

# Oracle Real Application Clusters (RAC) on Extended Distance Clusters

With updates for Oracle Database 11g Release 2

*An Oracle White Paper*

*August 2010*

# Oracle Real Application Clusters on Extended Distance Clusters

## TABLE OF CONTENTS

|  |    |
|--|----|
| Executive Overview .....   | 3  |
| Introduction .....   | 4  |
| Benefits of Oracle RAC on Extended Distance Clusters.....                  | 5  |
| Components & Design Considerations .....                                   | 6  |
| Full Oracle Stack .....  | 15 |
| Comparison with Local RAC + Data Guard.....                                | 17 |
| Conclusion.....  | 20 |
| Appendix A: Detailed Quorum Examples .....                                 | 21 |
| Appendix B: Customers Using Oracle RAC on Extended Distance Clusters ..... | 23 |
| Appendix C: Usage of separate Subnets for the Public Network .....         | 24 |
| References .....   | 25 |

# Oracle Real Application Clusters on Extended Distance Clusters

## EXECUTIVE OVERVIEW

Oracle Real Applications Clusters (RAC) is a proven mechanism for local high availability (HA) for database applications. It was designed to support clusters that reside in a single physical datacenter. As technology advances, customers are looking at the viability of using Oracle RAC over a distance.

Can Oracle RAC be used over a distance, and what does this imply? Oracle RAC on Extended Distance Clusters is an architecture that provides extremely fast recovery from a site failure and allows for all nodes, at all sites, to actively process transactions as part of single database cluster. While this architecture creates great interest and has been successfully implemented, it is critical to understand where this architecture best fits especially in regards to distance, latency, and degree of protection it provides.

The high impact of latency, and therefore distance, creates some practical limitations as to where this architecture can be deployed. This architecture fits best where the 2 datacenters are located relatively close (<~100km) and where the extremely expensive costs of setting up Oracle RAC dedicated direct connectivity between the sites has already been taken.

Oracle RAC on Extended Distance Clusters provides greater high availability than local Oracle RAC but it may not fit the full Disaster Recovery requirements of your organization. Feasible separation is great protection for some disasters (local power outage, airplane crash, server room flooding) but not all. Disasters such as earthquakes, hurricanes, and regional floods may affect a greater area. Customers should do an analysis to determine if both sites are likely to be affected by the same disaster.

For comprehensive protection against disasters including protection against corruptions, and regional disasters Oracle recommends the use of Data Guard with Oracle RAC as described in the Maximum Availability Architecture (MAA). Data

Guard also provides additional benefits such as support for full rolling upgrades across Oracle versions.

Configuring an extended distance cluster is more complex than a local cluster. Specific focus needs to go into node layout, quorum disks, data disk placement, network connectivity constraints, and other factors discussed in this paper.

Implemented properly, this architecture can provide greater High Availability than a local Oracle RAC database. This paper will address the necessary components, the benefits and limitations of this architecture, and will highlight some actual customer examples.

## **INTRODUCTION**

Oracle's Real Application Clusters (RAC) is designed primarily as a scalability and availability solution that resides in a single data center. It is possible, under certain circumstances, to build and deploy a Oracle RAC system where the nodes in the cluster are separated by greater distances. For example if a customer has a corporate campus they might want to place the individual Oracle RAC nodes in separate buildings. This configuration provides a degree of disaster tolerance, in addition to the normal Oracle RAC high availability, since a fire in one building would not, if properly set up, stop database processing. Similarly many customers have two data centers in reasonable proximity (<100km) which are already connected by direct, non-routed, high speed links and are often on different power grids, flood plains, and the like.

This paper discusses the potential benefits that attract customers to this type of architecture, covers the required components and design options that should be considered during implementation, reviews empirical performance data over various distances, covers supported and non-supported configurations, and reviews the additional advantages that Oracle Data Guard provides to this solution. Finally it looks at how extended Oracle RAC is being used by customers today.

Clusters, where all the nodes are not local, have been referred to by many names including campus clusters, metro clusters, geo clusters, stretch clusters and extended clusters. Some of these names imply a vague notion of distance range.

Throughout this paper this type of configuration will be referred as Oracle RAC on Extended Distance Clusters.

This paper is intended to provide a deeper understanding of the topic and to allow one to determine if this type configuration is applicable and appropriate.

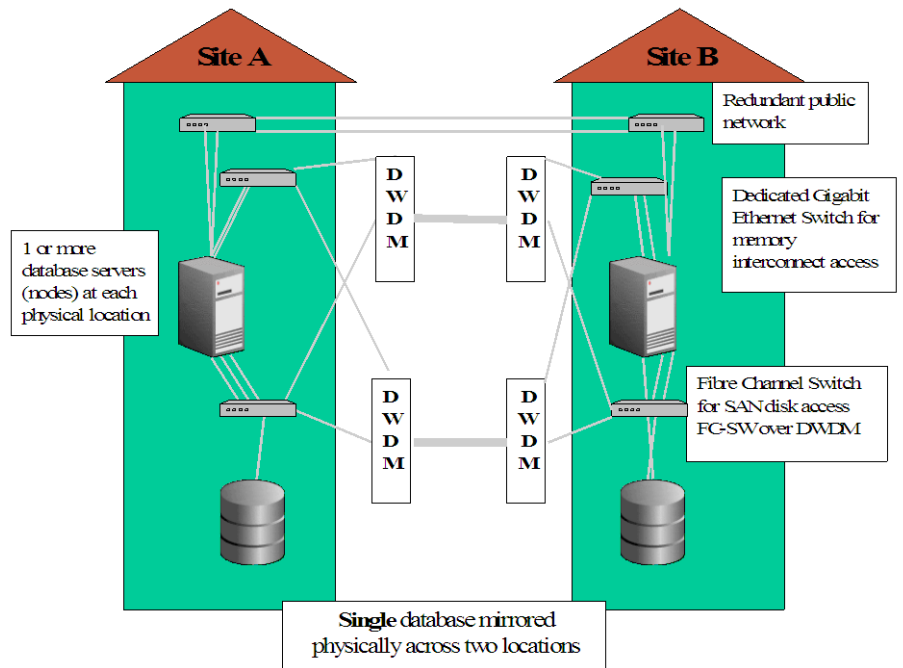


## COMPONENTS & DESIGN CONSIDERATIONS

Oracle RAC on an Extended Distance Cluster is very similar to a Oracle RAC implementation at a single site.

To build a Oracle RAC database on an Extended Distance Cluster environment you will need to.

- Place one set of nodes at Site A
- Place the other set of nodes at Site B
- Place the tie breaking voting disk at a third site
- Use host based mirroring to allow you to host all the data on both sites and keep it synchronously mirrored.
- Use fast dedicated connectivity between the nodes/buildings for Oracle RAC cross instance communication (for example a dedicated wavelength on Wavelength Division Multiplexing over Dark Fiber)



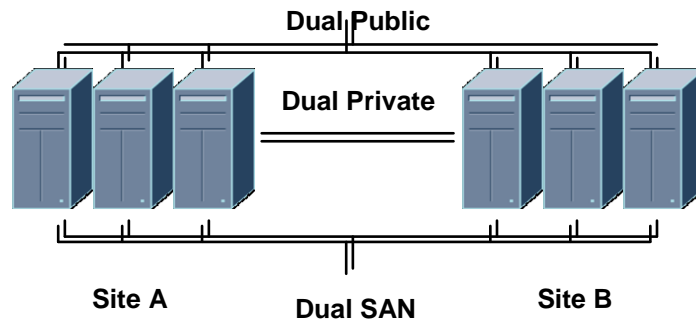
Details of the components, and design considerations, follow.

## Connectivity

Networking requirements for an extended distance cluster are much greater than those of a normal Wide Area Network (WAN) used for Disaster Recovery. This plays in two aspects: necessary connections and latency.

### **Necessary Connections**

Interconnect, SAN, and IP Networking need to be kept on separate *dedicated* channels, each with required redundancy. Redundant connections must not share the same Dark Fiber (if used), switch, path, or even building entrances. These channels should not be shared by any other communication channel or link. Keep in mind that cables can be cut, so physically separate paths should be part of the overall network design.



The SAN and Interconnect connections need to be on direct, non-shared, point-to-point cables (see effects of latency in the next section). Normal SAN and Ethernet connections are limited to 10km for point to point communications. WDM over Dark Fibre networks allow these to be much farther apart while still maintaining the low latency of a direct connection. The disadvantage of Dark Fibre networks is that they are extremely expensive, so generally they are only an option if they already exist between the two sites.

### **Notes of caution:**

Do not configure the Oracle RAC Interconnect over a WAN. These connections must be on the same subnet, so routing between Data Centers will not be possible. Network traffic that would use the same WAN would also cause performance degradations or even node evictions.

A single subnet on all nodes is required for the private interconnect.

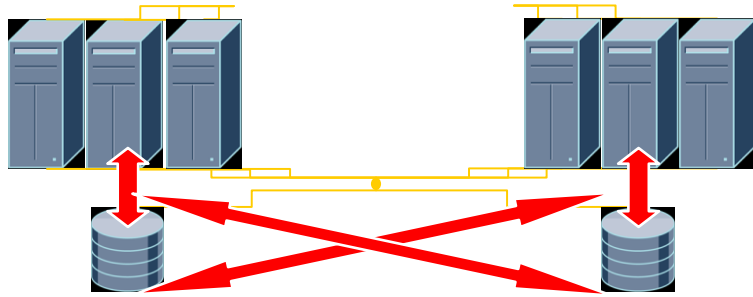
For the 'public' network, a single subnet is recommended as it best simulates a local cluster and allows full functionality to be provided. Please see Appendix C for a discussion on of separate Subnets for the Public Network.

Latency effects and performance implications of distances are discussed in the Latency & Empirical Performance Results Chapter.

## Storage

Oracle RAC on Extended Distance Clusters by definition has multiple active instances on nodes at different locations. For availability reasons the data needs to be located at both sites, and therefore one needs to look at alternatives for mirroring the storage.

### **Host Based Mirroring (Active/Active Storage)**



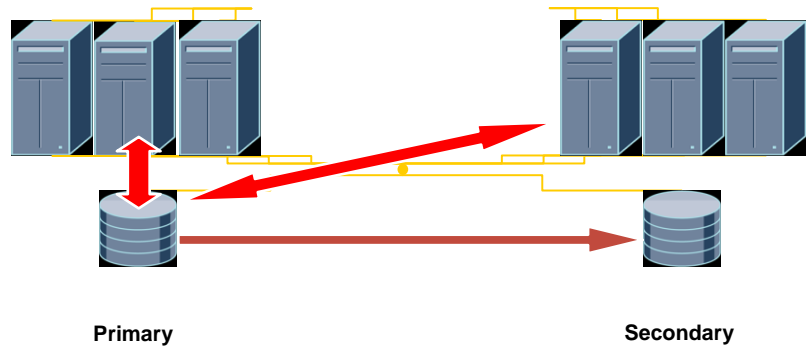
- Use two SAN/FC storage subsystems, one co-located with each node.
- Standard, cluster aware, host based mirroring software is implemented across both disk systems. With this, system writes are propagated at the OS level to both sets of disks, making them appear as single set of disks independent of location. These Logical Volume Managers (LVM) need to be tied closely with the clusterware. Examples of these include Oracle's Automatic Storage Management (ASM), Veritas CVM, & HP-UX MirrorDisk/UX..
- While there may be a performance impact<sup>1</sup> from doing host based versus array based mirroring, this is the preferred configuration from an availability perspective and the only configuration supported by Oracle from now on. When we refer to Oracle RAC on Extended Distance Clusters in this paper, it generally refers to this storage configuration.

---

<sup>1</sup> Host based mirroring requires CPU cycles from the host machines. Array based mirroring offloads this work to the storage layer. Advantage or disadvantage of this depends on which layer you either, have spare cycles, or it is more cost effective to add cycles.

### Array Based Mirroring (Active/Failover Storage)

**CAUTION:** Array Based Mirroring generally implies a primary/secondary storage site solution. Should the primary storage location fail, all instances will crash and need to be restarted once the secondary storage is made active. Array based mirroring requires a switch be made from receiving changes at the remote side to functioning as local disk. From an HA viewpoint it is recommended to instead do Host Based mirroring as it does not require a manual restart.



- Use two SAN/FC storage subsystems, one co-located with each node and each is cross cabled to both nodes
- One storage subsystem has all the live database files on it, all writes are sent to this system
- The second storage subsystem has an array based mirror mechanism (i.e. EMC's SRDF, HP's CA, etc.) of the first storage subsystems files
- Performance impacts in this case come from both doing additional work in the storage array for the mirroring, but more importantly by I/OS from the secondary site having to cross the 'distance' 4 times<sup>2</sup> before they return control.

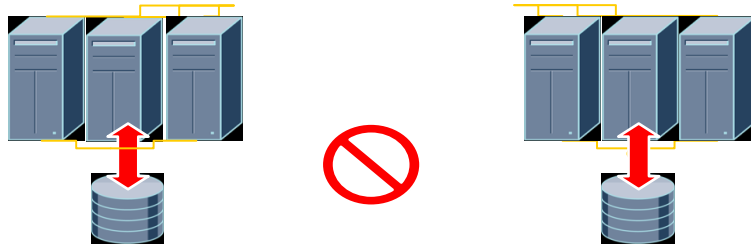
### Why not have just single storage locations?

While it is possible to implement Oracle RAC on Extended Distance Clusters with storage on only one site, should the site with the storage fail, storage is no longer available to any surviving nodes, and the whole cluster becomes unavailable. This defeats the purpose of having had the Oracle RAC nodes at different locations.

---

<sup>2</sup> The I/O will have to travel from the Secondary host to primary storage, then from the primary storage to secondary storage, then from the secondary storage to primary storage, and finally from the primary storage to secondary host. All need to be synched to ensure no data loss.

## Cluster Quorums, or Ensuring Survival of One Part of the Cluster:



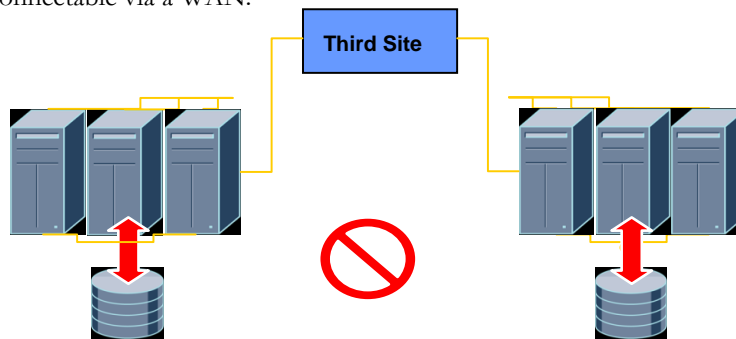
Cluster quorum mechanisms have a bigger impact on the design of an extended distance cluster than they would on a local cluster.

**CAUTION:** Extended RAC implementations without a third site for tie breaking quorum, require making one site a 'primary' site and the other a secondary. Then should the primary site fail, the secondary site will require a manual

When a local cluster is being built, one need not worry much about how quorum mechanisms work. Cluster software is designed to make the process fool proof, both for avoiding split brains<sup>3</sup>, and for giving the best odds for a portion of the cluster to survive when a communication failure between the nodes occurs.

Once the nodes of the cluster are separated, things are no longer so simple. They have a tie breaking mechanism that must be located someplace.

All clusterwares support putting a tie-breaker mechanism located at a third site. This allows both sites to be equal and the third site acts as an arbitrator should either site fail or connectivity be lost between the sites. Because of the HA implications, the 3 site implementation is highly recommended. With Oracle Clusterware the third site does not have the same connectivity requirements and is connectable via a WAN.



Setting up voting disks across sites should only be done directly via the clusterware software. They should not be mirrored remotely by other means otherwise this could potentially result in a dual active database scenario.

More detailed discussion and examples of quorum mechanisms, and the alternatives for implementing the third site are discussed in Appendix A.

---

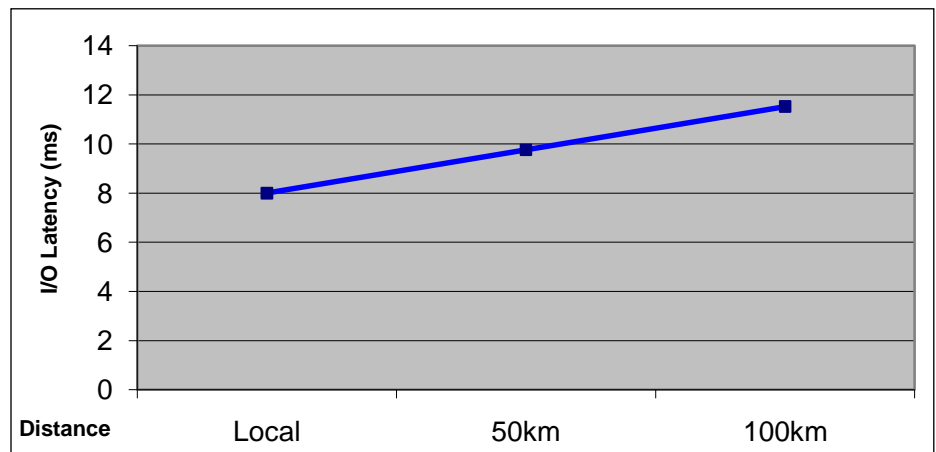
<sup>3</sup>A true split brain occurs when two portions of the cluster stop coordinating and start acting independently. This can easily lead to database corruption, so clustering software is carefully written to avoid a split brain situation from occurring. Should nodes start misbehaving or unable to communicate to each other, nodes will be evicted to ensure that at most only one subcluster survives.

## Latency & Empirical Performance Results

Oracle Real Application Clusters requires that the cluster interconnect (and thus Cache Fusion) be a dedicated, low latency network. The cluster interconnect can be conceived of as being the method used to extend the backplanes of the nodes into one logical unit since it is used to synchronize the various database caches. A dedicated network is required to ensure consistent response times and avoid the loss, or excessive delay of, the cluster heartbeat, which can cause nodes to be kicked out of the cluster. Interconnect latency directly affects the time it takes to access blocks in the cache of remote nodes, and thus it directly affects application scalability and performance. Local interconnect traffic is generally in the 1-2 ms range and improvements (or degradations) can have a big effect on the scalability levels of the application. I/O latencies tend to be in the 8-15ms range, and are also affected by the additional latencies introduced with distance.

Various partners have tested Oracle RAC on Extended Distance Clusters. These tests include ones done by HP and Oracle at 0, 25, 50, and 100 km; tests done by the EMEA Joint Solutions Center Oracle/IBM at 0, 5, and 20 km; and tests done by Veritas at 0, 20, 40 and 80km. All included a full OLTP application test and some included unit tests of the individual components.

The unit tests results from the HP/Oracle testing will be used to illustrate what happens at each component level.

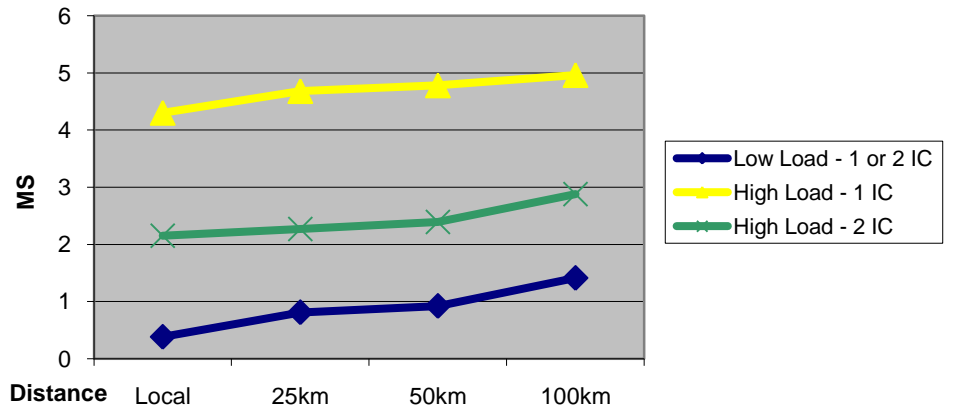


This figure shows the effects of distance on I/O latency with buffer-to-buffer credits (BBC). BBC allow a greater number of unacknowledged packets on the wire, thus allow greater parallelism in the mirroring process. As distances increase, especially with high traffic volumes, these BBC can make a huge difference. For example when the tests above were run without the additional BBC, I/O Latency at 100km was 120-270% greater than local, instead of 43% in the chart above.

These numbers are consistent with the results from the Oracle/IBM testing which had 20-24% throughput degradation on I/O Unit tests at 20km when BBC were not used.

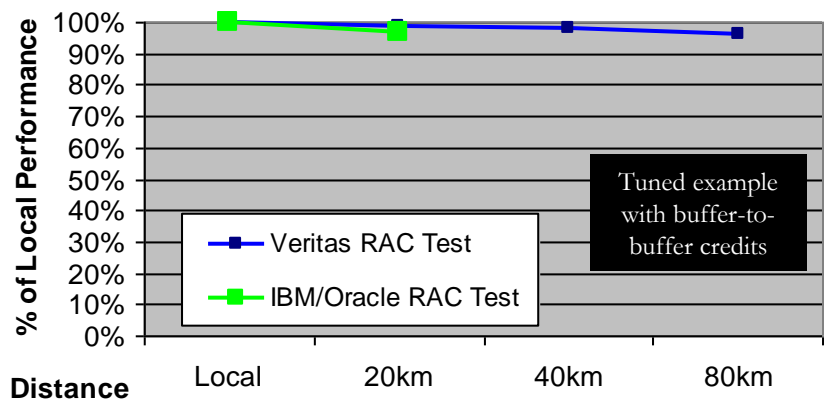
**Interconnect Traffic Unit Test Results**

Tests at both high and low load levels, and with one or two interconnects, show that there is a latency increase of about 1 ms at 100km. While Cache Fusion traffic is not as sensitive to distance as I/O latency, the effect of this latency increase can be as significant



**Overall Application Impact**

Unit tests are useful, but the final real impact comes down to how a full application reacts to the increased latencies induced by distance. Having three independent sets of tests provides a more complete picture than each individual test. A summarization of each test is provided, and full details can be seen in the paper by each respective vendor listed in references. An important note: these tests were performed with Oracle9i, and since then they have been many improvements to the cache fusion algorithms which could improve performance.

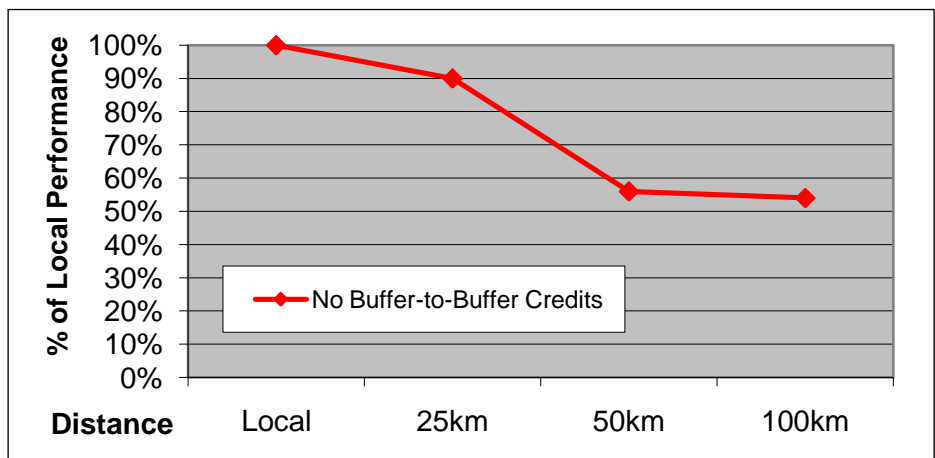


The IBM/Oracle tests performed a representative workload, which was accomplished by running the SwingBench workload with proper use of BBC. These tests at 20km showed 1% degradation for read transactions, 2-8% degradation for most write transactions. The average single transaction resulted in 2% degradation.

Veritas used another well-known OLTP workload, and set it up in a manner in which it was highly scalable. These tests done at 0, 20, 40, and 80km showed that the application suffered minimal performance loss (4% in their worst case at 80km).

Other tests were done without having buffer-to-buffer credits nor directing reads to the local reads. Combined with a very contentious application, this resulted in some impact at 25km (10%), but significant degradation at 50km-100km. Further testing is needed to determine why the 50 & 100km numbers are similar, but the 0, 25 and 100km numbers form a very nice linear slope. With appropriate BBC these numbers would be expected to significantly improve and be closer to the Veritas and Oracle/IBM numbers.

**⚠ CAUTION: Not using BBC can cause serious application performance degradation for greater distances**



Real life applications are expected at best to follow the IBM/Oracle & Veritas examples demonstrated earlier. In reality they will probably have more interconnect traffic and thus suffer slightly more from the distance.

Each of these results is for a particular application with a particular setup. Other applications will be affected differently, but the basic idea is that as distance increases, IO and Cache Fusion message latency increases. The limitations come from a combination of the inefficiencies and latency added by each switch, router

or hub a message must be handled by.<sup>4</sup> As was previously stated, Dark Fiber can be used to achieve connections greater than 10km without repeaters.

While there is no magic barrier to how far Oracle RAC on an Extended Distance Clusters can function, it will have the least impact on performance at campus or metro distances. Write intensive applications are generally more affected than read intensive applications. If a desire exists to deploy Oracle RAC at a greater distance, performance tests using the specific application are recommended.

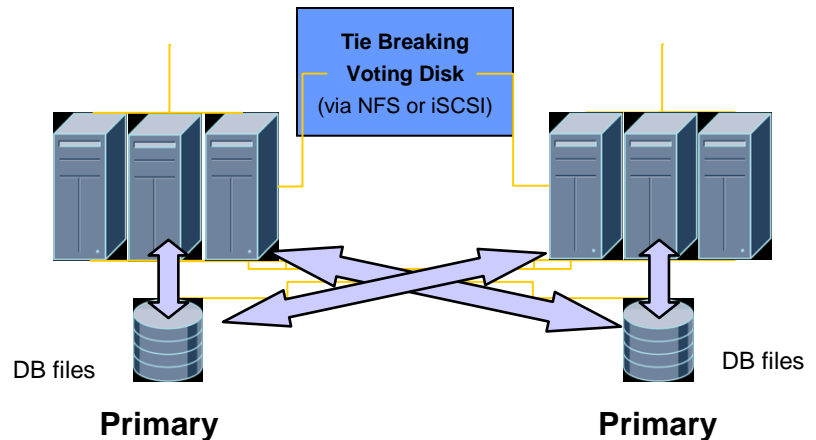
From these numbers I am extremely comfortable with a Oracle RAC on Extended Distance Clusters at distances under 50km, and recommend performance testing over 100km. There is no magic barrier for the distance; latency just keeps getting worse. Write intensive applications are generally more affected by distance than read intensive applications due to the efficiency of the intelligent caching by the database.

---

<sup>4</sup> To avoid large latencies the configuration should only have a switch at each site, WDM device, and a direct uninterrupted physical connection in between. No additional routed network.

## FULL ORACLE STACK

Since Oracle Database 10g Release 2, an extended cluster can be created on any OS using standard Oracle components. The Oracle Clusterware can be used for integrity and Automatic Storage Management (ASM) for mirroring.



## Oracle Clusterware

Since Oracle Clusterware 10g Release 2, Oracle provides direct support for mirroring of the Oracle Cluster Repository (OCR), as well as supporting multiple voting disks.

To setup an extended Oracle RAC with Oracle Clusterware:

1. OCR must be mirrored across both sites using Oracle provided mechanisms.
2. Preferably have two voting disks at each site and tie-breaking voting disk at a third site. This third site only needs to be a supported NFS device over a WAN. This can be done via a NetApp filer or on most platforms this can be done via standard NFS.<sup>5</sup> Starting in Oracle Clusterware 11g Release 2 this can be hosted on ASM on a dedicated Quorum Failure Group.

## ASM

Built in mirroring can be used to efficiently mirror all database files across both sites. Storage at each site must be setup as separate failure groups and use ASM mirroring, to ensure at least one copy of the data at each site.

---

<sup>5</sup> [Roland Knapp, Daniel Dibbets, Amit Das, Using standard NFS to support a third voting disk on an Extended Distance cluster configuration on Linux, AIX, HP-UX, or Solaris \(PDF\), December 2009](#)

From Oracle Database 11g onwards several enhancements are available with ASM to specifically provide better support for extended clusters:

1. **Partial Re-silvering:** With the fast resync option, full re-silvering is no longer required for ASM mirrors should a temporary loss of connectivity between the sites occur. The amount of time re-silvering information is maintained is configurable.
2. **Local Reads:** I/O read requests can be configured via the `ASM_PREFERRED_READ_FAILURE_GROUPS` parameter to go to the local mirror instead of going to any available mirror. Reading from both mirrors is better for shorter distances as all IO cycles are fully utilized. Local mirrors are better for further distances as all reads are satisfied locally.
3. **OCR and Voting Disk in ASM (11.2+):** These can now be located directly on ASM volumes. Care must be taken for voting disks that they are distributed across all 3 sites.

## COMPARISON WITH LOCAL RAC + DATA GUARD

Here is a comparison of a Oracle RAC over an Extended Distance Cluster versus a local Oracle RAC cluster for HA and Oracle Data Guard (DG) for DR.

### Comparison Summary

|                                      | Oracle RAC on Extended Distance Clusters   | Local Oracle RAC + Oracle Data Guard  |
|--------------------------------------|--|---|
| <b>Active Nodes</b>                  | All  | One Side Only<br>DG site can be used for reporting purposes   |
| <b>Recovery from Site Failure</b>    | Seconds, No Intervention Required  | Minutes, No Intervention Required <sup>6</sup>  |
| <b>Performance Hit</b><br>See charts | Minor to Crippling   | Insignificant to Minor in some Cases  |
| <b>Network Requirements</b>          | High cost direct dedicated network w/ lowest latency. Much greater network bandwidth | DG Sync - High cost direct dedicated network w/ lowest latency.<br><br>DG Async<br>Shared commercially available network. Does not have low latency requirements. |
| <b>Effective Distance</b>            | Campus & Metro   | Country and Continental-Wide distances  |
| <b>Disaster Protection</b>           | Host, building, and localized site failures, some local site disasters.              | Host, building, localized site failures,<br>Database Corruptions<br>Local and wider area Site Disasters   |
| <b>Costs</b>                         | High Network Costs   | Additional Nodes  |

---

<sup>6</sup> Assuming you are using Fast-Start Failover in Oracle10gR2 onwards

## **Strengths of Oracle RAC on Extended Distance Clusters**

### ***All Nodes Active***

One of the main attractions for an Extended Distance Cluster environment is that all nodes can be active, and dedicated nodes are not required for disaster recovery. Thus instead of a minimum of 2 Oracle RAC clusters required in full RAC+DG architecture, 1 Oracle RAC cluster can be used. One note of comment: in a RAC+DG architecture, the DR site can be used for other purposes including reporting and decision support activities.

In environments with larger number of nodes, some advantage is still gained from having all nodes able to be an active part of the same cluster

### ***Fast Recovery***

Prior to Oracle 10gR2 the biggest advantage of Oracle RAC on Extended Distance Clusters is that when a site fails, it is possible to recover quickly with no manual intervention needed. With Data Guard, when the primary site fails, failover is generally manually instantiated. In Oracle Database 10gR2, Fast-Start Failover was introduced as an Oracle Data Guard 10g Release 2 feature that automatically, quickly, and reliably fails over to a designated, synchronized standby database in the event of loss of the primary database, without requiring manual intervention to execute the failover. This also requires a third arbitrating site.

Now in the event of server failures, both Oracle RAC and Oracle Data Guard with Fast-Start Failover can accomplish the failover in a few seconds, requiring no manual intervention.

### ***Costs***

The biggest benefit of Extended Distance Clusters is in its potential to reduce costs. By being able to have all nodes active, it is possible to get scalability, very high availability and DR with just 2 nodes.

While one could get away with just one mirror copy of the data at each site, this would be a risky proposition when one site becomes unavailable. Local array mirrored copies should be kept at each site, totaling 4 copies of the data (same as w/ Oracle RAC + Data Guard).

Cost increments can be incurred by the higher bandwidth and specialized communication needs of an extended distance cluster environment. Additional costs can come from reduced performance, and the potential need to implement a third site<sup>7</sup> for the quorum disk.

---

<sup>7</sup> This can be negligible for large corporations with multiple locations, especially considering the third site can be accessible via a WAN.

## **Strength of local Oracle RAC + Oracle Data Guard at a remote site**

### ***Less Impact from Distance***

A Data Guard environment can be setup to be asynchronous, which allows data to be transferred across a great distance and still have from none to a minimal impact on the performance of the primary environment. Of course in an asynchronous configuration you no longer have the guarantee of zero data loss.

With RAC, the sites are much more tightly coupled, thus any latencies involved have a greater affect because of the separation of the two sites. Furthermore, the latency affects the data transfer between caches. Data Guard only sends redo data, and thus is less sensitive to network latency.

### ***Potential Greater Disaster Protection***

An Extended Distance Cluster scenario does not provide full disaster recovery (DR) as distance between the sites is limited.

In DR design it is important to avoid common utility failure (water, electricity), being on the same flood plain, or being part of a larger location that can all be damaged by the same jumbo jet. In an earthquake zone the general recommendation is 300 km at right angles to the main fault line. Hurricanes and wars can take out even larger areas. Terrorism brings more unpredictable effects.

So, for example, if the two sites are in a non-flooding non-earthquake zone, not under a flight path and each has independent automatic standby generators and self-contained cooling then 1Km may be ample except perhaps in times of war, terrorism, hurricanes, etc.

### ***Costs***

A Oracle RAC approach with only Oracle Data Guard at the remote site requires less network bandwidth and these networks do not need to be as redundant or with such extreme low latencies they would need for a Oracle RAC environment on Extended Distance Clusters.

## CONCLUSION

Oracle RAC on Extended Distance Clusters is an attractive alternative architecture that allows scalability, rapid availability, and even some very limited disaster recovery protection with all nodes fully active.

This architecture can provide great value when used properly, but it is critical that the limitations are well understood. Distance can have a huge effect on performance, so keeping the distance short and using costly dedicated *direct* networks are critical.

While this is a greater HA solution compared to local Oracle RAC, it is not a full Disaster Recovery solution. Distance cannot be great enough to protect against major disasters, nor does one get the extra protection against corruptions and flexibility for planned outages that a Oracle RAC and Oracle Data Guard combination provides.

While this configuration has been deployed by a number of customers, thorough planning and testing is recommended before attempting to implement.

## APPENDIX A: DETAILED QUORUM EXAMPLES

Clusters are designed so that in the case of a failure of communication between any 2 subsets of nodes of the cluster, at most one sub-cluster will survive and thus avoid corrupting the database.

The “At Most” in the last phrase is key. If the clusterware cannot guarantee after a failure that only one sub-cluster will survive, then all sub-clusters go down. You cannot assume that sub-clusters will be able to talk to each other (a communication failure could be the cause of needing to reform the cluster).

How the clusterware handles quorum affects how one should layout an extended cluster. Some cases require a balanced number of nodes at each of the 2 main sites, while all cases require a third site to locate the tie-breaking device for higher availability.

The following examples will help you to understand the details of these restrictions, as well as get a better understanding of how quorum works.

### **Oracle Clusterware example:**

The following example applies to when only Oracle Clusterware is used (i.e. when a third party clusterware is not used in conjunction with Oracle Clusterware).

By design, shared disk cluster nodes have 2 ways to communicate with each other, thru the interconnect network and shared disk sub system. Many vendor's clusterware monitor cluster availability only based upon the network heartbeat, but depend upon SCSI timeouts for detecting disk failures to one or all nodes, these timeouts can take up to 15 minutes.

Oracle Clusterware uses the concept of a voting disk and a heartbeat to monitor the cluster through both the disk subsystem and the interconnect. This helps Oracle Clusterware to resolve asymmetric failures very effectively without resorting to SCSI timeout mechanisms.

This method of using the voting disk actively helps protect against heterogeneous failures (where one node sees the cluster as being fine but others do not) but it also means that the ‘voting disk’ must be accessible at all times, from all nodes or the cluster will fail, and the location of ‘voting disk’ will make that site primary.

The ‘voting disk’ file should be mirrored locally for high availability purposes. Multiple voting disks setup via Oracle Clusterware are not mirrors, but members of a group for which you need to achieve a quorum to continue. Thus it is OK to mirror voting disks locally (as on RAID 0+1). They should *not* be mirrored remotely as part of an extended cluster as this could allow two sub-clusters to continue working after a failure and potentially lead to a split-brain or diverging database situation.

In the case of stretched clusters with stretched storage subsystems, a full site outage (where both network access and storage access to the voting disk are lost simultaneously) could result in a complete cluster outage at both sites if misscount is left at its default value. This will only happen if the storage takes longer than misscount seconds to reconfigure.

***HP ServiceGuard / Sun Cluster example***

Quorum is achieved here by giving each node a vote, and a quorum device (normally a disk or server) acts as a tiebreaker to make sure only one side gets the majority.

***Veritas DBE/AC example:***

With Veritas DBE/AC, nodes don't get votes but instead all nodes race for access to 3 coordinator disks. Because of the algorithm used, larger subsets of nodes will get to the coordinator disks quicker thus are more likely to survive.

In a 2-site environment, one would not want both sides to survive as this would quickly cause corruptions. Therefore one side must be able to form a quorum, and the tie breaking vote must exist on one side or the other. This ends up creating a primary and a secondary site. Should the primary site fail, the secondary site will not have a quorum and will shut down. In this case a manual reconfiguration is required and this should be practiced and well rehearsed.

In a 3-site implementation, quorum can be redistributed so that any 2 sites left can have a majority of votes or coordinator disks to ensure that the cluster survives.

## **APPENDIX B: CUSTOMERS USING ORACLE RAC ON EXTENDED DISTANCE CLUSTERS**

Extended RAC actually has a long history. The Rover Group completed the first known implementation with a similar architecture in the mid 1990's using Oracle7 Parallel Server. Since then other clients have implemented it with Real Application Clusters.

Extended Clusters started being widely adopted by customers in Oracle9i using various clusterwares and mirroring mechanisms. Today the majority of new customers implementing are doing so using ASM to mirror the data between the sites, and solely Oracle Clusterware for clustering.

The following are some overall stats taken from a survey of over 100 extended Oracle RAC customers running on Oracle Database 10g onwards. These show the wide wide variety of adoptions and usages for this architecture.

**Distance:** Most customers are under 25km, but some customers go up to 90km. Shortest distance is 200m.

**OS:** AIX, Solaris, HP-UX, Linux, Windows

**Platform:** Dell, Fujitsu, HP, IBM, Sun, Other

Both IBM and HP have done heavy investments to help customers do extended Oracle RAC implementations. This is reflected in adoption rates by customers on these platforms.

**Nodes:** 2 to 36

**Countries:** Australia, Austria, Belgium, Canada, Chile, Czech Republic, Finland, France, Germany, Greece, Hong Kong, India, Israel, Italy, Japan, Luxembourg, Netherlands, Saudi Arabia, South Africa, Sweden, Switzerland, United Kingdom, United States, Vietnam

2/3 of Extended Oracle RAC implementations are in Europe where a short distance can make quite a difference in electric networks, flood plains, etc.

**Applications:** Business Objects, Calidris, Custom , IBM, Inforsense, Interchain Interflex, J.D. Edwards, McKesson, Oracle EBS, Oracle Portal, Peoplesoft, SAP, Siebel

## **APPENDIX C: USAGE OF SEPARATE SUBNETS FOR THE PUBLIC NETWORK**

At this time separate subnets are NOT supported for extended RAC environments

A single subnet on all nodes is required for the private interconnect.

A single subnet is also required for the 'public' network as this best simulates a local cluster and allows full functionality to be provided. This configuration will require both the Public network and Interconnect network to be bridged between the two sites.

The following are a some reasons why separate subnets are not supported

1. Database Services cannot run across multiple subnets. This is due to existing dependency architecture of Services to VIPs in Oracle Clusterware 11g Release 2. Database Services are essential for environment flexibility, workload flexibility, etc.
2. Applications may not recover as quickly from node outages. VIPs are used to accelerate the response to failures by providing an immediate negative response (no-ack). But while the VIP resource will fail across the subnet, routers will not forward traffic to it as they change their routing tables. Unless clients receive the out of bound event provided by Fast Application Notification (FAN) for rapid reaction, they will have to wait for TCP/IP timeouts which can cause many minutes of 'hung' clients even after the database tier has recovered.
3. Similarly SCAN VIPs will only run on one side. If it is down, they won't run. Stretch clusters with separate subnets can't use SCAN name for client connections, they have to use node VIP address lists.
4. The Oracle Installer, is not aware of different subnets. Some of steps of configuration may need to be done while the nodes are all on one subnet and then modified via workarounds afterwards.
5. Policy Managed Databases, new in Oracle RAC 11g Release 2, may not work across separate subnets. This would remove some of the introduced advantages for large, dynamic clusters.

## REFERENCES

**Roland Knapp, Daniel Dibbets, Amit Das, Using standard NFS to support a third voting file for extended cluster configurations (PDF)**, December 2009

**HP-Oracle CTC, Building a disaster-proof data center with HP Extended Cluster for RAC**, 2007

**Jakub Wartak, Build Your Own Oracle Extended RAC Cluster on Oracle VM and Oracle Enterprise Linux**, 2008

**EMEA Joint Solutions Center Oracle/IBM, 10g RAC Release2 High Availability Test Over 2 distant sites on xSeries**, July 2005

**Paul Bramy (Oracle), Christine O'Sullivan (IBM), Thierry Plumeau (IBM) at the EMEA Joint Solutions Center Oracle/IBM, Oracle9i RAC Metropolitan Area Network implementation in an IBM pSeries environment**, July 2003

**Veritas, VERITAS Volume Manager for Solaris: Performance Brief – Remote Mirroring Using VxVM**, December 2003

**Mai Cutler (HP), Sandy Gruver (HP), Stefan Pommerenk (Oracle) Eliminate the Current Physical Restrictions of a Single Oracle Cluster**, OracleWorld San Francisco 2003



Oracle Real Application Clusters (RAC)  
on Extended Distance Clusters

August 2010

Author: Erik Peterson

Reviewers: Kevin Reardon,  
Markus Michalewicz

Prior Version Reviews: Daniel Dibbets, Bill  
Bridge, Joseph Meeks

Oracle Corporation  
World Headquarters  
500 Oracle Parkway  
Redwood Shores, CA 94065  
U.S.A.

Worldwide Inquiries:  
Phone: +1.650.506.7000  
Fax: +1.650.506.7200

oracle.com



Oracle is committed to developing practices and products that help protect the environment

Copyright © 2011, Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. UNIX is a registered trademark licensed through X/Open Company, Ltd. 1010

**Hardware and Software, Engineered to Work Together**