HUNKLER
GmbH & Co. KG

ORACLE WHITEPAPER
SEPTEMBER 2020

# WHY NOT ORACLE STANDARD EDITION 2?

Dbvisit
DISASTER RECOVERY SOFTWARE

# Contents

# Executive Summary

Why not look at Oracle Standard Edition 2? Sometimes we tend to just forget about it and we focus only on the top of the range products with all the latest and greatest features. Yet, on completion of a new solution implementation, you come to realize that you did not use any of these talked about features.

The objective of this paper is to assist you in:

- Gaining an understanding of some of the core features available in Oracle Database Standard Edition 2 and how these can be used to provide you with a robust environment at a lower cost.
- Learn the new rules around Oracle Standard Edition 2.
- Learn key points for implementing a high availability solution using Oracle Database Standard Edition 2.

# Target Audience

Whether you are a director, manager or just the junior database administrator in the team; understanding the capabilities of Oracle Database Standard Edition 2 is valuable and can assist you in making the correct decisions when implementing, designing or budgeting for new solutions.

# Introduction

Oracle Database Standard Edition 2 is sometimes the neglected younger brother of Enterprise Edition. Most will focus on the new features available in Enterprise Edition, but what about Standard Edition 2? People seem to forget that it is still the same database at the core. Oracle Standard Edition 2 can provide you with highly available solutions as well as help you establish a robust Disaster Recovery (DR) environment. One huge point is often overlooked; it is even possible to use Oracle Real Application Clusters (RAC) with Standard Edition up to Oracle 18c. From Oracle 19c Oracle RAC is no longer included with the Oracle SE2 code set - this is unfortunate, but there is still a good alternative option available to you - combine the Oracle SE2 Database with Oracle Linux and Oracle Clusterware - which is free with an Oracle Linux support agreement. Together you can create an Active/Passive cluster that can assist you in achieving higher availability. To top this all off, you gain the added advantage of a lower licensing cost. Whether you are a small business or enterprise, Standard Edition 2 may well suit your needs and as the cost saving can be substantial it is worth considering.

This white paper will take you on a journey into some of the options available with Oracle Standard Edition 2 - options which can provide you with a highly available solution with disaster recovery in place.

# Oracle Database Editions

Before we focus on some of the key Standard Edition 2 features, it is good to get an understanding of the key differences between the different Oracle database editions. Oracle provides four database editions. These are:

- Oracle Database Express Edition (XE)
- Oracle Database Personal Edition
- Oracle Database Standard Edition 2 (SE2) *
- Oracle Database Enterprise Edition (EE)

** Oracle Database Standard Edition One (SE1) and Oracle Database Standard Edition (SE) are no longer offered and were 'replaced' with Oracle Database Standard Edition 2 (SE2) in December 2015.*

If we look at these editions, we tend to classify them by company size. Enterprise Edition is seen as the big brother for all the enterprise companies and large systems out there. In some sense this is true, but there is no size limit on the Standard Edition 2.

The reason for its ‘classification’ as a database for smaller companies is not just its size; the feature limitations and most of all the licensing costs play a role in our perception that Standard Edition 2 is only for smaller companies or small databases.

What are these feature limitations and are they really limitations? To understand this better, it is recommended that you do a detailed review of the database editions available to make sure the features your application or business require are accessible.

In this paper we will not focus on Oracle Database Express Edition or Personal Edition, but we will provide a summary of these two at this point:

- **Express Edition**

The Express Edition is seen as the entry-level database edition, but that does not mean it is not worth looking at. It is based on the same core technology used for Enterprise Edition and the database we all came to love. It is free to download and use, quick, and easy to install and maintain. There is no limitation on the hardware it can be installed on, but it is limited to only use one CPU, 1GB of Memory and 11GB of user data. The size limitation of the database is key, but taking into account the technology at your fingertips, this is truly a powerful little database that can be used for small applications not needing a lot of storage. It is ideal for web based applications.

- **Personal Edition**

The Personal Edition is probably one of the least used editions. This is a powerful edition that includes all the features and options that are available in the Enterprise Edition, except the Real Application Clusters (RAC) option and the management packs are not included. The first key restriction on this edition is it is only available for a Single-User license, such as development or deployment environments that require similar features as what is available in Enterprise Edition. The second limitation is that it is only available on Windows and Linux platforms.

- **Standard Edition 2**

Standard Edition 2 can be licensed and used on servers with a maximum capacity of 2 CPU sockets. There is no limit to the number of cores. Oracle Database Standard Edition 2 automatically limits the maximum number of CPU threads to 16 at any one time. One of the most important factors of Standard Edition 2 is that it comes bundled with the Oracle Real Application Cluster (RAC) option from 10g up to 18c. This option provides the ability to implement high availability solutions using Oracle’s proven RAC technology at no additional charge. However be aware that when running RAC with SE2 (only available till 18c as Oracle RAC was removed from the 19c code set) you will be limited to 1 socket system as the maximum is 2 CPU sockets in total in the cluster. This is why looking at using Oracle 19c Database with Oracle Linux and Oracle Clusterware (if you have an Oracle Linux support agreement remember that Oracle Clusterware is then free to be used) to create a Active/Passive cluster is also a good alternative. You still get the full use of the 16 active threads as well as the option to fail-over to the second node in the cluster (This would normally be for a limited time up to 10 days per year, unless you licence both nodes).

- **Enterprise Edition**

Enterprise Edition is Oracle’s flagship database edition without CPU limitations and includes features that provide performance, availability, scalability and security. It should be noted that some of the optional features require additional licenses (at additional costs). This includes but is not limited to:

- Oracle RAC
- Partitioning
- Performance Tuning Pack
- Diagnostics Pack

# Key Differences Between Database Editions

Some of the key differences between the two Oracle database editions we are focusing on are highlighted in the table below. Note that this is only an extract of some key differences of which some will be discussed in more detail in this document.

| Database Features [1] | Standard Edition 2 | Enterprise Edition |
|---|---|---|
| Maximum CPUs | 2 CPU Sockets | No Limit |
| Oracle Real Application Clusters (RAC) - 10g up to 18c.<br>*Using an Active/Passive cluster with Oracle Linux and Oracle Clusterware with 19c is still an option | ✓<br>(Included with SE up to Max. Total of 2 Sockets) | ✓<br>(Extra License Cost Option) |
| Oracle Data Guard | ✗<br>(3rd party options available) | ✓<br>(Active Data Guard requires additional license) |
| Flashback Data Archive | ✓ | ✓<br>(Optimization Requires EE and Advanced Compression) |
| Flashback (Table, Database, Transaction) | ✗ | ✓ |
| Parallel Options (Parallel Query, Stats Gathering, Index Building and Parallel Data Pump) | ✗ | ✓ |
| Automatic Workload Repository | ✗<br>(Statspack and 3rd party options available) | ✓<br>(Extra License Cost Option) |
| Recovery Manager (RMAN) | ✓ | ✓ |

*[1]The table provides a summary of database features; a full detail listing can be found in the Oracle online documentation. From 19c Oracle RAC is not available with Oracle SE2 - however if you have an Oracle Linux support contract in place you can use Oracle Clusterware for free and create an Active / Passive cluster on which you can run 19c at one node with the option to fail-over to the second node.*

Reviewing the table above, it is clear that the big brother of them all is Enterprise Edition with all options available, although with some being available at an additional cost. Some key RMAN options not available in SE2 are parallel backups, fast incremental backups with Block Change Tracking and block-level media recovery. But it is important to know that even though Standard Edition 2 does not have all the features available in Enterprise Edition, it is still capable of providing a high availability solution for disaster recovery.

Even though Standard Edition 2 does not include Data Guard, it is still capable of providing standby database functionality with a third party tool.

The next section will focus on the different licensing options available and the possible cost savings when using Standard Edition 2.

# License Cost Comparison

Now that we have a bit of an understanding on the differences between the versions, it is always good to get an understanding of the licensing cost. To illustrate this, two examples will be used. One example will show you the licensing details for a single instance (single server) environment with two quad core Intel Xeon CPU sockets. The second scenario will look at an Oracle RAC implementation using two nodes and to add a little more complexity to it, we will add a single instance standby environment. But before doing this comparison, let's review the Oracle Licensing structure.

Oracle Database Licensing makes use of two metrics:

- **Named User Plus**

When looking at Named User Plus (NUP), the key is you can identify and count the users that will be making use of the database. One of the key factors to take into account is that these include both human and non-human devices. All of these counted users should be licensed. For example, if you have 20 users that can use the database either via their own PC or just a shared device, all 20 need to be licensed. If you have non-human operated devices, they will require a license as well. However, it should be noted that a licensed user (NUP) could access multiple instances with this license as long as the "minimums" are met for each server.

- **Processor**

If you cannot identify and count your users, such as Internet facing environments, then you need to look at the Processor License metric. All processors in the server where Oracle is running should be licensed. The Processor License can logically be split into two sections:

- **Processor License Standard Edition 2**

With Standard Edition 2, the licensing is socket based. Standard Edition 2 does not take into account the processor cores, but only the physical CPU sockets. With Standard Edition 2, you are limited to a maximum of 2 CPU sockets in total. Note: if multi-chip modules are used, each chip counts as a socket.

- **Processor License Enterprise Edition**

With Enterprise Edition the processor license is a little bit more complex. The first thing you need to be aware of is that it is taking CPU Cores into account. Oracle is using a "Processor Core Factor Table", see Appendix A, in which they assign a value to each specific CPU. This value should then be multiplied by the total number of cores on the CPU socket to identify the required Processor License. For example, an Intel E5-2609 Xeon CPU with 4 Cores has a factor of 0.5 used. This means that you will require 4 x 0.5 = 2 Oracle Processor Licenses.

- **Minimums**

It is important to take note of the minimums. When purchasing licensing you need to take into account the minimum number of licenses you will need to acquire. These minimums are especially important when buying Named User Plus licensing.

- **Standard Edition 2**

The minimum for Standard Edition 2 is 10 Named User Plus (NUP) licenses per server.

- **Enterprise Edition**

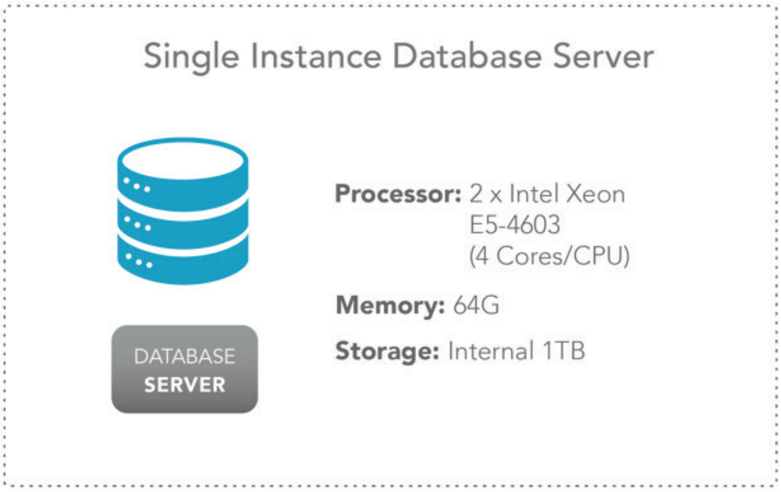
For Enterprise Edition, the minimum is 25 Named User Plus licenses per installed Processor where installed processor is equal to the value of processors being calculated taking into account the core factor. For example, if you have a 4 Core Processor with a 0.5 factor, you require 2 (4*0.5) Oracle Processor licenses. Taking the minimum of 25 Named User Plus into account, you will require 50 (2*25) Named User Plus licenses.

The next section will provide you with a license example to illustrate the lower cost of Standard Edition 2.

# Example A: Oracle Licensing Using Single Instance Database Environment

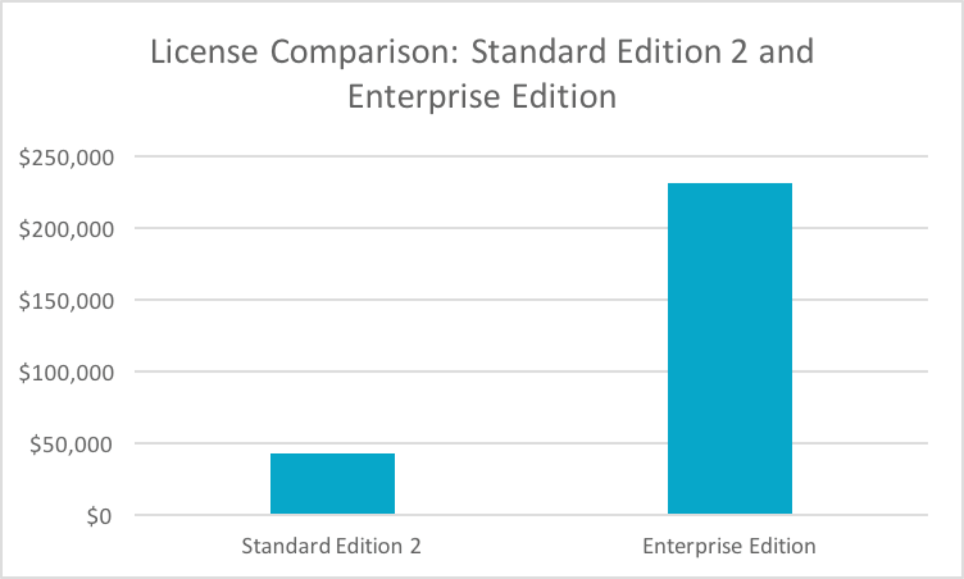
In the first example, we will be taking a single server with 2 CPU sockets using the Intel Xeon CPU with 4 cores. For licensing purposes, memory and storage are not taken into account. It is also assumed the environment will be web facing and processor licensing will be required as there is no way to specify or identify the exact users.

Note that for the Intel Xeon E5-4603 Quad Core CPU the processor factor is 0.5 when using Enterprise Edition licensing. Pricing below is in US$ and only includes the first year of support costs.


Single Instance Database Server
DATABASE SERVER
Processor: 2 x Intel Xeon E5-4603 (4 Cores/CPU)
Memory: 64G
Storage: Internal 1TB

| Standard Edition 2 | Enterprise Edition |
|---|---|
| 2 Socket License - (2 Sockets) | 4 Processor License - ((2x4)x0.5)=4 |
| $ 17,500 x 2 = $ 35,000 | $ 47,500 x 4 = $ 190,000 |
| First Year Support: $ 3,850 x 2 = $ 7,700 | First Year Support: $ 10,450 x 4 = $ 41,800 |
| Total Cost: $ 42,700 | Total Cost: $ 231,800 |

*Note: Pricing used was extracted from https://shop.oracle.com as at April 2020. Note that pricing does not include any discounts.*



# Example A Summary

As you can see there is a substantial difference in upfront costs as well as ongoing support between the different editions. It is important to state that the pricing above does not include any special discounts and that pricing is purely taken as it is stated on the Oracle website.

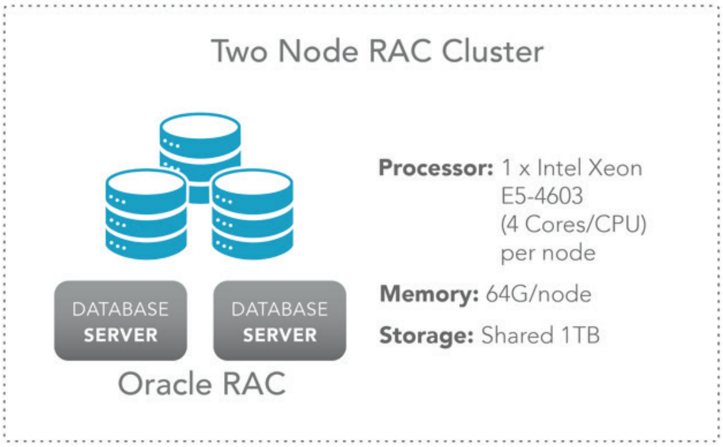
# Example B: Oracle Licensing Using Oracle RAC

In this test case, we will be looking at two scenarios. The 1st is using Oracle Enterprise Edition for the RAC cluster.

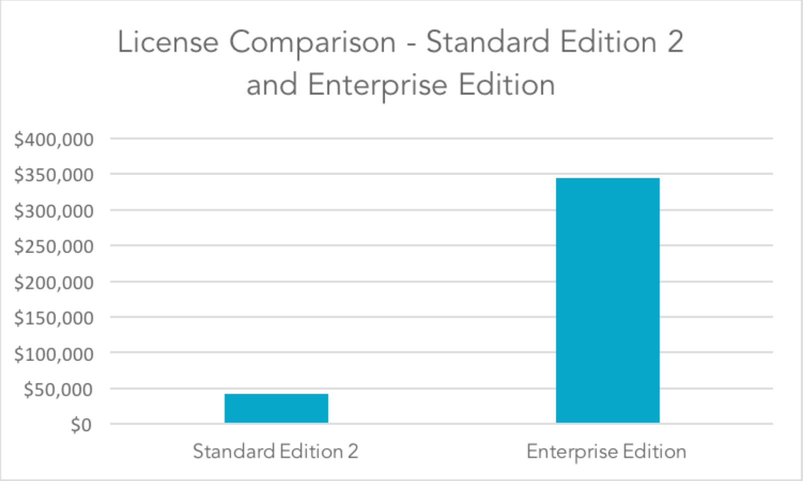
The 2nd option is implementing Standard Edition 2 for the RAC cluster - as mentioned before, Oracle RAC is only available with Oracle SE2 up to 18c.

If you were still looking to achieve higher availability on the primary side with Oracle 19c and above, you can make use of an Active / Passive cluster. This can be done if you have an Oracle Linux support agreement in place by installing a cluster with Oracle Linux and Clusterware, then configuring shared storage and installing Oracle 19c SE2 database on the shared storage. Adding the database as a cluster resource, you can then fail-over between the cluster nodes. Keep in mind though if you only license one of the cluster nodes for the database, you will not be able to switch to the second node more than 10 days in a year. For more detail on these requirements, please discuss it with your Oracle Account Manager. In summary, if you are using 19c with Clusterware your cost comparison will be similar to Example A above as no Oracle RAC will be used and Clusterware is free with an Oracle Linux support agreement in place.

Back to the Oracle RAC example. The environment consists of a primary site running an Oracle RAC cluster with two nodes. Each node contains 1 x Intel Xeon E5-4603 processor with 4 cores each. Note that pricing is in US$ and only includes the first year of support.



| Standard Edition 2 | Enterprise Edition |
|---|---|
| 2 Socket License - (2 Sockets)<br>$ 17,500 x 2 = $ 35,000<br>First Year Support: $ 3,850 x 2 = $ 7,700<br>Oracle Real Application Cluster Licensing: None<br>Total Cost: $ 42,700 | 4 Processor License - ((2x4)x0.5)=4<br>Oracle Database Licensing: $ 47,500 x 4 = $ 190,000<br>First Year Support: $ 10,450 x 4 = $ 41,800<br>Total Cost: $ 231,800<br>Oracle Real Application Cluster Licensing: $ 23,000 x 4 = $ 92,000<br>Oracle RAC First Year Support: $ 5,060 x 4 = $ 20,240<br>Total Cost: $ 344,040 |



# Example B Summary

Similar to Example A, in this example we can clearly see a big difference in the cost of using Standard Edition 2 vs. Enterprise Edition. Key to this cost difference is the fact that with Standard Edition 2 up to 18c, Oracle RAC is free of charge.

# Oracle Standard Edition 2 and High Availability

When thinking about high availability, there are a few options that come to mind. Some of the most common ones are listed below. But how do they fit in with Standard Edition 2?

- **Oracle Real Application Clusters (RAC)**

  Oracle RAC is available for use with Standard Edition 2 and it is 100% free since the release of 10g up to 18c, with the only limitation being that the cluster may only have a maximum of 2 sockets in total with only 16 active threads.

  Note: As mentioned earlier, from Oracle 19c Oracle RAC is no longer included in the SE2 code set. However, if you are looking to achieve higher availability - you can look at the option to use the Oracle 19c SE2 Database with Oracle Linux and Oracle Clusterware (which is free if you have an Oracle Linux support agreement). Using these in an Active/Passive configuration you can still end up with a great solution to meet your HA needs.

  This document will however still cover Oracle RAC, due to it still being available in 18c and Oracle RAC in Standard Edition and Standard Edition 2 is still in use by many.

  With Oracle RAC in a Standard Edition 2 environment, there are some strict requirements as well as valuable features:

- Oracle Advanced Security SSL/TLS is included.
- Maximum of 1 CPU socket per node
- Automatic Storage Management (ASM) must be used for all Oracle database file types.
- 3rd Party clusterware is not supported, you have to use Oracle Clusterware.
- No other clusterware software can be installed on the same system running the Oracle RAC configuration, this list includes OCFS, OCFS2 and any 3rd party cluster file systems.
- Maximum of 16 CPU threads running at any one time.

- **Standby Databases**

Another critical component in achieving high availability is standby databases, and YES this option can be implemented on Standard Edition 2, even when using Oracle Standard Edition 2 with Oracle RAC or if you have configured an Active/Passive cluster with Oracle Database, Oracle Linux and Oracle Clusterware. Creating and maintaining a standby database is done using 3rd party products or even custom scripts - if you have the time and knowledge to write them.

- **Backup and Recovery (Recovery Manager – RMAN)**

Oracle Recovery Manager (RMAN) can perform backup and recovery tasks in all the database editions, even in Oracle Express; but there are limitations when working with Standard Edition 2. Two of the key limitations are that parallel backup options and block-level recovery are not available. However, the core functionality of RMAN is fully available in all editions and makes this one of the most important components of implementing a highly available solution.

- **Flashback Technology (Flashback Database, Table, Transaction and Query)**

If we look at the flashback options, unfortunately most of the flashback options such as Flashback Database, Table and Transaction are only available in Enterprise Edition. But one of them, Flashback Query is available in Standard Edition 2, and even though it is not as powerful as the other Flashback options, it is still worth knowing it is there (it might come in quite handy one day). See "Appendix B: Flashback Query" for a quick summary of Flashback Query. When Oracle released 12c it was also noted that Flashback Data Archive (FDA) is also now part of Standard Edition 2. There are restrictions where optimization requires EE and for the Advanced Compression option. For more detail on this, please see the Oracle 12c online documentation.

Before we get into an example, it is good to have an understanding of what Oracle Standard Edition 2's capability with regards to Disaster Recovery is.

# Oracle Standard Edition 2 and Disaster Recovery

Three disasters that always get mentioned when talking about Disaster Recovery (DR) are:

- Natural Disasters
- Infrastructure (Hardware) Failure
- Human Error

To counteract these disasters, we implement processes, policies and procedures to provide comfort to the organization so that if a disaster struck, infrastructure, applications and databases could continue to function with limited downtime or could be recovered within agreed time frames.

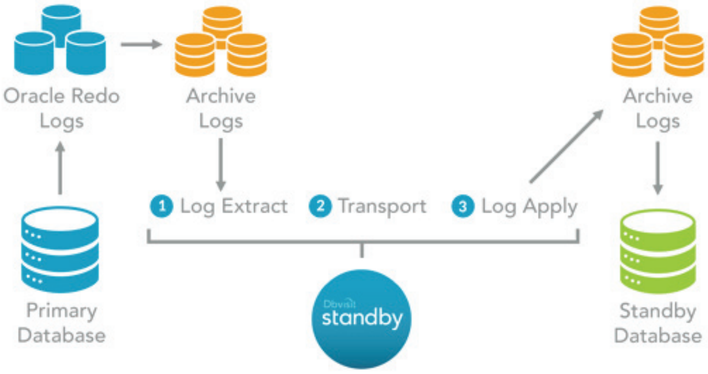
In most cases, this includes having a second datacenter in a different geographical location. Data is replicated using one of many methods between your primary datacenter to your "backup" secondary site - your DR site. All of this; the setup and configuration of the new infrastructure, as well as establishing the processes, policies and procedures are critical to ensure business continuity. This can be seen as the core reason behind Disaster Recovery processes.

What can Standard Edition 2 provide? Does it even cater for Disaster Recovery? The quick and easy answer to this question is YES. Standard Edition 2 does have Disaster Recovery solutions but one key option stands out; standby databases. Most don't realize that even though Data Guard is only available to Enterprise Edition, Standard Edition 2 can still provide you with standby database functionality. The main issue with this is that you have to build some of the components not available to achieve this.

Example processes that will need to be implemented manually include (but are not limited to):

- The creation of the standby database (clone of the primary database)
- The creation of the standby control file
- The automatic shipping of redo information from the primary database to the standby database
- The starting of recovery sessions to apply this redo to the standby database.
- Management of the archive logs after they were applied
- Performing role-switch operations where the primary and standby can switch roles
- Activating (failing over to) the standby database in case of disaster

Below is an image that provides you with a high level overview of a standby database.



In summary, the 3 key steps required for implementing and maintaining a standby database are:

- **Archive Log Extraction**

  Archive logs are key to database recovery and are also used to move redo information from your primary database to your standby database. The first step is to extract the archive logs from the primary environment as they could possibly be located in an ASM disk group.

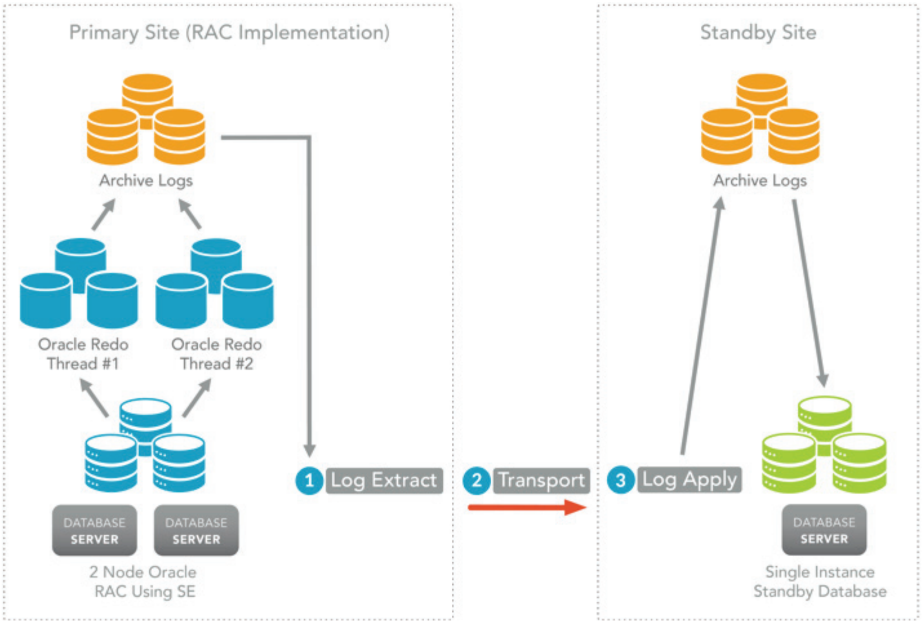
- **The Redo Transport**

  The second step is the transferring of the archive log between the primary and the standby server. Using Oracle Net for this is not available. Archive logs can be transferred using other network copy operations.

- **Redo (Archive) Apply**

  The last step in the process is applying the archive logs to the standby database. This process brings the standby database up to date with the primary. The key in this process is not to lose any archive logs as you would end up with an unrecoverable gap between the primary and the standby database.

Even if you are running an Oracle Standard Edition 2 environment with Oracle RAC, or using an Active / Passive cluster on the primary side, having a standby database is possible. The principles stay the same. The archive logs from the Oracle RAC database are shipped to the standby server, which will then apply the archive logs (redo) from both threads to the standby database. Below is a high-level demonstration of this:



The next section is an overview of how you can achieve disaster recovery in a cloud environment using a standby database, to provide you with a solution that not only ensures disaster recovery, but also higher availability.

# What About DR in the Cloud?

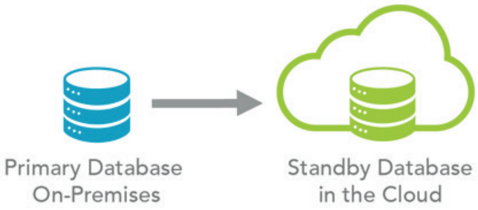
This is an area that is talked about more and more. “Cloud” is not just a buzzword anymore but it is a reality.  More and more companies are investigating and adopting the use of cloud computing for disaster recovery.  With the focus on Oracle databases, there are a variety of options available when looking at DR in the cloud with three key options that stand out, that is hybrid cloud, cloud to cloud, and inter cloud.

1. **Hybrid Cloud Solution**

This is probably the most popular way to go, where companies are using their own internal systems hosted in their own data centers as the production systems, and then they have an exact copy (DR environment) sitting in the cloud – using one of the many providers out there. This allows them the flexibility and the cost savings of not needing a second dedicated datacenter. Yes, there will be costs involved for the cloud-based solution, but in most cases they will be substantially less than establishing and maintaining your own datacenter.

This method allows a company to quickly and easily establish a DR site for their internal environment. If disaster strikes and the local copy of the environment is not available, the end users are directed to the copy in the cloud.

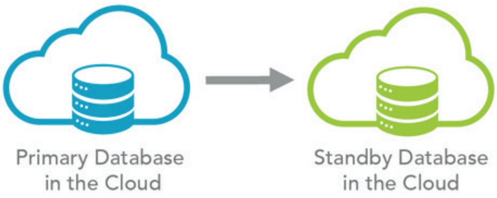
This option is fully supported with Oracle Standard Edition 2. When using Standard Edition 2, Dbvisit Standby can easily be configured and used to help you establish and maintain this type of configuration. It is also possible for the primary system to be located in the cloud and for the DR site to be located on-premises.



**2. Full Cloud-based Solution**

When using a full cloud-based solution, the idea is that both the primary and the DR site are located in the cloud. In this case, there would have been a "migration to the cloud" process if the environment were located on-premises, or it might be that the production environment was initially established in a cloud based environment. The DR site, or standby database for the primary database, is also created in the cloud. As most cloud providers now have datacenters in different geographical regions with high-speed networking to interconnect them – this can easily be established.

These options might not be as popular (at the current time) as the hybrid solution, but for some this is a good option and can result in substantial cost savings and of course, ease of management. Using Standard Edition 2 in these types of configurations is possible and, as with the hybrid solution, can easily be implemented and maintained using Dbvisit Standby.



**3. Inter Cloud**

There is also now a third option for customers, that is Inter Cloud. Rather than put their primary in Company A's Cloud and their DR site also in Company A's Cloud, they have opted to put their DR database into a different company's Cloud infrastructure. You might do this so that you are not completely locked into one cloud vendor, or to provide some redundancy in case of cloud outages. You might also be looking to test different cloud vendors to see what their capabilities are or you may find a different strategic reason for doing so.

# Oracle RAC and High Availability

Oracle RAC is in a league of its own, and is a powerful option that provides you with a highly available, scalable solution at an affordable cost as you only pay for the database license, not the RAC license. Implementing a standby database for an Oracle RAC environment is possible and as mentioned before can be done in two ways: writing your own custom scripts, or by using 3rd Party Products.

I also want to mention again that if you are using Oracle 19c SE2 and above, you will not be able to use Oracle RAC anymore as it was removed from the code set. However, you can still configure an Active/Passive cluster with Oracle Linux and Oracle Clusterware together with your Oracle 19c SE2 database and achieve a highly available solution. We are seeing more and more people looking at this and it is well worth investigating further if you are running Oracle RAC in older versions of the database. For more detail on how this can be implemented, please see the Dbvisit Standby version 9 Documentation under the Advanced section where a full detailed example is provided.

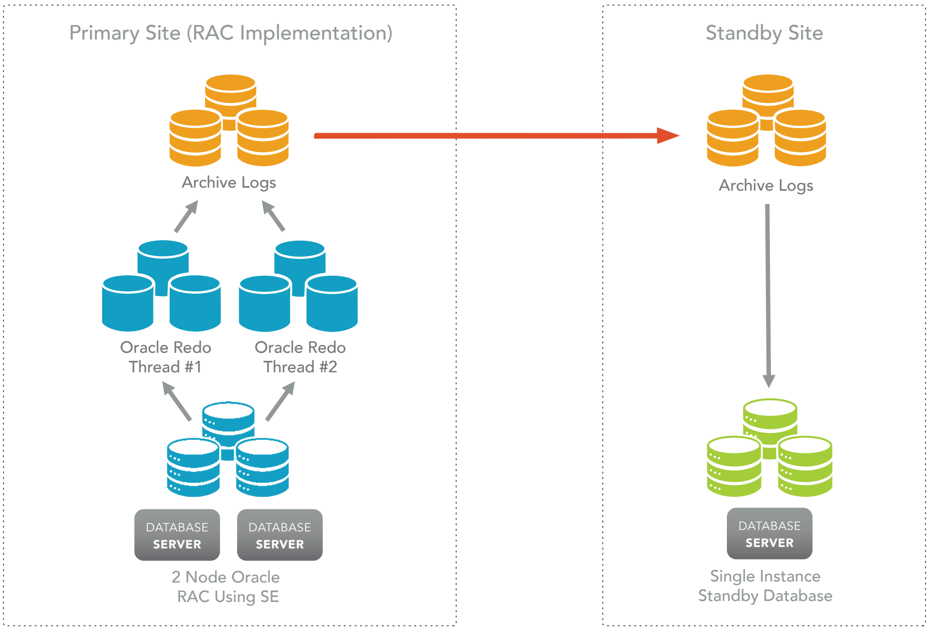
However as mentioned before as there are still a number of companies using Oracle RAC in Oracle SE and Oracle SE2 up to 18c I do want to continue with the test case below.

### Oracle RAC with Standby Database using Standard Edition 2: A Test Case

To implement a standby database in an Oracle RAC environment using a 3rd Party Product called Dbvisit Standby is not that complex, and I will provide you with the steps you need to know to get started. In this test case, I will provide a high level overview of how you can implement and maintain a standby database using Dbvisit Standby. The table below provides a high level summary of the test case environment:

| Primary Environment (Oracle RAC Cluster) | Standby Environment (Single Node) |
|---|---|
| 2 x Oracle Linux Servers | 1 x Oracle Linux Server |
| Oracle Clusterware configured with ASM storage | Oracle Grid Infrastructure with ASM storage |
| Two disk groups +DATA and +FRA | Two disk groups +DATA and +FRA |
| Oracle Database Standard Edition 2 | Oracle Database Standard Edition 2 |
| Oracle Home: /u01/app/oracle/product/12.2.0/db_1 | Oracle Home: /u01/app/oracle/product/12.2.0/db_1 |
| Primary Server Names: dbvrlin307.dbvisit.co.nz and dbvrlin308.dbvisit.co.nz | Standby Server Name: dbvlin309.dbvisit.co.nz |
| Database Name: DEV | Standby Database Name: DEV |
| Datafile Locations: +DATA | Datafile Locations: +DATA |
| Fast Recovery Area: +FRA | Fast Recovery Area: +FRA |
| Shared storage is configured using ACFS to be used for Dbvisit Standby installation - shared storage is mounted under: /acfs | |

The diagram below provides a high level overview of the implementation:



The steps below cover the high level key processes for Dbvisit Standby installation - for more detail please see the Dbvisit Standby Version 9 user guide:

- **Step 1. Perform Prerequisite Steps**

Before installing Dbvisit Standby, the following should be reviewed:

- Add entries to your `/etc/oratab` file (Unix environments) to allow you to quickly set your environment using the "`. oraenv`" script. An example entry in the `/etc/oratab` file is in format: `$ORACLE_SID:$ORACLE_HOME:<N|Y>`
- Ensure that your primary database is in Archivelog Mode
  - Review with "`archive log list`" command
  - Enable with "`alter database archivelog;`" while in mounted state (done from one instance in cluster)
- Enable Force Logging on the primary (not strictly required, but highly recommended)
  - Run command "`alter database force logging;`"
- Create the directory `/acfs/dbvisit` on the shared storage location (mounted under /acfs). This directory will now be available on all nodes. Make sure the Oracle software owner owns this directory
- Remember to create a virtual IP and assign a hostname alias to it - this will be used by Dbvisit Standby - in this example we call it **dbv-vip** and it will run on node 1. In summary, Dbvisit Standby will only run on one of the RAC nodes - if that node fails or is shutdown, the Virtual IP and Dbvisit Standby processes will be moved to the second node (action scripts and having Dbvisit Standby processes added as cluster resources will be required).
- **Step 2. Install Dbvisit Standby**

This is a simple process and the online documentation will explain all the steps required in detail. The high level steps of the installation can be summarized as follows:

- Download the latest Dbvisit Standby software into a temporary folder, for example: /home/oracle/dbv8
- Run the following to start the installer (as the Oracle software owner):
  - `cd /home/oracle/dbv8`
  - `unzip *.zip; tar xf *.tar; cd dbvisit/installer; chmod 755 dbvisit_install; ./dbvisit_install`
  - This will start the installer from where you can respond to the questions regarding the installation directory and the software owner.
  - Perform the above tasks on the primary RAC node where the virtual IP is running (dbv-vip) as well as on the standby database server. It is recommended to install the software on the standby database server into `/usr/dbvisit` as the Oracle database software owner, which in this case will be the UNIX Oracle account. During the installation of the software on the standby server, make sure you install the Standby Core Components (Dbvnet, Dbvagent, Standby CLI) - option 1, and the Central Console (Dbvserver) - option 2.
    - It is recommended to install the Dbvisit Standby Central console into a 3rd separate system, but in this example it will be installed on the Standby Database server.
- You now have two options, you can either do the configuration using the Central Console which would have been started on the standby server following the above installation (https://your_server_name:4433) or you can use the command line (run `/usr/dbvisit/standby/dbvctl -o setup`). This initial configuration needs to be run only on the primary RAC node, not on the standby server.
- During this configuration process you will be creating the Dbvisit Standby configuration files, also referred to as the DDC files. All the detail required to create this file is filled in while running the initial configuration.
- Once the installation is complete and you have created the DDC configuration file, you are ready to create the database.

- **Step 3. Create Standby Database**
- This process is also known as the CSD process. The process can either be started from the command line using the "`dbvctl -o setup --csd`" command or you can use the "Create Standby Database" option in the Central Console (GUI). This process will only take a few minutes and you will be asked to provide information about your standby server configuration. Once done, you can start the standby database creation process. Dbvisit Standby will make use of RMAN to back up the primary database and restore it on the standby server. The standby database creation process may take some time depending on the size of the primary database.
- Once this process is done, you will have a fully functional standby database on the standby server.

- **Step 4. Sending and Applying Logs**
- Once the installation is complete, you can start sending logs from the primary nodes to the standby site and applying the received archived redo logs on the standby.  For example, in this environment you would run the following:

```
Primary Node:  dbvrlin307
cd /acfs/dbvisit/standby
./dbvctl -d DEV

On the standby (single instance) environment: dbvrlin309
cd /usr/dbvisit/standby
./dbvctl -d DEV
```

- Once you confirm that the logs are shipping and applying, you can use the "-D start|stop|status" options to start the Dbvisit Standby Daemon (background) process to start auto shipping and applying of logs. The command should be run once on the primary and standby servers: `./dbvctl -d DEV -D start`
- **Step 5. Other Options**
- There are a number of options available to perform Failover, Graceful Switchover or even run internal Dbvisit reports. These options are available from the command line as well as the GUI. See Appendix A for an example of the GUI. The key commands to be aware of in the command line include:

```
o Send or Apply logs:  ./dbvctl -d <DDC>
o Perform Switchover: ./dbvctl -d <DDC> -o switchover
o Perform Activation: ./dbvctl -d <DDC> -o activate
o Run a log gap report:  ./dbvctl -d <DDC> –i
```

The focus of this whitepaper is to highlight that there are options available to help you create and maintain standby database configurations when using Oracle Standard Edition 2, such as Dbvisit Standby. For more details on the product, please review its online documentation at https://dbvisit.atlassian.net/wiki/spaces/DS9QSG/overview

# Migration from Enterprise to Standard Edition 2

So you are running Oracle Enterprise Edition (EE) and would like to switch to Standard Edition 2 (SE2) - how is this done? Firstly, even though at the core Standard Edition 2 and Enterprise Edition are the same, you should not just move your EE database to an SE software set without reviewing what you have in place first. This might appear to work initially, but remember the database dictionary is still EE and there might be EE features in use in the database and as soon as you attempt to use these features on the SE software, you will run into errors stating that the feature is not enabled. For example, if you move an EE database to an SE and you had partitioning in use, if you try and create an index on this partitioned table you will get an error - **ORA-00439: feature not enabled: Partitioning.**

To summarize, do not make the switch assuming it will work without reviewing what features you have in place already – you might find out later that things do not work and you might end up with features suddenly not available.

To review what database features you are using, you can make use of the view dba_feature_usage_statistics. There might be steps required to get the data refreshed, for more detail see My Oracle Support note – 1629485.1

Migration between the editions can easily be performed in two ways:

- Use Oracle provided Export/Import tools (datapump or the original exp/imp)

During this process a new clean SE database is created, the data from the EE database is exported and then loaded into the SE database. It is important to make sure EE database features are removed, as they will not be usable in the SE database. The recommendation is to make sure you test this process a number of times on your test systems and make sure that the applications are fully tested. This will ensure that when you perform these tasks in production, you are aware of the timing and know the steps.

# Are Two Sockets Really a Limitation?

One of the restrictions when looking at Standard Edition 2 is the amount of CPU sockets allowed to be used in the configuration. For Standard Edition 2, the limit is a maximum of two CPU sockets in a cluster even if the Oracle RAC option is used with Standard Edition 2. Now for most this might seem like a showstopper, but it might not be. In the last five years, the CPU modules improved drastically, running at much faster speeds and with much more cores and threads than before. And in many cases we see people just refreshing hardware to get newer faster processors to still get the most out of their licensing investment.

# Reasons to Consider Using Oracle Standard Edition 2

Standard Edition 2 is still the same core database we know and is rated at the top of its class. Oracle Database technology is at the forefront of new innovation providing exceptional performance, high availability, scalability, backup and recovery. Standard Edition 2 is part of this proven technology. For me, there are a few key reasons to consider using Standard Edition 2:

- Standard Edition 2 is proven database technology
- Affordable cost of ownership
- Standard Edition 2 pricing is truly attractive
- Oracle Real Application Clusters (RAC) option is a no charge option with Standard Edition 2
- Disaster Recovery with Standard Edition 2 is still possible with standby databases
- The Automatic Workload Repository, performance diagnostics and tuning options are not available for Standard Edition 2, but what about Statspack? It is powerful and still there for you to use as well as many other 3rd party products that can help you with performance tuning when using Standard Edition 2.

# Conclusion

There are a number of database editions available. Which one do you choose? Your decision should not be based on cost alone. If that was the case, then everyone would select Oracle Standard Edition 2. Before you select the database edition to use in your environment, you need to review the following:

- Oracle licensing structure and costs associated with each available option.
- Review each database edition's capability. Make sure you understand what is included and excluded.
- Then most importantly: review your business and application requirements. Make sure you know what the options and requirements are. You might be surprised to find that Standard Edition 2 is the perfect fit.

This whitepaper provided you with an overview of the different editions available, as well as some examples of cost comparisons to give you an idea of the cost benefit of using Oracle Standard Edition 2. More importantly, Oracle Standard Edition 2 might be the younger brother of Enterprise Edition, but with Oracle RAC - or using an Active/Passive cluster with Oracle Linux and Clusterware, and standby databases being available, you can still achieve a highly available environment with DR capability at an affordable cost.

# Appendix A: Oracle Processor Core Factor Table Extract

A full copy of the latest Processor Core Factor table document can be found here:

http://www.oracle.com/us/corporate/contracts/processor-core-factor-table-070634.pdf

# Appendix B: Flashback Query

Have you ever run a delete statement just to kick yourself two minutes later for doing that, as you still need the data? I know I have and this is something I am sure happens to most during their DBA career. But it does not have to be the end of the world. Oracle introduced a truly powerful option since Oracle 9i called Flashback Query. With Flashback Query, you can "go-back in time" and get the data back.

Flashback Query is making use of "undo" information to make this possible and the following is required:

- Have to use Automatic Undo
- Sufficient space in the Undo Tablespace
- Adjust the UNDO_RETENTION (in seconds) as required

In summary, Flashback Query allows you to view data as it was at a particular point in time, using TIMESTAMP or SCN. You can do this in multiple ways:

- Using the DBMS_FLASHBACK package functions:
  - get_system_change_number ();
  - enable_at_system_change_number();
  - enable_at_time();
  - disable();
- From 10g the "AS OF" clause can be used with TIMESTAMP and SCN:
  - select * from <table> AS OF TIMESTAMP <timestamp>;
  - select * from <table> AS OF SCN <scn>;

**Example A: Using Flashback Query**

In this basic example you can see the power of Flashback Query. First a table "test" is created in the schema "aels". Two records are inserted into this table. The current SCN number is extracted and then a row is deleted with a commit being executed after that. Flashback Query using "AS OF SCN" option is then used to get a view of the data as it was at that time (SCN number).

```
SQL> grant execute on dbms_flashback to aels;
Grant succeeded.

SQL> connect aels/aels
Connected.

SQL> create table test (id number);
Table created.

SQL> insert into test values (1);
1 row created.

SQL> insert into test values (2);
1 row created.

SQL> commit;
Commit complete.

SQL> select  dbms_flashback.get_system_change_number from dual;

GET_SYSTEM_CHANGE_NUMBER
------------------------
                 4300790
```

```
SQL> delete from test where id=1;
1 row deleted.

SQL> commit;

Commit complete.

SQL> select * from test;

        ID
----------
         2

SQL> select * from test as of scn 4300790;

        ID
----------
         1
         2

SQL> select * from test;

        ID
----------
         2
```

## References

http://www.oracle.com/us/corporate/contracts/processor-core-factor-table-070634.pdf

http://www.oracle.com/us/corporate/pricing/technology-price-list-070617.pdf

https://shop.oracle.com

http://www.oracle.com/us/corporate/pricing/sig-070616.pdf

https://docs.oracle.com/en/database/oracle/oracle-database/12.2/dblic/licensing-information-user-manual.pdf

https://docs.oracle.com/en/database/oracle/oracle-database/12.2/adfns/development-guide.pdf

**Dbvisit Software Limited** (www.dbvisit.com)

www.dbvisit.com

info@dbvisit.com

**USA:** 1-800-933-8007
**Rest of the world:** +64 9 950 3301

## ÜBER HUNKLER

Das Karlsruher Systemhaus HUNKLER wurde 1988 erster offizieller Partner von Oracle in Deutschland. Ein Team von rund 20 Mitarbeitern unterstützt Kunden aus Industrie, öffentlicher Verwaltung, Gesundheits- und Finanzwesen mit Beratung, Lösungsentwicklung und Managed Services.

Im Fokus von HUNKLER stehen leistungsfähige, wirtschaftliche Infrastrukturen für Oracle-datenbanken mit den Schwerpunkten Hochverfügbarkeit, Ausfallsicherheit und Zero Downtime Migration. Die integrierten Komplettlösungen der Produktfamilie Oracle Engineered Systems sowie der Datenbank-/Anwendungsbetrieb in der Oracle Cloud sind weitere Themenfelder, die das Unternehmen umfassend abdeckt.

HUNKLER
GmbH & Co. KG

**Hauptsitz Karlsruhe**
Bannwaldallee 32, 76185 Karlsruhe
Tel. 0721-490 16-0, Fax 0721-490 16-29
**Geschäftsstelle Bodensee**
Fritz-Reichle-Ring 6a
78315 Radolfzell
Tel. 07732-939 14-00, Fax 07732-939 14-04
**info@hunkler.de, www.hunkler.de**