Technical Report

Oracle Databases on Microsoft Azure

Using Azure NetApp Files

Jeffrey Steiner, Prabu Arjunan, Geert van Teylingen, NetApp
Andrew Chen, Tim Gorman, Kellyn Gorman, Microsoft
July 2020 | TR-4780 | Version 2

In partnership with

Microsoft

Abstract

This document provides best practices with respect to leveraging Azure NetApp® Files for Oracle on Azure virtual machine (VM) deployments. It also details the different use cases, specific performance, data protection, and migration considerations with Oracle on Azure NetApp Files.

NetApp®

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 Oracle on Microsoft Azure Overview

Today, many customers are using Microsoft Azure to accelerate their Oracle deployments to reduce costs and provide increased agility for their business processes. These issues are of paramount importance for IT leaders who have a Cloud First strategy. Moreover, integrating Oracle with the Azure suite of platform as a service (PaaS) services such as Azure Data Factory, Azure Internet of Things (IoT) Hub, and Azure Machine Learning can create business value and support digitalization.

More large enterprises choose Azure as the cloud platform of choice for their enterprise applications, including Oracle. Many customers embraced the DevOps paradigm by first moving their test and development systems. Recently, however, customers are choosing to migrate their complete Oracle infrastructures, including production, into the cloud.

Azure's vast compute offerings range from small to large virtual VMs for the most demanding database workloads. Microsoft introduced the Azure M-Series VMs in 2018 with up to 6TB of memory and, recently, 12TB VMs were announced. These colossal VMs are targeted at specific workloads such as high-performance computing (HPC), SAP HANA, and Oracle.

1.1 NetApp Values and Solutions on Microsoft Azure

For many customers, the NetApp storage and data management solutions, based on the NetApp ONTAP® software, are the foundation for their enterprise workloads such as Oracle. For more than 20 years, the ONTAP system and its NFS services have been used in many of the largest and most mission-critical Oracle deployments to enable secure and stable operations. This solution simplifies data management, accelerates projects, and reduces risk.

As a global Oracle technology partner, NetApp has a long history of providing excellent solutions and products with a deep integration into Oracle Database environments. This partnership enables customers to use NetApp Snapshot™ technology for fast, storage efficient, and reliable backup and recovery. It also provides fast and storage efficient cloning for quicker time to market while improving quality. These fully supported products help Oracle customers to automate a comprehensive backup and disaster recovery strategy while also considering other important workflows. You can focus on the complete Oracle application lifecycle management by using Snapshot-based cloning operations.

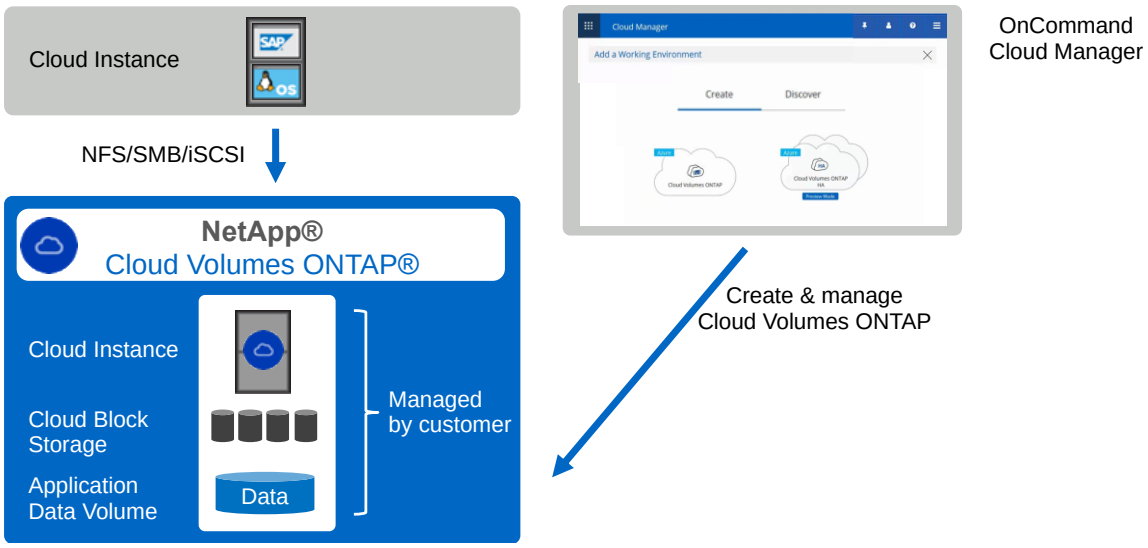
Many Oracle customers who want to move their systems to the cloud still want to use the NetApp storage benefits for their Oracle projects and operations. Customers do not want to give up on the performance, reliability, and enterprise data management capabilities when moving enterprise file-based workloads to the cloud. Not every cloud can offer a highly available, enterprise-grade, fast, reliable, feature-rich, but simple to manage shared file service based around NFS, which is required for those Oracle environments.

On Azure, customers can now benefit from two distinct ONTAP based offerings to build their Oracle systems on. Although the following sections provide a brief overview of both solutions, NetApp Cloud Volumes ONTAP and Azure NetApp Files, most of this document will focus on Azure NetApp Files only. For detailed description of Oracle on Cloud Volumes ONTAP, see TR-4691: Oracle Databases on ONTAP Cloud with Microsoft Azure.

Cloud Volumes ONTAP on Azure

Cloud Volumes ONTAP extends the trusted enterprise data management capabilities of ONTAP to leading cloud platforms such as Microsoft Azure. In Azure, it provides SMB/NFS/iSCSI-based services to hosted Oracle workloads. By leveraging the underlying Azure storage and compute resources Cloud Volumes ONTAP adds storage efficiency features such as thin provisioning, deduplication, compression, and now tiered storage to Azure Blob storage as well.

Figure 1) Cloud Volumes ONTAP.



Cloud Volumes ONTAP is NetApp's proven data management software running in a cloud instance using Cloud Storage. For the initial provisioning, customers need to install OnCommand Cloud Manager, as shown in Figure 1. You can then use Cloud Manager to deploy and manage multiple Cloud Volumes ONTAP instances, configured either as single nodes or highly available dual node configurations. Customers can use Cloud Manager to manage Cloud Volumes ONTAP in Azure, on-premises ONTAP systems and even Cloud Volumes ONTAP instances at other datacenters and even other cloud providers. When provisioning Cloud Volumes ONTAP, customers can select from different system classes and license types. This will define the maximum storage capacity and performance.

Customers can provision their 'data volumes' and shared files to the Cloud Instance to run their Oracle application and databases. Customers new to NetApp can use Cloud Manager for this provisioning while customers used to NetApp can use all the NetApp tools and workflows they are using in their on-premises data centers.

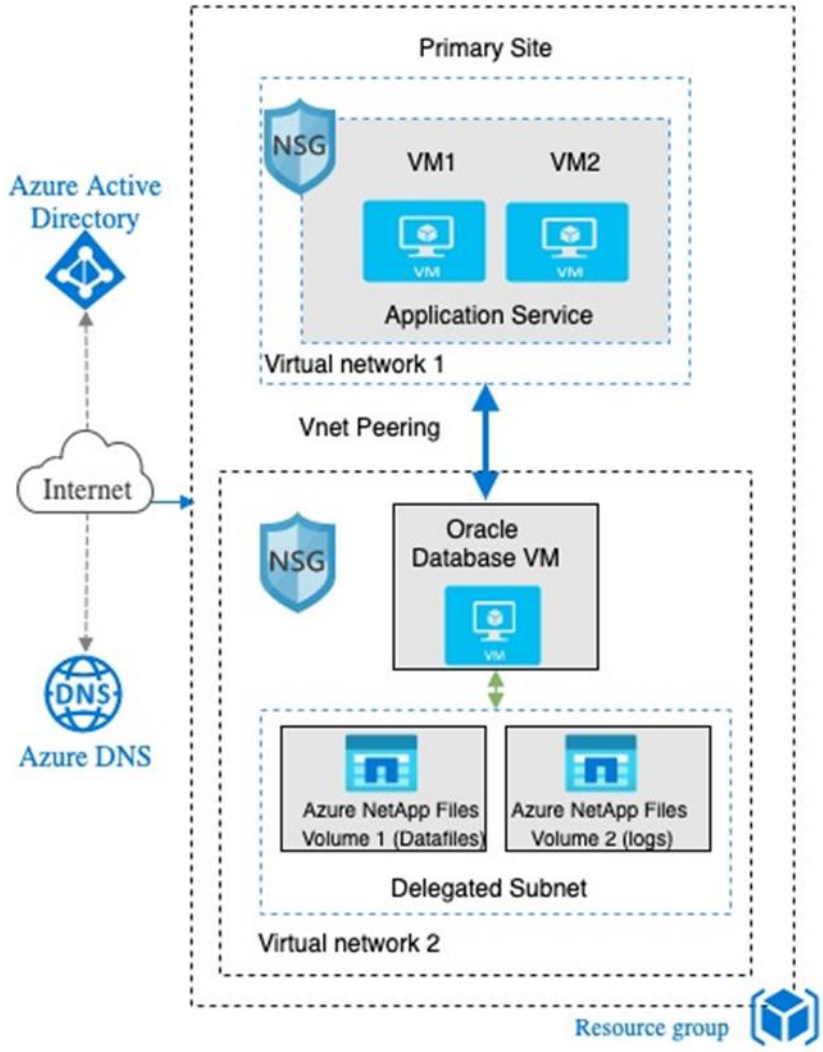
Azure NetApp Files

Azure NetApp Files delivers simplified data management and unmatched performance for Linux and Windows file-based applications. With Azure NetApp Files, you can get a fully managed, native file share service in the cloud that is simple to deploy and doesn't compromise performance, scale, availability, and resiliency.

Azure NetApp Files is completely integrated into the Azure DCs and portal, and customers can use the same comfortable graphical interface creating and APIs for creating and managing shared files as with any other Azure object, as shown in Figure 2. Azure NetApp Files provides NetApp enterprise-class storage and delivers many of the data management capabilities such as easy creation and resizing of volumes, adapting capacity and performance without downtime, creating space efficient storage snapshots and clones in seconds that are very valuable to use to optimize SAP operations.

As a comparison to Cloud Volumes ONTAP, Azure NetApp Files is built on the NetApp proven ONTAP storage hardware hosted in Azure data centers and operated and maintained by Microsoft directly. This results in high storage performance in combination with low latency I/O.

Figure 2 Azure NetApp Files.



Comparison

Table 1 shows the difference in level of management for the various options for ONTAP based systems, ranging from on-premises (NetApp AFF and FAS ONTAP) to IaaS (Cloud Volumes ONTAP) to PaaS (Azure NetApp Files). Clearly, Azure NetApp Files gives the best on-demand cloud service experience while providing on-premises-like performance. For users that require cloud-based storage with a higher level of management control and require access to all ONTAP features, Cloud Volumes ONTAP might provide the best experience.

Table 1) NetApp ONTAP offerings.

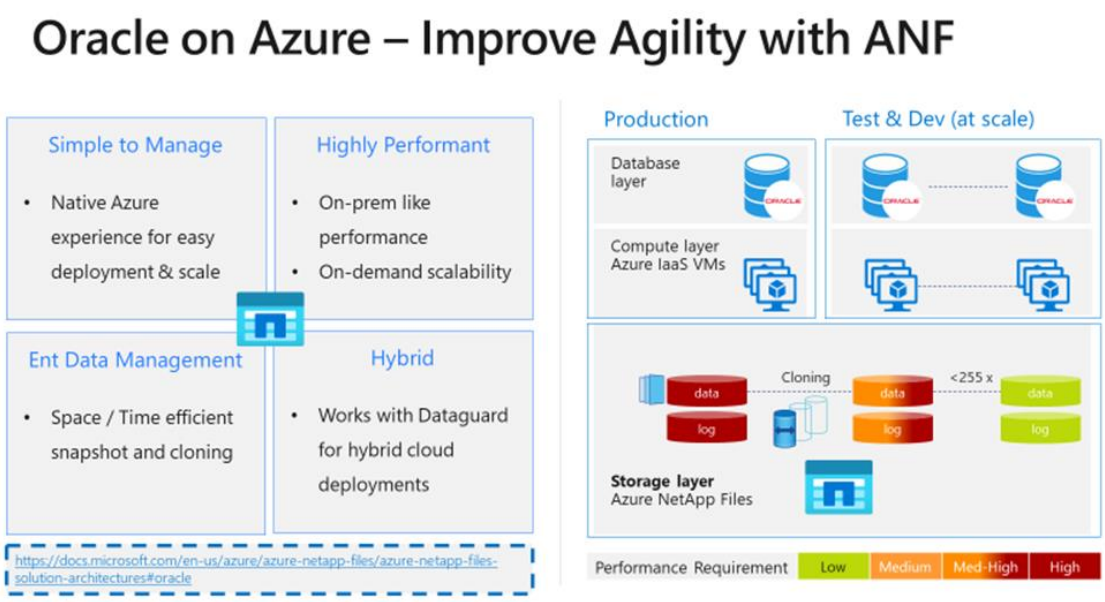
	On-Premises ONTAP AFF Full Control and Ownership	Cloud Volumes ONTAP Full Control and SW Ownership	Azure NetApp Files Service Consumption
Mount volumes	Customer	Customer	Customer

	On-Premises ONTAP AFF Full Control and Ownership	Cloud Volumes ONTAP Full Control and SW Ownership	Azure NetApp Files Service Consumption
Provision volumes	Customer	Customer	Customer
Performance	High	Low, medium	High
Space, time-efficient Snapshot copies	Yes	Yes	Yes
Space, time-efficient cloning	Yes	Yes	Yes
Deduplication and compression	Yes	Yes	No
Protocols	NFS, SMB, iSCSI, and so on	NFS, SMB, and iSCSI	NFS and SMB
Endpoint replication/migration	SnapMirror, Cloud Sync	SnapMirror, Cloud Sync	Cloud Sync
Encryption	Yes (customer-managed)	Yes (customer-managed)	Yes (Azure-managed)
Purchasing commitment	Typically, 3–5 years	• Paygo (hourly) • BYOL (yearly)	Paygo (hourly)
Disk capacity planning	Customer	Azure	Azure
ONTAP upgrades	Customer	Customer	Azure
ONTAP service deployment	Customer	Customer on Azure IaaS	Azure
Hardware deployment/refresh	Customer	Azure	Azure

1.2 Azure NetApp Files Key Value Proposition

Azure NetApp Files was designed to meet the core requirements of running high-performance workloads such as databases in the cloud, and provides performance tiers that reflect the real-world range of IOPS demands, low latency, high availability, high durability, manageability at scale, and fast and efficient backup and recovery and cloning. These capabilities are possible because Azure NetApp Files is based on physical all-flash NetApp ONTAP systems running within Azure data center environment.

Figure 3) Oracle on Azure – improve agility with Azure NetApp Files.



The result is an ideal database storage technology that can be provisioned and consumed just like other native cloud storage options:

- **Simple and reliable.** Azure NetApp Files is built as a simple-to-consume Azure native platform service with the power of the ONTAP reliability features. This feature enables customers to quickly and reliably provision enterprise-grade NFS volumes for their Oracle environments.
- **Enterprise performance.** High-performance, all-flash ONTAP enterprise systems are built into the Azure data centers and fully integrated into the Azure SDN and ARM frameworks. Customers get the on-premises like, high-IO/low-latency shared storage performance necessary for the highest demanding enterprise workloads like Oracle. Find Oracle benchmarks on the Azure NetApp Files Benchmarks page (click on Oracle tab) and on the Benefits of using Azure NetApp Files with Oracle Database page.
- **Enterprise data management.** This service is targeting the most demanding, mission-critical applications and workloads that typically require advanced data management capabilities. ONTAP's capabilities in this space – with Time-/Space efficient snapshot and cloning, on-demand capacity and performance scaling, efficient replication – are unmatched in the industry. Now with Azure NetApp Files the same capabilities of a native platform service are available in Azure.
- **Hybrid cloud.** Many customers have been on the journey into the cloud for quite some time or are even staying in a hybrid operating model for the foreseeable future. An efficient data replication/migration capability is paramount in these cases. Use of Oracle Data Guard and Azure NetApp Files' integration with NetApp Cloud Sync replication service extends this hybrid operational model, further playing into our Intelligent Cloud/Intelligent Edge strategy.

This document addresses the requirements for operating an Oracle Database on Azure NetApp Files in two ways. First, when a clear best practice exists, it is called out specifically. Second, this document reviews the many design considerations that must be addressed by architects of Oracle storage solutions based on their specific business requirements.

Increased Resilience with Snapshot Copies

You can easily create a snapshot copy of an Oracle Database using NetApp Snapshot technology. There are multiple ways to protect data, snapshots are only one of the options. Snapshot copies act as logical

backups. They're point-in-time representations of your data, with a rapid revert function that allows you to restore your database to an earlier point in time nearly instantaneously. You can create snapshot copies manually or schedule their creation using the Azure NetApp Files API or GUI. If there is a need to use a snapshot, a customer can rapidly revert using the API.

Snapshots can be restored in two different methods, one being "Revert volume" and another being restore to new volume. Snapshot copies are fast, plentiful, and nondisruptive. A snapshot copy in Azure NetApp Files simply manipulates block pointers, creating a "frozen" read-only view of a volume that enables your applications to access older versions of files and directory hierarchies without special programming. Snapshot copy creation takes only a few seconds (typically less than 1 second) regardless of the size of the volume or the level of activity within the environment. Since they are read-only, block-level incremental copies, you only pay for the space consumed by new data written.

Speed Up Time to Market: Spin Up Cloud Volumes in Seconds with Instant Copy

Most organizations need multiple copies of data for testing and development. Oracle landscapes are littered with system copies for variety of uses; creating and refreshing those copies are cumbersome. Typically, creating copies of Oracle landscapes is a time-consuming and tedious process. Azure NetApp Files allows you to instant copy the database files, drastically improving the process of copying, backing up, and reverting. The process takes almost no time, which ultimately leads to lower costs by way of a quicker time to market.

High Availability and Data Durability

With Azure NetApp Files – besides having a standard availability of 99.99% – data is protected not just against multiple drive failures, but also against numerous storage media errors that can harm your data durability and your data integrity.

Security and Encryption

Azure NetApp Files gives you FIPS-140-2-compliant data encryption at rest, role-based access control (RBAC), Active Directory authentication (enabled for SMB), and export policies for network-based access control lists. Azure NetApp Files also enhances data security by presenting mount points only within a virtual private cloud, and not as a public IP address. Azure NetApp Files is built to meet demanding Azure security standards, which has helped Azure achieve more compliance certifications than any other cloud provider.

Support for Hybrid Scenarios

Azure NetApp Files enables easy data migration across on-premises and cloud infrastructures using Cloud Sync, a NetApp service for rapid, security-enhanced data synchronization. Cloud Sync simplifies lift and shift migrations and DevOps use cases, with capabilities like instantaneous snapshot copy creation and restore, as well as Active Directory integration (SMB only). These features work as well—and in the same way—on-premises as they do in the cloud. Integrated data replication and backup features will be available in the near future. Learn more about Cloud Sync.

These capabilities are possible because Azure NetApp Files is based on physical all-flash NetApp ONTAP systems running within Azure data center environment. The result is an ideal database storage technology that can be provisioned and consumed just like other native cloud storage options.

This document addresses the requirements for operating an Oracle Database on Azure NetApp Files in two ways. First, when a clear best practice exists, it is called out specifically. Second, this document reviews the many design considerations that must be addressed by architects of Oracle storage solutions based on their specific business requirements.

1.3 Azure NetApp Files Enabled New Ways of Working in Public Cloud

Azure NetApp Files is crucial for success when deploying the Oracle Database on the Azure public cloud. On-premises, resources such as I/O and networking were never constrained and rarely metered. In general, the expectation was that hardware resources were to be used fully, and that unless chargeback was part of the business, there was no need for metering.

Azure public cloud has been designed from the ground up to manage those expectations to ensure that the customers get what they pay for. All resources are proactively constrained by design, preventing unexpected outages and slowdowns due to sudden exhaustion. Like on-premises, there is a limitation to the number of vCPUs provisioned in Azure, to the volume of vRAM provisioned and to the capacity of the storage provisioned. Another familiar expectation is the fact that the capacity of individual storage devices imposes constraints on the rate of I/O requests and the throughput rate of each storage device; larger storage devices permit more I/O requests and more data. All of this is close to our experiences on-premises, which means we can re-use assumptions and habits used across decades on-premises.

But wait! There is something different to consider.

VMs in public cloud have cumulative limits on the rate of I/O requests and cumulative limits on the total volume of data transferred.

This is a new and different challenge. Rather than having the freedom to continuously push a server to its limits on I/O, instead there is a constrained well short of the capabilities of the hardware. Also, typically consuming I/O incurs a significant cost, which becomes highly unpredictable relatively quick.

The limits are specific to each VM instance type. An 8-vCPU VM instance type might be limited to only 12,800 IOPS (I/O requests per second) and only 192MBps of data transfer for both reads and writes. For an Oracle Database running on such a server, these limits might be easily reached, and therefore present a performance risk.

However, on the same 8-vCPU VM instance type, you might notice that network bandwidth is 4000Mbps. With a little arithmetic, 4000Mbps translates to 500MBps, which means that network-attached storage over NFS or CIFS can perform 250% more data transfers than the permitted under the I/O limits on this VM instance type.

Right away, just as storage for database files, Azure NetApp Files (which runs on NFS) is a solution for I/O demands beyond the cumulative I/O limits for a VM.

And yet there's more to consider…

It is easy to forget to consider the impact of backups and database cloning operations on duration and the rate of I/O requests and the volume of data transferred.

Azure NetApp Files presents features which were very useful but optional on-premises, such as storage-level snapshots used for database backups and for database cloning. A large number of customers adopted these mechanisms and benefited from the advantages they provided, but an equally large number of customers continued to backup and clone as they always had, by copying huge volumes of data from one place to another.

Deploying Oracle databases on Azure NetApp Files presents customers with this choice once again, but now the ground rules have changed subtly but significantly. To date, most try to fit day-to-day database operations within the cumulative VM limits for I/O, and frequently tend to forget to consider how relatively infrequent and easily overlooked operations like backups, restores, recovery, and cloning can "blow the budget" represented by the resource limits and associated cost. For these reasons, Azure NetApp Files is a good consideration for the standard recommendation when deploying the Oracle Database on Azure.

2 Azure NetApp Files Performance

2.1 Understanding Azure NetApp Files Performance and Capacity Tiering

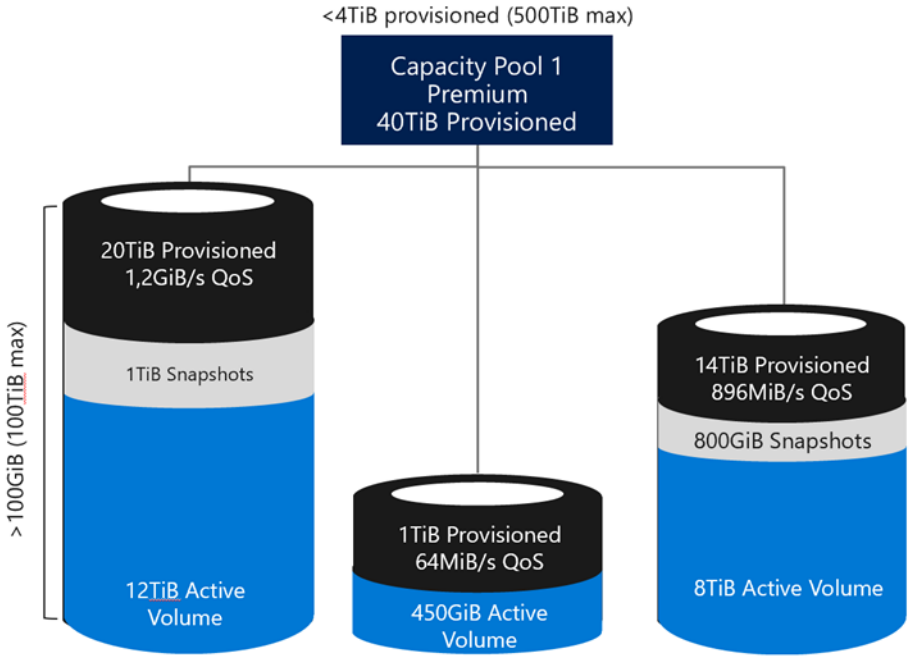
To understand how to optimize Azure NetApp File's capacity with regards to performance and costs, we need to take a closer look at how Azure NetApp Files is provisioned.

Azure NetApp Files volumes are allocated from a capacity pool the customer has to provision in his Azure NetApp Files storage account. Each capacity pool is assigned:

- To a service level that defines the overall performance capability.
- The initially provisioned storage capacity for that capacity pool.

Figure 4 illustrates the Azure NetApp Files performance and capacity tiering.

Figure 4) Azure NetApp Files performance and capacity tiering.



The performance of a volume is based on the capacity pool service level, in combination with the number of TiBs provisioned for that volume. Customers can dynamically grow and shrink the volumes and pool capacity, manage performance, and manage capacity. Billing is based on the provisioned capacity pool, on an hourly base. Each of the service levels available has an associated cost per provisioned capacity and includes a quality of service (QoS) level that defines the overall maximum throughput per provisioned space. For example, a 10TiB provisioned single capacity pool with premium service level will provide an overall available throughput for all volumes in this capacity pool of 10x 64MBps, so 640MBps, or 40,000 (16K) resp. 80,000 (8K) IOPs. For more information, see Cost model for Azure NetApp Files.

Within a capacity pool each volume is provisioned with a specific quota between 100GB up to the maximum volume size. This quota defines the maximum throughput/IOPs for this volume.

2.2 Service Levels for Azure NetApp Files

Service levels are an attribute of a capacity pool. Service levels are defined and differentiated by the allowed maximum throughput for a volume in the capacity pool based on the quota that is assigned to the volume.

Supported Service Levels

Azure NetApp Files supports three service levels: Ultra, Premium, and Standard.

- **Ultra storage.** This tier provides up to 128MiB/s of throughput per 1TiB of volume quota assigned.
- **Premium storage.** This tier provides up to 64MiB/s of throughput per 1TiB of volume quota assigned.
- **Standard storage.** This tier provides up to 16MiB/s of throughput per 1TiB of volume quota assigned.

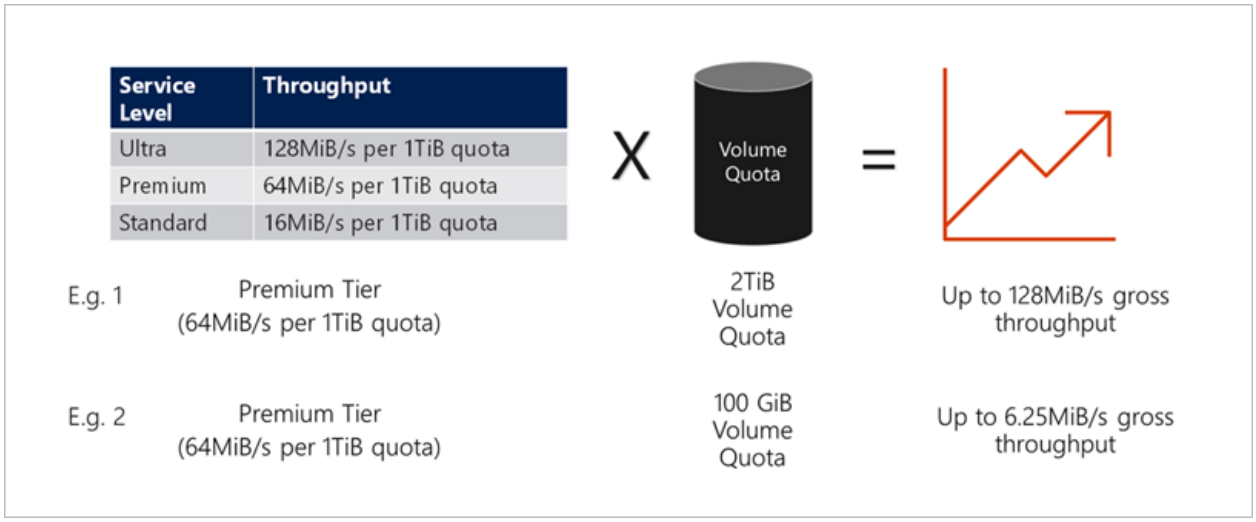
Throughput Limits

The throughput limit for a volume is determined by the combination of the following factors:

- The service level of the capacity pool to which the volume belongs
- The quota assigned to the volume

This concept is illustrated in Figure 3.

Figure 3) Throughput limits.



In example 1, a volume from a capacity pool with the Premium storage tier that is assigned 2TiB of quota will be assigned a throughput limit of 128MiB/s (2TiB * 64MiBps). This scenario applies regardless of the capacity pool size or the actual volume consumption.

In example 2, a volume from a capacity pool with the Premium storage tier that is assigned 100GiB of quota will be assigned a throughput limit of 6.25MiBps (0.09765625TiB * 64MiBps). This scenario applies regardless of the capacity pool size or the actual volume consumption.

The following sections dive into the specific considerations and best practices around networking, NFS protocol, Oracle configurations, performance, and sizing.

3 Oracle Deployments on Azure NetApp Files

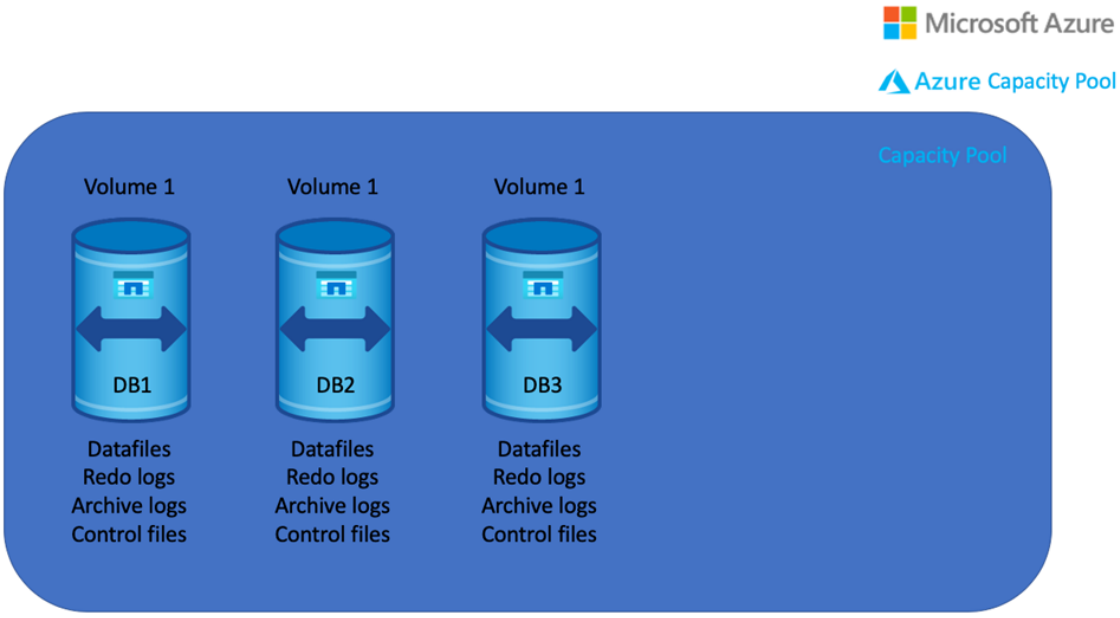
3.1 Azure NetApp Files Volume Layout for Oracle Databases

The volume layout for Oracle Database can be designed based on the criticality of the workload. You can choose between these approaches:

- Dedicated volume layout
- Shared volume layout

The simplest possible layout (as shown in Figure 6) is the shared volume layout where all the data files, redo logs, control files, and archive logs are kept in the same volume. This approach has limitations in terms of fully utilizing the advanced features of Azure NetApp Files but might be appropriate for smaller or less business-critical databases.

Figure 6) Single volume database layout.

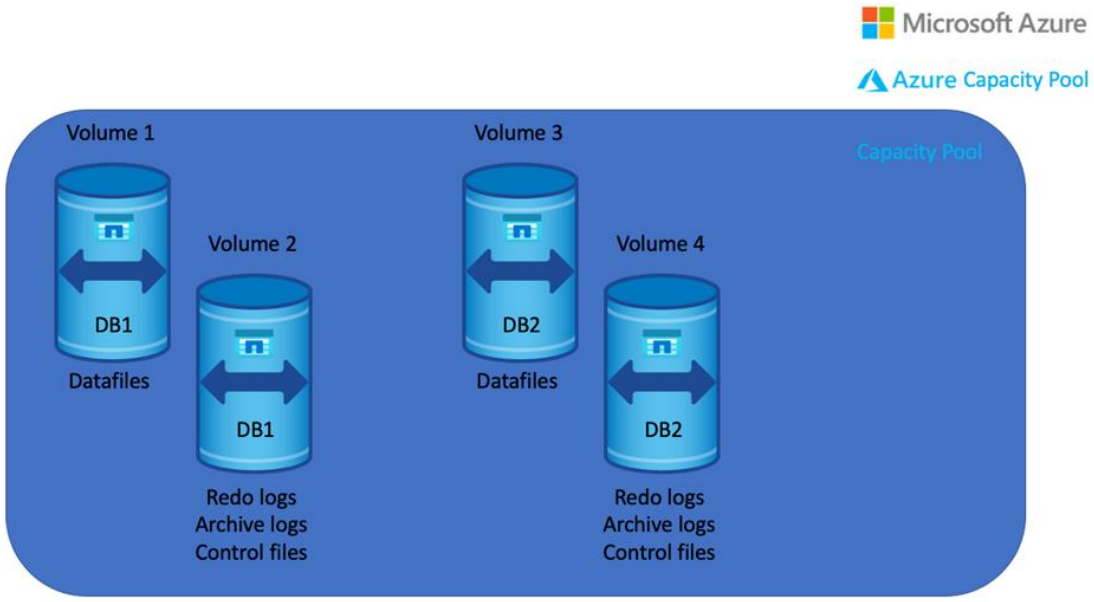


The primary benefit of this design is simplicity. All storage for a given database is sharing the same volume and associated performance capabilities.

It also addresses the provisioning challenge of hosting a database where some files consume a lot of space but have low performance needs while other files consume minimal space but are performance-intensive. Dividing a dataset into multiple volumes can strand performance or capacity on a volume where it could be better used by another volume.

The primary limitation of this approach is in backup and recovery procedures. Azure NetApp Files volumes can be instantly and efficiently backed up and restored, but if all the database files are in the same volume then restoration would restore not only datafiles but also critical redo and archive log data. This might not be a problem if a database only requires recoverability to the point of the backup itself, but almost all databases require specific point-in-time recovery. Datafiles could be restored individually and leave the logs required for replay untouched, but this is a more manual and time-consuming process. When speedy point-in-time recovery is required, the multivolume layout shown in the next example is superior and only requires a single additional volume.

Figure 7: Multivolume database layout.



Datafile Volumes

Datafiles are the most I/O-intensive file type in a database. Isolating the datafiles into a dedicated volume means the datafile volume QoS level can be set and dynamically adjusted based on IOPS needs.

Log Volumes

Redo log I/O is low compared to data file I/O, but redo log I/O is very latency-sensitive and the I/O can come in bursts. Averaging redo log I/O hides the true requirements during those bursts. Placing redo logs alone into a single volume is usually wasteful of capacity because the volume will need to be much larger than the logs themselves in order to deliver the required performance. In contrast, archive logs are written once and rarely touched again until their retention time is met and are then deleted.

Combining small, highly active files with large inactive files improves efficiency and reduces costs. Think of a volume as a pool of bytes and IOPS. The bytes required for archive log storage results in unneeded performance capacity that can be borrowed by the redo logs.

Placing the control files in the same volume enables Oracle snapshot-optimized backups. Snapshots of the log volume yields a perfectly consistent image of the redo, archive, and control file data. This can be used to easily make a snapshot of datafiles consistent, even if they were not in hot backup mode at the time of the snapshot. Instant and highly-efficient database backup and restore operations are possible with simple scheduled storage snapshots. No other software is required.

Finally, the two-volume approach delivers faster and more granular recoverability than a single volume approach. The procedure is simple – just restore the datafile volume to a point in time immediately before the desired recovery point, and then replay archive logs from the log volume. As discussed above, you don’t even need to use hot backup mode. Just use a snapshot policy with the frequency and retention times required for your desired RPO and RTO.

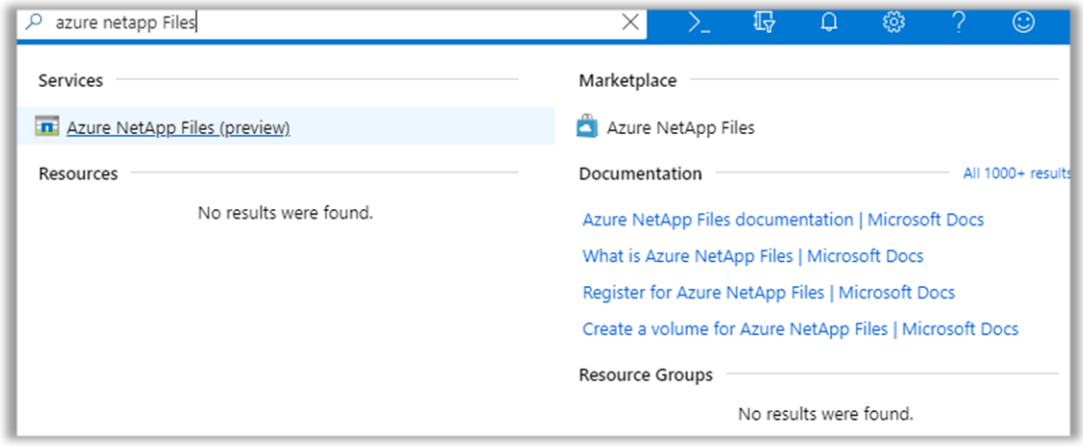
3.2 Azure NetApp Files Storage Provisioning for Oracle Database

Provisioning Azure NetApp Files storage for Oracle Database covers two parts:

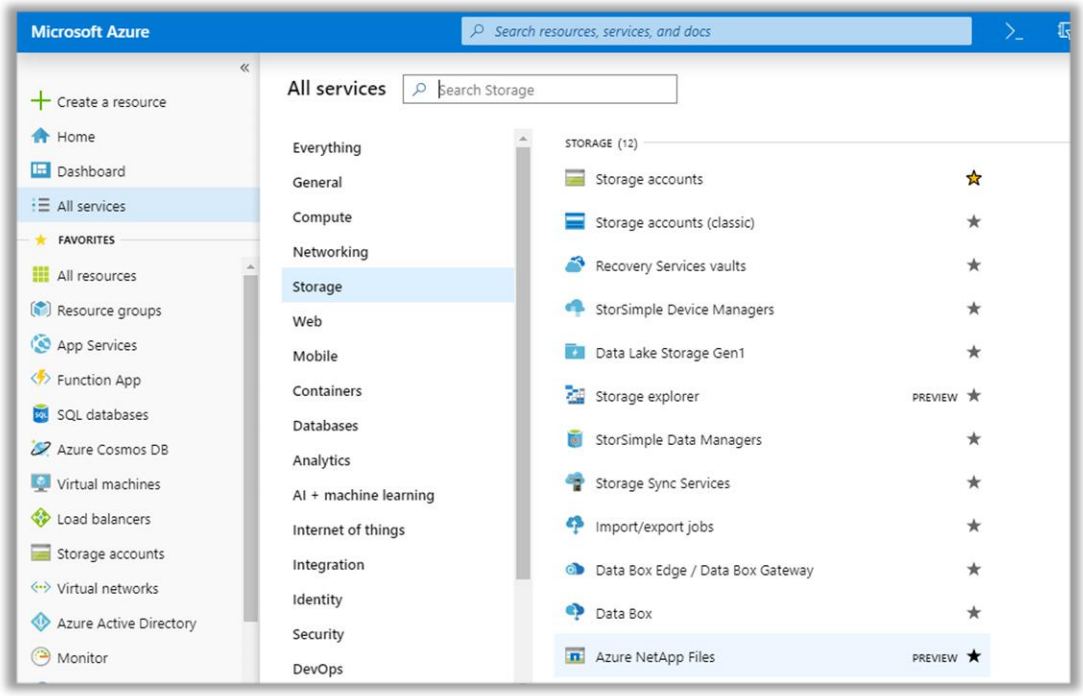
- Creating a NetApp account in Microsoft Azure portal
- Setting up capacity pools and volumes

Creating a NetApp account enables you to set up a capacity pool and then create a volume. You can log in to Azure portal and use any of the approaches to open a NetApp account.

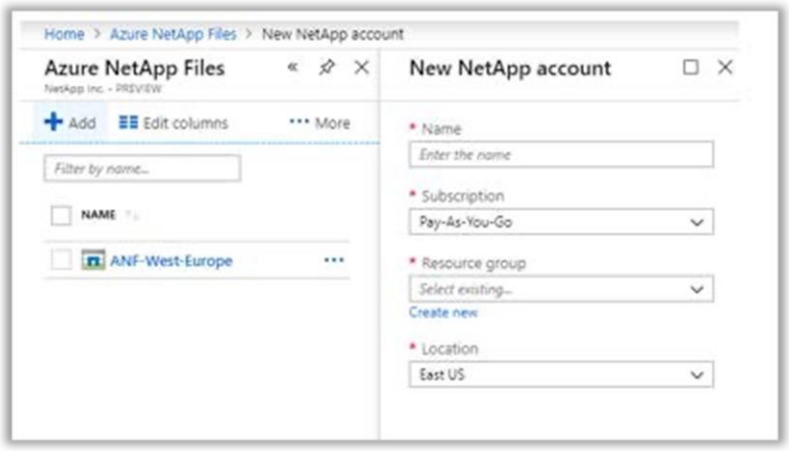
1. In the Azure portal search box, search for Azure NetApp Files.



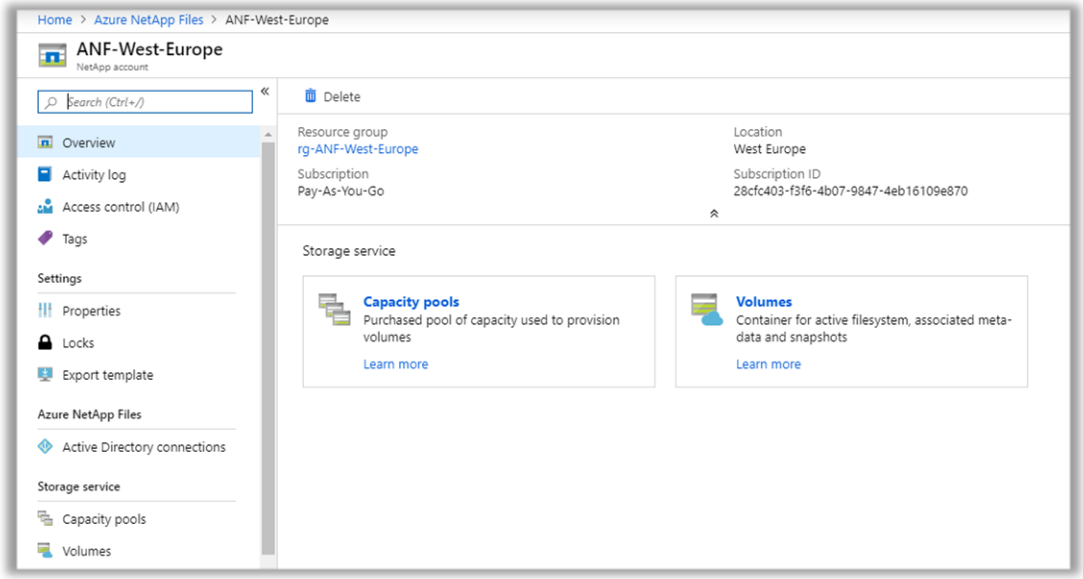
2. In the navigation pane, click All Services and then filter to Azure NetApp Files.



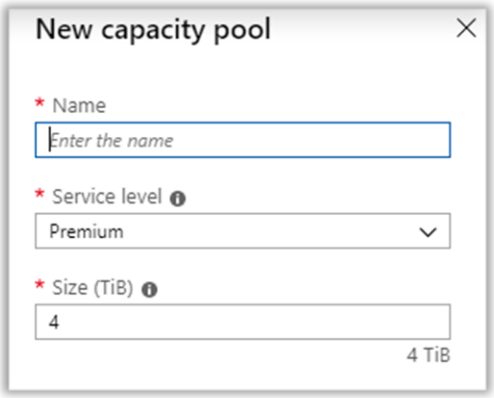
3. After launching the Azure NetApp files wizard, you can create an account by clicking Add.



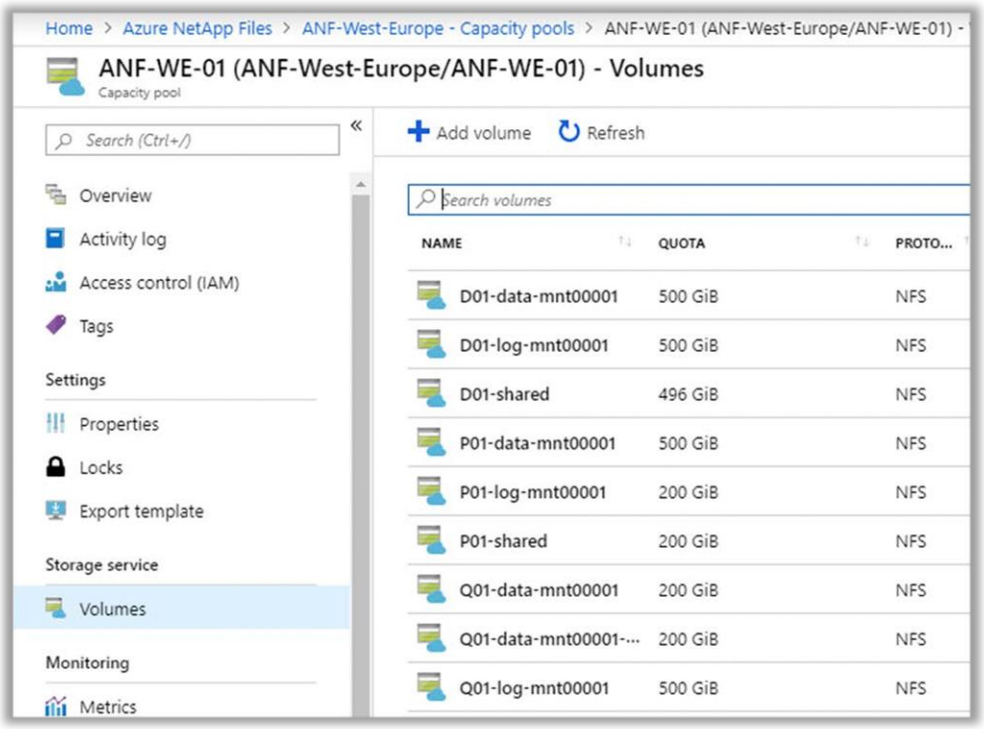
4. After the NetApp account is created, it should look similar to the following example:



5. Add a capacity pool and tag the appropriate service levels and then create the required volumes.



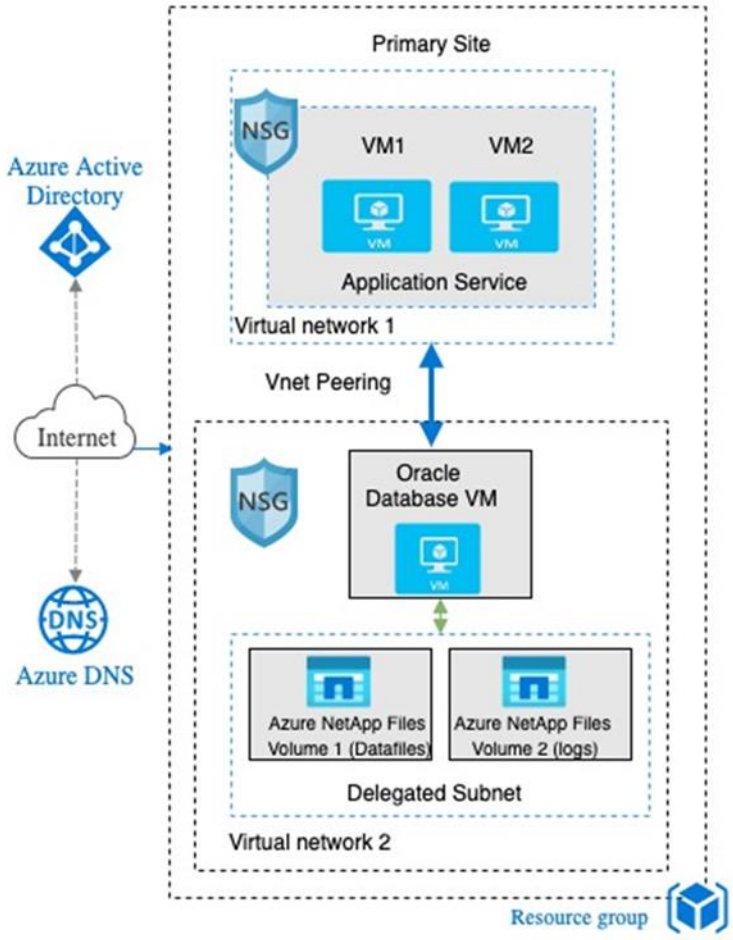
6. In this example, volumes are added to a given capacity pool. The volume layouts for any given workload should be based on best practices discussed in the next session.



3.3 Oracle Database on a Single VM

In this single VM architecture, you can see that Oracle Database data files and logs are configured on Azure NetApp Files. The dedicated volume layout, separate volume for data files and log files help the critical workloads that demand higher IOPS and lower latency. With the combination of snapshot copies, and right-sized throughput, you can easily host your high-performance database in the cloud with maximum data protection and nine 9s of data durability.

Figure 9) Oracle Database on a single Azure VM.

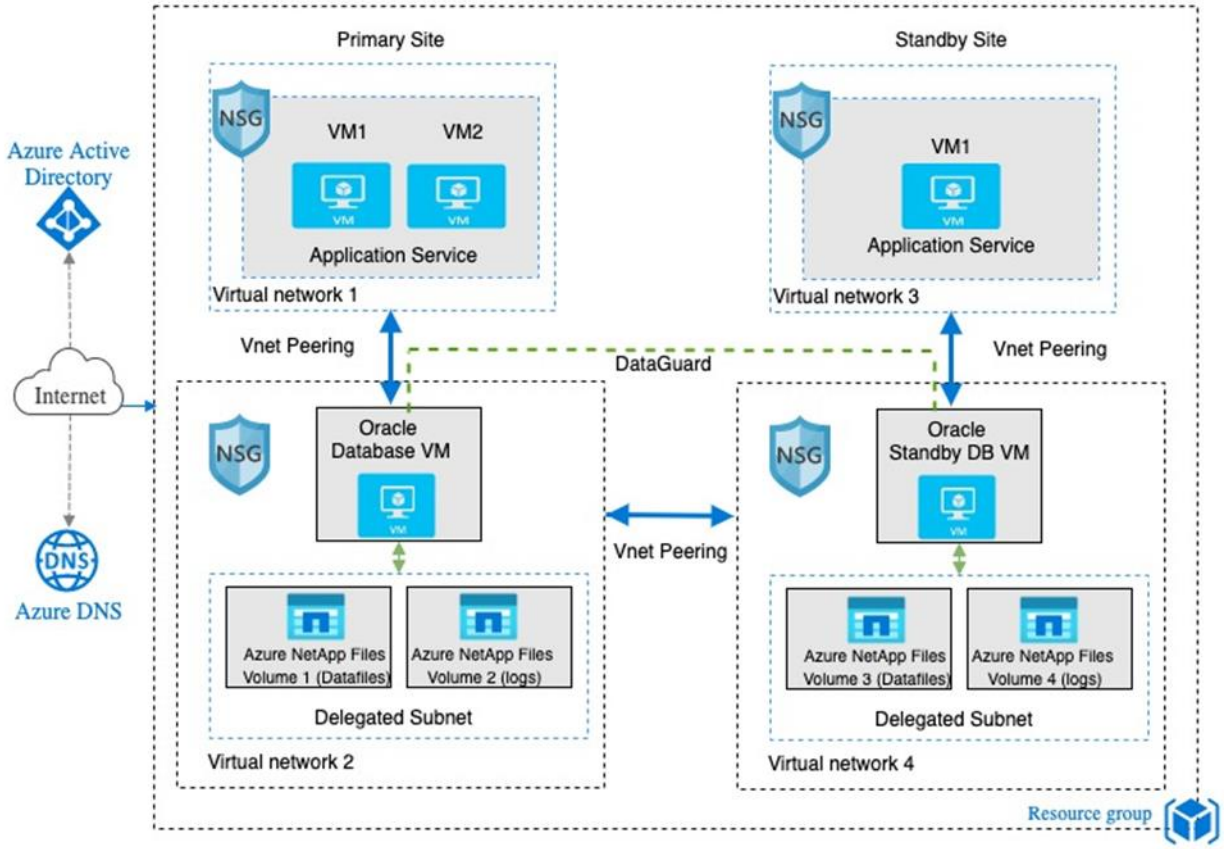


In Figure 9 you can see that Oracle Database is configured on an single Azure VM architecture. Single or multiple Azure NetApp Files volumes are used as the dedicated storage for the data files. An additional volume is dedicated to logs (archive logs, redo logs) and control files. The data file volumes are provisioned using the Ultra service level because that class provides the highest throughput at a manageable cost. A second volume is provisioned using the Premium service level.

3.4 Oracle Database High Availability on Azure NetApp Files

In this single VM architecture, you can see that Oracle Database data files and logs are configured on Azure NetApp Files. The dedicated volume layout, separate volume for data files and log files help the critical workloads that demands higher IOPS and lower latency. With the combination of snapshot copies, and right-sized throughput, you can easily host your high-performance database in the cloud with maximum data protection and nine 9s of data durability.

Figure 10) Oracle Database data files, archive logs, redo logs, and control files configured on Azure NetApp Files.



In Figure 10 you can see that Oracle Database data files, archive logs, redo logs, and control files are configured on Azure NetApp Files. The setup resembles the single instance Oracle Database diagram in Figure 9, except in this case, it includes a standby database. That database is set up on the second Azure VM, in a different virtual network, which was done by copying a primary database to the second instance. That availability increases when you have two cloud volumes. In tandem with Azure NetApp Files, the Oracle primary database is configured on an Azure VM in the first availability zone. The standby database is set up on the second Azure VM in a second availability zone by replicating the primary database to the second instance. A single cloud volume or multiple cloud volumes are used as the dedicated storage for the datafiles. An additional volume is dedicated to logs (archive logs, redo logs) and control files. The data volume is provisioned using the Ultra service level. The other volume is provisioned using the Premium service level.

Replication is managed by Oracle Data Guard. Options include guaranteed RPO=0 synchronous replication, synchronous replication that can drop back to unsynchronized mode in the event of the replication link is lost, and wholly asynchronous but very high-speed replication. In addition, Active Data Guard is a licensed option that allows the remote copy to be opened as a read-only database for reporting and other purposes. Data Guard also allows a database administrator (DBA) to easily reverse the primary-replica relation and is nearly transparent to running applications.

Data Guard does not deliver exactly the same HA capabilities as Oracle RAC, but RAC is not currently supported in any hyperscaler environment due to the lack of multicast IP support. For many customers, Data Guard is a comparable or even better replacement for Oracle RAC. Unlike RAC, Data Guard establishes two independent copies of a database. Failover times may be slower than RAC, but the data integrity is better protected by the second copy. For simple local node failure, the native capability of

Azure to bring up the Oracle VM on different hardware provides is generally sufficient. It's essentially a trade-off. Oracle in the Cloud with HA provides long-distance HA with better protection of the underlying data, at the cost of slightly longer recovery times for a simple VM crash.

4 Database Data Protection

Data protection is a critical part of any environment. The large part of a data protection strategy is ensuring that data can be restored quickly after any corruption or loss. The data protection architecture is defined by business requirements. These requirements include factors such as the speed of recovery, the maximum permissible data loss, and backup retention needs. The data protection plan must also take into consideration various regulatory requirements for data retention and restoration. Small changes in data protection and recovery policies can have a significant effect on the overall architecture of storage, backup, and recovery. It is critical to define and document standards before starting design work to avoid complicating a data protection architecture. Unnecessary features or levels of protection lead to unnecessary costs and management overhead, and an initially overlooked requirement can lead a project in the wrong direction or require last-minute design changes. By using Azure NetApp Files, customers can implement comprehensive data protection for their Oracle Database by using Azure NetApp Files' ONTAP storage-based Snapshot copies.

4.1 Solving the Challenges with Backups Using Snapshot Copies

The key benefits of the Azure NetApp Files storage-based Snapshot copies include:

- Snapshot copies can be created and kept on the service (for example, inside the original volume) with no performance impact on the storage service.
- Because these copies are created on the storage system, they don't consume database resources and no data needs to be copied out.
- Recovery from a data Snapshot copy is much faster than recovery from a data backup, resulting in aggressively short RTOs (down to minutes).
- Snapshot copies and restores are near-instantaneous, allowing frequent creation, which means even shorter RTO because only a few log backups need to be applied after restoring a recently created Snapshot copy.

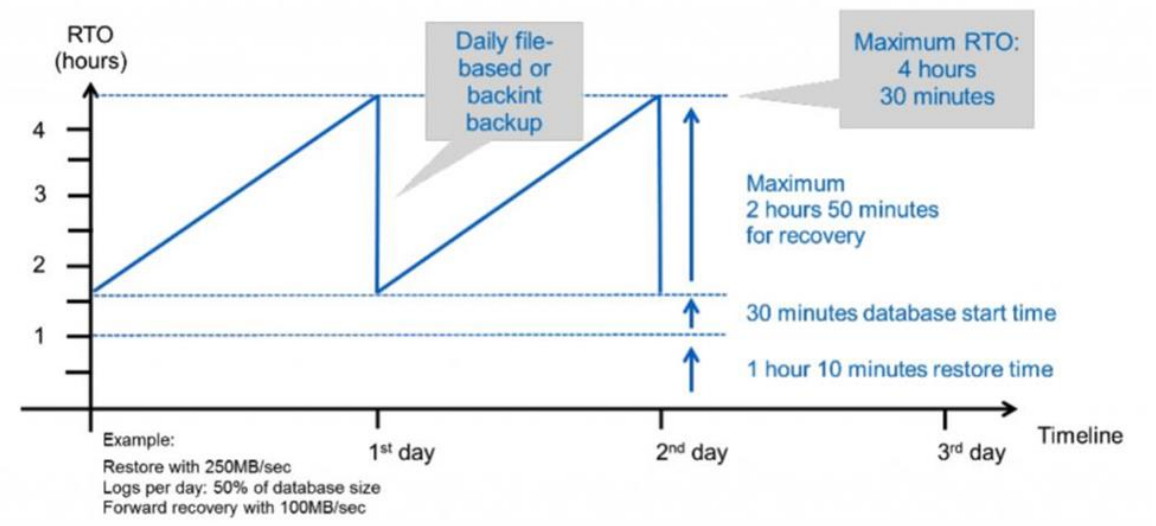
  Note: Snapshot copies serve as a first-line-of-defense and can account for the vast majority of the required restore operations of any given organization. However, Snapshot copies are not to be considered a complete backup. To cover all BU/R requirements typically external snapshot replicas and/or other backup copies must be created in a remote location.

The following sections illustrate these benefits.

Traditional Backup

In traditional backups, the complete database payload is streamed to the backup media. Given the large amount of data (and typically limited bandwidth), creation of the backups typically takes hours. As a result, often times it is possible to only complete one backup (or two backups at best) per day. In addition, restore operations usually take longer than the backup itself, so a lot of compute and network resources are being consumed, as shown in Figure 11.

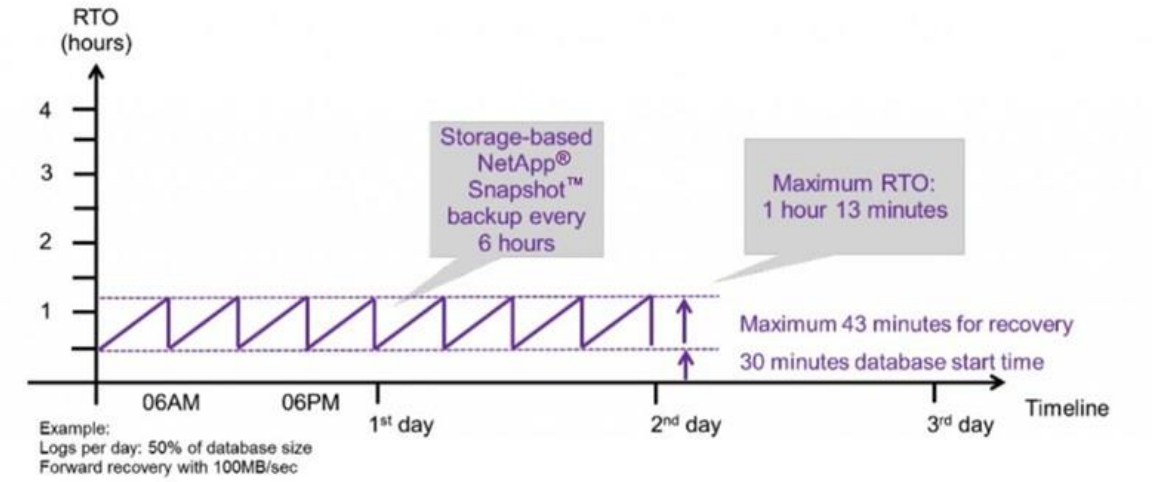
Figure 11) Restore operations timeline.



Snapshot Copy-Based Backup

With Snapshot copy-based backups, more backups can be created during the day. Data does not need to be copied on the storage service, therefore, creating a Snapshot copy takes only seconds. Regardless of data size, recovery from an ONTAP Snapshot copy is almost instantaneous and can be done from a fairly recent copy. Therefore, recovery time is shorter. Also, because of the more frequent creation of the Snapshot backups, fewer redo logs need to be applied after the Snapshot copy is restored.

Figure 12) Snapshot copy-based backup.



Regardless of the volume size, number of Snapshot copies kept, and the frequency of creation, Azure NetApp Files Snapshot copies based on ONTAP are fast (in seconds), space-efficient, and without performance impact. Instead of copying data, ONTAP marks the blocks on the active file system (volume) to be part of the new Snapshot copy and ensures that whenever a block is changed, the block is written to another empty location, preserving the snapped data block and avoiding any additional I/O. In other words, Snapshot copies are pointers to data blocks that allow restore operations to be fast as well, since only pointers are changed, and no data will be copied.

A Snapshot copy is a point-in-time file system image. Low-overhead Snapshot copies are made possible by the unique features of the WAFL® (Write Anywhere File Layout) storage virtualization technology that is part of Azure NetApp Files' Data ONTAP. Like a database, WAFL uses pointers to the actual data blocks on disk, but, unlike a database, WAFL does not rewrite existing blocks; it writes updated data to a new block and changes the pointer. An Azure NetApp Files Snapshot copy simply manipulates block pointers, creating a "frozen" read-only view of a WAFL volume that lets applications access older versions of files, directory hierarchies without special programming.

Because actual data blocks aren't copied, Snapshot copies are extremely efficient both in the time needed to create them and in storage space. An Azure NetApp Files Snapshot copy takes only a few seconds to create, regardless of the size of the volume or the level of activity on the Azure NetApp Files storage volume.

Meanwhile, the Snapshot copy of the data remains completely stable. An Azure NetApp Files Snapshot copy incurs no performance overhead; users can comfortably store up to 255 Snapshot copies per volume, all of which are accessible as read-only and online versions of the data (Figure 16).

Data in Azure NetApp Files Snapshot copies can be restored in three different ways:

- By copying files and directories from the read-only Snapshot copy folders in the `/.snapshot` directory of a volume
- By restoring a volume Snapshot copy to a new volume (thick clone)
- By reverting a volume from a Snapshot copy of a volume (Snapshot copy restore)

Snapshot technology forms the basis of a unique ecosystem of high-availability, disaster-tolerant, and data protection solutions, as shown in Figure 13 and Figure 14.

Figure 13) Snapshot copy creation.

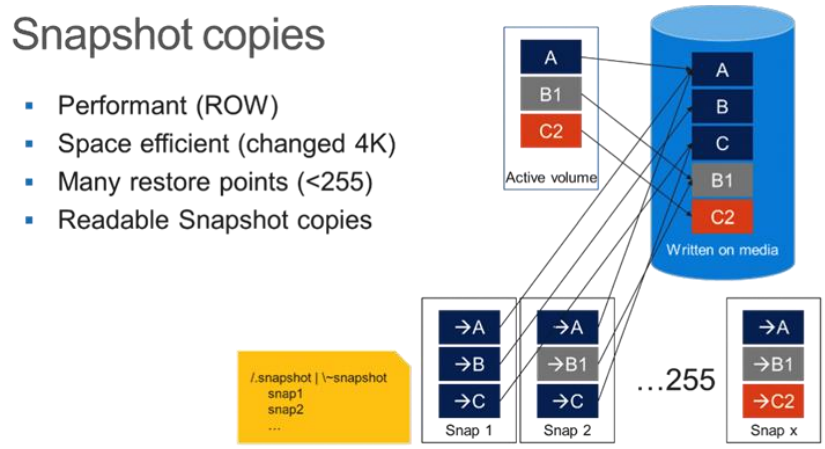
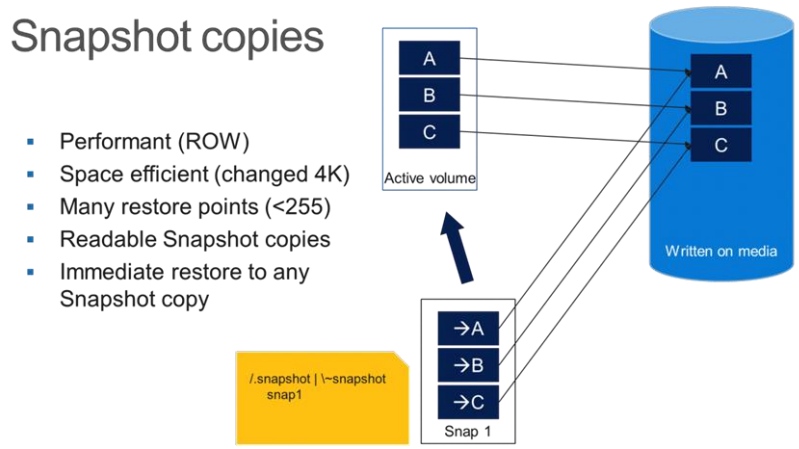


Figure 14) Snapshot restore operation.



4.2 Local Database Data Protection Architecture

ONTAP and Third-Party Snapshots

Oracle Doc ID 604683.1 explains the requirements for third-party snapshot support and the multiple options available for backup and restore operations.

The third-party vendor must guarantee that the company's snapshots conform to the following requirements:

- Snapshots must integrate with Oracle's recommended restore and recovery operations.
- Snapshots must be database crash consistent at the point of the snapshot.
- Write ordering is preserved for each file within a snapshot.

Note: ONTAP, Azure NetApp Files, and NetApp Oracle management products comply with these requirements.

Bring Back Your Oracle Database with In-Place Restore

The in-place help rolls back or fix corrupted storage through in-place restore without the need of spinning up a new volume. Customers can use this new capability by selecting Revert Volume from the Snapshot tab. The in-place restore or revert volume is comprised of the following steps:

1. Azure NetApp files take the snapshot of the volume based on the scheduled interval.
2. The snapshots are stored in the same volume.

Figure 4 shows the volume where Oracle Database is located.

Figure 4) Volume overview.

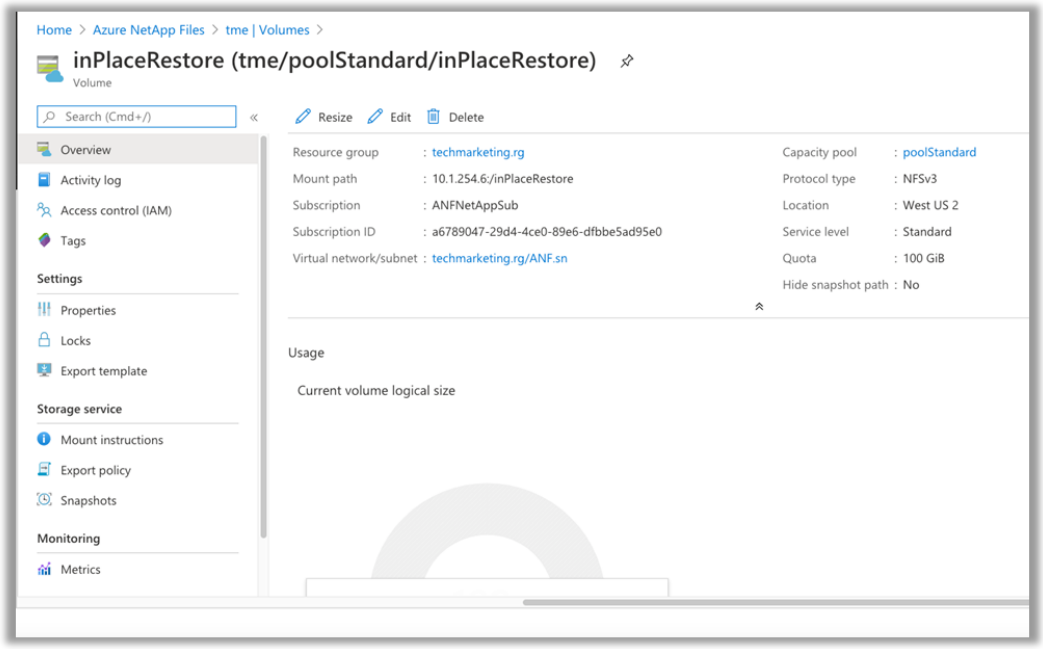


Figure 5 shows the volume mounted in the Oracle Linux machine.

Figure 5) List of mounted volumes.

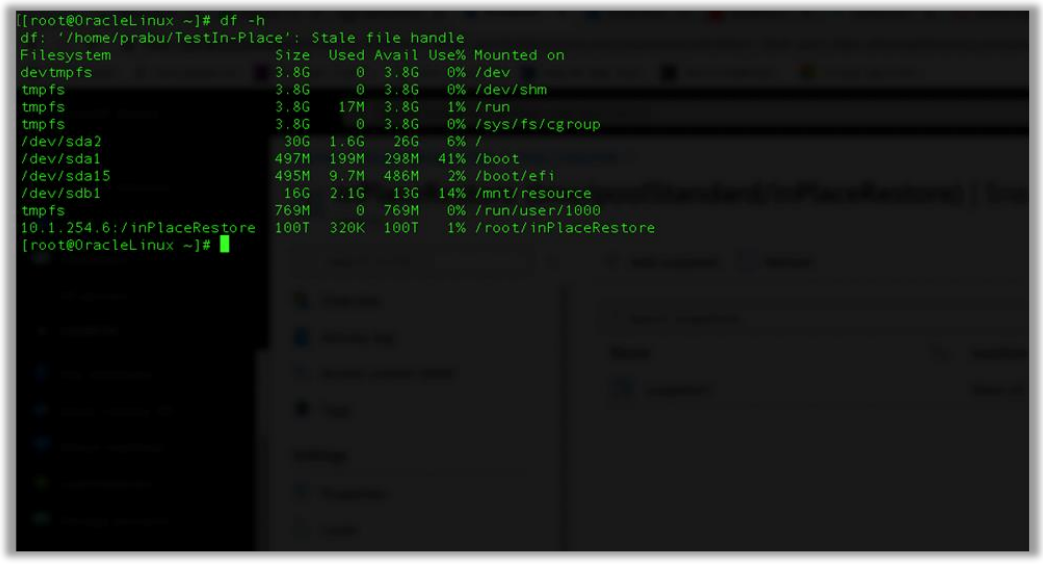


Figure 6 shows the Snapshots tab in the Azure NetApp Files.

Figure 6) Azure NetApp Files Volume Snapshot view.

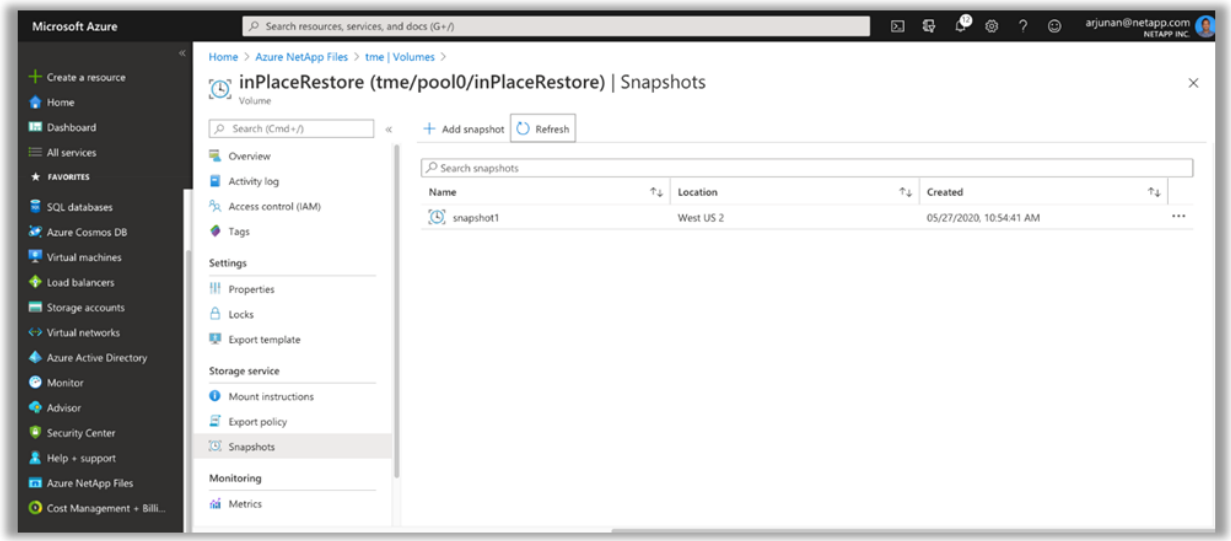
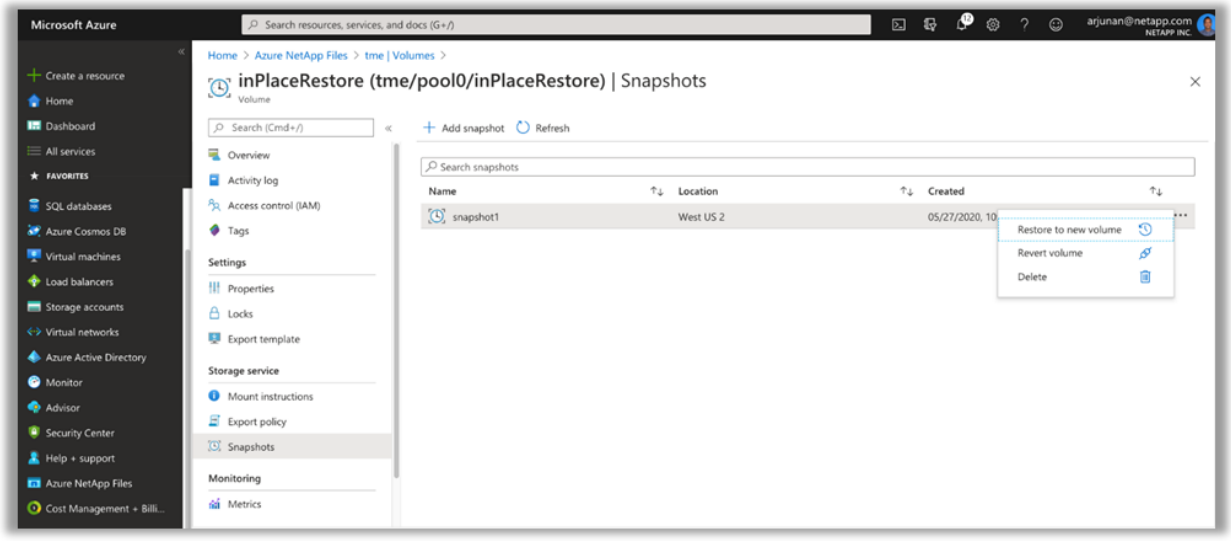


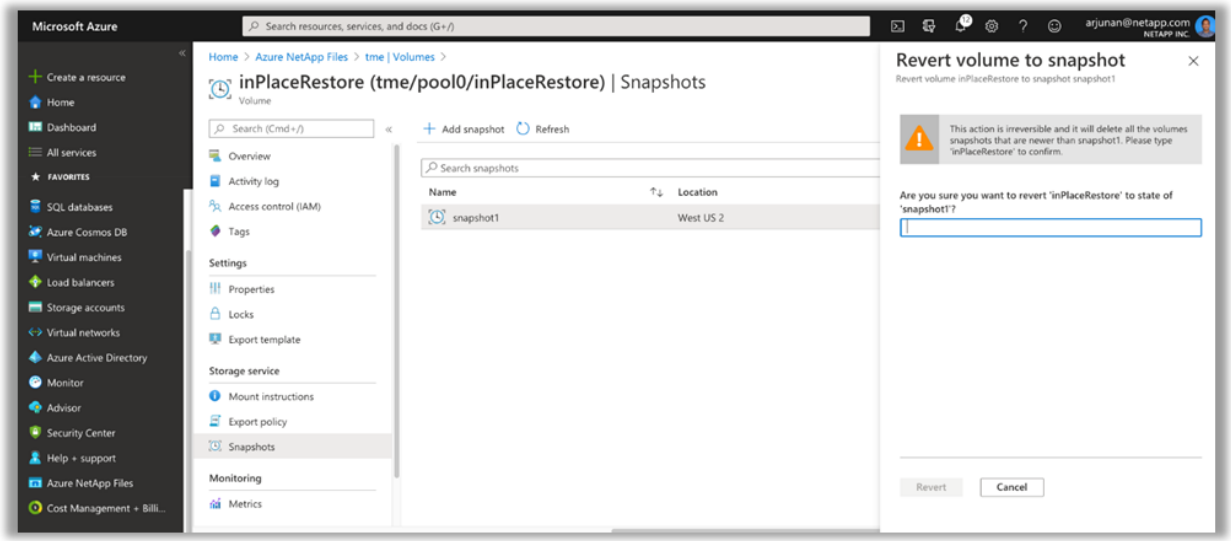
Figure 7 shows the option to take the in-place restore or revert volume to restore the volume using the chosen snapshot tab in the Azure NetApp Files.

Figure 7) Azure NetApp Files Volume restore options.



Note: One thing to remember before you start is that the Revert Volume action is irreversible and deletes all the volume snapshots that are newer than the current snapshot. To revert the changes, enter the volume name and click Revert.

Figure 8) Revert volume to snapshot.



Bring Back Your Oracle Database to the New Volume

Another option is to take advantage of the instant copy. The instant copy helps roll back or fix corrupted storage through to another volume. Customers can use this new capability by selecting Revert Volume from the Snapshot tab. The in-place restore or revert volume is comprised of the following steps:

1. Azure NetApp files take the snapshot of the volume based on the scheduled interval.
2. The snapshots are stored in the same volume.

Figure 9 shows the volume where the Oracle database is located.

Figure 9) Volume overview

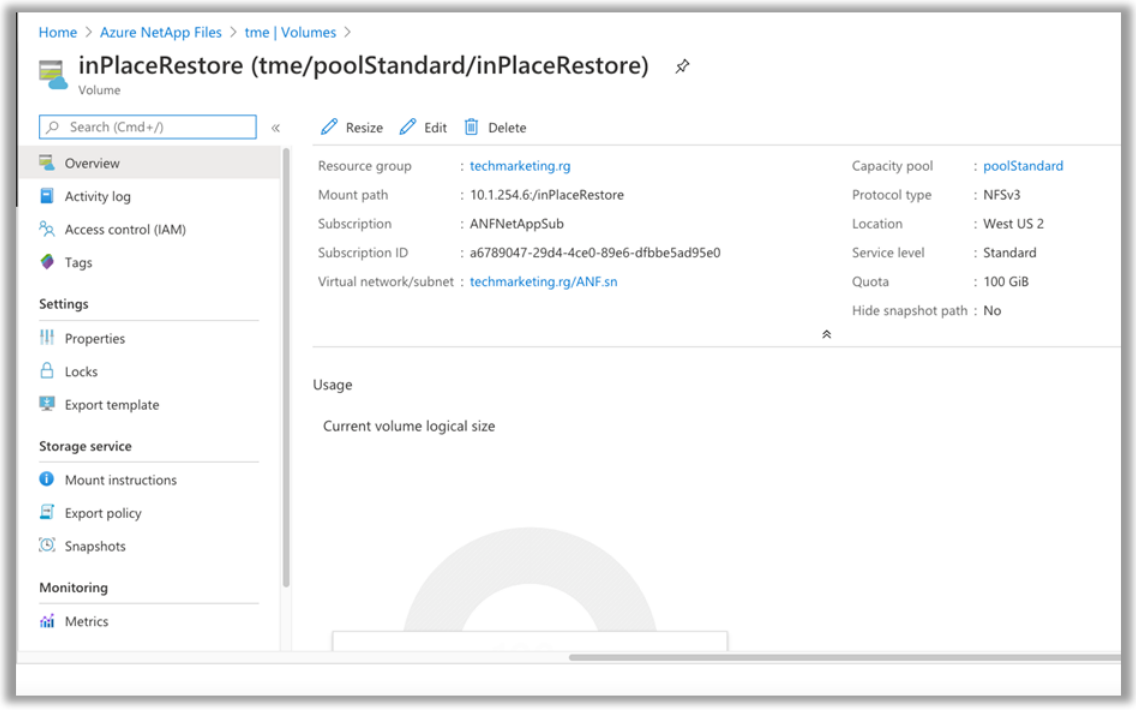


Figure 10 shows the volume mounted in the Oracle Linux machine.

Figure 10) Volume mounted in Oracle Linux machine.

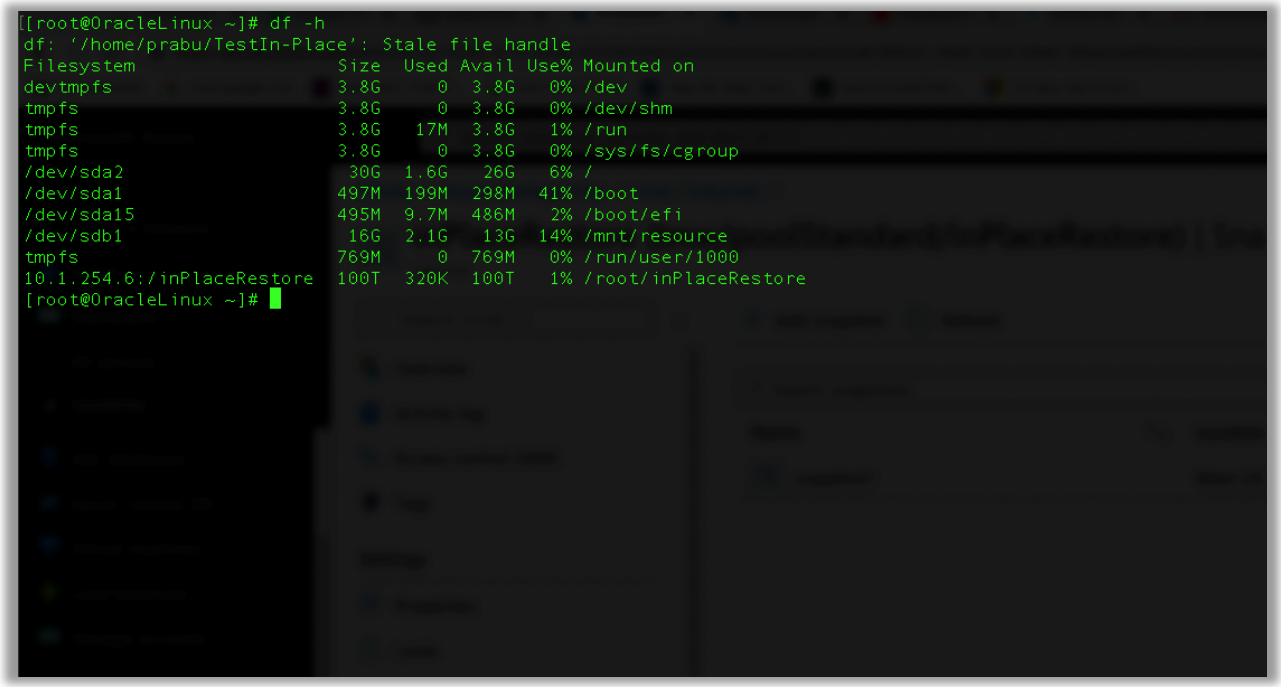


Figure 11 shows the Snapshots tab in Azure NetApp Files.

Figure 11) Azure NetApp Files Snapshots option.

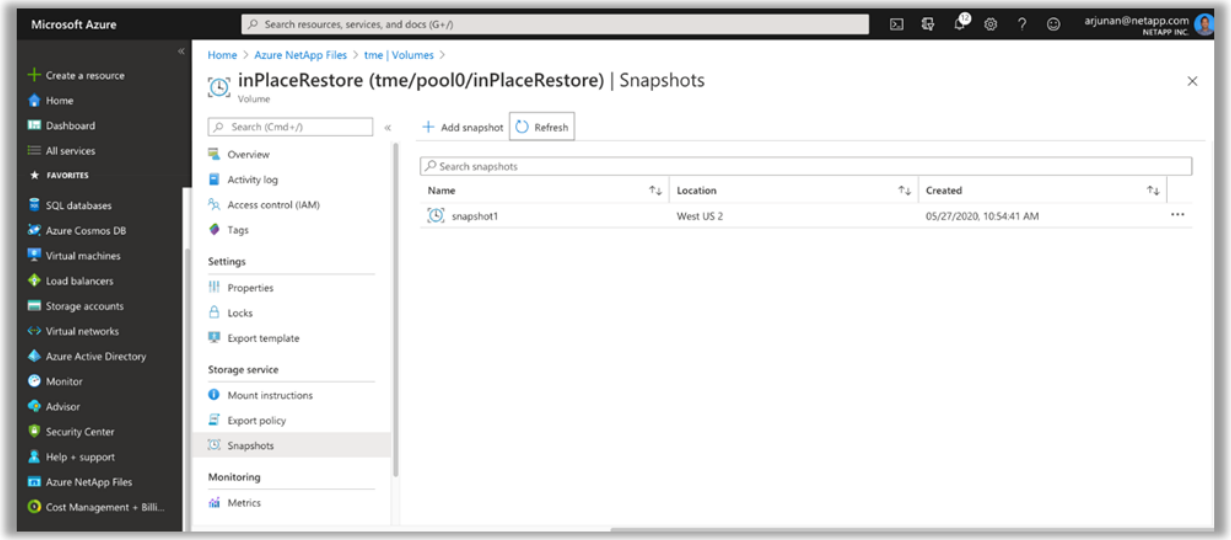
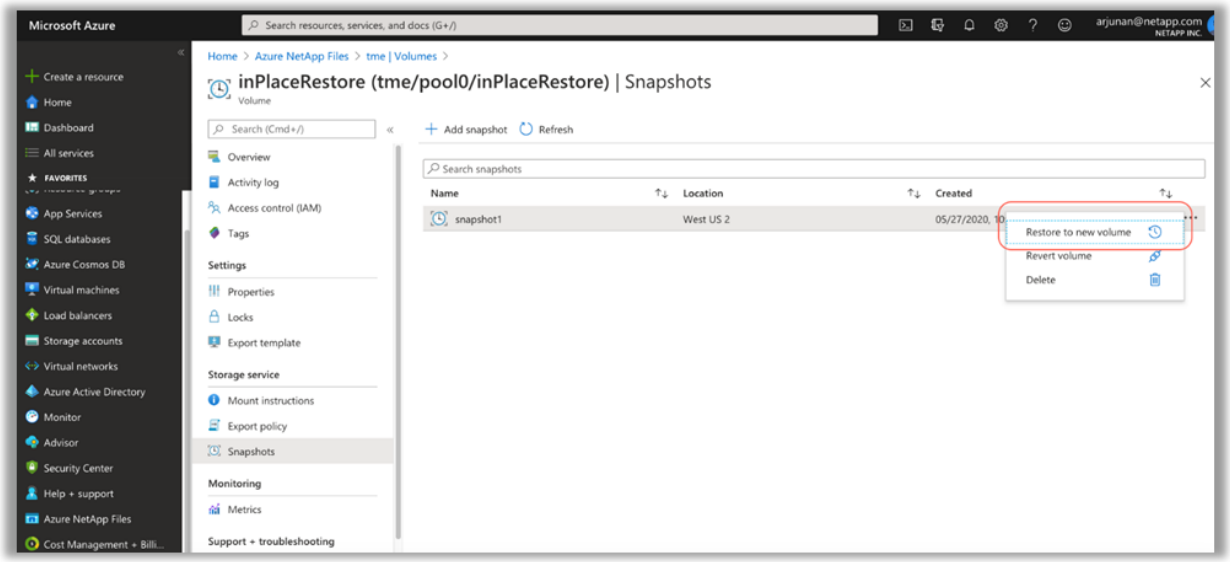


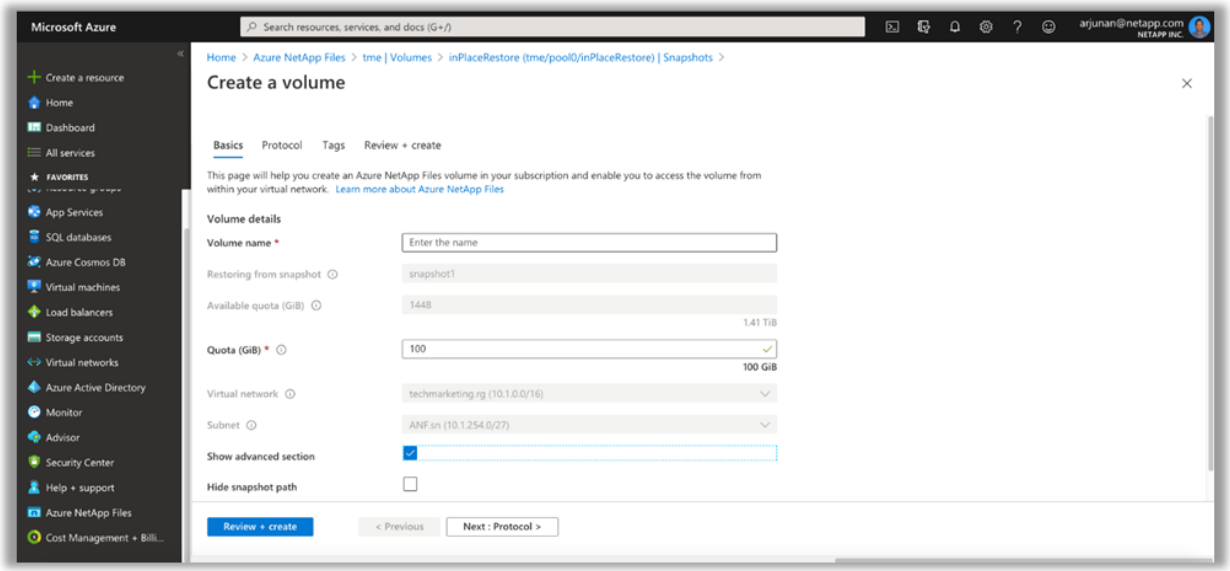
Figure 12 shows the option to take the in-place restore or revert volume to restore the volume using the chosen snapshot tab in the Azure NetApp Files.

Figure 12) Restore to new volume.



Unlike the Revert Volume action, the Restore to New Volume action is reversible. To create the new volume, click Review + Create.

Figure 13) Create a volume.



Is a Snapshot a Backup?

One commonly raised objection to the use of snapshots as a data protection strategy is the fact that the real data and the snapshot data are located on the same storage. Loss of that storage would result in the loss of both the primary data and the backup.

This is a valid concern. Local snapshots are used for day-to-day backup and recovery needs, and in that respect the snapshot is a backup. Close to 99% of all recovery scenarios in NetApp environments rely on snapshots to meet even the most aggressive RTO requirements.

Local snapshots should, however, never be the only backup strategy, which is why NetApp offers technology, such as CloudSync, to quickly replicate data to independent storage. In a properly architected solution with snapshots plus snapshot replication, the use of cheaper/lower tier storage (like e.g. Azure blob) can be minimized to perhaps a quarterly archive or eliminated entirely.

Local Recovery Procedure—NFS

The local recovery procedure can be driven manually or scripted. The basic procedure includes these steps:

1. Shut down the database.
2. Recover the datafile volumes to the snapshot immediately prior to the desired restore point.
3. Replay archive logs to the desired point.
4. Replay current redo logs if complete recovery is desired.

This procedure assumes that the desired archive logs are still present in the active volume. If they are not, the archive logs must be restored or `rman/sqlplus` can be directed to the data in the `.snapshot` directory.

In addition, for smaller databases, datafiles can be recovered by an end user directly from the `.snapshot` directory without assistance from automation tools or storage administrators.

4.3 Oracle Snapshot-Optimized Backup

Snapshot-based backup and recovery becomes even simpler with Oracle 12c because there is no need to place a database in hot backup mode. The result is an ability to schedule snapshot-based backups directly on a storage system and still preserve the ability to perform complete or point-in-time recovery.

Although the hot backup recovery procedure is more familiar to DBAs, it has, for a long time, been possible to use snapshots that were not created while the database was in hot backup mode. Extra manual steps were required with Oracle 10g and 11g during recovery to make the database consistent. With Oracle 12c, `sqlplus` and `rman` contain the extra logic to replay archive logs on datafile backups that were not in hot backup mode.

As discussed previously, recovering a snapshot-based hot backup requires two sets of data:

- A snapshot of the data files created while in backup mode
- The archive logs generated while the datafiles were in hot backup mode

During recovery, the database reads metadata from the datafiles to select the required archive logs for recovery.

Snapshot-optimized recovery requires slightly different datasets to accomplish the same results:

- A snapshot of the data files, plus a method to identify the time the snapshot was created
- Archive logs from the time of the most recent datafile checkpoint through the exact time of the snapshot

During recovery, the database reads metadata from the data files to identify the earliest archive log required. Full or point-in-time recovery can be performed. When performing a point-in-time recovery, it is critical to know the time of the snapshot of the datafiles. The specified recovery point must be after the creation time of the snapshots. NetApp recommends adding at least a few minutes to the snapshot time to account for clock variation.

For complete details, see Oracle's documentation on the topic, "Recovery Using Storage Snapshot Optimization" available in various releases of the Oracle 12c documentation. Also, see Oracle Document ID Doc ID 604683.1 regarding Oracle third-party snapshot support.

5 Oracle on Azure NetApp Files Configuration Best Practices

The TCP/IP settings required for Oracle database software installation are usually sufficient to provide good performance for Azure NetApp Files. There are a few exceptions.

5.1 TCP Parameters

Three settings are frequently misconfigured: TCP timestamps, selective acknowledgment (SACK), and TCP window scaling. Many out-of-date documents on the internet recommend disabling one or more of these parameters to improve performance. There was some merit to this recommendation many years ago when CPU capabilities were much lower and there was a benefit to reducing the overhead on TCP processing whenever possible.

However, with modern operating systems, disabling any of these TCP features usually results in no detectable benefit or might result in performance damage. Performance damage is especially likely in virtualized networking environments because these features are required for efficient handling of packet loss and changes in network quality.

The following settings are recommended:

- Enable TCP timestamps, SACK, and TCP window scaling on the host. Check the host operating system's network performance tuning guide on how to enable these parameters.

  Note: NetApp does not recommend enabling TCP timestamps for SAP on Oracle database deployments.

5.2 NFS Configuration

Operating Systems

The most common database platforms in cloud environments are Linux and Microsoft Windows. The Linux operating system includes native NFS capabilities and is the most common operating system.

For customers that prefer Windows, Oracle offers the direct NFS (dNFS) client, natively integrated into Oracle. For more information, see section 5.4, "Direct NFS and Host File System Access." The dNFS feature offers a path to the management benefits of NFS, including the ability to view files across environments, dynamically resize volumes, and use a less expensive IP protocol. See the official Oracle documentation for information about installing and configuring a database on Microsoft Windows using dNFS. No special best practices exist.

NFS Versions

Oracle has supported NFSv3 for over 20 years, and NFSv4 is supported with Oracle 12.1.0.2 and later.

TCP Slot Tables

TCP slot tables are the NFS equivalent of host bus adapter (HBA) queue depth. These tables control the number of NFS operations that can be outstanding at any one time. The default value is usually 16, which is far too low for optimum performance. The opposite problem occurs on newer Linux kernels, which can automatically increase the TCP slot table limit to a level that saturates the NFS server with requests.

For optimum performance and to prevent performance problems, adjust the kernel parameters that control the TCP slot tables.

Run `sysctl -a | grep tcp.*.slot_table`, and check the following parameters:

```
# sysctl -a | grep tcp.*.slot_table
sunrpc.tcp_max_slot_table_entries = 128
sunrpc.tcp_slot_table_entries = 128
```

All Linux systems should include `sunrpc.tcp_slot_table_entries`, but only some will include `sunrpc.tcp_max_slot_table_entries`. They should all be set to 128.

actimeo=0, noac

The presence of the following mount options in `ORACLE_HOME` causes host caching to be disabled, which badly damages performance for many workloads, especially installation and patching:

```
actimeo=0, noac
```

These mount options are unavoidable with Oracle RAC, but RAC is not supported in Azclient is critical for optimum performance in a cloud environment.

The dNFS client provides multiple benefits. It is designed to bypass the host NFS client and perform NFS file operations directly on an NFS server. Enabling it only requires changing the Oracle Disk Manager (ODM) library. Instructions for this process are provided in the Oracle documentation.

Using dNFS results in a general improvement in I/O performance and decreases the load on the host and the storage system because I/O is performed in the most efficient way possible.

When dNFS is used, it is critical that all patches described in Oracle Doc 1495104.1 are installed. If a patch cannot be installed, the environment must be evaluated to make sure that the bugs described in that document do not cause problems. Sometimes, an inability to install the required patches prevents the use of dNFS.

5.3 Handling Stale NFS Locks

If an Oracle Database server crashes, it might have problems with stale NFS locks upon restart. This only occurs with NFSv3. The problem is avoidable by paying careful attention to the configuration of name resolution on the server.

This problem arises because creating a lock and clearing a lock use two slightly different methods of name resolution. Two processes are involved, the Network Lock Manager (NLM) and the NFS client. The NLM uses `uname -n` to determine the host name, while the `rpc.statd` process uses `gethostbyname()`. These host names must match for the OS to properly clear stale locks. For example, the host might be looking for locks owned by `dbserver5`, but the locks were registered by the host as `dbserver5.mydomain.org`. If `gethostbyname()` does not return the same value as `uname -a`, then the lock release process did not succeed.

The following sample script verifies whether name resolution is fully consistent:

```
#! /usr/bin/perl
$uname=`uname -n`;
chomp($uname);
($name, $aliases, $addrtype, $length, @addrs) = gethostbyname $uname;
print "uname -n yields: $uname\n";
print "gethostbyname yields: $name\n";
```

If `gethostbyname` does not match `uname`, stale locks are likely. For example, this result reveals a potential problem:

```
uname -n yields: dbserver5
gethostbyname yields: dbserver5.mydomain.org
```

The solution is usually found by changing the order in which hosts appear in `/etc/hosts`. For example, assume that the hosts file includes this entry:

```
10.156.110.201  dbserver5.mydomain.org dbserver5 loghost
```

To resolve this issue, change the order in which the fully qualified domain name and the short host name appear:

```
10.156.110.201  dbserver5 dbserver5.mydomain.org loghost
```

`gethostbyname()` now returns the short `dbserver5` host name, which matches the output of `uname`. Locks are thus cleared automatically after a server crash.

Alternatively, manage customers use nolock in the mount options to prevent lock creation on the NFS server entirely. This creates some additional risk because it increases the chance two different database servers might try to open the same files simultaneously, but the risk can be minimized by ensuring a volume can be mounted only by a single, specific host IP address.

5.4 Direct NFS and Host File System Access

Using dNFS can occasionally cause problems for applications or user activities that rely on the visible file systems mounted on the host because the DNFS client accesses the file system out of band from the host OS. The dNFS client can create, delete, and modify files without the knowledge of the operating system. There might be a lag at times before a file change becomes visible to operating system users. Under very rare circumstances, actimeo=0 might be required to ensure changes are instantly recognized. The performance impact is significant unless dNFS is used.

Table 2 lists Linux NFSv3 mount options.

Caution

- Before using dNFS, verify that the patches described in Oracle Doc 1495104.1 are installed.
- Starting with Oracle 12c, DNFS includes support for NFSv3, NFSv4, and NFSv4.1. NetApp support policies cover v3 and v4 for all clients, but at the time of writing NFSv4.1 is not supported for use with Oracle dNFS.

Table 2) Linux NFSv3 mount options—single instance.

File Type	Mount Options
• Control files • Data files • Redo logs	`rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536`
• ORACLE_HOME • ORACLE_BASE	`rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536`

Generally, nondatabase files should be mounted with the same options used for single-instance data files, although specific applications might have different requirements. Avoid the mount options `noac` and `actimeo=0` if possible because these options disable filesystem-level read ahead and buffering. This can cause severe performance problems for processes such as extract, transform, load (ETL).

5.5 Nosharecache

One additional mount option, called `nosharecache`, is required in the following circumstances:

- dNFS is enabled.
- A source volume is mounted more than once on a single server.
- The volume mounts are nested.

  This `nosharecache` configuration is seen primarily in environments supporting SAP applications.

In general, a single volume is only mounted once but this practice is not mandatory. For example, an Azure NetApp Files volume might have a path at `/vol1234/base` and `/vol1234/home`. If

/vol1234/base is mounted at /oracle and /vol1234/home is mounted at /oracle/home, the result is nested NFS mounts that originate on the same source volume.

The operating system can detect the fact that /oracle and /oracle/home reside on the same volume, which is the same source file system. The operating system then uses the same device handle for accessing the data. Doing so improves the use of the operating system caching and certain other operations, but it interferes with dNFS. If dNFS must access a file on /oracle/home, it might erroneously attempt to use the wrong path to the data. The result is a failed I/O operation. In these configurations, add the nosharecache mount option to any NFS file system that shares a source volume with another NFS file system on that host. Doing so forces the Linux OS to allocate an independent device handle for that file system.

5.6 Automatic Storage Management

Automatic Storage Management (ASM) is supported with NFS. ASM uses the space inside one or more files and presents it to the database as a single pool of storage. Normally, these files are LUN devices at paths such as /dev/sdab or /dev/mapper/oracleasm/disk0, but they can also be paths to an NFS file system, including Azure NetApp Files volumes.

Re-virtualizing files on an NFS file system does not offer any significant benefits over placing Oracle files directly on NFS file systems, but if management practices make ASM desirable, then ASM over NFS may be used. Commands work as usual, and the ODM library used by ASM also includes the ability to use dNFS. Performance should be nearly identical.

5.7 Oracle Configuration

filesystemio_options

The Oracle initialization parameter filesystemio_options controls the use of asynchronous and direct I/O. Contrary to common belief, asynchronous and direct I/O are not mutually exclusive. NetApp has observed that this parameter is frequently misconfigured in customer environments, and this misconfiguration is directly responsible for many performance problems.

Asynchronous I/O means that Oracle I/O operations can be parallelized. Before the availability of asynchronous I/O on various operating systems, users configured numerous dbwriter processes and changed the server process configuration. With asynchronous I/O, the operating system itself performs I/O on behalf of the database software in a highly efficient and parallel manner. This process does not place data at risk, and critical operations, such as Oracle redo logging, are still performed synchronously.

Direct I/O bypasses the operating system buffer cache. I/O on a UNIX system ordinarily flows through the operating system buffer cache. This is useful for applications that do not maintain an internal cache, but Oracle has its own buffer cache within the System Global Area (SGA). In almost all cases, it is better to enable direct I/O and allocate server RAM to the SGA rather than to rely on the operating system buffer cache. The Oracle SGA uses the memory more efficiently. In addition, when I/O flows through the operating system buffer, it is subject to extra processing, which increases latencies. The increased latencies are especially noticeable with heavy write I/O when low latency is a critical requirement.

The options for filesystemio_options are:

- async. Oracle submits I/O requests to the operating system for processing. This process allows Oracle to perform other work rather than waiting for I/O completion and thus increases I/O parallelization.
- directio. Oracle performs I/O directly against physical files rather than routing I/O through the host operating system cache.
- none. Oracle uses synchronous and buffered I/O. In this configuration, the choice between shared and dedicated server processes and the number of dbwriters are more important.

- `setall`. Oracle uses both asynchronous and direct I/O.

In almost all cases, the use of `setall` is optimal, but consider the following issues:

- If a database has been using buffered I/O, a switch to direct I/O might also warrant a change in the SGA size. Disabling buffered I/O eliminates the performance benefit that the host operating system cache provides for the database. Adding RAM back to the SGA repairs this problem. The net result should be an improvement in I/O performance.
- Although it is almost always better to use RAM for the Oracle SGA than for OS buffer caching, it might be impossible to determine the best value. For example, it might be preferable to use buffered I/O with very small SGA sizes on a database server with many intermittently active Oracle instances. This arrangement allows the flexible use of the remaining free RAM on the operating system by all running database instances. This is a highly unusual situation, but it has been observed at some customer sites.

  Note: The `filesystemio_options` parameter has no effect in DNFS and ASM environments. The use of dNFS or ASM automatically results in the use of both asynchronous and direct I/O.

NetApp recommends the following:

- Set `filesystemio_options` to `setall`, but be aware that under some circumstances the loss of the host buffer cache might require an increase in the Oracle SGA.

db_file_multiblock_read_count

The `db_file_multiblock_read_count` parameter controls the maximum number of Oracle database blocks that Oracle reads as a single operation during sequential I/O. This parameter does not, however, affect the number of blocks that Oracle reads during any and all read operations, nor does it affect random I/O. Only sequential I/O is affected.

Oracle recommends that the user leave this parameter unset. Doing so allows the database software to automatically set the optimum value. This generally means that this parameter is set to a value that yields an I/O size of 1MB. For example, a 1MB read of 8KB blocks would require 128 blocks to be read, and the default value for this parameter would therefore be 128.

Most database performance problems observed by NetApp at customer sites involve an incorrect setting for this parameter. There were valid reasons to change this value with Oracle versions 8 and 9. As a result, the parameter might be unknowingly present in `init.ora` files because the database was upgraded in place to Oracle 10 and later. A legacy setting of 8 or 16, compared to a default value of 128, significantly damages sequential I/O performance.

NetApp recommends the following:

- The `db_file_multiblock_read_count` parameter should not be present in the `init.ora` file. NetApp has never encountered a situation in which changing this parameter improved performance, but there are many cases in which it caused clear damage to sequential I/O throughput.

Redo Block Size

Oracle supports either a 512-byte or 4KB redo block size. The default is 512 bytes. The best option is expected to be 512 bytes because this size minimizes the amount of data written during redo operations. However, it is possible that the 4KB size could offer a performance benefit at very high logging rates. For example, a single database with 50MBps of redo logging might be more efficient if the redo block size is larger. A storage system supporting many databases with a large total amount of redo logging might benefit from a 4KB redo block size. This is because this setting would eliminate inefficient partial I/O processing when only a part of a 4KB block must be updated.

It is not correct that all I/O operations are performed in single units of the redo log block size. At very high logging rates, the database generally performs very large I/O operations composed of multiple redo blocks. The actual size of those redo blocks does not generally affect the efficiency of logging.

NetApp recommends the following:

- Only change the default block size for cause, such as a documented requirement for a particular application or because of a recommendation made by NetApp or Oracle customer support.

6 Performance Optimization and Benchmarking

Accurate testing of database storage performance is a complicated subject. It requires not just an understanding of IOPS and throughput, but also understand of the following concepts:

- Differences between foreground and background I/O operations
- Effect of latency upon the database
- Multiple operating systems and network settings that affect storage performance
- Nonstorage database tasks

Note: There is a point where optimizing storage performance yields no useful benefits because storage performance is no longer a limiting factor for performance.

Any evaluation of database performance in a cloud environment must focus on real-world needs. There will nearly always be significant differences at maximum loading levels. The cloud environment will almost inevitably be slower at the top end, especially since nearly all cloud resources have multiple types of QoS controls, both visible and invisible.

Rather than testing maximums, identify the actual requirements and see whether they can be met in the Cloud.

6.1 Oracle Workload Repository and Benchmarking

The gold standard for Oracle performance comparison is an Oracle Automatic Workload Repository (AWR) report.

There are multiple types of AWR reports. From a storage point of view, a report generated by running the `awrrpt.sql` command is the most comprehensive and valuable because it targets a specific database instance and includes some detailed histograms that break down storage I/O events based on latency.

Comparing two performance arrays ideally involves running the same workload on each array and producing an AWR report that precisely targets the workload. In the case of a very long-running workload, a single AWR report with an elapsed time that encompasses the start and stop time can be used, but it is preferable to break out the AWR data as multiple reports. For example, if a batch job ran from midnight to 6 a.m., create a series of one-hour AWR reports from midnight–1 a.m., 1 a.m.–2 a.m., and so on.

In other cases, a very short query should be optimized. The best option is an AWR report based on an AWR snapshot created when the query begins and a second AWR snapshot created when the query ends. The database server should be otherwise quiet to minimize the background activity that would obscure the activity of the query under analysis.

Note: Where AWR reports are not available, Oracle statspack reports are a good alternative. They contain most of the same I/O statistics as an AWR report.

6.2 Oracle AWR and Troubleshooting

Starting with Oracle10g in the mid-2000s, every Oracle database contains a repository of automatically captured workload information. This is known as the AWR and it provides a powerful tool for diagnosing performance issues. This information is stored internally, captured from memory-based performance

views, usually with a 1-hour frequency, unless otherwise configured. By default, this data is retained for a week, unless otherwise configured.

The workload capture mechanisms are built in to the Oracle database enterprise edition. Except for configuration parameters, this mechanism is not exposed outside the database.

Reporting from the repository can be performed by using ad-hoc queries or using stored procedures in the built-in database PL/SQL package `DBMS_WORKLOAD_REPOSITORY`. This PL/SQL package can be called directly from tools such as Oracle Enterprise Manager or other database management consoles, or there is a list of SQL*Plus scripts on the database server in the `rdbms/admin` subdirectory within the `$ORACLE_HOME` software installation directory tree.

Select the best AWR for your needs:

- Use a standard AWR report for a single database instance (script `awrrpt.sql`)
- Use a standard AWR report summarizing all instances in a RAC database (script `awrgrpt.sql`)
- Compare two different time periods in a single database instance (script `awrddrpt.sql`)
- Compare two different time periods summarizing across all instances in a RAC database (script `awrgdrpt.sql`)

Each of these scripts prompts for a begin AWR snapshot ID and an end AWR snapshot ID; the time period between these snapshots comprises the resulting AWR report.

Each AWR snapshot is the time when workload information was captured, but there is no limit on the range between the begin and end snapshot IDs. The only restriction is that a database instance restart might not be included in the range of snapshots, because the memory-based performance views from which workload information is captured are reset to zero when the instance is restarted.

The following example calls the standard Oracle AWR report (such as `awrrpt.sql`) from SQL*Plus at the Linux command line:

```
$ cd $ORACLE_HOME/rdbms/admin
$ sqlplus / as sysdba
(SQL*Plus banner redacted for brevity)
SQL> @awrrpt
(the "awrrpt.sql" script will prompt for report output format, then for the number of days of
snapshot IDs to display, then the begin snapshot ID and end snapshot ID)
```

The HTML or text output file is generated to the present working directory. Be aware that the AWR report is very large, with thousands of data points.

For more information about using AWR, search for consult Oracle documentation as well as the following excellent blogs with information and up-to-date documentation references about AWR, including:

- ORACLE-BASE blog by Tim Hall
- Oracle Scratchpad blog by Jonathan Lewis

6.3 Scaling and Performance

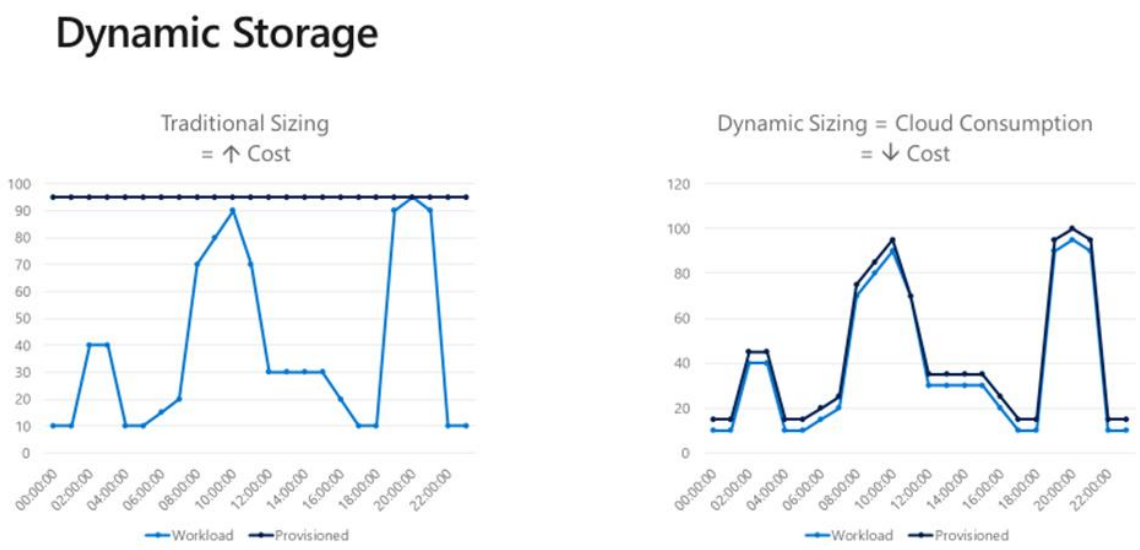
Volume Shaping

Oracle systems are often sized based on peak performance requirements such as regular batch run, data load or other peak workloads – wasting valuable resource for the remaining periods. While there is almost no other choice in on-premises setups, and even with most standard cloud deployments (while being flexible in assigning compute resources), the typical (fixed-block) storage offerings usually do not offer the same flexibility.

One of the innovative features of Azure NetApp Files is the capability to scale capacity and performance dynamically. Currently, the performance of any given volume can be increased and decreased

dynamically by resizing the volume quota, thus resizing the performance limits of the volume. This enables short-term (burst) performance increase of the volume and the databases hosted on it. This is especially useful if high-performance is only needed for short periods of time, for example, during specific data warehouse or batch workloads.

Figure 14) Dynamic storage sizing.



Therefore, Azure NetApp Files can shape capacity and performance on-demand, which can be changed at runtime without any reconfiguration at the operating system, NFS file system, or database level. These changes have an immediate influence on the Oracle database performance. This concept is known as volume shaping.

Database Volume Scale-Out

Another way to increase performance continuously is to distribute the database layout across multiple volumes, spreading the load and aggregating performance.

More detailed information about performance scalability of Azure NetApp Files can be found here:

- Benefits of using Azure NetApp Files with Oracle Database
- Performance benchmarks for Azure NetApp Files

6.4 Azure NetApp Files, Azure Premium Files, and Azure Managed Disks

Azure Premium Files GA was announced on June 26, 2019 (the announcement can be found here). It provides significant improvement over the Azure Files standard service level. Azure Managed Disks are the new and recommended disk storage offering for use with Azure VMs for persistent storage of data. You can use multiple Managed Disks with each VM. Azure offers four types of Managed Disks: Ultra Disk, Premium SSD Managed Disks, Standard SSD Managed Disks, Standard HDD Managed Disks. For more information, see What Disk Types are Available in Azure on the Microsoft Documentation page.

Table 3) Azure NetApp Files, Azure Premium Files, and Azure Managed Disks features.

Features	Azure NetApp Files	Azure Premium Files	Azure Managed disks
Performance	• Up to 300,000 IOPS in a single volume • Up to 32 IOPS/GB without burst	Azure Premium Files delivers 1 IOP/GiB, up to 100,000 IOPS. Premium Files can burst up to 3x (up to 100k IOPS) performance. Bursting has time-based limits varies based on the size of the share.	• Azure Ultra SSD support Max IOPS 160,000 • Azure Premium SSD support Max IOPS 20,000
Protocol support	Azure NetApp Files supports NFSv3, NFSv4.1 or SMB volumes.	Azure Premium Files only supports a subset of SMB features, unsupported features are listed here.	SMB volumes
• File size • File IOPs limit	• Azure NetApp Files support files up to 16TB in size • No limit	• Azure Files supports files up to 1TiB in size • Azure Premium Files supports up to 5,000 IOPS per file	n/a
Data management	Azure NetApp Files provides the following data management features using snapshot copies: • Create an on-demand snapshot for a volume or restore from a snapshot to a new volume. • Since snapshot copies are point-in-time copies, they take up the space in megabytes. • Rapidly create a new volume from a snapshot copy. The new volume does not share performance limits with the original volume snapshot. • Up to 255 snapshots per volume • Restoring a file from snapshot can be performed from a Linux host using an Azure NetApp Files NFS volume using the `.snapshot` fold	Azure Premium Files provides snapshot copies for SMB shares that can used for the restoration of individual files. **Note:** Snapshot copies can also be connected to through SMB from a host (Windows only). Azure Files supports up to 200 snapshot copies/shares.	Azure Managed disks provide: • A managed disk snapshot copy is a read-only crash-consistent full copy of a managed disk that is stored as a standard managed disk by default. • Premium SSD, standard SSD, and standard HDD support snapshots. For these three disk types, snapshots are supported for all disk sizes (including disks up to 32TiB in size). Ultra disks do not support snapshots. • Restore from the snapshot copies is not straightforward. You need to create a managed disk from a snapshot copy before it can be used with a VM. This means you can use a snapshot to create multiple VMs.

Features	Azure NetApp Files	Azure Premium Files	Azure Managed disks
			• Application-consistent backups for Windows Azure VMs and file-system consistent backup for Linux Azure VMs without the need to shut down VM.
Database requirement Database requires 20,000 8K IOPS and 2TB of capacity	• Azure NetApp Files requires a 4TB premium capacity pool (the minimum size for a pool) to satisfy this requirement. • Azure NetApp Files has a maximum file size of 16TB, so Azure NetApp Files meets the capacity and file size requirements of the database. • The performance requirement of 20,000 IOPS is met with a 2.5TB volume in the capacity pool.	• Azure Premium Files requires 20TiB (1 IOP/GiB sustained) of capacity to satisfy this requirement for performance. • Curveball: Remember that Premium Files only supports up to 1TiB of files and up to 5,000IOPS per file. Therefore, in order to work, the database must be split across multiple files prior to moving to Premium Files. This leaves you with a database split across four files in a share.	Azure Ultra SSD support with a maximum of 160,000 IOPS per disk.

7 How Does Oracle Licensing Work?

Oracle Database licensing on Azure is based on the number of vCPUs on the VMs on which the database is installed. In Microsoft Azure, two vCPUs are equivalent to one Oracle Processor license if hyperthreading is enabled, and one vCPU is equivalent to one Oracle Processor license if hyperthreading is not enabled. For more information, see Licensing Oracle Software in the Cloud Computing Environment.

The key to minimizing Oracle licensing costs is the same as on-premises: that is, reduce the number of Oracle Processor licenses by reducing the number of vCPUs. There are two important factors to consider:

- Always size the Azure VM instance type to the actual observed Oracle database workload used by the on-premises database. Never size to the specifications of the database server used by the on-premises database:
  - The on-premises database servers are sized to accommodate growth over a 3–5-year cycle of a hardware refresh. Therefore, the on-premises database server is likely over-allocated except toward the end of the hardware refresh cycle.
  - The VMs in the public cloud can be changed in minutes. We only need to size Azure VMs to accommodate growth over a 3–5-week cycle. In Azure, the customer do not need to over-provision as much as on-premises.
  - For that reason, the customer should employ the information in Oracle AWR reports to determine present actual workload, and then provision Azure VMs to accommodate that workload plus a small cushion for peak workloads.
- Azure NetApp Files provides an NFS server, which is less bound by network bandwidth limits on Azure VMs, and not than by cumulative IOPS and I/O throughput limits. In general, network bandwidth allows double or triple the data transfer of the cumulative IOPS and I/O throughput limits.

- So, in situations where the I/O workload from the Oracle database would exceed the cumulative IOPS and I/O throughput limits about Azure premium SSD or Azure UltraDisk with the Azure VM, the network bandwidth of that Azure VM is 2x or 3x higher, allowing us to avoid migrating to a larger VM instance type
- For example, let's say the customer have an Oracle database workload requiring 12 vCPUs of processing resources. In most cases, the customer would provision a VM instance type with 16 vCPUs to accommodate that processing requirement.
- However, suppose that the same Oracle database workload has a peak workload of 40,000 IOPS and 750 MB/s of I/O throughput? Most of the Azure VM instance types with 16 vCPUs cannot accommodate that much IOPS or I/O throughput.
- So, if the database storage is on Azure managed disk, a larger VM instance size is required, which will use more vCPUs, which will increase the Azure VM consumption and Oracle licensing cost
- However, if the database storage is on Azure NetApp Files, which is mounted using NFS over network bandwidth, the same 16 vCPU VM instance type can be used, thus minimizing Azure VM consumption and Oracle licensing cost significantly

This example only illustrates the simple scenario using one server, where the customer only has to license 16 vCPUs instead of 20 or 32 vCPUs. Now, consider the impact across many databases, relatively small databases. Individually, these databases can only use small Azure managed disks, each device of which does not provide adequate IOPS or I/O throughput service limits. Larger-than-necessary managed disk might often times have to be provisioned to meet I/O requirements, in addition to larger-than-necessary VM instance types. The ability to consolidate storage capacity and use network-attached storage for higher data transfer requirements makes Azure NetApp Files storage pools faster, economical, and easier to use.

Where to Find Additional Information

To learn more about the information that is described in this document, review the following documents and/or websites:

- Azure NetApp Files
  https://cloud.netapp.com/azure-netapp-files
- NetApp Product Documentation
  https://docs.netapp.com

Version History

Version	Date	Document Version History
1.0	April 2019	Initial release.
2.0	July 2020	Update with latest Azure NetApp Files data protection features and jointly reviewed with Microsoft.

Refer to the Interoperability Matrix Tool (IMT) on the NetApp Support site to validate that the exact product and feature versions described in this document are supported for your specific environment. The NetApp IMT defines the product components and versions that can be used to construct configurations that are supported by NetApp. Specific results depend on each customer's installation in accordance with published specifications.

Copyright Information

Trademark Information

TR-4780-0720