Last update: 25-Sep-2015 [Draft]

Reduce planned database down time with Oracle technology smart

Kirill Loifman
adidas Group AG
Herzogenaurach

Keywords:

Oracle, database, downtime, 12c, online, high availability, maintenance, SQL

Introduction

How to design an Oracle database system to minimize planned interruptions? That depends on the requirements, goals, SLAs etc. The presentation will follow top-down approach. First we will describe major types of planned maintenance, prioritize those and then based on the system availability requirements find the best cost-effective technics to address those. A bit of planning, strategy and of course modern database and OS technics including latest Oracle 12c features.

Downtime vs Availability

The key to determining which elements of high availability are appropriate for your site is how much down time you can tolerate; this includes unexpected as well as planned down time.
This type of metric expressed usually as System Availability in percentage, but is often more meaningful when expressed as hours. For example, 99.9% availability is roughly equivalent to 8 hours and 45 minutes of down time per year for you system.

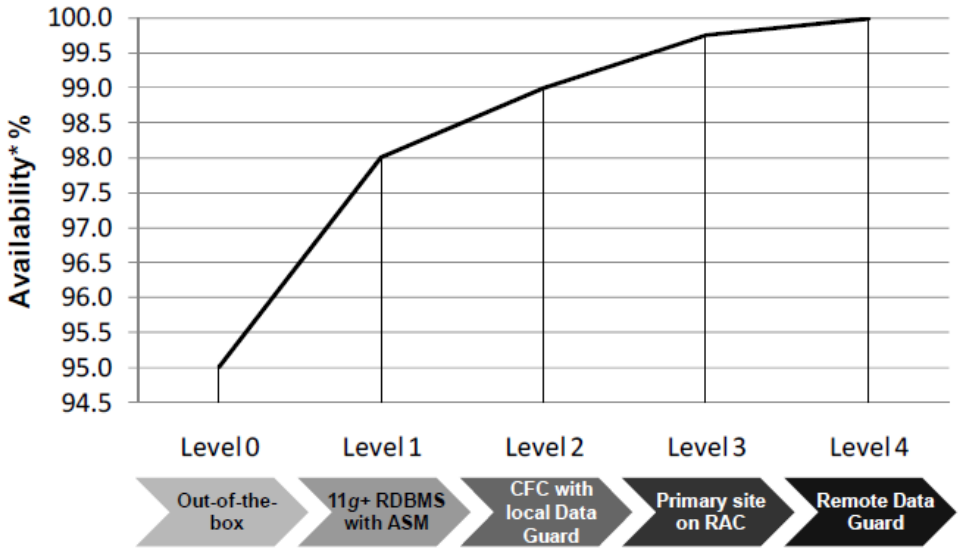

Illustration 1: Availability Levels graph

The graph depicts an example of the increases in availability that can be gained with the progression between levels of availability. It is not based on empirical data, and the percentage values used are for illustrative purposes only, matching the industry standards though.

Another way to interpret the y-axis scale is as a measure of acceptable down time—the lower end of the axis represents a reasonable amount of down time as tolerable, whereas the upper end of the axis represents even the smallest amount of down time as being intolerable.
Yet another way to think of the y-axis is in terms of costs. If the demanded availability is higher and if down time becomes accordingly less acceptable, there will be higher costs in achieving the desired level of availability.

Causes of Planned Down Time

One of the true challenges in designing a highly available solution is examining and addressing all the possible causes of down time. It is important to consider causes of both unplanned and planned down time. In this presentation we review only planned downtime.

Planned down time can be disruptive to operations, especially in global enterprises, that support users in multiple time zones, up to 24 hours per day. In these cases, it is important to design a system to minimize planned interruptions. As shown by the diagram in the slide, causes of planned down time include routine operations, periodic maintenance, and new deployments. Routine operations are frequent maintenance tasks that include backups, performance management, user and security management, and batch operations. Periodic maintenance, such as installing a patch or reconfiguring the system, is occasionally necessary to update the database, application, operating system middleware, or network. New deployments describe major upgrades to the hardware, operating system, database, application, middleware, or network. It is important to consider not only the time to perform the upgrade, but also the effect the changes may have on the overall application.

Categorize and Plan Maintenance

Let's clearly categorize planned maintenance and see how we can minimize it with newest technologies and minimum costs.

Maintenance Type	**Examples**
Hardware maintenance	Hardware upgrade Server/HBA Firmware patching
OS Maintenance	OS Patching / Upgrade
Database/Instance maintenance	DB files/parameters changes, backup, etc.
Database software maintenance	DB/GRID patching/upgrade
Database Schema maintenance	Objects/Segment maintenance, etc.
Application Maintenance	Application upgrade, etc.

Due to the fact that both planned and unplanned downtime affects the required availability SLA we need to minimize both to ensure we can reach them. How to minimize? Think in two dimensions:

1) Plan accordingly outages in every maintenance category so that to achieve adequate frequency

2) Minimize duration of every maintenance occurrence

Minimize planned maintenance on standalone systems

Let's start from the Availability Level 1 (Storage-Level Protection) that represents a single server and single database instance setup with the Oracle database version 11gR2 with protected storage (LVM/ASM). This provides an element of protection against a storage-level failure, but there is no redundancy for server components.

Below are some best practices and several Oracle 12c features that help minimize planned downtime on standalone database server configurations. Some of them will be described in detail.

Online System Reconfiguration
Online OEL Kernel Patching (OEL6+)
Online database patching (11g+) and patching best practices
Online database parameter changes
Online database file move/compression (12c+)
Online partition move/compression (12c+)
Online Table Redefinition including 12c enhancements
Online Statistics Gathering during Bulk-Loads (12+)
Online Application Upgrades with Edition-Based Redefinition
Online / Active Database Duplication (11g+) and 12c enhancements

Standalone Database patching best practices

To minimize downtime during patching of NON-RAC databases *Out of Place Patching* can be used. In this case time spent installing the software can be saved/reduced from the total database downtime required. However, some downtime is required for switching database services to the new Oracle database home. A basic overview of the steps include:

- Clone the existing database home online
- Apply required patch set to the cloned database home using OPatch
- Switch the database services to the cloned database home
- Complete the post installation tasks for the patch set applied

More details in MOS Note: 1389364.1

Online database patching (11g+)

- Available only for certain interim one-off patches and bug fixes
- OPatch utility to use
- Oracle 12c -> no improvements found

Online database file move/compression (12c+)

Oracle 12c Online Move data file feature provides the capability to move an online data file to another location, another storage system or even another compression level while the database is open and accessing the file. Queries, DML and DDL operations can be performed while the data file is being moved.

- Progress in `V$SESSION_LONGOPS`
- SQL Examples:

```
ALTER DATABASE MOVE datafile '/disk1/myexample01.dbf'
TO '/disk2/myexample01.dbf';

ALTER DATABASE MOVE datafile 5 TO '+DiskGroup3';
```

- NOT compatible with concurrent:
  - Data file OFFLINE
  - Media recovery
  - `data file RESIZE` (shrink) operation
  - `FLASHBACK DATABASE` operation

Online partition move/compression (12c+)

12c Online Partition maintenance enhancements provide the capability to move / split / merge / compress table partitions or sub-partitions online without preventing concurrent DML operations.

- Note:
  - DML allowed, not DDL
  - Global and local indexes maintained

- SQL examples:

```
ALTER TABLE ORDERS
MOVE PARTITION ORD_P1
TABLESPACE lowtbs
UPDATE INDEXES ONLINE;
```

```
ALTER TABLE ORDERS
MOVE PARTITION ORD_P1
ROW STORE COMPRESS
UPDATE INDEXES ONLINE;
```

Multiple Indexes on the Same Set of Columns (12c+)

- Different index types: unique vs non-unique; Local vs Global
- Only one of the indexes visible at a time.

Invisible Table Columns (12c+)

The property of whether a column is visible can be controlled by the user. Invisible columns are not seen unless specified explicitly.

Online Table Redefinition

Following enhancements introduced in Oracle 12c:

- Support for redefinition of multiple partitions in a single redefinition session.
- Support for tables with VPD policies
- New parameter dml_lock_timeout to handle pending DML in FINISH_REDEF_TABLE procedure
- New procedure REDEF_TABLE allows a one-step operation to redefine a table or partition
- Improved performance of sync_interim_table and resilence of finish_redef_table

More online DDL (12c+)

Several schema maintenance DDL operations no longer require blocking locks in 12c.
Following operations do not block end user DML statements (restrictions apply):

- `DROP INDEX ONLINE`
- `DROP CONSTRAINT ONLINE`
- `ALTER INDEX UNUSABLE ONLINE`
- `SET COLUMN UNUSED ONLINE`
- `ALTER INDEX [VISIBLE | INVISIBLE]`

Online System Reconfiguration

- Online addition or removal of disks (ASM/LVM)
- Online addition or removal of clustered nodes (RAC/GRID)
- Online addition or removal of CPUs on SMP servers that have this online capability
- Online grow and shrink of shared memory and database cache.

Active Database Duplication

Following Oracle 12c enhancements introduced to Active database Duplication feature of RMAN:

- Default use of backup sets for active duplication (less blocks to read -> faster)
- Choice of compression, section size, and encryption (again faster)
- Option to end database duplication with database in mounted, but not opened state (useful for upgrades)
- Duplication of pluggable databases

Online Statistics Gathering during Bulk-Loads

In Oracle Database 12c, the database automatically gathers table statistics during the following types of bulk loads:
– `CREATE TABLE AS SELECT`
– `INSERT INTO ... SELECT` into an empty table by using a direct path insert.

• Statistics are available immediately after load.
• No additional table scan is required to gather statistics.

Metadata-Only DEFAULT Column Values for NULL Columns

The default values of columns are maintained in the data dictionary for columns specified as NULL. Adding new columns with DEFAULT values no longer requires the default value to be stored in all existing records.

How Oracle Multitenant can reduce planned downtime

The Oracle 12c Multitenant architecture can help in certain areas to improve planned downtime. Some of them below:

- Fast database provisioning
    - Pluggable PDB creation from PDB$SEED, by plug non-CDB in as PDB;
    - Clone PDB from another PDB into the same or another CDB
    - Plug an unplugged PDB into a same of another CDB; even to higher 12c version

- Time saving patching, upgrading and migrations
    - Upgrading CDB with its PDBs in one go (upgrade time increases with more PDBs though)
    - Unplug a PDB from one Oracle release, and then plug it into a CDB from a higher release.
    - Migrate a 11.2.0.3 non-CDB into new 12.1 CDB as PDB using full transportable Exp/Imp

Minimize planned maintenance on clustered systems

Starting with System Availability Level 2 a database system must introduce redundant components that are usually based on Cluster configurations, Cold and Hot database failover and data replications technics. These technologies usually require considerably more investments and maintenance efforts. The minimum downtime from Oracle perspective you can achieve building on so-called Oracle Maximum Availability Architecture.

Rolling Upgrades in Multi-node Configurations

The big advantage of database multi-node configurations is that you can do maintenance on one cluster node while the others are running. This type of cluster configuration can reduce planned downtime considerably and can be used for patching database with Patch Set Updates (PSUs), Critical Patch Updates (CPUs), one-off database and diagnostic patches using OPATCH; operating system upgrades, and hardware upgrades.
In spite Oracle refers in this respect usually to *Rolling Patch Upgrades using Oracle RAC* other Cold Failover Cluster configurations can be used to achieve similar results.

Data Guard Standby-First Patch Assurance (11gR2)

Data Guard Standby-First Patch Assurance (Oracle Database 11.2.0.1 onward) enables physical standby to support different software patch levels between a primary and standby database for the purpose of applying and validating Oracle patches in rolling fashion.

Database Rolling Upgrades using Active Data Guard (12c+)

The transient logical database rolling upgrade process uses a Data Guard physical standby database to install a complete Oracle Database patch set (Oracle 11.2.0.1 to 11.2.0.3), or major release (Oracle 11.2 to 12.1), or perform other types of maintenance that change the logical structure of a database. Oracle Database 12c, improves the former upgrade procedure by replacing forty-plus manual steps required to perform a rolling database upgrade with three PL/SQL packages that automate much of the process.

Summary

At the end after evaluating and testing the features and technics above, you need to apply those that are appropriate for your particular database configuration based on the availability requirements.

Contact address:

Kirill Loifman
Olympia Ring 100
91074, Herzogenaurach

Phone:	+49(0)160 884 2541
Email	dadbm1@gmail.com
Internet:	www.dadbm.com
	www.twitter.com/loifmkir
	www.google.com/+dadbm
	www.facebook.com/pages/DaDBm/425726914237370
	www.youtube.com/dadbman