



ORACLE

# The New Generation Oracle RAC

August, 2024, Version 1.2  
Copyright © 2024, Oracle and/or its affiliates  
Public

TABLE OF CONTENTS

<b>Executive Overview</b>	<b>3</b>
<b>High Availability &amp; Scalability as Part of the Database Design</b>	<b>4</b>
Oracle RAC Components Overview	4
Oracle RAC Scale-out Architecture	5
Oracle Clusterware	6
Oracle ASM	6
Application Continuity	7
Fleet Patching and Provisioning	7
A New Generation of Oracle RAC	8
Oracle Real Application Clusters 19c	8
<b>Engineered Systems – Designed with Oracle RAC in Mind</b>	<b>9</b>
Exadata Scale-out Architecture	10
Exadata Extensions Benefiting Oracle RAC	10
<b>Extending Oracle RAC Benefits Into the Oracle Cloud</b>	<b>11</b>
Oracle RAC and Autonomous Database	11
<b>Comparison of Oracle RAC to other Scale-out Database Approaches</b>	<b>13</b>
<b>Conclusion</b>	<b>15</b>

## EXECUTIVE OVERVIEW

The success of enterprises worldwide depends on *mission-critical applications that run* quickly and reliably. Businesses suffer or stop when these critical applications suffer from availability or scalability issues. An application's availability and responsiveness depend upon the reliability and responsiveness of the underlying database. Oracle has invested thousands of years in engineering development and enhancement of its Real Application Clusters (Oracle RAC) technology to enable the transparent database availability and scalability that mission-critical applications need. This technology is unique in the industry and is used by most of the leading enterprises worldwide to run both their internal and critical customer-facing applications.

This white paper describes Oracle RAC and how it meets the essential requirements of critical applications:

1. **High availability** – A database supporting critical applications must continuously service application requests, even in the face of hardware or software failures. It must also enable performing planned maintenance at all levels, including hardware, OS, database software, and database schema, without disrupting the application.
2. **Workload scalability** – A database supporting critical applications must dynamically adapt to increasing application workloads without disruption. It must transparently scale compute, storage, memory, connections, users, and application complexity.
3. **No application changes** – A database supporting critical applications must meet the above requirements without requiring changes to the application. Enterprises have invested thousands of engineering years in developing their applications, and rewriting existing applications is simply not cost-effective.

Oracle RAC has long been the premier technology to address these essential requirements. It provides near-linear scalability and availability across up to one hundred nodes in a cluster without requiring changes to application code. Combined with Oracle Exadata Database Machine, Oracle RAC databases successfully run the world's largest and most demanding OLTP and Analytics workloads.

Oracle Database 19c implements significant improvements in the RAC architecture that enable applications to achieve many times better availability and scalability than just a few years ago. Together, these improvements create a new generation of Oracle RAC technology ready to meet the needs of next-generation workloads and applications.

HIGH AVAILABILITY & SCALABILITY AS PART OF THE DATABASE DESIGN

Oracle has architected and integrated high availability and scalability concepts into the design of the Oracle Database from day one. This includes providing an ACID compliant database, sound backup and recovery, a replication solution (Oracle Data Guard) as well as providing Oracle RAC as a local high availability and scalability solution.

Integrating high availability and scalability into the database design, especially with Oracle RAC, enabled a never before seen ability to scale any feature of the converged Oracle Database<sup>1</sup> and any database application without changes as the examples below show:

- Oracle RAC scales all Oracle Database features and architectures:
  - Oracle Multitenant Databases as well as non-CDB databases
  - Oracle Parallel SQL to accelerate analytics and batch
  - Oracle Database In-Memory across instances in the cluster
  - Oracle Data Guard using Multi-Instance-Redo-Appl (MIRA)
- Oracle RAC scales complex OLTP, DWH and analytics workloads:
  - SAP, Oracle EBusiness Suite, Peoplesoft, Siebel and many other business applications commonly run on top of RAC
  - More than 15,000 customer workloads are successfully using Oracle RAC
- Oracle RAC provides best-in-class high availability for 24/7 systems:
  - Most Fortune 1000 banks, telecoms, airlines, and e-commerce companies use RAC for their critical applications
  - RAC provides continued database availability through most hardware and software failures and maintenance events

The remaining part of this paper describes how Oracle RAC uses a combination of function shipping and distributed caching to achieve industry leading scalability and availability and how the latest version of Oracle RAC has been improved to provide high availability and scalability for a *new generation of applications and DBAs* to come.

**Oracle RAC Components Overview**

Oracle RAC implements an application transparent scale-out database architecture in which an application connects to any database instance (defined as a collection of processes and memory on a server that are used to execute the Oracle database) that is part of a cluster of up to one hundred databases instances hosted on different servers. All servers must have access to a shared storage system that holds the actual data (the set of database files). The servers must be connected via a dedicated high-speed network, which is commonly referred to as the RAC Interconnect.

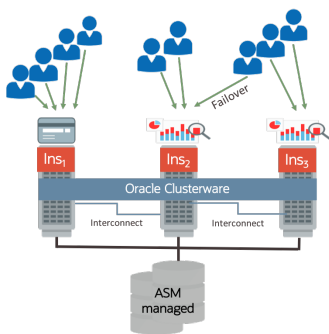


Figure 1: Simplified Oracle RAC Architecture Overview

<sup>1</sup> For more information about Oracle’s Converged Database Architecture, see: <https://youtu.be/9d76-LhgMOs>

The interconnect is a key element of the architecture as it helps to ensure that the scale-out Oracle RAC database is seen as a “single system” from the application perspective, in which all database instances are able to access and update any data in a coordinated and coherent fashion. This means an application does not need to be aware of which database instance is processing a workload request. In addition, Oracle RAC utilizes five key software components:

1. **Scale-Out Database Instances.** It is not sufficient that multiple database instances have concurrent access to database data; they must have the intelligence for coordinated access and updates. This intelligence within the database layer is provided by Oracle’s unique **Cache Fusion technology**, which is the algorithm that implements coherent distributed caching of data across the nodes of the cluster for efficient horizontal scaling.
2. **Oracle Clusterware** transforms a collection of servers into a high availability cluster. From a management perspective, the cluster is managed as a single entity as opposed to a pool of independent servers. Additionally, Oracle Clusterware provides failure detection, failover, and node membership management for the cluster.
3. **Automatic Storage Management (ASM)** scales storage by enabling concurrent access to database files from all of the individual Oracle RAC database instances. Additionally, Oracle ASM provides volume management capabilities, including optimized database file access and data mirroring to protect from storage failures.
4. **Application Continuity (AC)** enables automatic and transparent failover of active application connections to a surviving instance by performing a replay of partially executed in-flight requests and transactions in a non-disruptive and rapid manner.
5. **Fleet Patching and Provisioning (FPP)** greatly automates and simplifies one of the former Oracle RAC challenges: patching and provisioning. Oracle Fleet Patching & Provisioning (formerly known as Oracle Rapid Home Provisioning) orchestrates provisioning, patching, and upgrade to maximize database availability to applications.

## Oracle RAC Scale-out Architecture

There are two fundamental ways a database system can be scaled horizontally across servers: 1) by bringing data to the server running the SQL that wants to access the data (sometimes called data shipping or shared disk), 2) by sending SQL to remote servers to access the data locally (sometimes called function shipping or shared nothing). Oracle RAC is often incorrectly described as a data shipping shared disk architecture, whereas Oracle RAC really uses a combination of these two approaches to deliver a best-of-both-worlds scale-out architecture.

Oracle RAC uses function shipping (shared nothing) to execute long running analytic and batch SQL. RAC parallelizes long running SQL automatically by breaking up SQL statements that operate on large amounts of data into SQL fragments that are run in parallel across the database instances in a cluster. Unlike traditional shared-nothing databases, Oracle RAC instances are not confined to specific subsets of database data since they can read any data from storage. Therefore, RAC is able to dynamically distribute workload across the nodes of a cluster, avoiding large skews where one slow instance slows down the entire operation.

While function shipping works well for analytics, it works poorly for OLTP. OLTP requests generally access a small number of rows and require very fast response times. It is very slow and expensive to send an RPC call to a remote node to access one row or traverse one index. Also, every time the same data is accessed by the application, the request must be re-sent to the remote node since the data may have changed since the last request. Further, sending requests to the instance where the data resides requires running complex and expensive distributed transaction and 2-phase commit protocols to coordinate the instances. In addition, shared nothing does not work for complex applications like ERP, CRM, HCM, due to its inability to scale to many thousands of tables and indexes. These flaws have caused the transparent shared nothing databases approach to be abandoned as an OLTP solution, even by its traditional advocates. There are no shared-nothing architectures today running complex OLTP applications like ERP, CRM, and HCM.

Oracle RAC executes short running OLTP SQL directly in the instance that receives the application request. Data requested by OLTP SQL that is already cached locally in the instance is instantly accessed directly from memory. Data that is not local to the instance is brought into the instance from storage or a remote instance using high-performance data transfer protocols and then *cached locally*. By caching the data locally, future accesses to the same data are dramatically accelerated and messaging is avoided. Caching is a well-known and highly proven approach that often eliminates 90% to 99% of remote access and greatly improves response time. Due to transactions running locally to an instance, there is also no need for expensive distributed transactions or 2-phase commit protocols.

In Oracle RAC, the application does not need to be aware of where the data resides (in the memory of one of the database instances or on disk), as the RAC protocols automatically bring the data to the instance to which the application is connected (location transparency of data). The algorithm that implements the high-performance distributed caching protocol while maintaining full data consistency is called Oracle Cache Fusion.

Cache Fusion leverages the private cluster interconnect to provide shared access to data cached on any of the Oracle RAC database instances, virtually creating a single “fused” cache across the whole cluster. Cache Fusion implements direct memory-to-memory transfers of data blocks across instances for high-performance. Oracle RAC database instances only need to read data from storage if the block is not already present in the combined caches across the entire Oracle RAC configuration. Cache Fusion also implements locking protocols to prevent multiple instances from updating the same data block simultaneously, ensuring consistency while providing optimal concurrent data access across the Oracle RAC cache distributed across independent instances in the cluster.

By using a combination of function shipping, data shipping, and distributed data caching, Oracle RAC is uniquely able to optimize analytics, OLTP, batch or any combination of these workloads. Oracle has recently implemented significant enhancements to the Oracle RAC algorithms, further enhancing their performance.

## Oracle Clusterware

Oracle Clusterware is the technology used in the RAC architecture that transforms a collection of servers into a highly available unified system. Oracle Clusterware provides failure detection, node membership, node fencing and optimal resource placement. It provides cluster-wide component inter-dependency management for RAC and other applications in the cluster. Clusterware uses resource models and policies to provide high availability responses to planned and unplanned component downtime.

For more information on Oracle Clusterware visit <http://www.oracle.com/goto/clusterware>

## Oracle ASM

Automatic Storage Management (ASM) is a file system and volume manager integrated into the database that enables sharing and scaling of storage capacity across thousands of storage devices with 24/7 availability. ASM provides storage management across all servers of a cluster for Oracle RAC databases. It stripes data to prevent hot spots and to maximize I/O performance. ASM allows the online addition or removal of storage capacity. ASM can maintain redundant mirror copies of data to provide fault tolerance, or it can be used on top of vendor supplied reliable storage arrays. Data management is done by selecting the desired reliability and performance characteristics for classes of data rather than with human interaction on a per database file basis.

ASM solves many of the practical management problems of large clustered databases. As the size of a database and associated data increases towards thousands of storage devices, or dozens of servers, the traditional techniques for storage management do not scale efficiently and become prone to human error. Other tasks, such as manual load balancing, also become highly complex. Oracle ASM solves these issues for Oracle RAC as well as for Oracle single instance databases. For more information on Oracle ASM visit <http://www.oracle.com/goto/asm>

## Application Continuity

Oracle RAC is an active-active database management system in which the database service remains continuously available, even after a RAC database instance or server failure. This is in contrast to technologies utilizing an active-passive configuration that must restart and resume processing after the active database instance failed, normally leading to a failover followed by a brownout that can last for many minutes. In both scenarios, SQL and any uncommitted transactions that are being processed at the point of failure are aborted and rolled back by the database management system. Complex application logic must be written for every application to handle these failures. Since failures are rare, however, this logic is often ignored or rarely tested, leading to application failures and the potential for logically corrupted data.

Utilizing Application Continuity as part of Oracle RAC enables automatic and transparent failover of active application connections to a surviving instance. Application Continuity performs replay of partially executed in-flight requests and transactions in a non-disruptive and rapid manner. After a successful replay, the application transparently continues as if the failure never happened. Utilizing Application Continuity, application developers do not have to worry about protecting the application from accidentally executing a transaction twice, nor do they need to code complex error handling and retry or worse, find ways to recover failed requests manually. With Application Continuity, applications become much more robust to failures, and are easier and faster to code.

For more information on Oracle Application Continuity visit <http://www.oracle.com/goto/ac>

## Fleet Patching and Provisioning

In the past, provisioning and patching Clusterware, ASM, and RAC databases was complicated. Oracle's Fleet Patching and Provisioning (FPP) greatly automates and simplifies this process. Oracle Fleet Patching & Provisioning (formerly known as Oracle Rapid Home Provisioning) orchestrates provisioning, patching, and upgrade to maximize database availability to applications.

FPP is fully integrated with the Oracle RAC architecture. FPP maintains a space-efficient repository of standardized software homes (gold images) that can be provisioned to any number of target machines. Any number of homes can be provisioned from a given gold image, and FPP maintains lineage information so that the provenance of deployed software is always known.

These and additional features make FPP the ideal fleet software patching system to be used on-premises and in the Oracle Cloud. FPP is used by Autonomous Database to fully automate the patching process. FPP is part of the Oracle RAC license, so on-premises Oracle Databases get the same benefits free of additional charge.

For more information on FPP see: <https://www.oracle.com/database/technologies/rac/fpp.html>



## A New Generation of Oracle RAC

Oracle has continuously enhanced Oracle RAC technology, making it easier, faster, more scalable, and more available culminating in the breakthrough Oracle RAC 19c release that enables RAC to meet the needs of all next generation workloads and applications.

### Oracle Real Application Clusters 19c

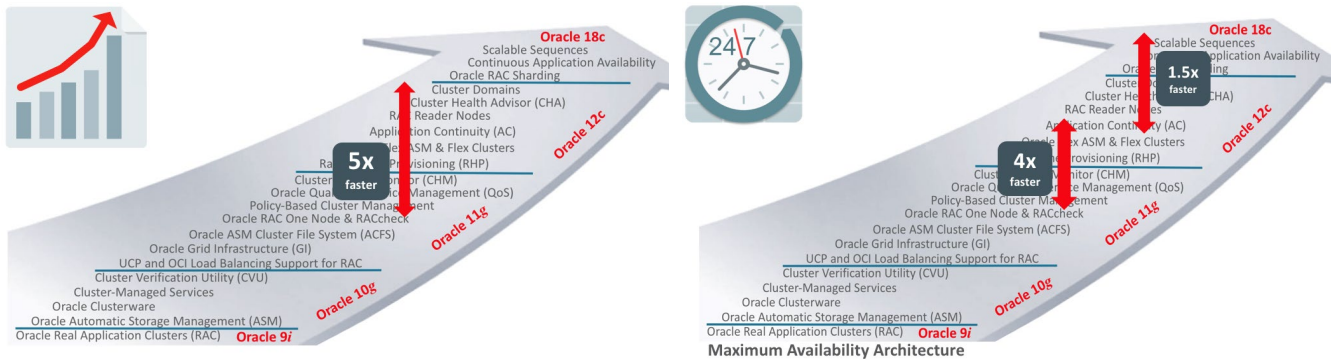


Figure 2: Better Scalability (left) and High Availability (right) with Oracle RAC 19c

Figure 2 illustrates why Oracle RAC 19c is the most scalable and reliable Oracle RAC version ever. Oracle RAC has improved its performance for high contention workloads by 5 times since Oracle RAC 11g Release 2 while the brownout time during cluster-reconfiguration and after an instance failure has been reduced by 4 times between Oracle RAC 11g Release 2 and Oracle RAC 12c Release 2 and by yet another 1.5 times between Oracle RAC 12c Release 2 and Oracle RAC 19c. This means that an application which experienced a brownout time of one minute with Oracle RAC 11g Release 2 may find itself fully operational after a failure in less than 15 seconds with Oracle RAC 19c.

Recovering from a RAC database instance or server failure requires a cluster reconfiguration. During a cluster reconfiguration access to data that was open for modification at the time of failure may be delayed until instance recovery has been performed. Instance recovery for the failed instance in a RAC database is performed by one of the remaining instances. In Oracle RAC versions before Oracle RAC 12c Rel. 2, this recovery process entailed five steps, including the election of the database instance to perform the recovery after a failure as well as standard redo log read and apply.

Starting with Oracle RAC 12c Release 2, two out of those five steps, among other improvements, have been fully eliminated by the Recovery Buddies feature, which eliminates the need to elect the instance that is performing the instance recovery by pre-assigning the recovering instance whenever a new instance starts up. In addition, a new process has been introduced that continuously sends redo log recovery data directly over the interconnect to the assigned Recovery Buddy instance. The Buddy instance then stores this information in-memory to eliminate the need to read redo logs when performing recovery, reducing the time to re-enable full data access for new transactions.

The 5 times improvement in performance for high contention workloads cannot be explained by referring to a few features only; instead, a variety of features shown in figure 3 below has contributed to this important improvement. Explaining how each of the performance enhancing features listed in figure 3 has helped and contributed to improving the performance for high contention workloads is beyond the scope of this paper.

Most applications, including complex applications such as SAP, Oracle’s EBusiness Suite, Siebel and generally most OLTP as well as analytic workloads on an Oracle RAC system will achieve near linear scalability assuming that they scale vertically on a single instance system. More than 20 years of running the most complex workloads across all industries around the world has proven this fact.



If an application shows a scalability rate of less than 80%, this application is likely subject to a performance-inhibiting contention. The causes for such contention are one of the following three or a combination thereof:

1. The application performs frequent transactional changes to the same data blocks (“write hot spots”).
2. Distribution of new block changes to other instances is slowed down by the need to write the redo log (“redo log latency”).
3. The application causes contention on metadata (“right growing index and index contention”).
  - Most OLTP write hot spots occur on indexes.

Oracle RAC 19c uses a variety of features that significantly reduce the performance impact of these sources of contention.

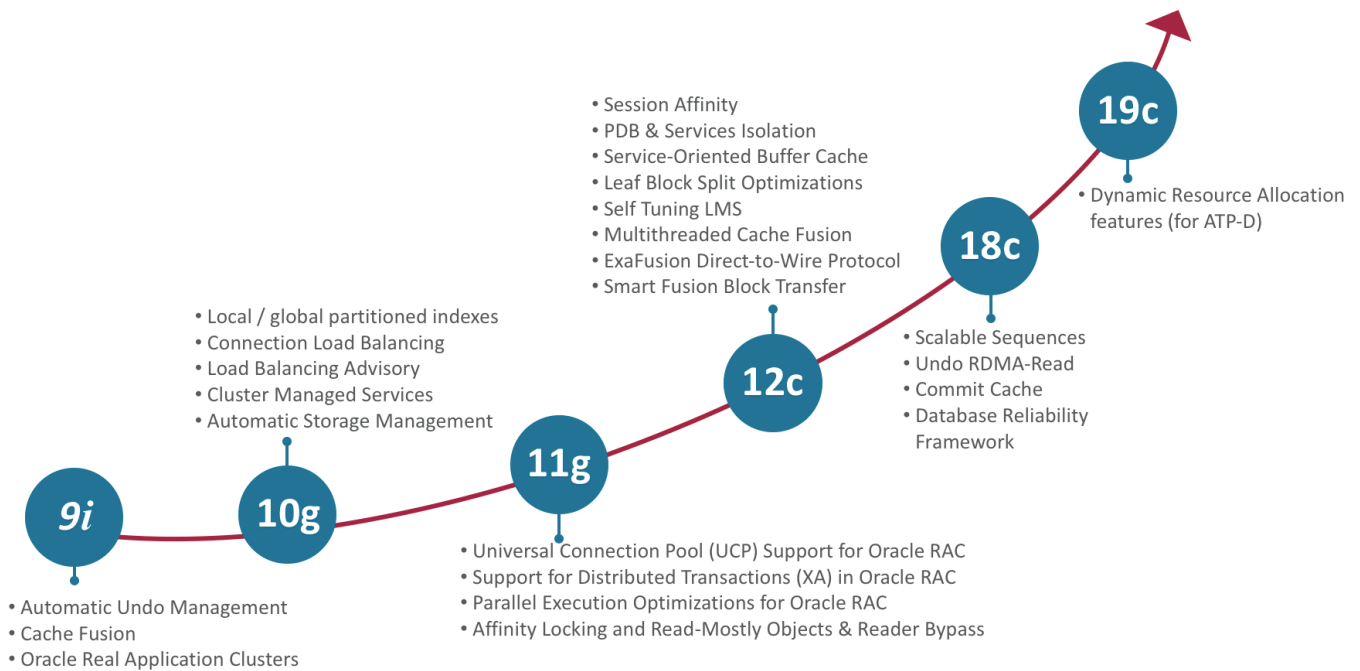


Figure 3: Oracle RAC Performance Enhancing Features Across Releases

### ENGINEERED SYSTEMS – DESIGNED WITH ORACLE RAC IN MIND

Over ten years ago, Oracle embarked on a new strategy for delivering high-performance and robust database infrastructure commonly referred to as *Oracle Engineered Systems*. An Oracle Engineered system is a pre-configured, pre-tuned, and pre-tested integrated system of servers, networking and storage, optimized around the Oracle Database. Two of the most important engineered system offerings are Exadata Database Machine (Exadata) and the Oracle Database Appliance (ODA). ODA is the simpler of the two systems and principally designed to simplify RAC deployments. ODA hardware is RAC-ready out of the box and ODA tooling orchestrates deployment and patching of the operating system, firmware, Oracle Clusterware, and Oracle RAC database software. Exadata is a more scalable and high-performance offering providing many capabilities that are not achievable by customers deploying a database on generic hardware systems.

Exadata is Oracle’s preferred platform delivering extreme performance for database applications including OLTP, DWH, Reporting, Batch Processing, Consolidation, and mixed database workloads. Oracle Exadata offers superior price-performance, availability, and supportability. Oracle Exadata frees users from the need to build, test, and maintain systems and allows them to focus on higher value business problems. Furthermore, because Oracle provides and integrates all elements of an Exadata Database Machine, Oracle is able to develop many unique software features to achieve benefits unavailable on generic platforms.

## Exadata Scale-out Architecture

Oracle Exadata Database Machine uses a scale-out architecture for both database servers and storage servers. As an Oracle Exadata Database Machine grows, more database CPUs, storage and networking are added in a balanced fashion, ensuring scalability without bottlenecks. Exadata complements the scale-out database compute provided in RAC with scale-out compute in the storage tier to accelerate data intensive operations. Storage compute scale-out works seamlessly and transparently to applications.

The Oracle Exadata architecture is purpose built for the Oracle Database management system. Over many Oracle Database releases, more and more features and functions have been added to the core database to leverage the rich capabilities of Exadata's architecture. Some examples of the synergistic features used on an Oracle Exadata Database Machine in conjunction with the Oracle Database include: storage indexing, SQL offload processing by storage servers, Hybrid Columnar Compression, and Smart Cache Flash. While these features benefit single instance databases as well as Oracle RAC databases, there are several enhancements to the database and Exadata that provide unique benefits for Oracle RAC databases.

## Exadata Extensions Benefiting Oracle RAC

- Exadata Instant Failure Detection

An Oracle RAC system consisting of independent servers operating as a coordinated cluster must be able to quickly detect and recover from the failure of servers in the cluster. For this reason, Exadata has incorporated several enhancements that enable node failure detection and allow reconfiguration to happen in as short as two seconds. The ability to engineer this capability is the result of Oracle controlling the end-to-end database and networking infrastructure in Exadata.

- Cache Fusion Enhancements

Oracle RAC performance is dependent on the speed of the underlying communication. Oracle Cache Fusion on Exadata benefits from the low-latency and high-bandwidth cluster interconnect that Exadata provides. Further, Exadata implements the Exafusion network protocol to enable Oracle processes to bypass the operating system kernel and directly perform RDMA reads and writes between database servers. This dramatically improves performance as the processes do not suffer from the overhead of traversing the network stack and incurring context switches.

Another Cache Fusion optimization available exclusively on Exadata is the Exadata Commit Cache. This feature is an in-memory cache that records transaction commit times using RDMA protocols. A RAC database instance must ensure data consistency of an ongoing transaction, even though another database instance is actively operating on the same data. The optimization used on Exadata utilizes a commit cache to quickly determine if the first database instance needs to read any UNDO data. With some workloads, data block traffic for UNDO data is reduced by 60 percent as a result of this feature.

- In-Memory Fault Tolerance

Oracle Database In-Memory provides in-memory columnar and vector processing database functionality to existing databases and transparently accelerates analytics by orders of magnitude while simultaneously speeding up mixed-workload OLTP. Deploying Oracle Database In-Memory with existing databases, including Oracle RAC databases, is simple as there are no requirements for application changes.

If a database instance fails in a generic Oracle RAC configuration, the in-memory data on that instance becomes unavailable. While queries can continue to run on surviving database instances, it takes time to repopulate the in-memory data from storage and during this time analytic queries will run much slower. To address this, Exadata provides a fault-tolerant version of Oracle Database In-Memory that eliminates this slowdown by optionally duplicating data across instances in a RAC cluster. Just as storage subsystems stripe and mirror data across disks to achieve high performance and high availability, Oracle Database In-Memory also distributes and duplicates in-memory data across RAC instances in an Exadata Database Machine. Consequently, if a server or database instance fails, in-memory queries can transparently use the duplicate copy of data on surviving servers.

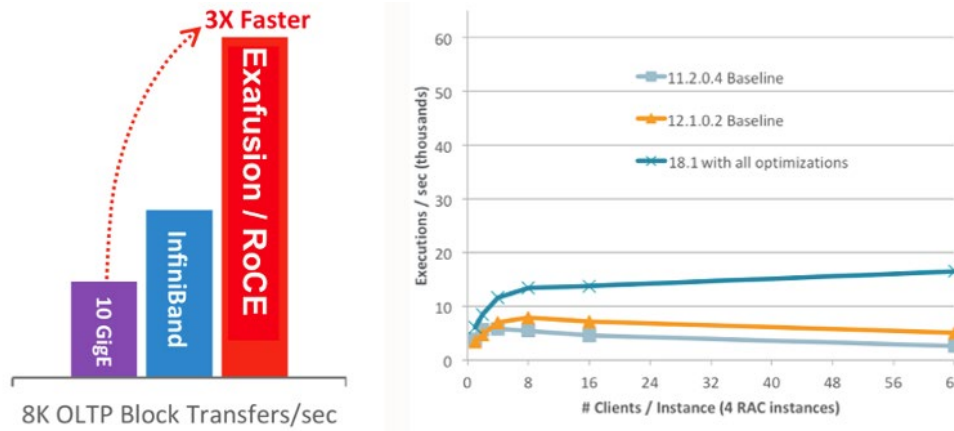


Figure 4: Overview of Exadata-based Oracle RAC Improvements and Their Impact on Performance

### EXTENDING ORACLE RAC BENEFITS INTO THE ORACLE CLOUD

Oracle RAC is an important element in Oracle’s cloud offerings. Oracle RAC is available in the Oracle RAC Database Cloud Service (DBCS), Oracle Exadata Service, and Autonomous Database. Oracle RAC is also part of the underlying database infrastructure for Oracle SaaS. Customers can run Oracle RAC Databases in the Oracle Cloud with the same extreme performance and availability experienced by thousands of organizations deploying Oracle RAC on-premises. These database services are 100% compatible with databases deployed on-premises, ensuring a smooth transition to the Oracle Cloud and an efficient hybrid cloud strategy.

When customers deploy an Oracle RAC configuration in the Oracle Cloud, all the underlying components of the RAC infrastructure, including Clusterware and ASM are utilized, just as for on-prem deployments. The difference is that the Oracle Cloud deployment of Oracle RAC is highly automated and completes within minutes.

Oracle is the only cloud vendor to create an infrastructure that fully supports Oracle RAC. These steps include the efficient sharing of storage between virtual machines and providing an Oracle RAC-compatible network infrastructure. For these reasons, Oracle RAC is only supported on Oracle’s cloud. Please, see MOS note 2093394.1 for more details about deploying RAC on Cloud Infrastructure.

## Oracle RAC and Autonomous Database

In order to provide the high availability and scalability that mission critical applications need, Oracle has chosen Oracle RAC as the foundation for its flagship database cloud service – the Autonomous Database Service, as shown in figure 5 below.

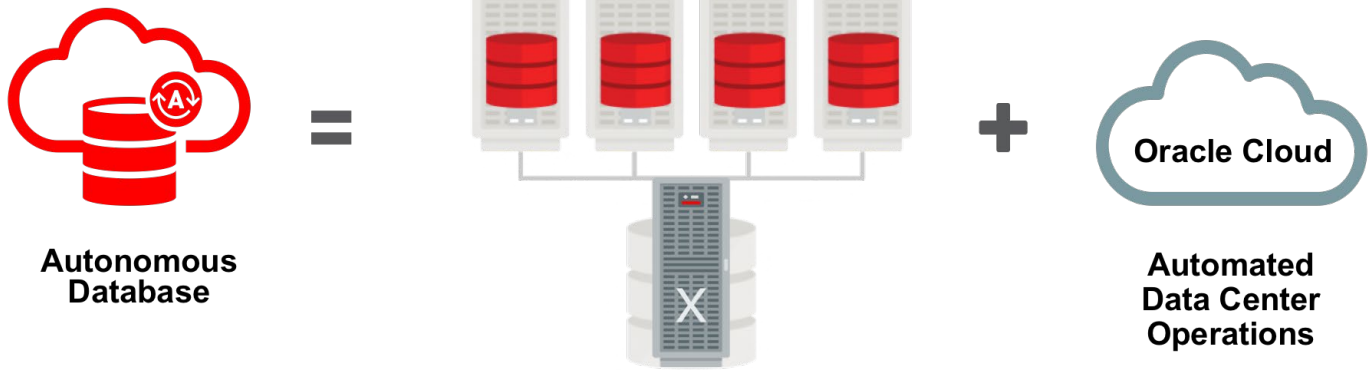


Figure 5: Oracle RAC is the Foundation for the Autonomous Database

Oracle RAC on Exadata provides transparent scale-out for workloads run on Autonomous Database. All aspects of configuring and operating Oracle RAC are fully automated. Databases that run using significant numbers of CPUs are automatically configured to run across multiple servers in the Oracle RAC cluster. RAC rolling patching and Application Continuity are used to avoid application downtime during server, OS, Clusterware, and database maintenance. Application modules that would cause contention if spread across the nodes of a cluster can be configured to run on the same instance using *collocation tags* in the connect string.

COMPARISON OF ORACLE RAC TO OTHER SCALE-OUT DATABASE APPROACHES

Given the number of databases available on the market today and the variety of high availability and scalability solutions that can be used with those databases, the question arises, why use Oracle Database with Real Application Clusters as the solution of choice? The answer to this question is simple: Oracle uniquely combines the best elements of all existing scalability and availability architectures to meet the needs of any type or scale of application.

At a high level, database high availability and scalability solutions can be categorized using the approaches listed in Table 1.

Table 1: Database High Availability and Scalability Solutions – High Level Overview

Technology	Advantages	Disadvantages	vs. Oracle RAC
<b>Cluster Failover – database execution is transferred to another server if the current running server fails.</b>	Relatively simple to use and implement.	Simple solution that fails to scale or provide transparent maintenance.	RAC provides a superset of failover cluster capabilities by adding scalability, much faster failover, and transparent maintenance.
<b>Shared-Nothing – database runs transparently across multiple computers that do not share storage.</b>	Scales across multiple nodes without the need to implement shared storage. Works well for analytics workloads.	Poor performance and scalability for OLTP workloads due to high messaging and transaction coordination costs.  Does not work for highly complex applications like ERP, CRM, HCM, due to inability to scale to thousands of tables and indexes.	Combines the best aspects of shared-nothing with the best aspects of shared-storage.  <i>Uses SQL function shipping</i> in a similar fashion to shared-nothing to automatically parallelize long-running SQL across the nodes of a cluster. This allows analytic workloads and batch to transparently scale across nodes.  <i>Uses data shipping and distributed caching</i> to provide excellent response times and transparent scaling for OLTP workloads, even for ultra-complex (ERP) applications.
<b>Database Sharding – logical database is built from multiple physical databases with requests routed based on a sharding key.</b>	Independence of physical databases enables high scalability and fault isolation.	Not transparent to applications.  Only works well for applications that can be partitioned by a sharding key.  Poor or no cross-shard operations.	Oracle Globally Distributed Database implements sharding across multiple physical Oracle Databases delivering highest scalability and fault isolation for applications that can be partitioned using a sharding key.

		<p>Does not work for highly complex (ERP) applications.</p> <p>Poor or no analytics.</p>	<p>Oracle RAC can be used within each sharded data base to provide high availability and planned maintenance for the shard to combine the benefits of both technologies.</p>
<p><b>Master-Slave Replication – data replicated from master database to one or more slave databases.</b></p>	<p>Provides disaster protection, failover, and in some solutions read-scalability.</p>	<p>No write scalability.</p> <p>Usually loses some data on failover.</p> <p>Usually loses data consistency in read-replicas.</p> <p>Poor connection and user scalability.</p> <p>No analytics scalability across replicas.</p>	<p>Combining RAC’s transparent scalability and failover with the disaster protection provided by replication gives the best of both worlds.</p> <p>Oracle Active Data Guard can be used to provide additional read scalability using the replica databases, and, with Oracle Database 19c, transparent updates on the replica database.</p>
<p><b>Active-Active Replication – data is replicated across multiple fully active databases.</b></p>	<p>Very high availability, disaster protection, and good read scalability.</p>	<p>Writes on multiple replicas can conflict and cause data consistency issues.</p> <p>Not transparent to applications.</p> <p>Requires multiple copies of the database (files).</p> <p>No analytics scalability across replicas.</p>	<p>Using Oracle RAC within each replica provides write scalability and analytics scalability and allows fewer replicas which helps avoid conflicts.</p> <p>The Industry leading GoldenGate replication is fully integrated with Oracle RAC.</p>

As can be concluded from this discussion, Oracle RAC uniquely provides high availability and scalability for any combination of OLTP, analytics, and mixed workloads and seamlessly integrates with Oracle’s native Sharding, Data Guard, and replication technologies to provide disaster recovery and even higher scalability and availability. Oracle RAC implements the best attributes of all current database architectures while minimizing or eliminating their drawbacks.

## CONCLUSION

Oracle has invested thousands of engineer years developing and enhancing Oracle Real Application Clusters (RAC) to enable the transparent database availability and scalability that mission critical applications need. This technology is unique in the industry and is used by most of the leading enterprises in the world to run both their internal applications and their critical customer facing applications.

Oracle has recently implemented major improvements in the Oracle RAC architecture that enable applications to achieve many times better availability and scalability than just a few years ago. Together, these improvements create a new generation of RAC technology that is ready to meet the needs of next generation workloads and applications.

Engineered Systems such as Exadata Database Machine, designed with Oracle RAC in mind, further improve the level of high availability and scalability to never seen before levels. At the same time, Engineered Systems simplify the deployment and management of Oracle RAC Systems.

The Oracle Cloud, especially Oracle's Autonomous Database, fully utilizes Oracle RAC as well as Engineered Systems to provide the best database service available on the market. The Oracle Cloud also simplifies the deployment and the management of Oracle RAC, allowing even large deployment estates to be managed simply and efficiently.

Utilizing features such as Application Continuity and Fleet Patching and Provisioning that are part of the Oracle RAC architecture on-premises as well as in the Oracle Cloud further allows those management tasks to be executed fully automatically and most importantly transparently for the application and the user.

Concluding, Oracle RAC databases are ideal for mission critical application environments. The most important requirements for critical applications: high availability, dynamic scalability, and no application changes, are fully met by RAC, especially with Oracle RAC 19c.



## Connect with us

Call **+1.800.ORACLE1** or visit **oracle.com**. Outside North America, find your local office at: **oracle.com/contact**.

 [blogs.oracle.com](https://blogs.oracle.com)

 [facebook.com/oracle](https://facebook.com/oracle)

 [twitter.com/oracle](https://twitter.com/oracle)

Copyright © 2024, Oracle and/or its affiliates. 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, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.