PURESTORAGE®
Uncomplicate Data Storage, Forever

WHITE PAPER

FlashArray ActiveCluster for Oracle Database

A solution for higher availability, read scalability and non-disruptive storage-level migration for Oracle Databases.

Contents

Executive Summary

Neglecting to implement a robust High Availability solution at storage layer in conjunction with Oracle High Availability solutions like Oracle RAC databases and Oracle standalone databases with no high availability can have wide-ranging consequences. These include increased vulnerability to data loss, potential extended periods of system downtime in the event of underlying storage failures, heightened operational complexity, and the need for manual intervention during failover scenarios.

Recovery Point Objectives (RPO) and Recovery Time Objectives (RTO) may be less efficient, and conventional backup methods become the primary means of data protection, introducing the risk of data corruption and data availability. For critical applications, this lack of high-availability measures can seriously jeopardize business continuity and service delivery. Thus, the adoption of a suitable high-availability solution is crucial to safeguard data, minimize downtime, and ensure the resilience of Oracle databases.

Pure Storage® ActiveCluster™ provides synchronous replication between two FlashArray™ systems. Arrays participating in a replication keep identical copies of volume(s) being replicated. Writes directed to replicated storage volumes in either array are executed on both.

In the event of a site-wide disaster, ActiveCluster guarantees your Oracle databases remain accessible and up to date, safeguarding your business operations and minimizing potential data loss.

Deploying Oracle RAC on ActiveCluster doubly ensures business continuity, as your critical Oracle databases are always accessible, updated, and protected. This is essential for applications and databases that are vital to your organization's operations.

Leveraging the capabilities of ActiveCluster technology, offered by Pure Storage, ensures that data is replicated synchronously, guaranteeing zero data loss, and enabling transparent storage failover for minimal downtime.

How to Use This Document

This document serves as a comprehensive resource designed to offer insights into the effective utilization of Pure Storage FlashArray's ActiveCluster feature. It highlights the capabilities of ActiveCluster in providing redundancy and high availability at the storage layer for Oracle Clusters, ultimately minimizing downtime and ensuring uninterrupted operations.

In addition to addressing storage resilience, this document explores how the ActiveCluster feature can be harnessed for the seamless migration of Oracle databases. By leveraging the combined power of ActiveCluster and Oracle database migration, you can enhance the reliability and continuity of your critical data infrastructure. Throughout this document, you will find detailed guidance and best practices to support your endeavors in achieving a robust and resilient storage environment for your Oracle databases.

Solution Overview

FlashArray ActiveCluster provides synchronous replication of volumes across different arrays, ensuring identical volume identities presented to Oracle hosts from any participating array. This is particularly valuable for Oracle database nodes, which can be zoned to ActiveCluster FlashArray systems to provide high availability at the storage layer.

This section provides a detailed description of the solution's components.

FlashArray ActiveCluster

Pure Storage Purity FA ActiveCluster is an advanced solution for data replication, offering a fully symmetrical, bidirectional replication capability that ensures zero Recovery Point Objective (RPO) and zero Recovery Time Objective (RTO). It achieves this by replicating data synchronously and enabling automatic, seamless failover.

ActiveCluster is designed to work across multiple locations, allowing for the deployment of clustered arrays and hosts in a flexible and active/active datacenter configuration. This means that data can be accessed and utilized from different sites, providing high availability and disaster recovery capabilities.

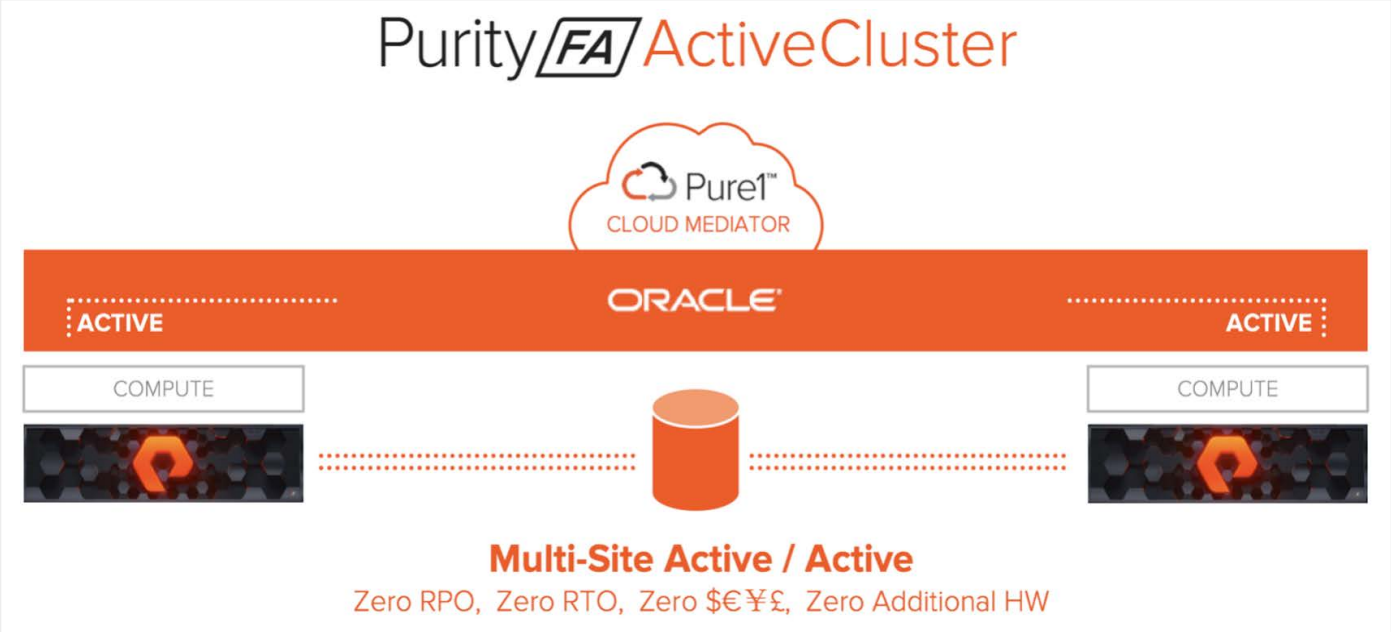

FIGURE 1 Purity ActiveCluster architecture

ActiveCluster Features

ActiveCluster offers several key benefits, including:

Synchronous Replication—When data is written, it's simultaneously copied to both storage arrays and stored in non-volatile RAM (NVRAM) on each array. Only after this duplication process is complete is the write acknowledged to the host system. This ensures data consistency and high reliability.

Symmetric Active-Active—Allows data to be read from and written to the same volumes on either side of the mirror. This means you can access and update data on both storage arrays equally, and there is an option to be aware of the host-to-array site, which enhances control and flexibility.

Transparent Failover with Preferences—Ensures that failover between synchronized arrays and sites happens seamlessly and without causing disruption. Users can choose their preferred failover conditions, and the system will automatically handle the failover process. After failover, it automatically re-establishes synchronization and recovers the data, maintaining high availability and data integrity.

Active-Active Asynchronous Replication—Includes an integrated feature that allows asynchronous replication with a target orchestration. This means it can efficiently replicate data to a third site (out of the region) to ensure data protection and resilience. This approach simplifies the process while maintaining the integrity of your data.

No Bolt-Ons & No Licenses—ActiveCluster eliminates the need for additional hardware or costly software licenses. Transitioning to an active/active setup is achieved simply by upgrading the Purity Operating Environment, without incurring additional expenses.

Simple Management—Additionally, ActiveCluster offers simplified data management capabilities. Users can perform various data management operations from either side of the replication mirror. This includes provisioning storage, connecting hosts, creating snapshots, and generating clones, all through an intuitive and user-friendly interface, enhancing operational efficiency and flexibility.

Integrated Pure1® Cloud Mediator—Seamlessly includes a passive mediator that is automatically configured. This mediator not only enables transparent failover but also safeguards against split-brain scenarios, all without the necessity of deploying and managing an additional component. This integration streamlines high availability and disaster recovery management while reducing operational complexity.

ActiveCluster Core Components

Purity ActiveCluster stands on three pillars: the Pure1 Mediator, paired active/active clustered arrays, and extended storage containers. The Pure1 Mediator ensures seamless communication, while the paired arrays provide the infrastructure for bidirectional data access.

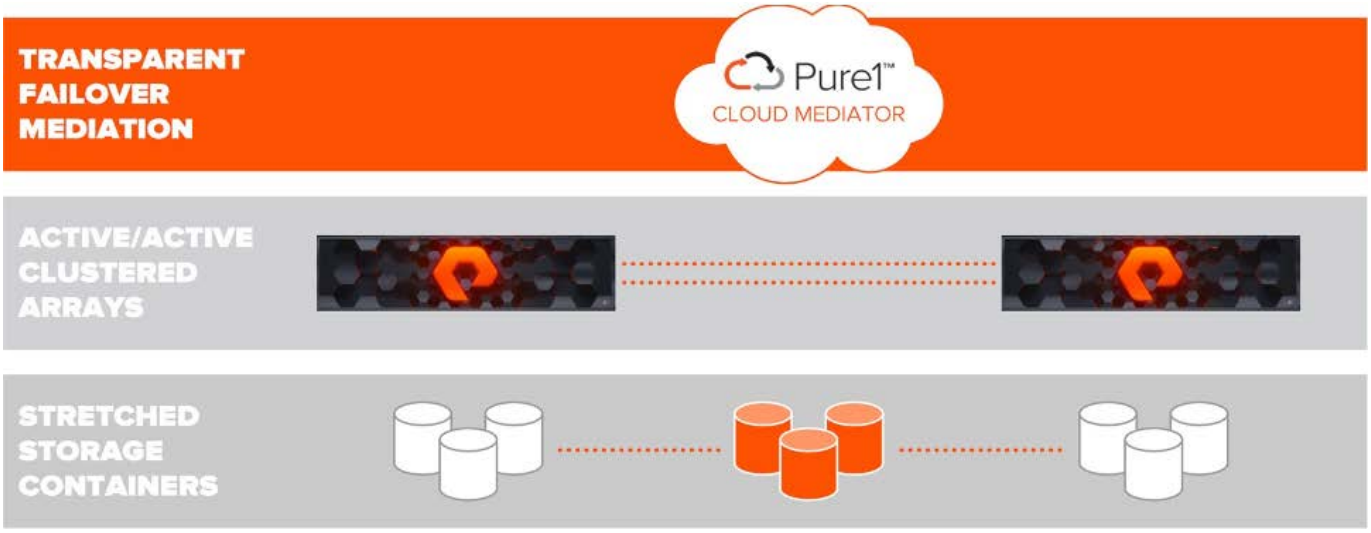

FIGURE 2 Purity ActiveCluster components

The Pure1 Cloud Mediator—An essential component within the solution. It plays a critical role in determining which storage array will assume data services in the event of an outage, ensuring the continuity of operations within the environment.

Active-Active Clustered FlashArray systems—Leverage synchronous replication to uphold identical data copies on each array and present them as a unified and consistent copy to hosts connected to either or both of the arrays. This approach ensures data accessibility and consistency across the clustered arrays.

Stretched Storage Containers—Specialized management containers that organize storage objects, such as volumes, into groups that span across two arrays. These containers not only facilitate the grouping of storage objects but also ensure a consistent IO continuation behavior for the storage objects contained within them, enhancing data availability and reliability.

Administration

ActiveCluster introduces a new management entity known as "Pods." A Pod is a stretched storage container that delineates a group of objects that undergo synchronous replication together and specifies between which arrays this replication occurs.

An array can accommodate multiple Pods, and these Pods can exist on a single array or span across two arrays concurrently, maintaining synchronous replication. When Pods are synchronously replicated between two arrays, they are referred to as being "stretched between arrays." This innovative feature enhances the flexibility and management capabilities of ActiveCluster.

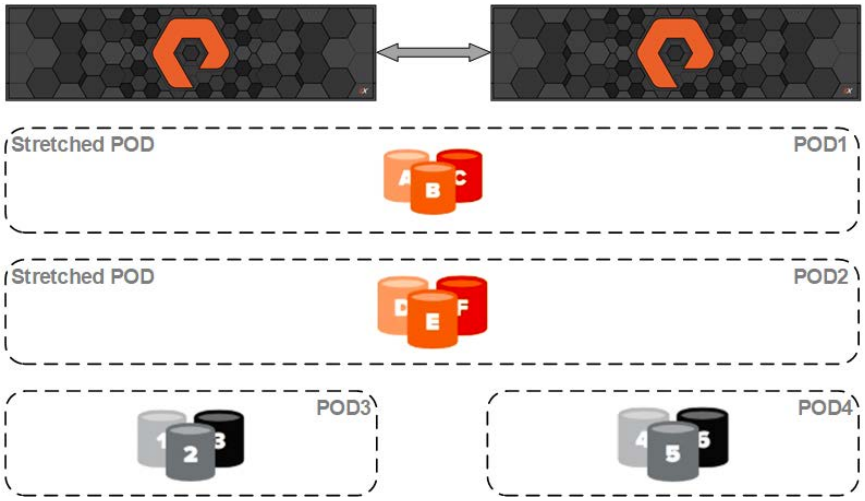

FIGURE 3 Purity ActiveCluster PODs

POD Consistency Group

Pods can contain various components, including volumes, protection groups, and configuration details specifying the association between volumes and hosts. Essentially, a Pod functions as a consistency group, guaranteeing that multiple volumes housed within the same Pod maintain a strict consistency in terms of the order in which write operations are executed. This ensures data integrity and coherence for the interconnected components within the Pod.

POD Namespaces

Pods also introduce the concept of volume namespaces, which means that different volumes can share the same volume name as long as they are located in different Pods. In the depicted scenario, the volumes within Pod3 and Pod4 are distinct from those in Pod1, which is a stretched active/active Pod. This architectural choice enables the migration of workloads between arrays or the consolidation of workloads from multiple arrays onto a single one, without encountering conflicts related to volume names. It enhances flexibility and simplifies data management in diverse operational scenarios.

Mediation

A mediator is a critical component that helps manage and coordinate data replication and failover between two FlashArray systems in an Active/Active configuration. ActiveCluster is a technology that enables synchronous replication of data between two FlashArray storage systems, typically located in separate data centers, to provide high availability and disaster recovery capabilities.

Pure1 Cloud Mediator

The Pure1 Cloud Mediator plays a crucial role in ensuring that, in the event of communication loss between arrays, only one array remains active for each Pod. When the arrays can no longer communicate over the replication interconnect, both arrays will temporarily halt I/O operations and contact the mediator to determine which array can continue serving the synchronously replicated Pods.

The array that reaches the mediator first is granted permission to maintain its synchronously replicated Pods online. Conversely, the second array to reach the mediator is required to cease servicing I/O to its synchronously replicated volumes to prevent a scenario known as "split brain." Importantly, this entire process is designed to occur within standard host I/O timeouts, ensuring that applications experience no more than a brief pause and resume of I/O, minimizing disruption.

On-premises Failover Mediator

Failover mediation for ActiveCluster can also be facilitated by deploying an on-premises mediator, which is distributed as an OVF (Open Virtualization Format) file and can be set up as a virtual machine. The failover procedures remain consistent with those previously described, regardless of whether the Pure1 Cloud Mediator or the on-premises mediator is employed.

The on-premises mediator must adhere to certain fundamental requirements:

- Virtual Machine Deployment: The on-premises mediator can only be deployed as a virtual machine on virtualized hardware; it is not installable as a standalone application.
- High Availability: To ensure high availability, the mediator's hosting hosts should implement a reliable failover mechanism, such as VMware high availability or Microsoft Hyper-V high-availability clustering.
- Immutable Configuration: Storage hosting the mediator must be configured to prevent any rollback of the mediator's configuration to prior versions. This includes scenarios like storage snapshot restores or situations where the mediator resides on mirrored storage.
- Configuration Setting: The storage arrays must be configured to utilize the on-premises mediator instead of the Pure1 Cloud Mediator.
- Third-Site Deployment: The mediator should be deployed in a third site, within a separate failure domain that remains unaffected by failures in either of the sites where the arrays are installed.
- Independent Network Connectivity: Both array sites must have distinct network connections to the mediator. This ensures that the failure of one network connection does not impede both arrays' access to the mediator, enhancing redundancy and reliability.

Transparent Failover

ActiveCluster's failover is automatic, requiring no manual intervention from administrators. It happens within standard time limits, similar to upgrades in a single array. This ensures high availability and prevents data mismatches between arrays serving the same volume.

To maintain availability across two sites, a component called a witness or voter is needed to manage failovers and prevent data mismatches. ActiveCluster uses the Pure1 Cloud Mediator (or On-premises Failover Mediator) for this purpose, making failover and site changes seamless and automatic in the event of issues, without the need for manual actions.

Uniform and Non-uniform Host Access

In ActiveCluster, hosts can be configured in two ways: non-uniform host access, where hosts interact solely with their local storage array, and uniform host access, where hosts can access both arrays, regardless of their physical location.

Uniform Host Access

A standardized storage access model is applicable within environments featuring host-to-array connectivity via either Fibre Channel (FC) or Ethernet (for iSCSI), along with array-to-array Ethernet or FC interconnectivity across two separate sites. In this deployment scenario, each host possesses access to identical volumes through both the local and remote arrays. This solution is adept at facilitating the connection of arrays even when the round-trip time (RTT) latency between them reaches up to 11 milliseconds, ensuring seamless and consistent data access and management.

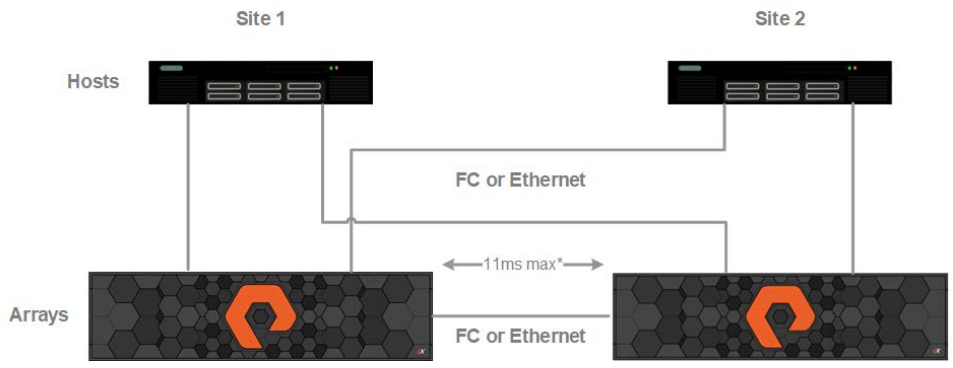

FIGURE 4 ActiveCluster Uniform host access

Figure 4 provides a representation of the logical pathways connecting hosts and arrays, including the replication link connecting the two arrays, within the context of a uniform access model.

In a uniform storage access model, where all hosts regardless of their site location can access both arrays, it's important to note that different latency characteristics will be observed.

Specifically, paths from hosts to the local array will exhibit lower latency, while paths from each local host to the remote array will demonstrate higher latency due to the geographical separation between sites. This diverse latency profile should be considered when optimizing data access and ensuring efficient data management across the environment.

For instance, in the image below, if an application called “App Service—B” writes to stretched volume A through Array A, it might take longer compared to writing through Array B. This is because it has to travel the longer path between the host and Array A and then cross the replication path between Arrays A and B. It's important to be aware of these latency differences when deciding how to route data for better performance.

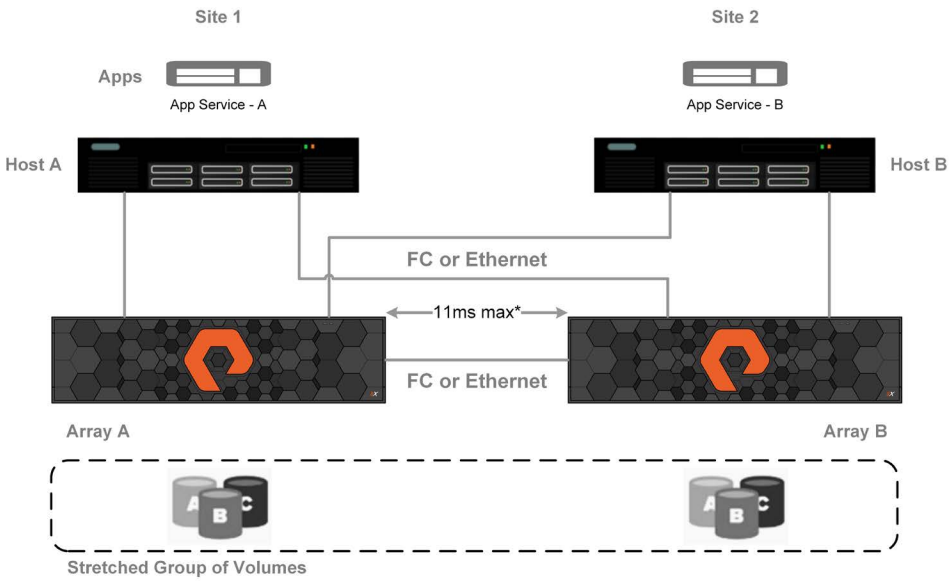

FIGURE 5 Application Uniform access

ActiveCluster uses a feature called SCSI ALUA (Asymmetric Logical Unit Access) to encourage hosts to use local paths when accessing FlashArray volumes. Array administrators can select a preferred array for a host-volume connection, making it the best choice for local hosts. Remote paths are available but not as fast. While both local and remote paths can be used for reading and writing, hosts prefer the fast local paths and only use the slower remote paths when there are no fast ones available. This helps data access stay quick and efficient.

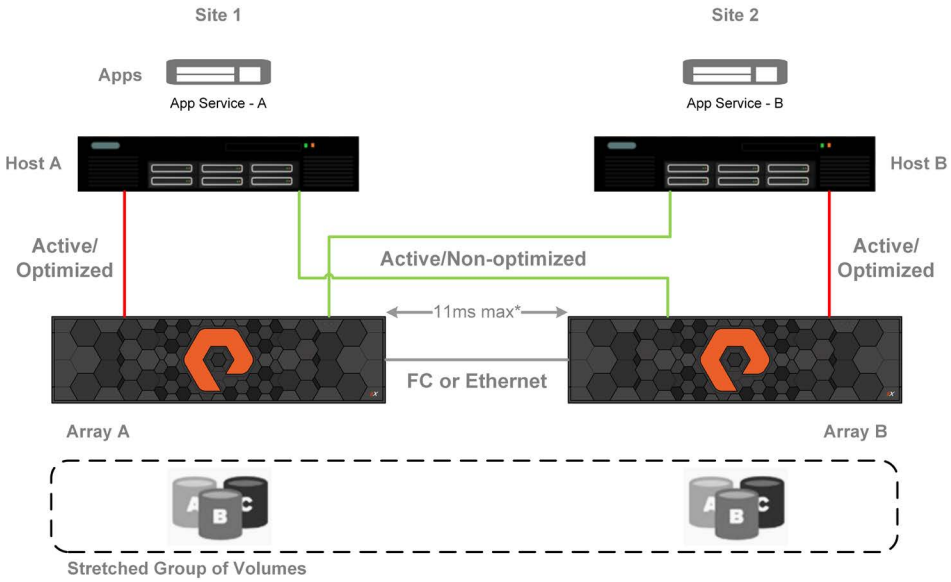

FIGURE 6 ActiveCluster Uniform host access with path optimization

As shown in Figure 6 Array A is local to Host A, setting the preferred array to Array A is important when there is a big difference in latency between hosts and their local array compared to the remote array. In cases where datacenter or small campus setups have similar latencies for host-array communication, utilizing all paths can enhance overall performance.

Non-Uniform Access

In environments where host-to-array connections are either Fibre Channel (FC) or Ethernet (iSCSI) and limited to local site communication, a non-uniform storage access model is applied. However, an Ethernet connection is still needed for the array-to-array replication link between distant sites. In this setup, each host can only access a volume through the local array and not the remote array. This approach supports array connections even with up to 11 milliseconds of round-trip time (RTT) latency between arrays.

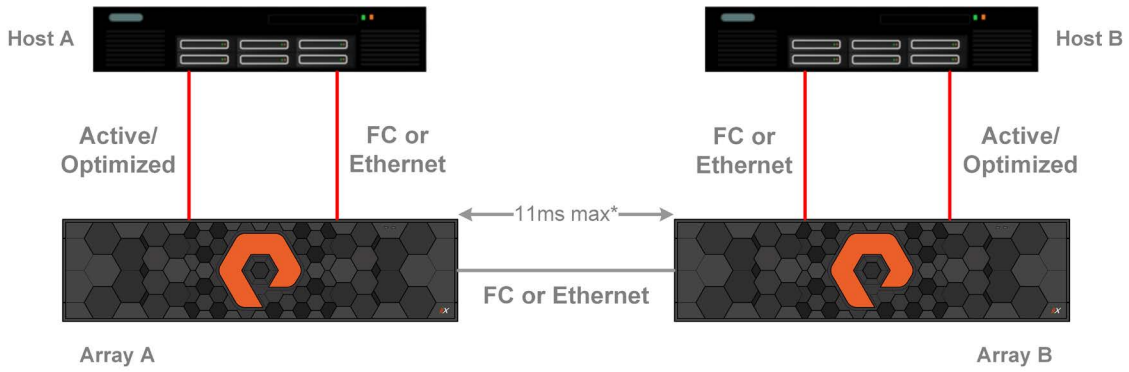

FIGURE 7 ActiveCluster Non-Uniform host access

In a non-uniform access setup, hosts direct I/O operations exclusively to local storage, as local paths are the active and optimized paths available. While it's possible to configure Extended RAC Clusters with non-uniform host access to data, it's not the recommended approach. This is because if an array fails, it leads to an outage for the RAC nodes local to that array, which can disrupt operations.

The knowledge base article ActiveCluster Requirements and Best Practices describes the requirements in detail.

Oracle Database Overview

Oracle Database, an industry-leading database management system, stands out for its capacity to handle extensive and varied datasets. Designed for scalability and performance, it serves as a foundation for enterprise applications, encompassing tasks like transaction processing, business intelligence, and data warehousing.

Oracle Real Application Clusters (RAC)

Oracle RAC is a purpose-built feature to deliver scalability and high availability within the confines of a single data center, making it an ideal choice for organizations seeking to meet the demands of mission-critical applications. However, in today's dynamic business landscape, where resilience against unforeseen disruptions is paramount, Oracle RAC can be extended to further fortify data protection strategies. This extension involves configuring Oracle RAC nodes in separate geographic locations within the same metropolitan region, creating what's known as an "extended distance cluster."

By spreading the Oracle RAC cluster across multiple locations, organizations can mitigate the risk of site-wide disasters and maintain business continuity in the face of unexpected events. This approach not only enhances disaster recovery capabilities but also ensures that critical databases remain accessible, even when one location experiences a disruption.

The extended distance cluster configuration empowers enterprises to strike a balance between scalability, high availability, and disaster resilience, thereby enabling them to address the evolving challenges of modern business operations while continuing to rely on Oracle RAC's core capabilities for robust database management.

Oracle Standalone Instances

In contrast to Oracle RAC's emphasis on cluster-based operations, standalone Oracle databases are designed for simplicity, cost-effectiveness, and streamlined management. These databases are ideal for organizations where high availability and scalability are secondary considerations, and a single-instance configuration serves their operational needs effectively. Standalone Oracle databases do not provide redundancy against failures, which calls for a need to have redundancy at the storage layer.

Storage Considerations

In the realm of Oracle databases, the choice of storage plays a pivotal role in shaping performance and dependability. This technical white paper delves into essential considerations for Oracle storage, elucidating strategies to enhance performance, guarantee high availability, and enable scalable operations. Additionally, it highlights the potential drawbacks that organizations might face if they forgo the implementation of FlashArray ActiveCluster, an integral component for maximizing storage efficiency and fortifying Oracle database solutions.

Storage Considerations for Oracle RAC databases

Oracle RAC relies on shared storage, ensuring that every cluster node can fully access the database storage. Extended RAC Clusters follow the same approach, requiring all nodes at each site to have complete access to the storage. Oracle offers support for two replication methods, host-based and storage array-based, to enable this data accessibility and synchronization across distributed clusters.

Storage Considerations for Oracle Extended RAC databases

In Extended RAC Clusters, all RAC nodes need to be actively connected to storage across all sites, which is referred to as active-active replication. FlashArray ActiveCluster supports this by enabling synchronized replication between arrays at two sites. This ensures that all RAC nodes maintain active connections to both arrays at both sites, helping to keep data consistent and accessible in multi-site RAC setups.

Oracle suggests using its Automatic Storage Management (ASM) for host-based replication rather than array-based replication. The reason for this preference is that many array-based replication solutions offer active-passive capabilities, meaning that nodes on one site don't maintain active connections to storage on the other site. Oracle's ASM provides a more reliable and consistent data synchronization method, ensuring redundancy and availability in Oracle environments.

Symmetric active-active replication with ActiveCluster

Through ActiveCluster, FlashArray systems offer symmetric active-active replication. This means that any data written to one of the replication partner arrays instantly and synchronously appears on the other. Furthermore, volumes created on one array become immediately accessible and usable for hosts connected to the other array. Hosts also have the capability to connect to both arrays, effectively balancing their data access. This robust approach ensures high redundancy and optimal performance in storage operations.

Hosts have the flexibility to connect to both arrays and evenly distribute write operations between them. Alternatively, they can choose to primarily access the array that's local to them, only switching to the remote array when the local one becomes unavailable. This adaptability in host behavior enhances performance and resilience in data access and storage operations.

Combined with Oracle RAC, FlashArray systems with ActiveCluster protect not only against host failure, but also against array failure and even failure of an entire site without the need for “voting disks” at a third site.

Deploying RAC on ActiveCluster eliminates the need for extended cluster configuration. Unlike a traditional stretched RAC setup, which requires configuration at the ASM level and utilizes ASM failure groups, RAC on ActiveCluster operates seamlessly without any knowledge of the two-site infrastructure. All synchronization occurs at the FlashArray layer with the aid of OS multipathing, reducing complexity and simplifying deployment. This approach not only minimizes administrative overhead but also halves the storage capacity requirement for stretched RAC by eliminating the need for ASM failure groups. This makes ActiveCluster a compelling solution for organizations seeking a simplified, high-performance, and cost-effective disaster recovery solution for Oracle RAC databases.

Configuring and Managing ActiveCluster

One of the significant advantages of employing an ActiveCluster stretched storage solution is its simplicity. Prior to proceeding, it is important to verify that the environmental configuration aligns with the requirements outlined in ActiveCluster Requirements and Best Practices. This confirmation ensures a smooth and straightforward implementation process.

Creating a Synchronous Connection

To set up ActiveCluster, start by creating a synchronous connection with another FlashArray. It's worth mentioning that you can use either FlashArray for this connection—both work equally well for the task.

Obtaining a Connection Key from target Array

For ActiveCluster to work, a TCP/IP connection must exist between the two FlashArray systems involved in replication.
To establish this connection

Obtain the connection key from the other FlashArray using the command mentioned below.

```
pureuser@FA-B> purearray list --connection-key
Connection Key
eb435b0d-ebbf-7cd6-ec7a-405ce1e2629b
```

Connecting the Arrays for Sync replication

On the corresponding FlashArray, utilize the connection key and the management IP address to establish a synchronization replication relationship. The provided example uses Ethernet as the replication transport, but it is equally feasible to employ Fibre Channel (FC) if preferred.

Run below command to connect the arrays

```
pureuser@FA-A> purearray connect --management-address FA-B --type sync-replication
--connection-key
Enter the connection key of the target array: eb435b0d-ebbf-7cd6-ec7a-405ce1e2629b
Name            ID                          Version Management Address  Replication Address
Status     Throttled Type           Transport
FA-B c35d2639-27cf-4c5a-aa52-b0bb676b8c52 6.4.5    FA-B                10.21.126.31
connected False     sync-replication    ip
                                                                       10.21.126.32
                                                                       10.21.126.33
                                                                       10.21.126.34
```

Verify the connection between arrays

Execute the following command to confirm that the arrays are successfully connected.

```
pureuser@FA-A> purearray list --connect
Name ID                                  Version Management Address    Replication Address
Status     Throttled Type          Transport
FA-B c35d2639-27cf-4c5a-aa52-b0bb676b8c52 6.4.5  FA-B                  10.21.126.31
connected  False    sync-replication  ip
                                                                        10.21.126.32
                                                                        10.21.126.33
                                                                        10.21.126.34
```

Managing a POD

ActiveCluster replicates storage objects like volumes, snapshots, and schedules within consistency groups, referred to as pods. Each array can accommodate multiple pods, with each pod acting as an independent namespace for the objects it houses. An array administrator has the capability to stretch (replicate) a pod across two arrays, enhancing data redundancy and availability.

An entire pod is replicated in sync, ensuring everything in it stays consistent. If replication is interrupted, a winner takes control of the entire pod.

It's a good idea to put related volumes in the same pod. This is useful for volumes with similar needs or when you want them to be consistent. It also makes administration easier. For volumes with different requirements, use separate pods to keep things organized.

If pre-existing database volumes are present, the pod should be created on the array that is local to.

Execute below command to create a POD.

```
pureuser@FA-A> purepod create AC-RAC-POD
Name       Source Array  Status Frozen At Promotion  Status Link Count Quota Limit
AC-RAC-POD -    FA-A     online -        promoted     0      -
```

Adding Volumes to a ActiveCluster Pod

Before stretching a pod (enabling synchronous replication to another array), it's possible to move existing volumes into it. After the pod is stretched, you cannot move existing volumes into it, but you can create new volumes within the stretched pod.

Moving Existing Volumes into a Pod

Existing volumes can be moved into the pod as long as the pod is not in a stretched state. When a volume is moved into a pod, it is renamed by adding the pod name followed by two colons as a prefix to its original name. This naming convention helps maintain clarity and organization within the pod.

Execute following command to move existing volume into a POD

```
pureuser@FA-A> purevol move ACVOL-ORACRS AC-RAC-POD
Name                        Size Source Created                Serial
AC-RAC-POD::ACVOL-ORACRS 200G -     2023-10-05 06:04:15 PDT D4ECA33C401C4C0B0001C44A
```

Creating a New Volume in a Pod

To create a new volume in a pod, whether stretched or unstretched run below command.

```
pureuser@FA-A> purevol create --size 500G AC-RAC-POD::ACVOL-ORAFRA
Name                        Size Source Created                Serial
Protection
AC-RAC-POD::ACVOL-ORAFRA 500G -     2023-10-05 06:15:31 PDT D4ECA33C401C4C0B0001C455 AC-RAC-
POD::pgroup-auto
```

Stretching a Pod

Expanding a pod initiates the replication of volumes added to it and conducts the initial baseline copy of these volumes from the source to the target array. Once this baseline process is finished, the pod achieves high availability on both arrays, ensuring data redundancy and accessibility.

Execute below command to stretch a POD

```
pureuser@FA-A> purepod add --array FA-B AC-RAC-POD
Name           Source Array       Status  Frozen At Promotion Status Link Count Quota Limit
AC-RAC-POD -   FA-B             offline     -     promoted            0      -
                 FA-A              online     -
```

Verifying Pod's Status

Run below mentioned command to verify PODs status

```
pureuser@FA-A> purepod list AC-RAC-POD
Name       Source Array       Status  Frozen At Promotion Status Link Count Quota Limit
AC-RAC-POD -     FA-B        online  -       promoted            0        -
                  FA-A        online   -
```

Host Connectivity

FlashArray provides support for both iSCSI and Fibre Channel host connections, and these options are fully compatible with ActiveCluster. To facilitate storage provisioning to a host, an array administrator is required to create a corresponding host object. Additionally, administrators have the flexibility to group hosts with shared storage provisioning needs, such as RAC cluster nodes, into host groups, or "hgroups." This organizational feature streamlines storage management and allocation.

Hosts, Host Objects and Preferred Arrays

A FlashArray host contains a list of a host's initiators, which are identified through worldwide names (Fibre Channel WWN) or iSCSI qualified names (IQN). Prior to allowing the I/Os for a host on a volume, a connection must be established between the host object and the specific volume. This connection is a prerequisite for enabling data access and management.

Preferred Arrays

The FlashArray Preferred Arrays feature allows to designate specific paths between a host and a volume as preferred or optimized routes for routing I/O commands. This customization enhances control and efficiency in data transfer operations.

To create a host object and set preferred array run following command:

```
pureuser@FA> purehost create HOST-A --preferred-array FA-A --iqnlist iqn.1988-12.com.
oracle:93a6f1afbdeb
Name     WWN  IQN                                                    NQN  Preferred
Array
HOST-A   -        iqn.1988-12.com.oracle:93a6f1afbdeb  -        FA-A
```

Host Groups

To simplify the provisioning of storage for a cluster, host objects that correspond to the cluster's nodes can be grouped together, known as an host group (hgroup). This approach simplifies the management of storage resources for the entire cluster, enhancing efficiency and organization.

Execute the following command to create a host group and add hosts to it.

```
pureuser@FA-A> purehgroup create AC-RAC-GROUP --hostlist HOST-A,HOST-B
Name           Hosts
AC-RAC-GROUP HOST-A
               HOST-B
```

Oracle Real Application Cluster Hosts with ActiveCluster

Setting up Oracle RAC hosts to work with ActiveCluster replicated volumes follows the same procedures as configuring a local RAC cluster with FlashArray volumes. Therefore, the best practices outlined in Oracle Database Recommended Settings for FlashArray remain applicable. However, this document will highlight some additional considerations.

Each Oracle RAC node needs to connect with a local array for local storage access and to a remote array(s) to have redundancy and performance at the storage layer.

The diagram below illustrates an Oracle RAC setup integrated with FlashArray ActiveCluster, with no preferred array configured for the hosts. This configuration not only ensures high availability at the storage level but also enhances performance by facilitating simultaneous I/O operations across both arrays.

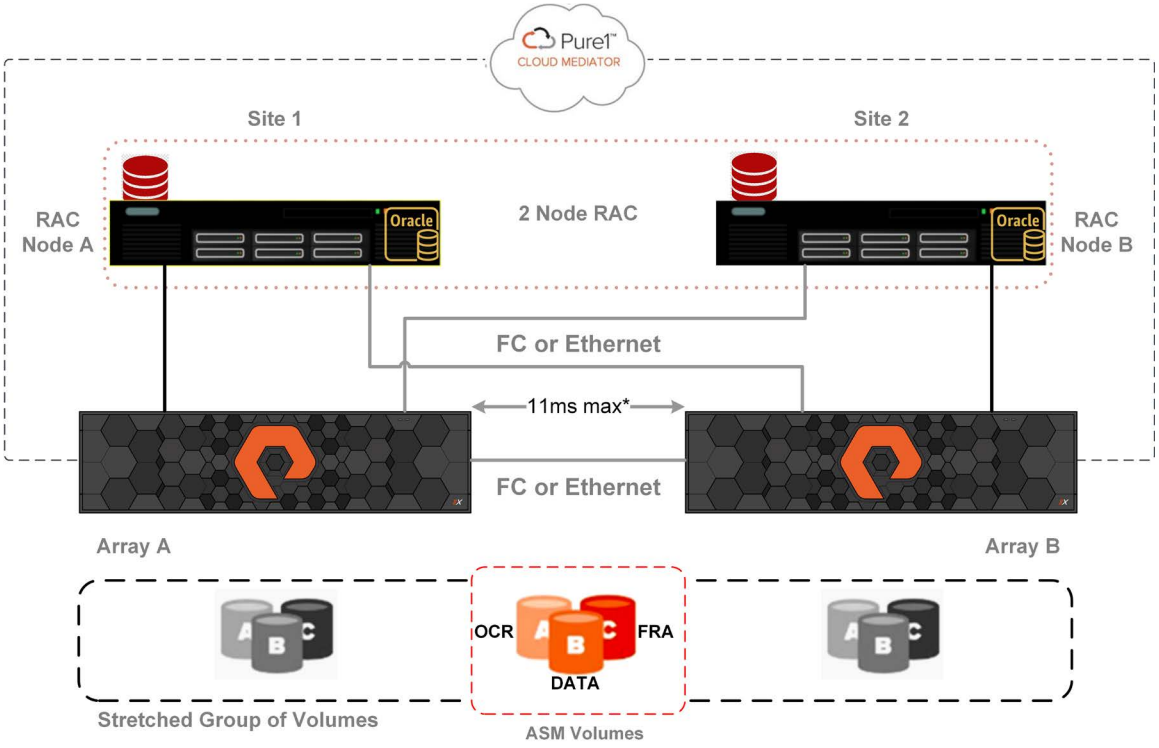

FIGURE 8 Oracle RAC with ActiveCluster

The Host Connectivity section provides detailed guidance on establishing connectivity between your Oracle hosts and the FlashArray systems, including configuring preferred array settings.

Configure RAC Clusters with uniform access. Ensure that all hosts within a cluster have corresponding host objects on both arrays. These host objects should be organized into a host group on each array. In a non-uniform access configuration, only hosts connected directly to an array should be added to its host group.

Network Connectivity

In the context of Oracle RAC, cluster nodes need to maintain constant communication, ensuring each other's availability and responsiveness. When it comes to Extended Distance Oracle RAC clusters, a stretched layer 2 network is imperative between sites. This network serves the dual purpose of facilitating RAC node interconnections and supporting the ActiveCluster replication links. Multipathing is strongly recommended to have network redundancy.

The knowledge base article: ActiveCluster Requirements and Best Practices describes the required networking configuration.

Storage Connectivity

RAC Clusters require uniform access configurations, meaning all cluster nodes should have paths to both ActiveCluster arrays. To maximize availability, redundant paths are essential. Keep in mind that network performance between sites is generally lower. Therefore, it's advisable for hosts to primarily use the local array for I/O to pod volumes and switch to the remote array only when the local one is down.

The FlashArray's preferred array feature lets administrators designate certain host connections as preferred or optimized for volumes within a stretched pod. Most multipathing I/O systems support Asymmetric Logical Unit Assignment (ALUA), a SCSI protocol feature that helps hosts select optimized paths for better performance and uses non-optimized paths only as a backup when needed. This strategy enhances storage operation efficiency and reliability.

For ActiveCluster storage configuration along with multipathing configuration guidance refer to ActiveCluster Requirements and Best Practices.

Uniform Access Configuration

In a uniform access configuration, all hosts have connections to both FlashArray systems and can therefore see paths to stretched pod volumes on each FlashArray.

Follow below steps to have uniform storage access to all the nodes in Oracle RAC, refer to the section Configuring and Managing ActiveCluster for detailed steps.

- Step 1: Create synchronous connections between arrays
- Step 2: Create Database Volumes on FlashArray
- Step 3: Create a Pod and Move Volumes into it
- Step 4: Stretch the Pod to a Second Array
- Step 5: Connect Volumes with one RAC Cluster Hosts
- Step 6: Discover Volumes on other RAC Cluster Hosts
- Step 7: Configure multipathing
- Step 8: Verify multipath configuration
- Step 9: Optionally specify Preferred Array for Each Host on both the Arrays

To ensure high data availability and optimal performance, it's crucial to establish connections from all hosts to both arrays. Additionally, configure each host object to have its preferred array set to the local array it is connected to.

Once the preferred array for each volume on both arrays is configured, execute the multipath command on each host. Confirm that half of the paths are assigned a priority of 50 (optimized) with an "active" status, while the remaining half should be designated with a priority of 10 (non-optimized) and an "enabled" status. This balance ensures an efficient and resilient storage setup.

ActiveCluster for Oracle Extended RAC

Setting up an Extended RAC Cluster closely resembles configuring a single-site Oracle RAC cluster, with the main difference being that cluster nodes are distributed across two sites. To guarantee that I/O commands are executed within standard host timeout periods, it's essential to note that Pure Storage facilitates ActiveCluster replication between arrays that have a communication latency of 11 milliseconds or less in roundtrip duration. This low-latency communication ensures efficient data transfer and operational reliability.

Below figure depicts an Oracle Extended RAC implementation with FlashArray ActiveCluster.

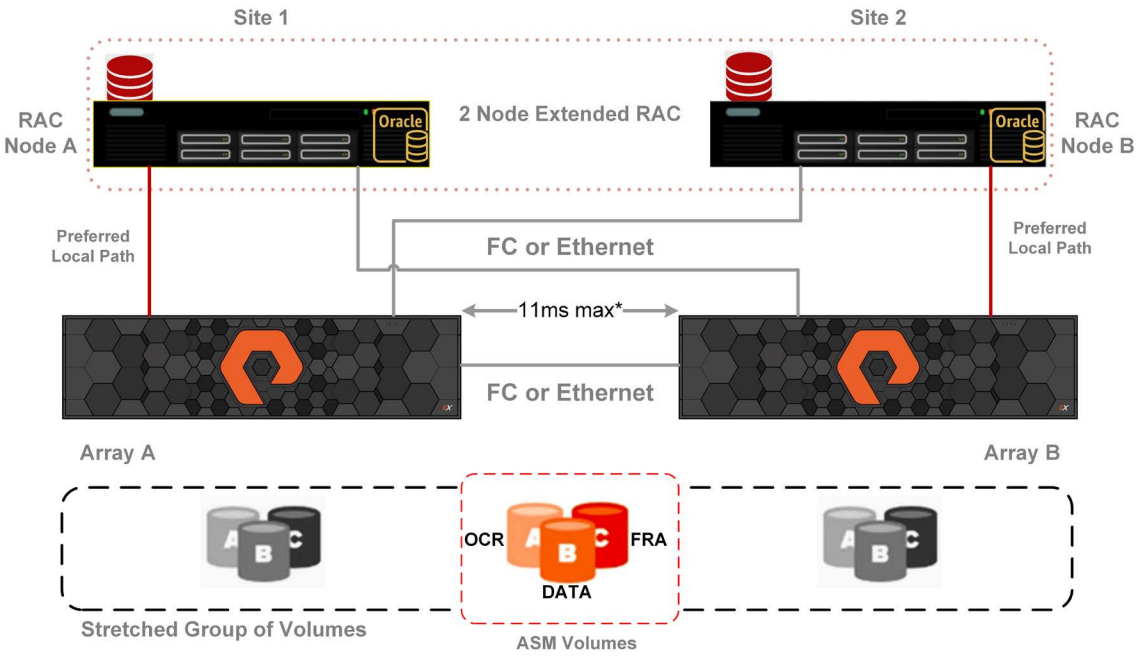

FIGURE 9 Oracle Extended RAC with ActiveCluster

With ActiveCluster, Oracle Extended RAC becomes a standard RAC setup. Its synchronous replication seamlessly handles storage availability, removing the need for complex ASM configuration and management associated with geographically separated storage. This results in simplified deployment, improved performance, and reduced costs, making it ideal for optimized RAC environments.

Refer to the section Configuring and Managing ActiveCluster for detailed steps on how to configure ActiveCluster.

For the best Oracle Extended RAC setup with ActiveCluster, follow these recommendations:

Network Considerations:

- Each site needs to be equipped with a FlashArray running Purity//FA Version 5.0 or later.
- IP based 10 Gbps (or more) replication interconnect between site 1 and site 2 FlashArray.
- The latency on the replication network must not exceed a round trip latency of 11 milliseconds.
- High-speed, dedicated connections are established between RAC nodes at both sites.
- Configure hosts in uniform access, ensuring that all RAC nodes are connected to both arrays.

Mediator Considerations:

- ActiveCluster mediation is managed by the Pure1 Cloud Mediator, unless user policies dictate otherwise. An alternative is a private mediator virtual machine situated at a third site, accessible by both FlashArray systems.

FlashArray Configuration Considerations:

- A POD that contains all volumes needed for Oracle RAC and stretches across both FlashArray systems.
- Host group that includes all RAC nodes on both FlashArray systems.
- Configure Preferred Array settings for all RAC nodes on both FlashArray systems.

Host Considerations:

- Configure Oracle RAC nodes with ALUA MPIO for optimal performance.
- ASM disk Groups using FlashArray volumes configured with external redundancy

Configuring Oracle Standalone hosts with ActiveCluster volumes

Oracle Standalone instances do not provide redundancy, with FlashArray ActiveCluster Oracle Standalone instances can leverage storage redundancy. Configuring Oracle Standalone hosts with ActiveCluster volumes provides storage redundancy and availability for enterprise storage environments.

In deploying standalone Oracle instances with FlashArray ActiveCluster, you have flexibility. You can choose arrays at a single site for enhanced storage redundancy and performance. In this configuration, Oracle hosts can utilize paths from all arrays for data operations. Alternatively, you can opt for arrays at different sites, ensuring data redundancy at the site level. The setup caters to diverse needs in storage and redundancy strategies.

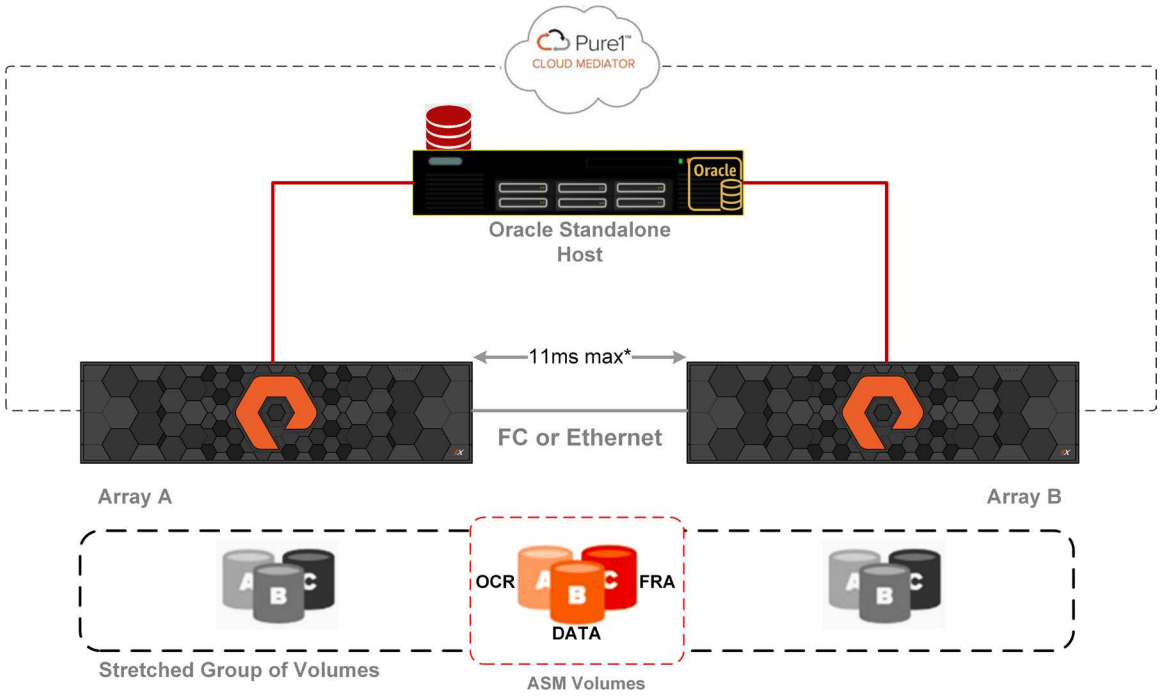

FIGURE 10 Oracle standalone database with ActiveCluster

By following the guidelines outlined below, system administrators can confidently set up Oracle Standalone hosts to take full advantage of ActiveCluster volumes, ultimately delivering improved data protection, system reliability, and peace of mind in the face of unforeseen events.

Refer to section Configuring and Managing ActiveCluster for detailed steps on how to configure ActiveCluster.

To set up Oracle Standalone hosts with ActiveCluster volumes, follow below recommendations:

- **Fast Network Connection**: Ensure a high-speed network (10 Gbps or more) between the two FlashArray systems. The network should have low latency, not exceeding 11 milliseconds for a round trip.
- **ActiveCluster Mediation**: ActiveCluster mediation is typically managed by the Pure1 Cloud Mediator. However, you can opt for a private mediator virtual machine if your organization's policies require it. This mediator should be accessible by both FlashArray systems.
- **Use a POD**: Create a POD that includes all the volumes required for Oracle RAC. This POD should span across both FlashArray systems for seamless data protection.
- **Uniform Access**: Configure your Oracle host to have uniform access. This means making sure it's connected to both FlashArray systems for improved data availability.
- **Preferred Array Settings**: If your FlashArray systems are in different locations, set up Preferred Array settings on the local FlashArray for your Oracle host. This ensures optimal performance.
- **ALUA MPIO**: Configure your Oracle host with ALUA MPIO for better performance and reliability.
- **ASM Disk Groups**: When creating ASM Disk Groups for your Oracle databases, use FlashArray volumes with external redundancy for enhanced data protection.

To configure an Oracle standalone database with FlashArray ActiveCluster, adhere to the following step-by-step guide, refer to section Configuring and Managing ActiveCluster for detailed steps:

- **Step 1: Create Synchronous Connections between Arrays:** Establish synchronous connections between the FlashArray arrays to enable ActiveCluster functionality.
- **Step : Create Database Volumes on any one of the FlashArray's:** Generate the necessary database volumes on the FlashArray storage to store Oracle database data.
- **Step 3: Create a Pod and Move Volumes into It:** Organize the volumes within a Pod, a grouping that facilitates synchronization and availability.
- **Step 4: Stretch the Pod to a Second Array:** Extend the Pod across a second FlashArray to ensure redundancy and disaster recovery capabilities.
- **Step 5: Optionally Specify Preferred Array for Each Host on Both Arrays:** If required, designate a preferred array for each host on both FlashArray systems to optimize data access.
- **Step 6: Connect Volumes with Oracle Host:** Establish connections between the volumes and the Oracle host for seamless data access.
- **Step 7: Configure Multipath:** Set up multipath configurations to enhance I/O performance and redundancy.
- **Step 8: Verify Multipath Configuration:** Validate the multipath configuration to ensure a robust and optimized setup for Oracle standalone databases using FlashArray ActiveCluster.

By following these steps Oracle Standalone hosts work seamlessly with ActiveCluster volumes, enhancing the database availability.

ActiveCluster for Read Performance

ActiveCluster provides synchronous replication capabilities that allow Oracle databases to be simultaneously accessible from multiple arrays. This unique approach significantly improves read performance, enabling organizations to harness the full potential of their databases while ensuring data redundancy and high availability.

With ActiveCluster, read and write requests from Oracle databases are distributed evenly across multiple arrays, whether they are located in the same site or dispersed across different sites. This load-balancing mechanism optimizes database access and reduces latency, resulting in enhanced query response times and a seamless end-user experience.

It is versatile and compatible with various Oracle implementations, including both Oracle standalone and Oracle RAC/Extended RAC setups.

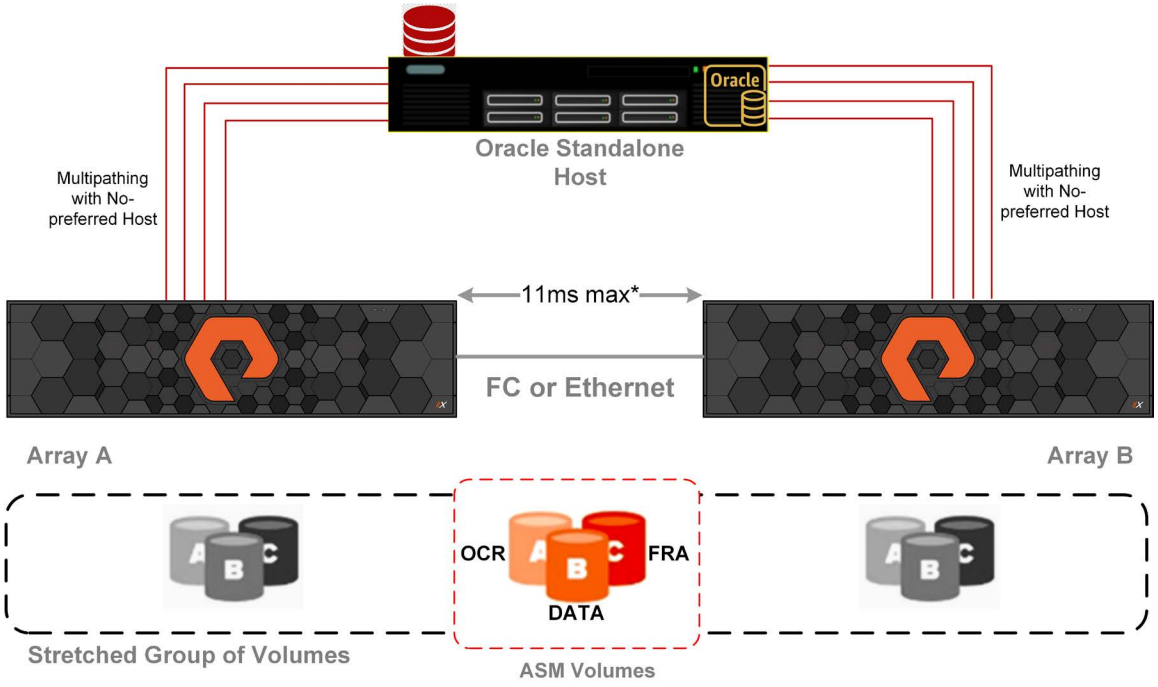

FIGURE 11 Oracle database for read performance

For optimal performance in Oracle deployments, ensure concurrent, active access to all storage by avoiding preferred arrays for hosts on participating FlashArray systems. Refer to section Configuring and Managing ActiveCluster for detailed steps on how to configure ActiveCluster.

Improving the performance of an Oracle database using FlashArray ActiveCluster involves several important strategies and best practices.

Here are some crucial recommendations that should be carefully considered to optimize the performance and reliability of your Oracle databases:

- Leverage ActiveCluster's active-active architecture to balance read and write workloads across multiple FlashArray systems. Make sure to have the exact same connectivity to all the arrays.
- Configure your Oracle database to use uniform access, ensuring that all hosts can access data from all the FlashArray systems.
- Do not set the Preferred Array for Oracle hosts on FlashArray systems. This ensures that hosts can leverage all available paths for I/O operations from all the FlashArray systems. This distributes workload and minimizes latency, leading to improved overall performance for Oracle database.
- Configure hosts with Asymmetric Logical Unit Access (ALUA) Multipath I/O (MPIO). ALUA helps hosts identify optimized paths to volumes, ensuring efficient data retrieval.
- Use FlashArray volumes with external redundancy for Oracle ASM disk groups.
- Ensure that your network connectivity meets the performance requirements. High-speed, low-latency connections between FlashArray systems (10 Gbps or more) are essential for efficient data replication.
- Keep an eye on replication network latency, as it should not exceed an 11 ms round-trip latency. Lower latency leads to faster data synchronization between FlashArray systems.

By following above recommendations and configuring ActiveCluster with Oracle best practices, you can significantly enhance the read performance of your Oracle database. ActiveCluster's active-active replication and failover capabilities, combined with efficient load balancing and access configurations, contribute to a high-performing, highly available Oracle database environment.

ActiveCluster for Data Migration

With the ever-increasing demand for data availability and business continuity, organizations often require the flexibility to move their critical Oracle databases for maintenance, disaster recovery, or load balancing. ActiveCluster steps in as a reliable, real-time data replication and failover solution that simplifies the database migration process.

FlashArray ActiveCluster facilitates seamless and secure migration of Oracle databases across sites or arrays located in the same site or across sites. It streamlines the migration process, reduces downtime, and ensures that critical Oracle databases remain accessible and responsive during the migration process.

This solution is a versatile solution applicable to both Oracle RAC and Oracle standalone databases. This flexibility ensures that regardless of your Oracle database architecture, ActiveCluster can seamlessly facilitate the migration process, providing a unified and efficient approach.

When migrating data using ActiveCluster, a carefully planned phased approach ensures a smooth and efficient transition. Follow these key stages to execute a successful data migration:

Stage 1: Assessment and Planning:

- Start with a thorough assessment of your Oracle database migration requirements.
- Plan the migration process, including the source and target arrays, data replication, and failover procedures.

Stage 2: Environment Setup:

- Ensure that your Oracle database environment, including the source and target sites, meets the hardware and software requirements for ActiveCluster

Stage 3: ActiveCluster Configuration:

Configure ActiveCluster by connecting the arrays and setting up synchronous replication to have Oracle database replicated to the target array. Refer to section Configuring and Managing ActiveCluster for detailed steps on how to configure ActiveCluster.

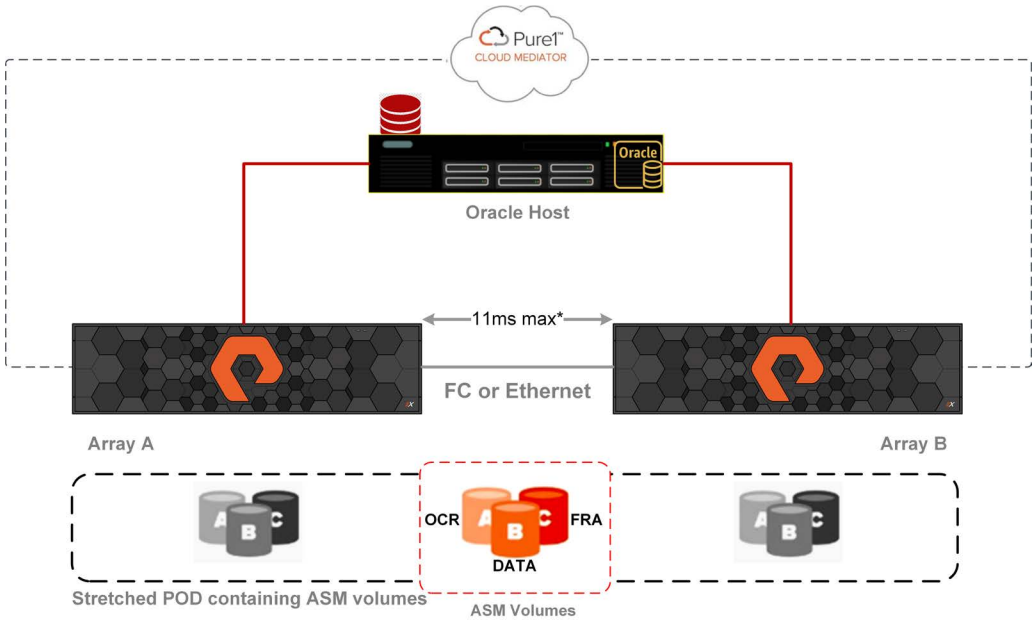

FIGURE 12 Oracle standalone database with stretched POD

- Setup ActiveCluster on the source and target arrays.
- Do not configure preferred arrays for Oracle host(s).
- Configure the necessary synchronous replication relationship between the sites or arrays to ensure real-time data replication.
- Configure your Oracle host to have uniform access to both the Arrays. Ensure to configure multipathing.

Stage 4: Testing:

- Perform comprehensive testing to validate the migration process and the failover mechanisms by disabling the paths to source array and validate data availability from target array.
- Test data accessibility and responsiveness of array at the target site.

Stage 5: Migration Execution:

- When ready to migrate, initiate the failover from the source to the target site or array. The failover process is as simple as removing the paths of the source array from the Oracle host(s). This ensures minimal downtime.

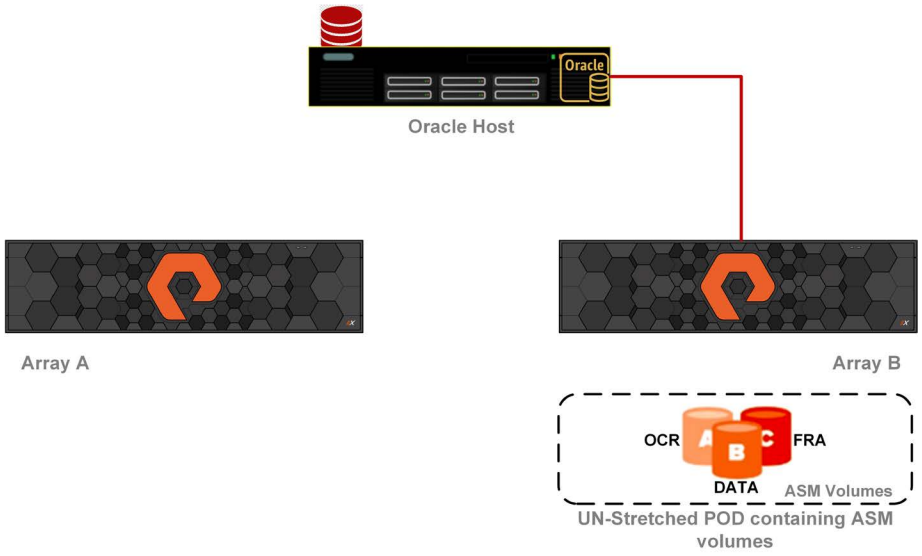

FIGURE 13 Oracle standalone database with un-stretched POD

- Un-stretch the POD.
- Remove Oracle host(s) to Array-A paths
- Remove synchronous replication between arrays
- Oracle host(s) now can access the database from Array-B.

Stage 6: Post-Migration Verification:

- After the migration is complete, verify the integrity of the Oracle database on the target site to ensure that data remains consistent and accessible.

Failure Scenarios

This section examines the availability of storage and network resources in different failure scenarios.The following table lists storage failure scenarios that can occur with ActiveCluster and how data availability to hosts is affected.

ActiveCluster Component Failures

Components				Access to Storage	Oracle Database
Array (FA-A)	Array (FA-B)	Replication Link	Mediator		
Up and Running	Up and Running	Up and Running	Up and Running	Available on both arrays	Up and running on all the nodes with access to both arrays
Up and Running	Down	Up and Running	Up and Running	Available on the surviving array	Up and running on both nodes with access to FA-A
Up and Running	Up and Running	Down	Up and Running	Available on one array	Up and running on both nodes with access to chosen array
Up and Running	Up and Running	Up and Running	Down	Available on both arrays	Up and running on both nodes with access to both arrays
Up and Running	Down	Down	Up and Running	Available on one array	Up and running on both nodes with access to chosen array
Up and Running	Up and Running	Down	Down*	Unavailable	Down as replication link and mediator both are down
Up and Running	Down	Up and Running	Down*	Unavailable	Down as replication link and mediator both are down along with an array
Down	Down	Not applicable (N/A)	Up and Running	Unavailable	Down as both the arrays are down

TABLE 1 ActiveCluster component Failures

Rows labeled with asterisks (*) denote failures of components other than the mediator when it's not accessible. In the event that the mediator becomes inaccessible after an array or the replication link has already experienced a failure, storage remains accessible on one array.

This section assesses the availability of storage and network resources in various failure scenarios, illustrating how FlashArray systems manage the most probable ActiveCluster failure situations.

Figure 14 illustrates a scenario involving Array failure. Following the failure of Array B in a 2-node RAC setup, RAC Node A seamlessly maintains access to Array A through a local connection. Meanwhile, RAC Node B intelligently redirects IOs to Array A via a remote path, ensuring the uninterrupted flow of normal Oracle operations.

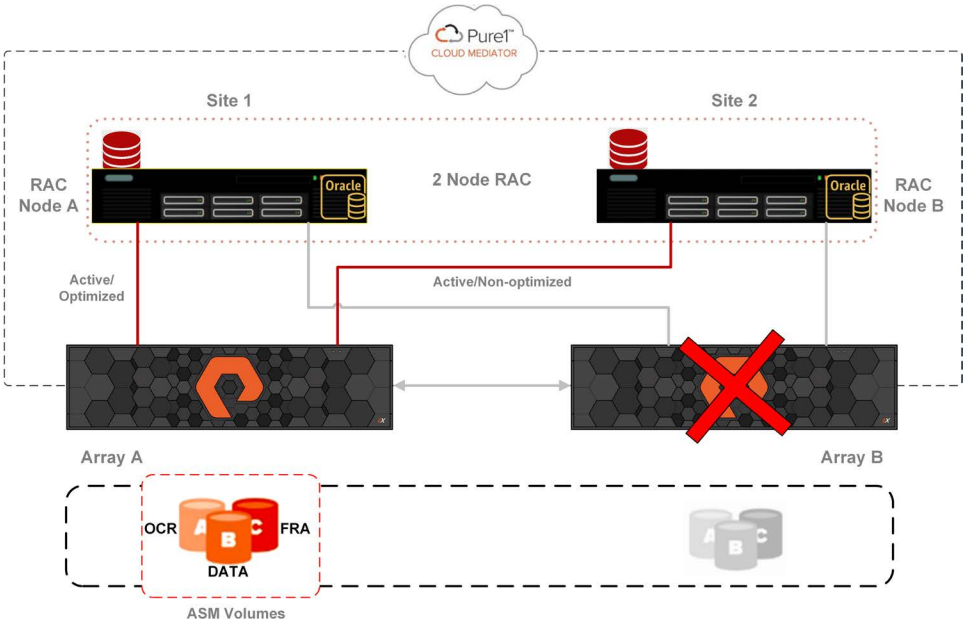

FIGURE 14 Array failure scenario

Figure 15 depicts the replication network failure scenario. After the replication link failure following a brief pause, host I/O to volumes within the affected pod resumes on the first array to establish contact with the mediator. The failover process is automatic and transparent, requiring no administrator intervention. Subsequently, after a brief pause, both local and remote host I/O resumes through the array that successfully contacts the mediator first. Simultaneously, storage paths leading to the array that loses the race to the mediator are rendered inactive.

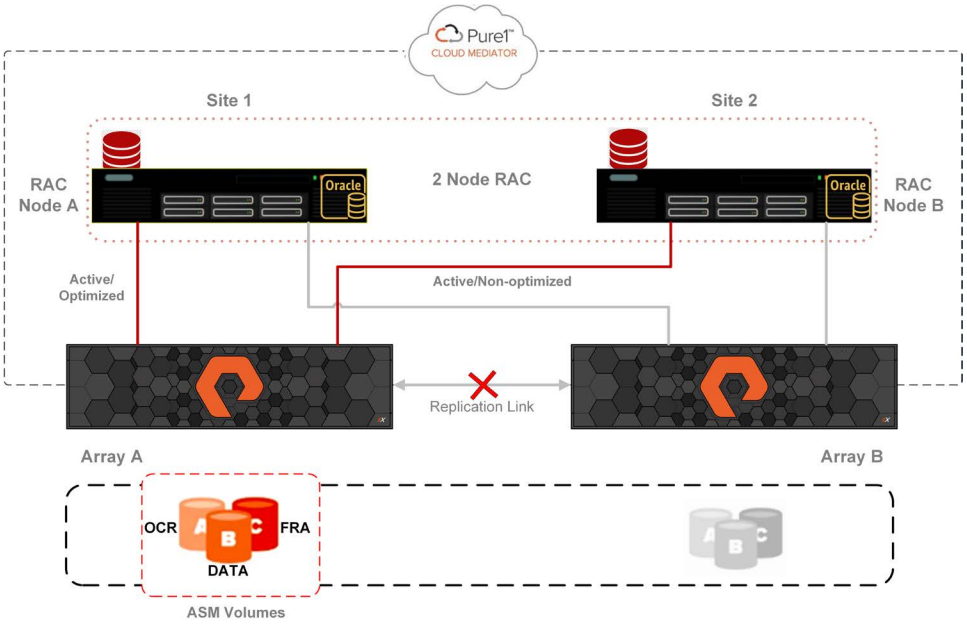

FIGURE 15 Replication link failure scenario

Storage Network failures

Failure Scenario	Behavior
Stretched SAN fabric outage (FC or iSCSI) (failure of SAN interconnect between sites)	Host I/O continues automatically on local paths within the local site. Uniform connected hosts: • Encounter some storage path failures to the remote array but maintain I/O on paths to the local array. • In each site, retain access to local volumes with only a brief pause in IO. Non-uniform connected hosts: • This scenario is not applicable as there is no SAN interconnect between sites.
SAN fabric outage in one site	Applications can automatically switch to hosts on the other site connected to a different array, driven by host cluster software such as VMware HA, Oracle RAC, SQL Cluster, etc. Uniform connected hosts: • In the site without a SAN issue, there may be some storage path problems for paths to the remote array, but operations continue on paths to the local array. • In the site with the SAN problem, there is a complete loss of access to volumes, and applications need to switch to the other site as mentioned above. Non-uniform connected hosts: • In the site without a SAN problem, there is continued access to local volumes. • In the site with the SAN problem, there is a total loss of access to volumes, and applications must switch to the other site as mentioned above.

TABLE 2 Storage Network failures

Array, Replication Network, and Site Failures

Failure Scenario	Behavior
Controller failover within an array	After a short pause for local high availability (HA) failover, host I/O continues on both arrays without compromising RPO-Zero. Asynchronous replication may resume from a different array than the one before the failover.
Replication link failure	After a short pause, host I/O for volumes associated with each affected pod resumes on the initial array that establishes contact with the mediator. Failover is automatic and seamless, requiring no administrator intervention. Uniform connected hosts: • Following a brief pause, local and remote host I/O resumes through the array that successfully contacts the mediator first. • Storage paths to the array that loses the race to the mediator will fail. Non-uniform connected hosts: • Hosts linked to the site that successfully contacts the mediator resume I/O after a brief pause. • Hosts linked to the site that loses the race to the mediator lose access to volumes. • Host cluster software may automatically restart applications on a host in the mediator-winning site.
Mediator failure or loss of access to mediator	No effect. Host I/O continues through all paths on both arrays as normal.
Failure of an entire array (surviving array is able to contact the mediator)	Following a short pause, host Input/Output (I/O) automatically resumes on the remaining array. Failover occurs seamlessly and automatically, requiring no administrator intervention. Uniform connected hosts: • After a brief pause, local and remote host I/O recommences through the operational array. • Storage paths through the failed array will cease functioning. Non-uniform connected hosts: • After a brief pause, hosts at the surviving array site resume I/O through the operational array. • Hosts at the failed array site lose access to volumes. • Host cluster software may restart applications on hosts at the surviving array's site, potentially automatically.

TABLE 3 Array, Replication Network, and Site Failures

Host failures

Oracle RAC efficiently manages host failures, regardless of ActiveCluster. When an Oracle RAC node fails, all sessions running on the affected node are terminated unless transparent application continuity is configured and enabled at the service level. The RAC, Database, and Application services seamlessly transition to other hosts within the same site or to hosts in the other site connected to the alternate array, as guided by Oracle Clusterware.

For Oracle Standalone setups, the host represents a single point of failure, requiring the implementation of a disaster recovery plan for Oracle. Host failure behavior remains consistent, whether the Oracle is configured in a uniform or non-uniform access topology.

Conclusion

ActiveCluster provides a compelling solution for organizations seeking a robust and cost-effective disaster recovery solution for Oracle databases. Its simplified deployment, seamless operation, and reduced storage capacity requirements make it the ideal choice for maximizing application uptime and data availability.

Below are several notable advantages of using ActiveCluster:

- **High Availability:** ActiveCluster ensures continuous application uptime with automatic, transparent failover between two geographically separated FlashArray systems.
- **Synchronous Replication:** Data consistency is guaranteed with synchronous replication, minimizing recovery time objectives (RTOs) in case of disasters.
- **Simplified Deployment and Management:** ActiveCluster integrates seamlessly with existing infrastructure and requires minimal configuration, reducing administrative overhead.
- **Enhanced Performance:** By utilizing all available paths for data access, ActiveCluster can deliver improved performance for demanding RAC workloads.
- **Reduced Storage Capacity Requirements:** ActiveCluster eliminates the need for dedicated disaster recovery storage, resulting in significant cost savings.
- **Flexibility:** ActiveCluster can be deployed within or across multiple sites, providing organizations with the flexibility to meet their specific needs.

By leveraging ActiveCluster, organizations can achieve the high availability, disaster recovery, and performance required to support their mission-critical applications.

In addition to the benefits mentioned above, ActiveCluster also offers a number of other advantages, including:

- **Simplified licensing:** ActiveCluster is included with Purity//FA software, eliminating the need for additional licenses.
- **Non-disruptive upgrades:** Existing FlashArray customers can upgrade to ActiveCluster without any downtime or data migration.
- **Seamless integration with Pure1 Cloud Mediator:** ActiveCluster integrates with Pure1 Cloud Mediator, a cloud-based service that provides centralized management and monitoring of Pure Storage arrays.

With its comprehensive set of features and benefits, ActiveCluster is the clear choice for organizations seeking a reliable, high-performance, and cost-effective solution for Oracle database.

References

- ActiveCluster Requirements and Best Practices
- Oracle Database Recommended Settings for FlashArray

PS2540-01-en 12/23