces (if the paths of datafiles are different)
- Import specific schemas or all schemas



# CTAS

- SQL statements
- CREATE TABLE .. AS SELECT
- INSERT .. SELECT

# Step 2 – Test Upgrade Process

- 2.1 Create a test environment that will not interfere with the current production database.
- 2.2 Practice upgrading using the test environment.
- 2.3 Perform upgrade test on an exact copy of the database to be upgraded.

# Step 3 – Test Upgraded Database

- 3.1 Perform planned tests.
- 3.2 Compare the results, noting anomalies. ( you will always get queries which are slow, same or fast than the previous version. The CBO changes in every version. Try using outlines, hints etc to work on problematic queries)
- 3.3 Repeat the test upgrade as many times as necessary.
- 3.4 Test the upgraded test database with existing applications to verify that they operate properly.

# Step 4 – Upgrade to 10g

- 4.1 System Considerations and Requirements
- 4.2 Install Release 10.1 Oracle Software
- 4.3 Upgrade Using the Database Upgrade Assistant
- 4.4 Upgrade Manually

# Which Upgrade Approach To Use?

- Generally Oracle recommends you use DBUA as it will upgrade installed Server options in the proper order, check disk space requirements, and will optionally perform a full cold database backup.
- DBUA runtime is directly related to the installed Server options, but not database size, number of tablespaces, etc.
- Interactive DBUA provides more upgrade options and is most useful when a few databases need to be upgraded in a short period of time.
- Use Export/Import when changing operating systems, database block size, etc OR when database size is small.

## 8.1.7 Upgrade Notes

- You need 170MB of free space in the SYSTEM tablespace to complete the upgrade. Interactive DBUA will require you to satisfy this requirement, but silent mode DBUA will appear to run successfully without sufficient free space yet the alert log will contain many “failed to extend segment” errors.
- DBUA will leave compatible=9.2 after the upgrade to 10.1 ( So the max you can downgrade after 10g is 9.2 )
- The upgraded 8.1.7 database will still be using rollback segments.
- You must run the Character Set Scan utility prior to the upgrade if it involves changing character sets. ( This utility will check whether current character set is compatible with new character set)
  - Never use “Alter database ..... INTERNAL\_CONVERT “ for changing character set
- DBUA leaves the upgraded instance in NOARCHIVELOG mode.

## 9.2 Upgrade Notes

- DBUA will leave compatible=9.2 after the upgrade to 10.1 so that you can downgrade to 9.2 if necessary.
- Undo management wont change during upgrade.
- Use DBUA, not Export/Import, to upgrade an 9.2 database using Replication.
- [Some discussion on how to replicated in various replicated environments]
- DBUA leaves the upgraded instance in NOARCHIVELOG mode.

# Best Upgrade Practices (10g)

- Check the latest Support Status and Alerts for 10g (MetaLink article 263719.1)
- Follow the 6 major upgrade steps summarized in this presentation, presented in detail in the 10g Upgrade Guide.
- Upgrade to the latest patch set (10.1.0.3 expected in August)
- Use the Database Upgrade Assistant (DBUA) whenever possible.
- Use the Export/Import upgrade method to introduce new 10g features.
- Perform other upgrades (O/S, application, 3rd party products) separately from your database upgrade unless absolutely necessary.
- Oracle8, Oracle8*i*, Oracle9*i*, and Oracle Database 10g clients can connect to the upgraded database.



# Best Usage Practices (10g)

- Utilize Oracle 10g's performance diagnosis and monitoring technology. Refer to the following technical whitepapers included in 10gUpgradePackage.zip:
- The Self-Managing Database: Automatic Performance Diagnosis (TWP\_manage\_automatic\_performance\_diagnosis.pdf)
- The Self-Managing Database: Automatic Health Monitoring and Alerting (TWP\_auto\_health\_monitoring.pdf)
- The Self-Managing Database: Guided Application & SQL Tuning (TWP\_manage\_automatic\_SQL\_tuning.pdf)

# Recommended Resources

- Oracle 10g Database New Features Guide
- Oracle 10g Database Installation Guide for UNIX
- Oracle 10g Database Quick Installation Guide for Solaris
- Oracle 10g Database Administrator's Reference for Solaris
- Oracle 10g Database Release Notes for Solaris
- Oracle 10g Database Upgrade Guide
- Oracle 10g Overview and Additional Enhancements presentations
- 10g Upgrade Notes (by platform and Oracle version) series
- 10g System Requirements (by platform and Oracle version)
- 10g Tuning technical whitepapers

# Oracle Database 10g Upgrade Concepts

- Database upgrade is the process of transforming the internal Oracle database dictionary from one release to another
- Driven primarily through SQL and PL/SQL scripts
- Very careful ordering of execution and dependencies
- Must complete cleanly and correctly before the database can be used

# 10g Database Upgrade: Enhancements

- Reduced Complexity
- Component integration
- Consolidation of steps
- Single script to upgrade all components
- Enhanced GUI tool to assist the upgrade
- Database Upgrade Assistant (DBUA)

# 10g Database Upgrade: Enhancements

- Analysis of database before upgrade to increase likelihood of successful upgrade
- Pre upgrade information tool
- Upgrade Performance Enhancement
- Parallel recompilation of invalid pl/sql database objects on multiprocessor CPUs
- Verification of database upgrade
- Post upgrade status tool

# Reduced Complexity

- Automatically determines what components (e.g. Java, Spatial...) need to be upgraded and then performs all necessary upgrades in correct dependency order
- All components in DBMS\_REGISTRY

# Prepare to Upgrade

- Become familiar with the features of the new Oracle Database 10g release
- Determine the Upgrade Path to the new release
- Choose an upgrade Method
- Choose an Oracle Home Directory
- Prepare a Backup Strategy
- Develop a test plan

# Choose an Upgrade Method

- Database Upgrade Assistant (DBUA)
  - The Database Upgrade Assistant is a graphical user interface (GUI) tool that interactively steps the user through the upgrade process and configures the database to run with the new Oracle Database 10g release.
- Manual Upgrade
  - A manual upgrade consists of running SQL scripts and utilities from the command line



# Pre-Upgrade Analysis

- Checks for certain known database, init.ora, etc settings that may cause upgrade to fail, and generates warnings
- E.g. shared\_pool\_size init.ora parameter is too low
- Utility runs in “old server” & “old database” context
- Based on “posterity” database knowledge + needs of Oracle Database 10g
- DBUA automatically performs any required actions

# Gather Dictionary Statistics

- SQL>exec dbms\_stats.gather\_schema\_stats ('SYS',options=>'GATHER', estimate\_percent => DBMS\_STATS.AUTO\_SAMPLE\_SIZE, method\_opt => 'FOR ALL COLUMNS SIZE AUTO', cascade => TRUE);
- Repeat it for all the other users
- Save it in a file
- We later import them to new database

# Pre Upgrade Analysis

- Redo logs <4MB
- Updated Parameters e.g. shared\_pool\_size
- Deprecated (renamed) parameters
- Obsolete parameters
- Cluster check
- Account Check

# Pre Upgrade Analysis

- Names of Components in database e.g. JAVAVM, Spatial, etc
- Tablespaces
- Increase in size recommendations
- New SYSAUX tablespace
- Character set incompatibilities
- Installed options
- Upgrade time estimate

# Key DBUA Features

- The DBUA automates the upgrade process by performing all the tasks you normally accomplish manually
- Invokes the Pre Upgrade Information Utility and automatically makes appropriate adjustments (including tablespace, redo logs, etc)
- Checks for adequate resources (disk space, rollback segments)
- Creates mandatory tablespace SYSAUX
- Optionally, backs up all necessary files.
- Creates new configuration files (init.ora, spfile, listener) in the new Oracle home

# Key DBUA Features

- During the upgrade process, the DBUA:
- Runs all necessary scripts (pre-upgrade information utility, “u” script, post upgrade status utility)
- Shows upgrade progress
- Write detailed trace and logging files
- Can use multiple CPUs to recompile objects
- Creates restore script (db\_name.sh)

# Key DBUA Features

- Settings during upgrade
- DBUA disables archiving during upgrade phase
- Security features
- DBUA locks new users in the upgraded database
- RAC-DBUA
- RAC-DBUA facilitates upgrade of 8.1.7, 9.0.1, 9.2 to 10.1.0
- Upgrade done on the cluster. All database and configuration files are upgraded on all the nodes in the cluster.

# Upgrade Validation

- New “Post Upgrade Status Utility” – utlu101s.sql
- Queries component REGISTRY to determine upgrade status
- Provides information about invalid/incorrect component upgrades
- DBUA calls this script to display status and then takes corrective action



# Upgrade Validation

- Automatically handles setting certain system parameters that can otherwise cause problems during upgrade (E.g. set `job_queue_processes = 0`)
- `DBMS_REGISTRY` package and system table are used to record and manage information about all the component upgrades
- E.g. `SQL> select ... from dba_registry;`

# DBUA Related Bugs on Linux

- 3944180 ...does not check free space properly
- 3631276 ...dbua restart fails in silent mode
- 3334108 ...dbua runs forever
- 4123532 ...SGA more than 1.7 GB on x86\_64
- etc...

# Manual Upgrade

- Install Oracle Database 10g
- Analyze the existing database using pre upgrade information tool @rdbms/admin/utlu101i Make necessary adjustments based on feedback
- Shutdown database
- Switch to New Oracle Home

# Manual Upgrade

- Startup Database SQL> startup upgrade
- Create SYSAUX tablespace
- Invoke the proper upgrade “u” script SQL>  
@rdbms/admin/u0801070
- Shutdown database
  - Startup database
  - SQL> startup

# Manual Upgrade

- Run utlrp SQL>@rdbms/admin/utlrp
  - run utlrp at least 3 times .....
- Check upgrade status using post upgrade status tool SQL>@rdbms/admin/utlu101s

# Important Steps

- `_sys_trig_enabled=false`
- If `_SYSTEM_TRIG_ENABLED` is not set to `FALSE` then you may encounter [BUG: 1362374] which causes ORA-604 , ORA-6553 [BUG: 1747065] can give ORA-600 [16201] errors while opening a database
- `Aq_tm_process=0`
  - To avoid bug [BUG: 1421476]
- etc...

# u0902000.sql

- **STAGE 1:** call i0902000.sql:
  - This loads all tables that are necessary to perform basic DDL commands.
- **STAGE 2:** call utlip.sql
  - invalidate views, procedures, packages...
- **STAGE 3:** call c0902000.sql:
  - This performs all necessary dictionary upgrade to bring the db from 9.2.0 to the new release.
- **STAGE 4:** call a0902000.sql:
  - This performs all necessary upgrade using anonymous blocks.
- **STAGE 5:** call cmpdbmig.sql
  - This calls the upgrade scripts for all of the components that have been loaded into the database

# Changing Word Size

- Migrate/upgrade does it automatically
  - For within the same release need to use
  - Utlirp
- (provide workaround for [BUG: 2555795] before running utlirp)

*So what is the diff between utlrp and utlirp?*



# Utlip.sql

- This script must be run when it is necessary to regenerate the compiled code because an action was taken that caused the old code's format to be inconsistent with what it's supposed to be, e.g., when migrating a 32 bit database to a 64 bit database or vice-versa.
- STEPS:
  - Invalidate all stored PL/SQL units (procedures, functions packages, types, triggers, views).
  - Reload PL/SQL package STANDARD and package DBMS\_STANDARD.
  - update obj\$ set status = 6

(obj\$ status column has only 5 status and 6 means that all are invalid !!!)

# Utlrp.sql

- This script will validate all remaining invalid objects. It will also run a component validation procedure for each component in the database.
- These script must be run using SQL\*PLUS.
- You must be connected AS SYSDBA to run this script.
- There should be no other DDL on the database while running these script's. Not following this recommendation may lead to deadlocks.

# Some Discussion About Upgrade When Using HA SETUP ( Dataguard and RAC/OPS)

# Discussions

- About the Rolling upgrade and Orainventory
- About how to get information about one off patches on older version ( $\leq 8.1.7$ ) using “ar -x”
- What is OPatch ?

# Topics

- Parameters
- Processes
- SYSAUX Tablespace
- Listeners
- Clients & Net
- Export & Import
- Performance
- Lost & Found

# Parameters

- Classes of parameters
- Deprecated / obsolete parameters
- New parameters
- New defaults

# Parameters

- Parameters
- 28 basic
- 228 advanced
- \_ (underscore parameters)

(Select ksppinm from x\$ksppi )

- To get list all the parameters.

( Select view\_name from v\$fixed\_view\_definition)

- To get list of all the views GV\$ and V\$ views

# 28 Basic Parameters

- NLS\_LANGUAGE
- NLS\_TERRITORY
- OPEN\_CURSORS
- PGA\_AGGREGATE\_TARGET
- PROCESSES
- REMOTE\_LISTENER
- REMOTE\_LOGIN\_PASSWORDFILE
- ROLLBACK\_SEGMENTS
- SESSIONS
- SGA\_TARGET
- SHARED\_SERVERS
- STAR\_TRANSFORMATION\_ENABLED
- UNDO\_MANAGEMENT
- UNDO\_TABLESPACE
- CLUSTER\_DATABASE
- COMPATIBLE
- CONTROL\_FILES
- DB\_BLOCK\_SIZE
- DB\_CREATE\_FILE\_DEST
- DB\_CREATE\_ONLINE\_LOG\_DEST\_n
- DB\_DOMAIN
- DB\_NAME
- DB\_RECOVERY\_FILE\_DEST
- DB\_RECOVERY\_FILE\_DEST\_SIZE
- INSTANCE\_NUMBER
- JOB\_QUEUE\_PROCESSES
- LOG\_ARCHIVE\_DEST\_n
- LOG\_ARCHIVE\_DEST\_STATE\_n



# Deprecated init Parameters

- SQL> select name from v\$parameter where isdeprecated='TRUE';
- NAME
- -----
- buffer\_pool\_keep (replaced by DB\_KEEP\_CACHE\_SIZE)
- buffer\_pool\_recycle (replaced by DB\_RECYCLE\_CACHE\_SIZE)
- drs\_start (replaced by DG\_BROKER\_START - 9.2)
- fast\_start\_io\_target (replaced by FAST\_START\_MTTR\_TARGET - 9.0)
- global\_context\_pool\_size
- lock\_name\_space
- log\_archive\_start
- max\_enabled\_roles
- parallel\_automatic\_tuning
- parallel\_server (replaced by CLUSTER\_DATABASE - 9.0)
- parallel\_server\_instances (replaced by CLUSTER\_DATABASE\_INSTANCES)

# Obsolete init Parameters

- SQL> select name from v\$obsolete\_parameter;
- DBLINK\_ENCRYPT\_LOGIN
- HASH\_JOIN\_ENABLED
- LOG\_PARALLELISM
- MAX\_ROLLBACK\_SEGMENTS
- MTS\_CIRCUITS
- MTS\_DISPATCHERS
- MTS\_LISTENER\_ADDRESS
- MTS\_MAX\_DISPATCHERS
- MTS\_MAX\_SERVERS
- MTS\_MULTIPLE\_LISTENERS
- MTS\_SERVERS
- MTS\_SERVICE
- MTS\_SESSIONS

# Continued ...

- OPTIMIZER\_MAX\_PERMUTATIONS
- ORACLE\_TRACE\_COLLECTION\_NAME
- ORACLE\_TRACE\_COLLECTION\_PATH
- ORACLE\_TRACE\_COLLECTION\_SIZE
- ORACLE\_TRACE\_ENABLE
- ORACLE\_TRACE\_FACILITY\_NAME
- ORACLE\_TRACE\_FACILITY\_PATH
- PARTITION\_VIEW\_ENABLED
- PLSQL\_NATIVE\_C\_COMPILER
- PLSQL\_NATIVE\_LINKER
- PLSQL\_NATIVE\_MAKE\_FILE\_NAME
- PLSQL\_NATIVE\_MAKE\_UTILITY
- ROW\_LOCKING
- SERIALIZABLE
- TRANSACTION\_AUDITING
- UNDO\_SUPPRESS\_ERRORS

# New init Parameters

- asm\_diskgroups
- asm\_diskstring
- asm\_power\_limit
- db\_flashback\_retention\_target
- db\_recovery\_file\_dest
- db\_recovery\_file\_dest\_size
- db\_unique\_name
- ddl\_wait\_for\_locks
- fileio\_network\_adapters
- gcs\_server\_processes
- instance\_type
- ldap\_directory\_access
- log\_archive\_config
- log\_archive\_local\_first
- plsql\_code\_type
- plsql\_debug
- plsql\_optimize\_level
- plsql\_warnings
- resumable\_timeout
- sga\_target
- skip\_unusable\_indexes
- smtp\_out\_server
- sp\_name
- sqltune\_category
- streams\_pool\_size

# Continued...

- New / important defaults
- Parameters
- LOG\_ARCHIVE\_FORMAT: should include %s, %t, and %r
- PGA\_AGGREGATE\_TARGET:
  - not set: enabled with 20% of the SGA
  - 0: disabled
  - Value set: PGA=Value
- STATISTICS\_LEVEL = TYPICAL [Default]
- collects data and saves it to the AWR (see also: v\$statistics\_level)
- FAST\_START\_MTTR\_TARGET = 0 [Default]
- not set: enables automatic checkpointing
- 0: disables automatic checkpointing
- Value set: enables automatic checkpointing

# Parameter COMPATIBLE

- At least has to be 9.2.0 in a 10g database
- No way back once 10.0 has been set ( As compatible goes to redo log header)
- Supported downgrade only down to 9.2.0.3

# Processes

- New processes
- Obsolete processes
- Active processes

# Processes

- New
  - For Monitoring purposes
  - MMAN == Memory Manager ( SGA)
  - MMON == Manageability Monitor ( AWR)
- MMNL == Manageability Monitor Light ( AWR)
  - For Backup&Recovery and Flashback
  - CTWR == Change Tracking Writer ( CTF/RMAN)
  - RVWR == Recovery Writer ( Flashback Logs)



# Processes

- New
  - For Automatic Storage Manager
  - ASMB == ASM Background
  - RBAL == ASM Rebalance master
- ARBn == ASM Rebalance 0..n
  - For DataGuard
  - NSV == Data Guard Net Slave
  - INSV == Data Guard Instance-Slave

# Listeners

- New Porting
- Conflicts with Existing/Older Listeners
- Security of Listeners

# Listeners

- Oracle Listeners utilizing the default configuration file LISTENER.ORA will attempt to configure and service all instances.
- Oracle Listeners on the default port 1521 will automatically manage load and resource balance for you

( Check on new listener and tnsnames configuration method)

- Oracle recommends utilizing the latest version of the Listener, (from the latest installed Oracle Binary stack), for all instances of a certified version.
- If 2 Oracle XML-DB instances are running on the same server, the HTTP and FTP ports must be moved from default or they will collide.

Q & A