PURESTORAGE®
Uncomplicate Data Storage, Forever

REFERENCE ARCHITECTURE

Microsoft SQL Server on VMware vSphere with FlashArray

Architectural guidance and best practices for deploying Microsoft SQL Server workloads in a combined VMware vSphere and Pure Storage® FlashArray™ environment.

Contents

Executive Summary

Virtualization is a common choice for deploying most production and non-production workloads in many companies. Virtualization is the emulation of computer hardware in a software form. Virtualizing servers are known as virtual machines (VMs). The software packages used to virtualize workloads are known as hypervisors, and one of the most popular is vSphere from VMware. vSphere is a virtualization platform that contains a suite of server virtualization products, such as the ESXi hypervisor and vCenter management software.

As server virtualization has become more popular, deployments of Microsoft SQL Server (SQL Server)—one of the most common database platforms in use today—have increased, as well. SQL Server can run on Microsoft's own Windows Server, on SQL Server 2017 and, later selected Linux distributions. Because of its popularity, SQL Server is considered business-critical in many environments. SQL Server must be architected to have the right performance, availability, and security to support the business. This means all layers from the hardware up through the hypervisor and into the VM (including SQL Server itself) must be architected and implemented properly. Storage architecture and performance are key aspects of any physical or virtual implementation of Microsoft SQL Server. A well-architected, correctly-configured virtual environment can help ensure that business operations run smoothly and allows for rapid troubleshooting in the event of any issues.

This reference architecture is intended to be used as a guide for the implementation of Microsoft SQL Server using VMware vSphere with Pure Storage FlashArray providing storage for the virtual infrastructure. The target audience for this document includes systems architects, database administrators, storage administrators, virtual infrastructure specialists, and IT professionals.

Benefits of Running SQL Server on VMware vSphere with FlashArray Storage

Virtualizing Microsoft SQL Server and using FlashArray storage enables the best outcomes for the infrastructure stack and application environment. At the storage layer, FlashArray is simple, easy to use, and storage space-efficient,with a range of data services that enhance any VMware vSphere environment's existing benefits. At the virtual infrastructure layer, vSphere allows for server consolidation and application flexibility through a range of innovative utilities and functions. The combination of VMware vSphere and FlashArray allows for even greater consolidation, operational efficiency, and cost reduction for growing organizations as both products share the same goal: simplifying application environments in a cost-effective way at little to no cost to application flexibility or operational capabilities.

Placing SQL Server into this environment allows organizations to focus on the core aspects of their business while benefiting from the combination of these components.

How to Use This Reference Architecture

This reference architecture focuses on VMware vSphere virtual infrastructure architecture with Pure Storage FlashArray providing storage for Microsoft SQL Server deployments. A diverse range of concepts, such as block storage, hypervisors, operating systems, and database instance management, are covered to provide a complete view of the desired state configuration. Data services, such as volume snapshots and replication with FlashArray, are also covered for each product's use cases to achieve the best outcome for virtualized Microsoft SQL Server deployments with VMware products.

The concepts, architectures, and methodologies set out in this guide can be used for any FlashArray model.

Hardware Compatibility

This guide has been validated against the components documented in Table 1. Operating systems and the versions of SQL Server they run on reflect versions currently in mainstream support by Microsoft as of the writing of his paper. Older versions of SQL Server and Windows Server, such as SQL Server 2016 and Windows Server 2016, are in extended support, meaning that only critical fixes will be released. Customers should strive to deploy versions in mainstream support where possible to receive end-to-end support from all vendors. See the links later in this paper to see which versions of SQL Server are supported for your deployment scenario.

Component	Compatibility
Storage Hardware	FlashArray//X™ FlashArray//XR2 FlashArray//XR3 FlashArray//XL™ FlashArray//C™
Storage Software	Purity //FA 6.0 Purity //FA 6.1 Purity //FA 6.2 Purity //FA 6.3 Purity //FA 6.4
VMware vSphere	ESXi 7.x (all supported versions and updates) ESXI 8
Operating Systems	Windows Server 2019 Windows Server 2022 Red Hat Enterprise Linux 8.0–8.6 SUSE Linux Enterprise Server(SLES) 12 (SP3–SP5), 15(SP1–SP3) Ubuntu 16.04 LTS, 18.04 LTS, 20.04 LTS
Microsoft SQL Server	SQL Server 2019 SQL Server 2022

TABLE 1 Compatibility Matrix

Additionally, the Microsoft Support Matrix for Pure Storage products page provides an overview for supported integrations and the associated operating systems.

Microsoft SQL Server Editions and Features

Microsoft SQL Server is available in multiple editions. Below is a high-level overview of the features of each edition:

- **Enterprise**: Delivers high-end data center capabilities with blazing-fast performance, unlimited virtualization (with Software Assurance), and end-to-end business intelligence. The highest service levels for mission critical workloads can be achieved with this edition.
- **Standard:** Delivers lower-cost data management and business intelligence to run applications to environments that do not need all the features or capabilities of Enterprise Edition. Standard Edition has most of the features of Enterprise Edition but has some restrictions.
- **Web**:A lower-cost option for web hosting providers. Provides scalability, affordability, and manageability capabilities for small to large-scale web environments.
- **Developer:** Allows developers to build any kind of application on top of SQL Server. Includes all the functionality of Enterprise Edition but is licensed for development and test. It should not be used for a production deployment. Developer Edition is also supported on desktop operating systems such as Windows 10 and 11 in addition to Windows Server.
- **Express:** A free, entry level database offering. Ideal for learning and building desktop and small server-based, data-driven applications.

SQL Server is more than just a relational database platform. There are optional components that can be installed which enable business intelligence, reporting, and more. This reference architecture is focused on traditional deployments of SQL Server.

VMware vSphere

Similar to SQL Server, vSphere has multiple editions which have different features and capabilities:

- Standard
- Enterprise Plus
- vSphere+

See this documentation from VMware to compare the different editions.

VMware vSphere is made up of the following components and technologies:

- **VMware ESXi**: A hypervisor installed onto physical servers. It is responsible for abstracting processors, memory, storage, networking, and other resources into multiple virtual machines.
- **VMware vCenter Server**: A management tool providing a singular interface for managing one or more ESXi hosts.
- **VMware vSphere Client**: A graphical web-based management tool built into vCenter Server.
- **VMware vSphere Software Development Kits**: Interfaces through which users can manage vSphere components.
- **vMotion**: A feature that enables the live migration of virtual machines between ESXi hosts in a data center.
- **Storage vMotion**: A feature that enables the live migration of virtual machines between datastores.
- **vSphere High Availability**: A utility that ensures virtual machines are restarted on other available servers when a failure occurs.
- **VMware Distributed Resource Scheduler (DRS)**: A utility that manages resource utilization across multiple ESXi hosts and balances virtual machines across a collection of hardware resources.
- **Fault tolerance**: A feature that creates a duplicate of a selected workload on a different host within the cluster to provide continuous availability.
- **VMware vSphere distributed switch**: Enables a single virtual network to be available across many ESXi hosts. This enables the centralization of network configurations.

Unified Block and File on FlashArray

Pure Storage FlashArray is a software-defined, unified block and file storage product. This storage solution offers an effortless and consistent experience and behaves in an efficient manner by offering data reduction without an impact on performance. All Pure Storage products offer an Evergreen™ product model to increase capacity and performance without the need to purchase new storage products. Lastly, Pure Storage FlashArray enables businesses and organizations to drive out direct carbon usage in their data storage systems by up to 80% compared to competitive all-flash systems and even more against magnetic disk.

The FlashArray product line caters towards multiple business needs and use cases with these distinct offerings as shown in Figure 1:

- FlashArray//C: An all-QLC FlashArray with consistent performance at 2–4ms latency for capacity-oriented workloads.
- FlashArray//X: Provides latency as low as 150µs to power critical applications and business operations.
- FlashArray//XL: Enterprise-grade performance and scalability for demanding workloads

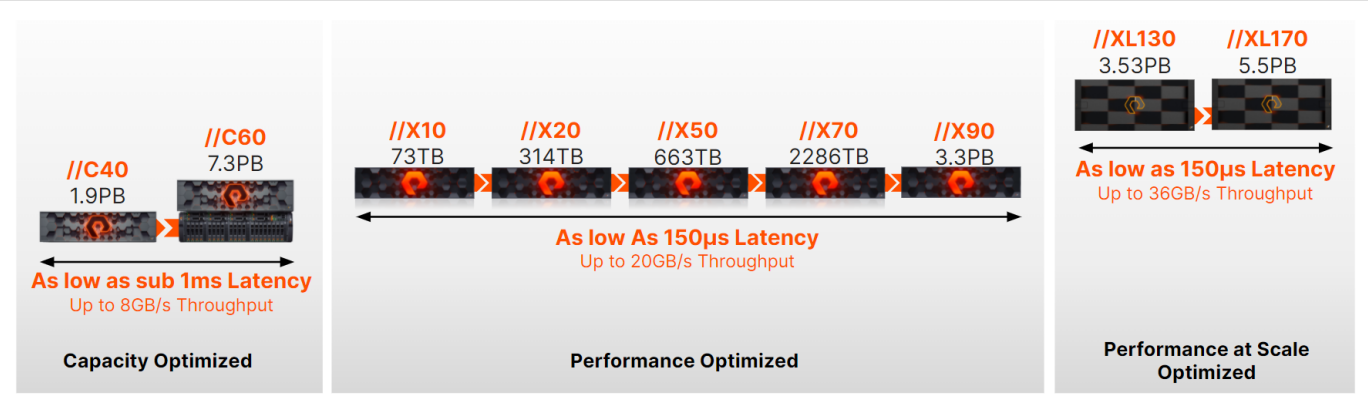

FIGURE 1 Pure Storage Flash Array Suite of Products

FlashArray and Pure Cloud Block Store™ systems are powered by Purity for FlashArray. Purity delivers rich, enterprise-level data services that ensure data is stored in the most secure and efficient way while providing additional functionality to extend storage capabilities.

- **Data reduction**: Purity averages an industry-leading 5:1 data reduction with a total efficiency of 10:1 (including thin provisioning).
- **High availability**: Purity protects against concurrent dual-drive failures and initiates re-builds automatically within minutes.
- **Always-on ransomware remediation**: Cost-efficient, portable, SafeMode™ Snapshots prevent cyber attackers from tampering with or maliciously destroying critical recovery data.
- **On-demand data portability**: Quickly and easily move data where it most cost-effectively meets service level agreements to satisfy your customers: between both physical and virtual machines.
- **Rich data services**: Data service functionality such as ActiveCluster™, ActiveDR™, and asynchronous replication provide increased resilience, availability and enhance applications workflows.

Figure 2 shows the Purity for FlashArray data services architecture.

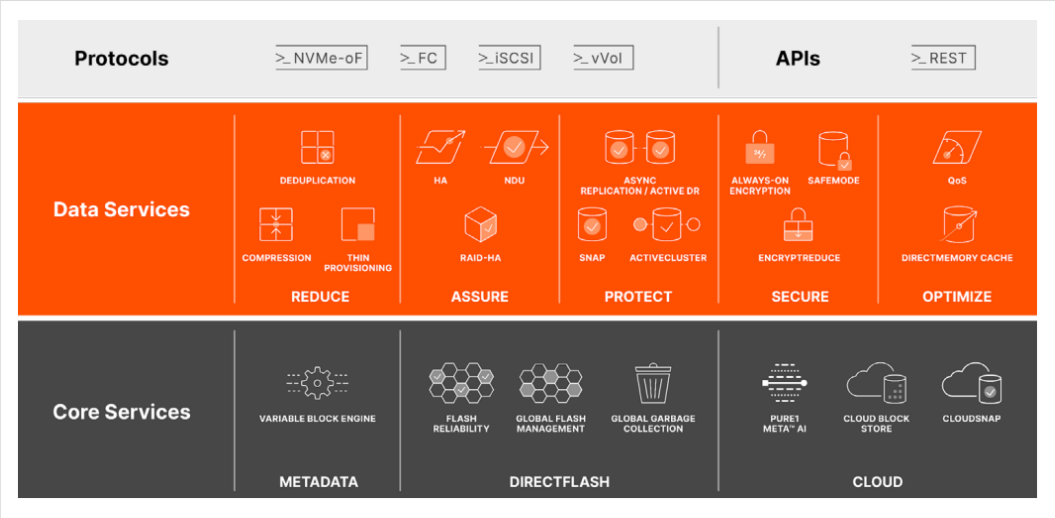

FIGURE 2 Purity for FlashArray data services architecture

Component Architecture and Recommended Configuration

Component Overview

The purpose of this reference architecture is to provide clear guidance on how to optimize the interaction between Microsoft SQL Server, VMware vSphere virtual infrastructure, and the underlying storage provided by Pure Storage FlashArray. Figure 3 shows this relationship.

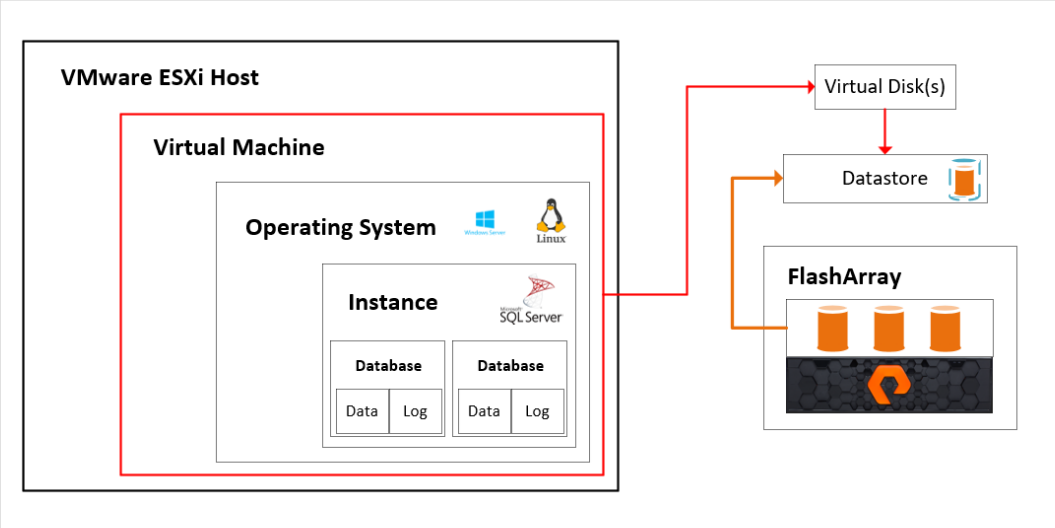

FIGURE 3 Component interconnectivity diagram

The relevant components in this reference architecture are set out below:

Component	Definition
Database	A database contains a structured set of data held on a computer. This data is stored in one or more data files. Microsoft SQL Server databases are made of up at least two files, data files and log files: • Data files contain data, as well as objects such as tables, indexes, stored procedures, and views. These can be either a primary (contains startup information for the database and uses the .mdf extension) or a secondary (optional user-defined data files with the .ndf extension) data file type. Data files can be grouped together into filegroups for allocation and administration purposes. • Transaction log files contain the information that is required to recover all transactions in a database. These files usually use the .ldf extension. See the Microsoft SQL Server Database Files and Filegroups page for more information. There are several system databases managed by an instance, as well as user-defined databases.
SQL Server Instance	An instance is an installation of the database engine. SQL Server and its associated services, such as SQL Server Agent run as services in the operating system. Applications and end users connect to an instance to gain access to the databases within them. There are two types of instances: default and named. For standalone installations of SQL Server, a default instance takes on the name of the underlying server. There can only be one default instance per server. Linux deployments only allow one installation of SQL Server which can only be a default instance. A named instance is the combination of a server name with a unique name associated with it. This is how SQL Server allows multiple installations on the same operating system if using Windows Server. An instance is responsible for managing one or more user databases and several system databases. System databases are not shared if there are multiple instances. Applications connect to an instance to perform work in a database managed by the instance. See the Microsoft SQL Server Database Engine Instances (SQL Server) page for more information.
Operating System	The operating system (OS) is a system software that manages computer hardware, software resources, and provides common services for computer programs. SQL Server is supported for use on Windows Server and selected Linux operating systems. See the Hardware and Software Requirements page for more information on SQL Server on Windows Server. See the SQL Server on Linux page for more information on SQL Server on Linux.
Virtual Machine	A virtual machine is a compute resource that uses software representations of physical hardware to run programs and deploy applications. One or more VM (sometimes known as guests) run on a physical ESXi host. Each VM runs its own operating system independently of any other VM on the same host. Virtual machines are provided with resources such as networking, storage, CPU, and memory. These resources can be reconfigured in the event of evolving requirements. One VM's workload can affect the performance of another VM running on the same hypervisor host. More information on VMware vSphere virtual machines can be found in the VMware glossary.

Component	Definition
VMware ESXi Host	An ESXi Host is a physical server installed with VMware ESXi. VMware ESXi is an enterprise-class, type-1 hypervisor with a small installation footprint. It forms a part of the vSphere Enterprise Workload Platform where many ESXi hosts can be managed and clustered for high availability by a centralized vCenter Server. All the virtual resources provided to virtual machines must be physically present in the ESXi hosts hardware. More information on ESXi hosts and hypervisors can be found in the VMware glossary.
Datastore	Datastores are logical containers that hide the specifics of physical storage from virtual machines and provide a uniform model for storing virtual machine files. Providing storage resources to a virtual machine is done through the addition of a virtual disk being attached to a SCSI or NVMe storage controller. The virtual disk resides on a specific datastore but can be migrated to a different one. The non-disruptive live migration of a VMware virtual machine's storage resources is called storage vMotion. There are different datastore types that can be utilized for virtual machine storage: • Virtual Machine File System **(VMFS version 5 and 6)**: A special high performance file system optimized for the storage of virtual machines created on block storage volumes. • Network File System **(NFS version 3 and 4.1)**: This datastore uses the NFS protocol over TCP/IP to access a designated NFS volume located on a network attached storage server. The ESXi host mounts this volume as an NFS datastore. • VMware vSphere Virtual Volumes: This datastore type represents a storage container. All virtual disks in the container receive the benefit of policy-based, granular storage configuration. Raw Device Mapping is not a datastore but is still a way in which virtual machines can be provided with storage resources. This storage capability uses raw LUNs or logical disks to provide volume storage directly to one or more virtual machines. More information about datastore types can be found in the VMware vSphere library.
FlashArray Volumes	Block storage volumes are created on any FlashArray and then connected to an ESXi host, where datastores can be created on them for virtual disk storage. When using vVols there is no difference between a volume on FlashArray and a virtual disk (located in the vVol datastore) attached to a virtual machine. This results in a 1:1 relationship between the virtual disks and volumes on FlashArray. The result of this is that array-based features can be used at the virtual disk level. Volumes on FlashArray can be connected to ESXi hosts using Fiber Channel, iSCSI, NVMeoF-RoCE, NVMeoF-FC or, NVMEoF-TCP. vVols can only be connected to via Fiber Channel or iSCSI. More information on using VMware with FlashArray can be found in the Pure Storage VMware Platform Guide.

TABLE 2 Useful terms and definitions

FlashArray Storage and Virtual Infrastructure

FlashArray volumes can be used with VMware virtual infrastructure in several ways:

FlashArray volumes as vVols: VMware vSphere Virtual Volumes (vVols) are a storage technology that provides policy-based, granular storage configuration and control of virtual machines. Through API-based interaction with an underlying array, VMware administrators can maintain storage configuration compliance using native VMware interfaces. vVols offer the ability to remove abstractions between applications running on virtual systems and the underlying storage. This concept ensures that the focus moves away from the infrastructure and focuses more intently on the application running in the virtualized environment.

With FlashArray, vVols allow for virtual disks and regular block storage volumes to be seen as one and the same. A datastore is presented to the ESXi host and any virtual disks migrated into it will have a corresponding FlashArray volume that can be seen and managed from the array interfaces. Virtual disks are limited to 62TB in size and any virtual machine with many of these would need to be divided amongst many VMFS datastores. With vVols, all these large virtual disks can be contained within the single vVol datastore, vastly simplifying management at this scale. This can be seen in Figure 4.

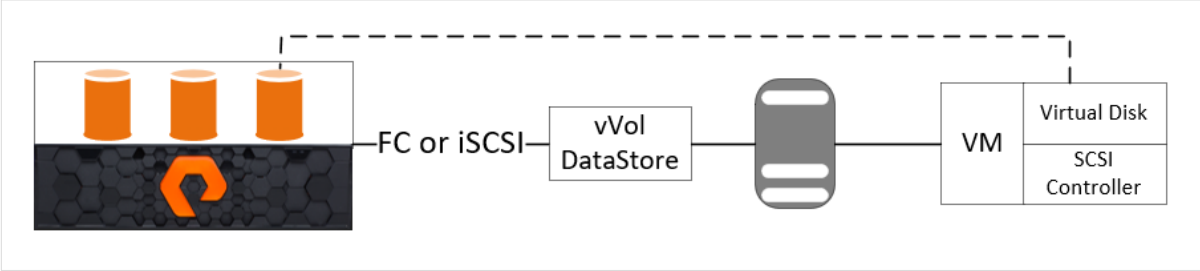

FIGURE 4 vVols allows for a volume to be created on FlashArray that directly represents the virtual disk

Detailed instructions on the configuration of vVols with FlashArray is beyond the scope of this document. The below table provides a high-level overview of the configuration items with links to detailed descriptions and best practices in the Web Guide: Implementing vSphere Virtual Volumes with FlashArray.

vVol Configuration Item	Description and Recommendations
vVol Requirements and Recommendations	vVols are only available on specific versions of VMware vCenter, ESXi, Purity//FA, and FlashArray. Best practices include: • Use Purity//FA 6.2.10 or later • Use VMware vCenter 7.0 Update 3f (build 20051473) or later • Use VMware ESXi 7.0 Update 3f (build 20036589) or later • When registering the VASA provider, use a local FlashArray user account • Do not run VMware vCenter VMs on vVols. Visit this section of the Failure Scenario KB for more information. • The protocol endpoint should be connected to host groups and not individual hosts. • Configure snapshot policies for all config vVols. • Use virtual machine hardware version 15 or later.
Configuring the FlashArray VASA Provider	The vSphere APIs for Storage Awareness (VASA) is the interface that VMware vCenter and ESXi use to interact with storage arrays. FlashArray supports VASA version 3.0, which was introduced with VMware vSphere 6.5. As such, the vSphere cluster must be running vSphere 6.5 or higher. A FlashArray VASA provider is a service that is registered with VMware vCenter. Specific steps must be followed to register the FlashArray VASA provider: 1. Create and manage VASA provider certificates 2. Register the FlashArray VASA provider using the Pure Storage vSphere plugin, the VMware vSphere Client, or PowerShell. 3. Verify the VASA provider registration. VMware administrators can also remove and de-register VASA providers using the vSphere Client and PowerShell.
Configuring Host Connectivity	A host object is a collection of initiators that refer to a physical host. A FlashArray host object must have a one-to-one relationship with an ESXi host, and every active initiator for an ESXi host must be added to the FlashArray host object. A host group is a collection of host objects.We recommend you deploy your ESXi hosts into a vSphere cluster within vCenter, with each cluster that uses FlashArray storage having its own host group. Each ESXi host in the cluster must have a corresponding host that is added to a host group. Pure Storage also recommends that the number of ESXi hosts in a host group match the number of hosts in the vSphere cluster. Each ESXi host object should have the ESXi host personality enabled. Once you've created host objects and host groups, you can then connect FlashArray volumes to the host objects or host groups using the Pure Storage vSphere plugin.
Protocol Endpoints	VMware uses protocol endpoints (PEs) to avoid limitations of LUNs, such as the 512-device limit per host and time-consuming I/O interconnect rescans. Each PE allows up to 16,383 vVols bound to each ESXi host simultaneously and doesn't require an I/O rescan when a new binding is created. FlashArray administrators can manage PEs through either the FlashArray UI or CLI. VMware administrators can view PEs that are connected to ESXi hosts using the VMware vSphere Client.

vVol Configuration Item	Description and Recommendations
vVol Datastores	vVols replace LUN-based VMware datastores that are formatted with VMFS. vVol datastores do not contain a file system, and vVol-based virtual disks are not encapsulated within files like they are on VMFS. vVol datastores are mounted to an ESXi host with access to a PE on the array that hosts the vVol datastore. Mounting a vVol first requires registering a VASA provider, and connecting a PE to the ESXi host. Once these tasks are complete, VMware administrators can use a number of methods to mount vVol datastores.
Types of vVols	FlashArray organizes vVols associated with a VM as a volume group. When a VMware administrator creates a vVol-based VM, the FlashArray that hosts the VM creates a volume group with the following naming schema: vvol-{VM Name}-{unique 8-character string}-vg. vVols do not change the structure of a virtual machine: • Each VM has a VMX configuration file that describes the VMs virtual hardware and settings • Each powered-on VM has a swap file • Each VM's virtual disks are storage objects that limit guest OS disk capacity • Each VM has a vmem memory file that stores snapshots of its memory state. While vVols do not contain a file system, VMware vSphere makes the structure appear to be that of a conventional VM on a VMFS datastore. Additionally, vVol-based VMs use four types of vVols: • Configuration vVol, which stores the VM's VMX configuration file, pointers to the virtual disk VMDK files, and other files. • Data vVol, which corresponds to a virtual disk • Swap vVol, which contains the VMs swap data • Memory vVol, which contains VM suspension or snapshot data. Note that the VM snapshot data is different from FlashArray snapshots. VMware administrators can view and manage each vVol using the VMware vSphere Client.
Recovering Deleted vVols	VM data and configuration vVols can be recovered with 24 hours of deletion using a variety of methods. vVol recovery can use both FlashArray and vSphere tools.
vVol Binding	One of the primary goals of using vVols is the ability to scale. Using PEs instead of LUNs helps achieve that goal. With PEs, ESXi hosts can bind and unbind (connect and disconnect) vVols dynamically as needed. ESXi hosts can also provision VMs and power them on and off even if VMware vCenter is not available. vVols are bound to specific ESXi hosts for as long as they are needed. vVols are bound to an ESXi host under the following circumstances: • The VM is powered on • A folder is navigated to a vVol datastore using the VMware vCenter GUI, SSH session, or console • A VM is moved using VMware vSphere vMotion • A VM is created, deleted, or reconfigured • A VM clone is created from a VM or template • A snapshot is taken of a VM Note that binding and unbinding is automatic. No administrator input is required.

vVol Configuration Item	Description and Recommendations
Snapshots of vVols	vVols allowVMware administrators to snapshot individual VMs using storage array snapshot capabilities instead of VMware's snapshot capabilities. VMware and Pure Storage recommend using vVol snapshots instead of VMware snapshots, as VMware snapshots introduce latencies that negatively impact performance. As with VMware-based snapshots, array-based managed snapshots that are created through the VMware vSphere Client can include snapshots of the VM's memory and snapshots that quiesce the file system. Administrators can revert VMs to a snapshot and delete snapshots. Snapshots that are created outside of the VMware vSphere Client using tools such as the FlashArray GUI, CLI, or REST interfaces are called unmanaged snapshots. This type of snapshot can be used in a VMware environment, but with certain restrictions. VMware administrators can manage vVol snapshots using the FlashArray plugin, including viewing vVol details, creating managed snapshots, restoring a vVol from a snapshot, and creating copies of a vVol from a snapshot.
Storage Policy-based Management	One of the unique benefits of using vVols is the ability to apply granular storage policies to each vVol. Since vVols are native to VMware vSphere and integrated, VMWare administrators do not need storage vendor-specific tools to manage vVols, as array functionality is integrated directly with VMware vSphere. Administrators use native VMware vSphere tools, such as the VMware vSphere Client or PowerCLI. Storage policies are collections of storage capabilities that are defined by VMware administrators. Storage capabilities are specific to arrays. When a storage policy is applied, VMware filters out non-compliant storage so that only compliant options are presented for configuring storage for VMs or vVols. FlashArray provides several storage capabilities that are available and cannot be disabled, including data at rest encryption, deduplication, compression, RAID protection, and flash storage. Other capabilities are available and can be configured by VMware administrators. Using native VMWare vSphere tools, VMware administrators can also create a storage policy, check VM storage policy compliance, and assign storage policies to a VM or virtual disk.
Replicating vVols	FlashArrays can replicate vVols to other FlashArrays. VMware vCenter is aware of replicated VMs and can fail them over to a FlashArray at a remote location. VMware administrators can manage replication using tools such as the VMware Web Client, PowerCLI, REST APIs, and VMware vRealize Orchestrator. FlashArray vVol replication is also supported by VMware Site Recovery Manager and is certified by both VMware and Pure Storage. More information is available in SRM User Guide: vVol Periodic Replication SRM Workflows.

vVol Configuration Item	Description and Recommendations
vVol Reporting	vVol integration with FlashArray gives VMware vSphere insights into vVol storage. Additionally, the same integration gives FlashArray the ability to recognize and report on both entire vVol-based VMs that are implemented as volume groups, and individual virtual disks that are implemented as volumes. For example, the Volumes tab in the FlashArray Storage pane lists and array's volume groups. The volume group naming schema follows the following pattern: vvol-VMname-vg. The VM name is set when the VM is first created. When a VM is renamed through VMware vCenter, the volume group is not automatically renamed on the FlashArray. The volume name can be renamed or updated using a manual workflow. A FlashArray can also report on VM and vVol performance history. The history of a VM's or vVol's IOPS, latency, and data throughput (bandwidth) can be viewed through the FlashArray GUI.
Migrating VMs to vVols	VMware administrators can use VMware Storage vMotion to migrate VMs from VMFS, NFS, or raw device mapping (RDM) disks to vVols. When migrating, VMware administrators select a FlashArray vVol datastore as the migration target. During migration, administrators can select a storage policy for the migrated VM that provides additional features. When migrating from VMFS or NFS datastores, since the administrator is using Storage vMotion, the VM remains online and available during the migration. RDM migration is more involved and might require VM downtime.
Data Mobility with vVols	Since a vVol-based VM's storage VMDKs is directly correlated with volumes on FlashArray, the VM's data can be easily shared and moved. A data vVol is a virtual block device that is attached to a VM. Therefore, a data vVol or a volume created by copying a snapshot of it can be read by software that understands its contents, such as an NFS or XFS file system created on the vVol by the VM. As such, it is possible to present a data vVol or the volume created from a snapshot of a data vVol to a physical server. Additionally, a volume created by a physical server can be presented to a vVol-based VM as a vVol.

TABLE 3 vVol configuration recommendations for use with Pure Storage FlashArray

FlashArray volumes with VMFS. A single volume can serve as a VMFS datastore of up to 64TB in size. This volume can then contain many virtual disks for one or more virtual machines that collectively do not exceed the datastore size. Figure 5 shows how the volume can relate to a virtual disk.

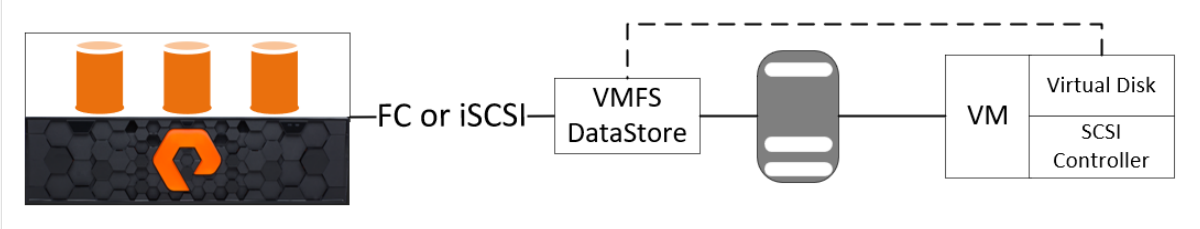

FIGURE 5 Many virtual disks can be contained within a single VMFS datastore which maps to a single volume

FlashArray volumes as mapped devices. A single volume can serve as a mapped device with an RDM. This provides up to 64TB of storage for a single virtual machine. The mapped device consumes the entire volume which can then be shared between one or more virtual machines. There are a number of considerations and limitations for RDMs which are listed in the VMware documentation library. RDMs are often used for special cases with Microsoft SQL Server, such as an Always On Failover Cluster Instance where other options for shared storage may not be an option. Figure 6 shows what an RDM looks like from a FlashArray perspective.

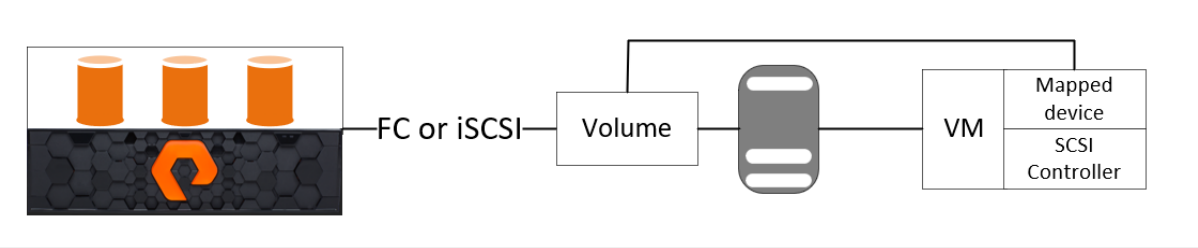

FIGURE 6 A mapped device can be used by one or more virtual machines but the entire volume for the RDM is consumed

Configuration and Best Practices

This section highlights the recommended configuration per component in this reference architecture.

VMware provides a comprehensive best practice guide for Microsoft SQL Server on VMware which should be used as a further reference for the recommendations below.

Network Optimization

Pure Storage recommends optimizing network settings for Fibre Channel, iSCSI, and host-specific implementations. The below configuration items should be taken into consideration when reviewing network optimization settings:

- SAN design and setup: The SAN design and setup is an important consideration for optimal performance. Pure Storage recommends specific standards when using Fibre Channel, iSCSI, or host-specific protocols. The web guide "FlashArray VMware Best Practices" contains the specific standards and links to host-specific best practice guides.
- Topology: When designing a SAN network, more hops equal higher latency, which negatively impacts storage performance. Pure Storage recommends a flat fabric where the FlashArray is only one hop away from applications that run from it. For example, iSCSI should not be routed. Also, watch for topological bottlenecks and blade server chassis oversubscription.
- Physical paths: This applies to Fibre Channel and iSCSI protocols. Storage devices are often oversubscribed, so you should use all of the FlashArray's ports to balance connections between hosts and the FlashArray. This provides more pathways, resiliency, and performance; mitigates physical problems; and takes better advantage of CPU allocation.
- Clean paths: A clean physical cable plant is a requirement for optimal performance. Make sure cable tips are clean and monitor switches for physical layer errors using tools such as Brocade's portErrShow tool. Avoid patch panels if possible.

- Port connection speeds -This applies to Fibre Channel and iSCSI protocols. For best performance, your network should use 8 Gb/s, or 16 Gb/s host bus adapters (HBAs). If you have 2GB or 4Gb/s HBAs, you should add more physical paths and use multipathing, or fix ports speeds to one step down if you are two or more port speeds down from your highest speed HBA. For example, if you are running 2Gb/s in an 8Gb/s SAN, you should step down to 4Gb/s on the FlashArray. Various methods are available for checking port speed on FlashArray and switches. The minimum recommended ethernet port speed for iSCSI is 10Gbe.
- Zoning -This applies to Fibre Channel. You can zone any single initiator to as many FlashArray ports as you need. Configuring the same initiator to multiple targets is acceptable, while configuring multiple initiators to multiple targets is not recommended.
- Jumbo frames: Jumbo frames allow Ethernet frames larger than 1,518 bytes. While the default maximum transmission unit (MTU) for most devices is 1,500 bytes, FlashArray can support up to 9,000 bytes. Jumbo frames must be enabled across the full path from initiator to target, including HBAs and switches.

FlashArray

The following recommendations should be followed when using FlashArray with VMware vSphere:

- **Use host groups that correspond to VMware clusters**: FlashArray uses the Hosts term to refer to a collection of initiators for a physical host inside a single management domain. These collections of host objects in FlashArray are known as Host Groups.
- **Ensure that each ESXi Host's initiators are set to use the "ESXi Host Personality":** Setting the behavior for a specific host in FlashArray causes the array to change some of its behavior for specific host types. This ensures that various configuration options for VMware ESXi storage are interpreted correctly.
- **Connect volumes to host Groups**: When grouping ESXi hosts into clusters within VMware–providing functions such as High Availability and Dynamic Resource Scheduling–it is recommended to connect volumes to a host group.

For more detail on these recommendations see "FlashArray VMware Best Practice User Guide: FlashArray Configuration."

Datastore

The following recommendations should be followed for datastores in VMware vSphere built upon FlashArray volumes:

- **Use the latest version of VMFS that is permitted by the ESXi host**: VMFS-6 should be used for datastore provisioning where possible.
- **Increase HBA or iSCSI initiator queue depth limits when necessary**: The queue depth dictates how many I/Os can be outstanding for a specific device before queueing in the ESXi kernel. A balance should be found for the value set here and is workload dependent as if the value is too high virtual machine I/O fairness will be affected and if too low IOPS and throughput can be limited.
- **Provision additional datastores if the default queue depth per datastore is overwhelmed**: This is attributed to the Disk Schedule Number Requests Outstanding (DSNRO) value that manages the queue depth for an individual storage device. Provisioning additional datastores and migrating virtual disks to the new datastore will alleviate the bottleneck.

For more detail on these recommendations and other insights, see the FlashArray VMware Best Practice User Guide – Datastore Management.

VMware ESXi Host

The following recommendations should be followed for ESXi Host FlashArray storage configuration in VMware vSphere:

- Use the Round Robin Path Selection Policy for FlashArray volumes. This is set automatically in ESXi 6.5 Update 1 and later. For ESXi 7.0 and later the latency-based sub policy is automatically applied.
- (iSCSI connectivity) Set the Login Timeout to 30 seconds
- (iSCSI connectivity) Disable DelayedAck iSCSI option
- (iSCSI connectivity) Use jumbo frames, which requires end-to-end configuration of network infrastructure such as switches and port configuration.
- (iSCSI connectivity) Decrease failover times for sensitive applications. The official best practice is to utilize the default iSCSI configuration for failover times; sometimes performance sensitive applications require the failover time to be lowered.

For more detail on these recommendations and other insights see the FlashArray VMware Best Practice User Guide – iSCSI Failover Times.

For VM performance, ensure that the following are set:

- For the physical host, in UEFI/BIOS, ensure that the below options are set. These options will be named differently depending on the hardware manufacturer. You may also want to see if the hardware vendor has a specific VMware best practices guide.
  - Enable Turbo Boost
  - Enable Hyperthreading
  - NUMA Enabled
  - Enable VT-x/VT-d
  - Power Management set to OS Controlled
  - Disable all processor C-states (C1E halt states
- For each ESXi host, ensure that the power management policy is set to High Performance.

Virtual Machine

The following recommendations should be followed for Virtual Machines utilizing FlashArray backed datastores with Microsoft SQL Server deployed on them:

- **Use the maximum number (four) of supported Paravirtual SCSI (PVSCSI) adapters**: A single virtual machine can be provisioned with up to four PVSCSI adapters. The goal is to spread the I/O load as evenly as possible. One example model is as follows:
  - Adapter 1: Virtual disks for the operating system, system databases and database backup volumes.
  - Adapter 2: Virtual disks for database data files.
  - Adapter 3: Virtual disks for database transaction log files.
  - Adapter 4: Virtual disks for tempdb data and log files.

As with a physical server, a single adapter can become overloaded. You may want to do things like put data files on different disks, not just ones that are attached to Adapter 2. Know your workload and configure your disks accordingly to achieve better performance.

- **Queue depth for PVSCSI adapters**: On Pure Storage, most customers can benefit from changing the default queue depth of 32 to 64 but that should be tested in your environment.
- **(Optional) Provision separate datastores for each database**: This becomes functionally more useful when utilizing FlashArray data services such as high availability, continuous replication, or volume snapshots.
- **(Optional) Provision separate virtual disks for each database**: This becomes functionally more useful when using vVols and FlashArray data services such as volume snapshots and capacity management.
- **Take special note of VMDK and VMFS limits**: When creating virtual disks larger than the supported maximum (62TB) either vVols or multiple smaller virtual disks across multiple datastores should be used.
- **Networking**: Use the VMXNET3 network adapter
- **Virtual machine hardware version:** Ensure that the virtual machine is upgraded to the latest version of virtual machine hardware
- **Memory reservations:** If looking to fully optimize a VM for performance, reserve memory to remove page swapping and sharing.
- **CPU configuration**: Be mindful of assigning too many logical CPUs and crossing NUMA boundaries, only SQL Server Enterprise is NUMA aware. For both memory and CPU, avoid overallocation of resources at the ESXi host level.

Operating System

Windows Server. The following recommendations should be followed for SQL Server using Windows Server:

- **Folder structure**: Use either virtual disks with drive letters or mount points, but ensure that specific locations for each database file function are provisioned similar to the following example.
  - SQL Server binaries: Install SQL Server on the operating system drive. Databases and backups will be placed on other drives.
  - Data files: All databases and backup files should not be placed on the same disk where the operating system and binaries exist. Whether using drive letters or mount points, use drive letters and folders with naming conventions that make sense for your environment. The ones used here are just examples. Do not put SQL Server data and log files on the root of a drive or a mount point.
    - For example, use a folder on the D drive such as D:\DATA or a mount point under the operating system drive such as C:\SQL\DATA. All of the data files can be located in a single virtual disk, or each database can have a virtual disk provisioned for it (recommended) and then mounted under this structure. Example for the data files of "UserDB" would be to have a virtual disk mounted at C:\SQL\DATA\UserDB or D:\DATA\UserDB.
  - **Transaction log files**: For example, a folder named LOGS on the L drive) **L:\LOGS** or a mount point under the operating system drive such as **C:\SQL\LOGS**. All of the transaction log files can be located on a single virtual disk, or each database can have a virtual disk provisioned for it (recommended) and then mounted under this structure. Example for the log files of "UserDB" would be to have a virtual disk mounted at **C:\SQL\LOGS\UserDB** or **L:\LOGS\UserB**.

  - **tempdb**: Provision a single virtual disk to contain the data and log files for tempdb. Either **T:\ tempdb** or a mount point under the operating system drive such as **C:\SQL\tempdb**.
    NOTE: *SQL Server data files cannot be created on the root of a filesystem in Windows Server, additional folders will need to be created to contain them.*
  - **Database files spread across multiple volumes:** If leveraging the FlashArray SQL Server Management Studio Extension ensure the database layout avoids a database layout with a structure of having multiple databases, with each dedicating its data file to a separate volume/folder but all of the transaction log files for the multiple databases on a single volume/folder. A good database design to use with the extension is one that either has a consistent deployment of data and transaction log files for multiple databases or completely segregates the database files by using dedicated volumes for a single database.

- **File system and disk formatting**: Format all SQL Server volumes with the NTFS or ReFS filesystems (most use NTFS and a minimum of 64KB Allocation units. This must be done before placing anything on the disks and is part of formatting.
- **Block alignment**: Check if the block alignment is configured properly as per the Windows File Systems for FlashArray knowledge article.
- **Anti-virus and threat detection systems**: At a minimum, ensure that systems running SQL Server exclude files with the extension MDF, NDF, LDF, and BAK. Other folders and files may need to be excluded. Follow Microsoft's recommendations in this article.
- Set the power plan to high performance
- **Configure database instant file initialization:** Enable instant file initialization as per Database Instant File Initialization.
- **Page file size**: The best practice for any server running SQL Server is to have enough memory so that it does not page to disk. Use the Microsoft Windows best practice to determine an appropriate page file size. However If the page file resides on the operating system volume, which itself is also on a FlashArray datastore (vVols or VMFS) then it is recommended that the SCSI controller for that device is set to be a Paravirtual SCSI Adapter.

Linux. Linux is only supported as a platform from SQL Server 2017 onwards.

The following recommendations should be followed for SQL Server on Linux:

- **Folder structure**: The default location for all Microsoft SQL Server data files is **/var/opt/mssql**. Default folders for data, transaction log and tempdb files are created during installation (DATA - **/var/opt/mssql/data**, LOG - **/var/opt/mssql/log**, tempdb - **/var/opt/mssql/tempdb**).

  - The tempdb database files should be migrated to a separate virtual disk to achieve the best performance. This virtual disk can be mounted at the default tempdb location after the files have been migrated to it.
  - Any additional user databases should have separate folders for data and transaction log files created for them.

- **Filesystem, volume formatting and mount options**: Format virtual disks with the XFS or EXT4 filesystems and mount them with the "**rw,attr2,noatime**" attributes.
- **Open file limitations**: Set a soft limit of 16000 and a hard limit of 32727.
- Forced Unit Access I/O subsystem capability For Linux operating systems that support Force Unit Access configure **control.writethrough =1 and control.alternewritethrough = 0**.

- **Disk settings:**
  - IO Scheduler: Use the none scheduler for all virtual disks that contain SQL Server Data Files. See the Red Hat Solution on How to use the Noop or None IO Schedulers.
  - Disk readahead: set the readahead setting to 4096.
  - vm.swappiness: Set vm.swappiness to 1 for high performing systems.
  - vm.dirty_*: Set vm.dirty_ratio = 80 and vm.dirty_background_ratio = 3
  - kernel.sched_*: Set kernel.sched_min_granularity_ns = 15000000 and kernel.sched_wakeup_granularity_ns = 2000000

More detail on the recommendations for Linux operating systems can be found in the Linux Recommended Settings knowledge article.

Microsoft SQL Server Instance

The following recommendations should be following for configuring the SQL Server Instance:

- **Service accounts**: Since most environments using SQL Server use Active Directory Domain Services, the SQL Server service accounts should use a domain-based service account. A Group Managed Service Account (gMSA) is recommended since the password is managed by Active Directory Domain Services. This means it is more secure. Ensure that this is the same user who has the correct permissions to the mount points or drive letters being used for the various SQL Server data, transaction log, and backup folders and files.
- **tempdb**: For an initial configuration, create one tempdb data file per vCPU up to eight cores. Ensure they are evenly sized and grow capacity at the same rate. Know your usage of SQL Server to create the right tempdb configuration.
- **Optimizations**:
  - Set Max Memory to recommend 75% of server memory. There is no need to set the min value. Microsoft recently released their recommendations on memory settings and you can find more information about it in this guidance. **NOTE:** *there are always exceptions based on workload and features in use.*
  - Max Degree of Parallelism (MAXDOP) set to less than or equal to the number of cores in a NUMA node of less than or equal to eight. This is an advanced setting. It is recommended that the Microsoft documentation be read before changing this setting.
  - Cost Threshold for parallelism should be optimized while a workload is running but start at a value of fifty (50).
  - Lock pages in memory when set can prevent external pressure to deallocate memory SQL Server processes.
- **(SQL Server 2019 onwards)Turn on Accelerated Database Recovery:** Accelerated Database Recovery improves database availability , especially in the presence of long running transactions.
- **Enable query store**: Query store was introduced in SQL server 2016, it collects information about query processing and runtimes and stores it. In most environments, Query Store should be enabled but test with your workloads first. It does not work in some scenarios such as Always On Availability Group (AG) readable replicas yet. It does add minimal overhead but the benefits outweigh the cost. In 2019 and later which queries, the number of executions and how long they are executed can all be configured. Check out more details on setting this up here.

- **Ad-hoc workload optimization**: This is a setting that will limit execution plans from being cached until it is executed a second time. The first time only a stub is saved. This prevents memory bloat from execution plans. It is recommended that this be turned on.
  - Additional information on the feature can be found in Microsoft learn.
- **Auto update statistics asynchronously**: This is recommended to be turned on; this database level setting allows queries to continue to execute during the update of index statistics.

FlashArray Data Service: Volume Snapshots and Asynchronous Replication

Volume snapshots and asynchronous replication are data services included in every FlashArray. These data services can be used with virtualized SQL Server instances for a range of use cases such as data protection, seeding AGs and cloning databases.

Volume Snapshots

Block storage volumes on FlashArray support the creation of snapshots which are immutable, point-in-time images of the contents of one or more volumes. Volume snapshots can be created individually or as a part of a Protection Group. When creating a Protection Group snapshot all of the member volumes will have snapshots created for them which are consistent with one another. Volume snapshots on FlashArray are unique for a number of reasons. They:

- **Can be instantly created or recovered.** Volume snapshots are created using the redirect-on-write technique meaning that only the blocks in a volume that have changed since the most recent snapshot are saved.
- **Are storage space-efficient.** Volume snapshots are always thin-provisioned, deduplicated, compressed, and require no snapshot capacity reservations. All volumes (and by extension snapshots) are thin provisioned and a part of a global deduplication namespace, further reducing the storage space consumed.
- **Are portable and transport-efficient.** Volume snapshots are portable which allows for them to be transferred to one or more FlashArray or Pure Cloud Block Store instances. All transfers are done using space efficient copies accompanied by the volume metadata allowing for any offloaded snapshot to be recovered to any FlashArray or Pure Cloud Block Store instance.
- **Can be used for many use cases.** Volume snapshots can be used for many different use cases to make business operations more efficient and effective. More information on these use cases is found in the white paper: Using Databases with FlashArray Volume Snapshots.
- **Protect data from ransomware with SafeMode Snapshots.** SafeMode Snapshots ensure that volumes and snapshots are not eradicated from a system using a policy-driven retention mode policy. The policy can only be changed by Pure Storage support with instruction from a trusted contact.

Asynchronous Replication

Asynchronous replication is the copying of volume snapshots from one FlashArray to another. The transfer is efficient as only the data that doesn't already exist on the target will be transferred over the wire. Asynchronous replication is fantastic for applications requiring a recovery point objective as low as five minutes or for a data mobility strategy which requires data to be copied across sites or to different FlashArray or Cloud Block Store instances. Figure 7 shows what this process looks like.

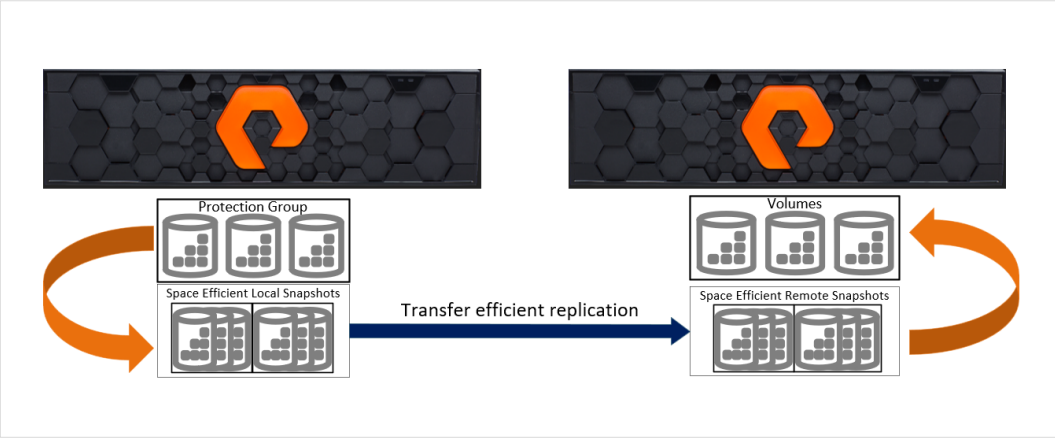

FIGURE 7 Asynchronous replication of Volume Snapshots from a Source to a target

Pure Storage FlashArray Management Extension for Microsoft SQL Server Management Studio

The Pure Storage FlashArray Management extension supports the creation of application consistent volume snapshots using the Pure Storage Volume Shadow Copy Service (VSS) Hardware Provider for both physical and virtualized Microsoft SQL Server implementations.

Information on how to acquire and use the FA SQL Server Management Studio (SSMS) FlashArray Management Extension can be found in the Microsoft Platform guide. An example of what this looks like in SSMS is shown in Figure 8.

The extension supports the following features:

- Taking application consistent volume snapshots as recovery points of running databases.
- Restoring databases from snapshot using restore with No-Recovery (allowing for point in time recovery) or database mount (restore to a different location or database instance) options.
- Automating operations using the Pure Storage PowerShell Backup SDK
- Volume snapshots can be created for databases with virtual disks in physical raw device maps (pRDMs) and virtual volumes.

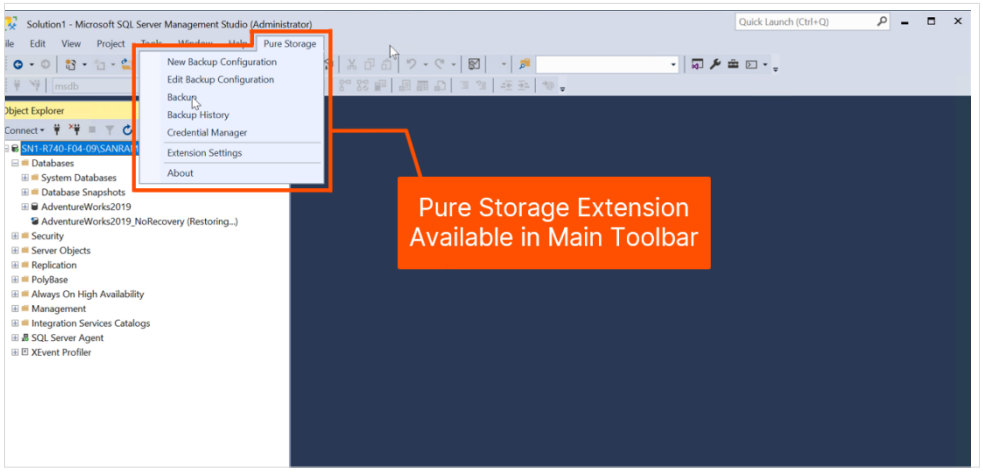

FIGURE 8 The Pure Storage FA SSMS extension integrated directly into the SSMS interface

Using Volume Snapshots with Databases That Are Part of a Virtualized SQL Server Instance

Creating and using volume snapshots with databases that are part of virtualized Microsoft SQL Server implementations is dependent on the datastore types used for the virtual disks on which the database files reside.

Volume snapshots of databases can be created using one of two methods: application consistency (known as a volume shadow copy snapshot) or crash consistency.

Application consistent volume snapshots can be created using the SSMS FlashArray Management extension; this ensures that the snapshot is captured alongside any operations required to make it consistent with the application at any one point in time and allow the database to achieve transaction level point-in-time restore with SQL Server native backups. Application consistency with SQL Server databases is achieved through the use of the Volume Shadow Copy Server (VSS) and SQL Writer (SQL Server 2019 and below) or Transact-SQL snapshot backups (SQL Server 2022 onwards). Combining application consistent snapshots with transaction log backups allows for the administrator to control how far to roll the database forward from when the snapshot was taken using the transaction log backups. To create application consistent snapshots for SQL Server versions up to 2019 the volume shadow copy process in Windows Server needs to be used. From SQL Server 2022 onwards volume snapshots can be created using the Transact-SQL backup process for Microsoft Windows and Linux based deployments.

Crash consistent volume snapshots work independently from an application and will always present valid databases from the point in time of the snapshot but with a lower recovery point objective. The choice of consistency is reliant on the granularity level of restoring a database, application consistency provides point-in-time while crash consistency can achieve recovery point objectives as low the snapshot creation time.

Special note needs to be taken on the difference between a volume snapshot and a virtual machine snapshot. A virtual machine snapshot preserves the state and data of a VM at a specific point in time. It includes the memory contents (when the virtual machine is powered on) and virtual disk state. The preservation of the virtual disk state is accomplished through the creation of a delta disk–where all changes post the snapshot creation are recorded. As the snapshot is retained the associated delta disk will grow in size and only once the snapshot is destroyed will its contents be merged or discarded. Virtual machine snapshots should not be used as a backup for any virtual disk associated with it and running a virtual machine for an extended period of time with a snapshot can cause instability and data loss. More information on virtual machine snapshots can be found in the Overview of virtual machine snapshots knowledge article. Due to the focus only on the underlying storage for a virtual machine, volume snapshots do not suffer from the same limitations as a virtual machine snapshot making them ideal for long term retention.

Virtual Disks in a Virtual Machine File System

If the database files reside in one or more virtual disks located on one or more VMFS datastores then the following recommendations should be followed when using volume snapshots:

- Only the virtual disks located on the VMFS datastore are for a single database or for all of the databases in a single instance.
- It is recommended that larger databases are isolated onto their own virtual disks and VMFS datastores. These databases should not share VMFS datastores with other databases, even from the same instance.
- The virtual disks used for data and transaction log files can be held on separate VMFS datastores, but snapshots need to be created from a protection group that contains all applicable volumes to preserve consistency.

Example: Creating a volume snapshot for a single database with virtual disks on a single VMFS datastore. The following is an example of hosting a single database on two separate virtual disks, but both within the same VMFS datastore and then taking a crash consistent volume snapshot of the underlying volume.

This is a database made up of five files. Four of these files are database data files located in a mount point at C:\SQL\Data and one is a transaction log file located in a mounted filesystem at C:\SQL\Log. The structure can be seen in Figure 9.

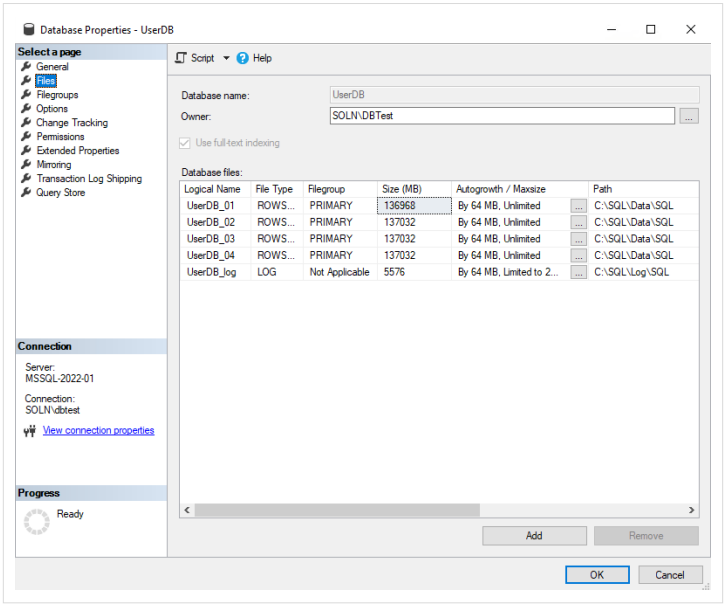

FIGURE 9 The database properties view for UserDB

Figure 10 shows what this looks like from a Windows Server perspective. The disks are formatted with NTFS and have relevant names.

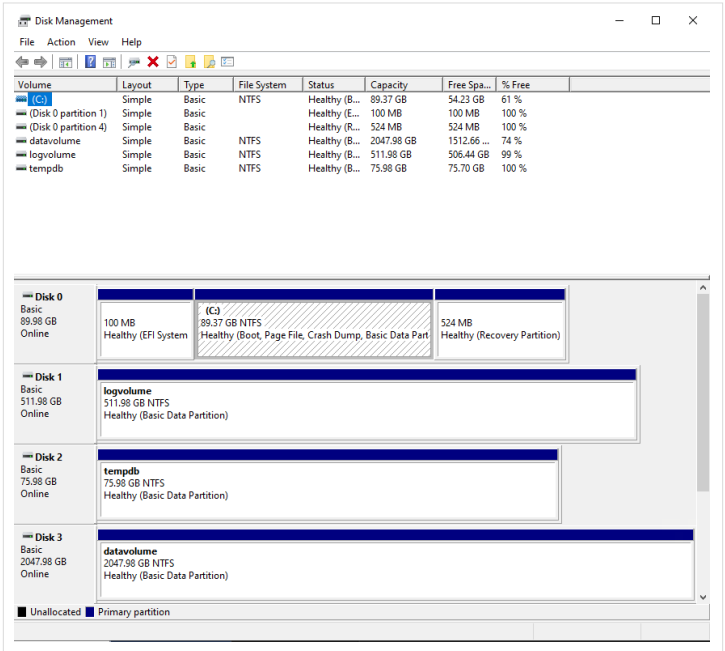

FIGURE 10 The volumes on which the UserDB data and log files.

Figure 11 shows what this looks like when configuring the VM in vCenter.

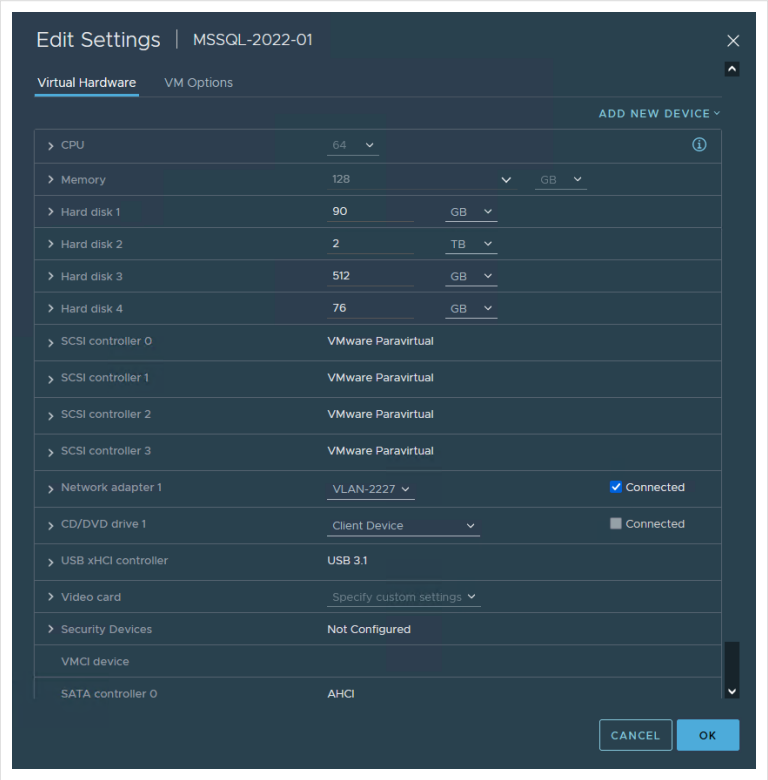

FIGURE 11 The virtual machine configuration with the virtual disks for the UserDB data and log files

Figure 12 shows the virtual disks, or VMDKs, associated with the VM. They are located on a single datastore "MSSQL-2022-01-DS".

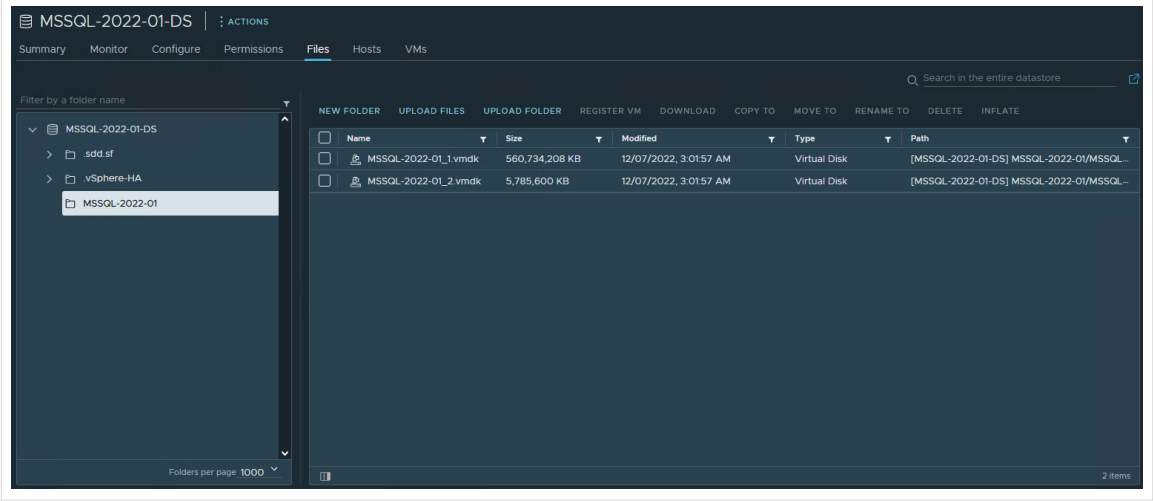

FIGURE 12 The virtual disks associated with the UserDB data and log files

To create a volume snapshot of the datastore with the virtual disks containing the UserDB data and log files, follow these instructions.

1. Navigate to the volume in the FlashArray graphical user interface and identify the **Volume Snapshots** section. In this section select the **+** to create a volume snapshot. An example is shown in Figure 13.

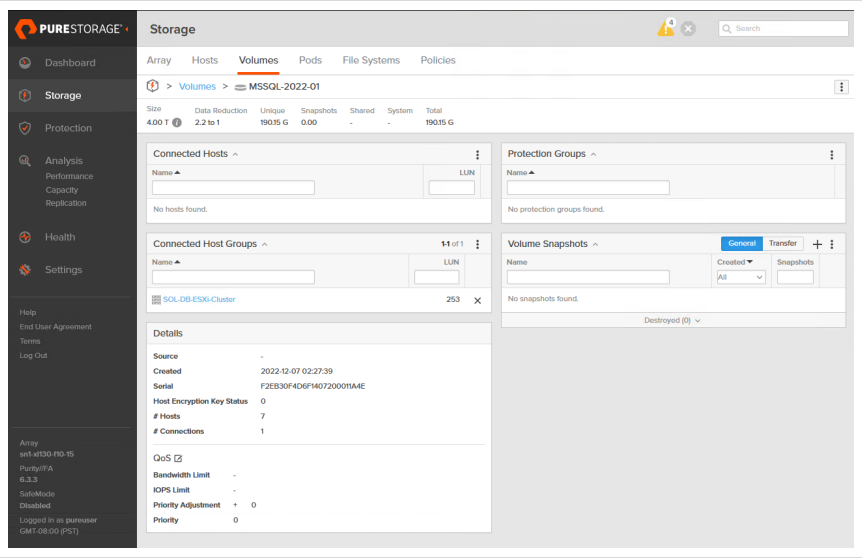

FIGURE 13 The volume details view in the FlashArray graphical user interface

2. Provide an optional suffix for the snapshot to identify it and then select Create as shown in Figure 14.

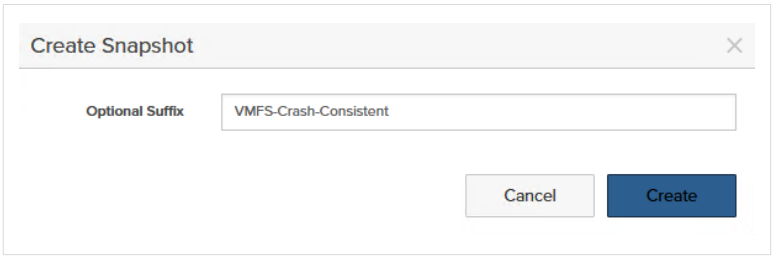

FIGURE 14 The Create Snapshot dialog

3. Once the volume snapshot has been created it will show up in the list of volume snapshots in the volume detail view as shown in Figure 15.

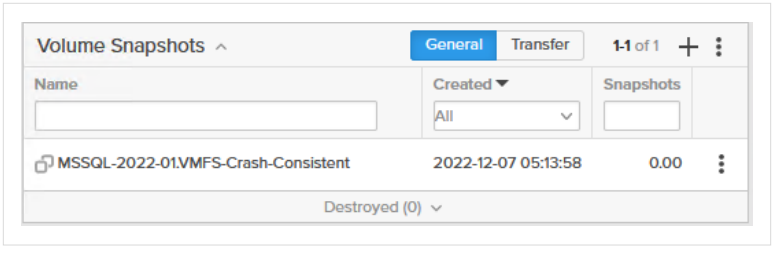

FIGURE 15 The list of volume snapshots in the volume detail view

Example: Recovering a volume snapshot for a single database with virtual disks on a single VMFS datastore. Using the snapshot created in the "Creating a volume snapshot for a single database with virtual disks on a single VMFS datastore" example the following steps showcase how a SQL Server database can be recovered to the same system using volume snapshots.

1. In the volume detail view of the FlashArray graphical user interface identify the snapshot to recover and select the three ellipses next to it to bring up the context menu. In this context menu select **Copy...** as shown in Figure 16.

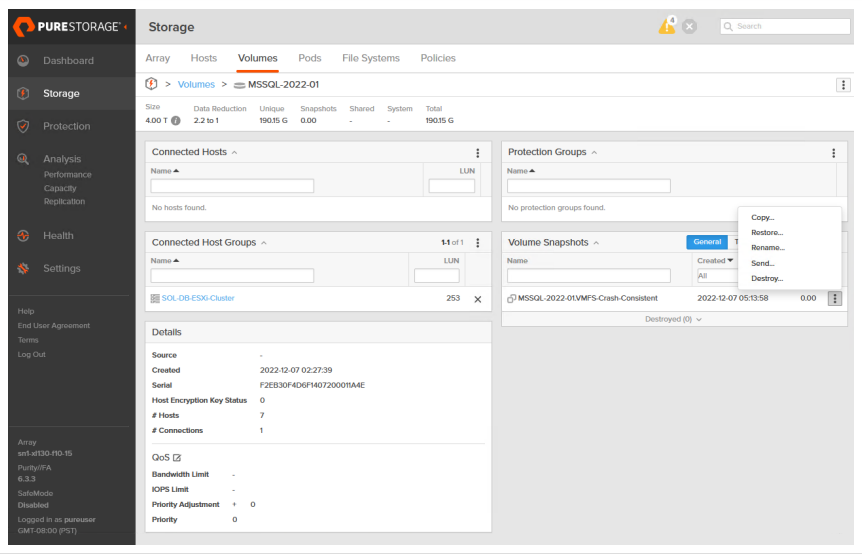

FIGURE 16 The context menu for the volume snapshot Restoring is also an option but overwrites the original volume.

2. In the Copy Snapshot dialog shown in Figure 17, provide a name for the new volume and then select **Copy**

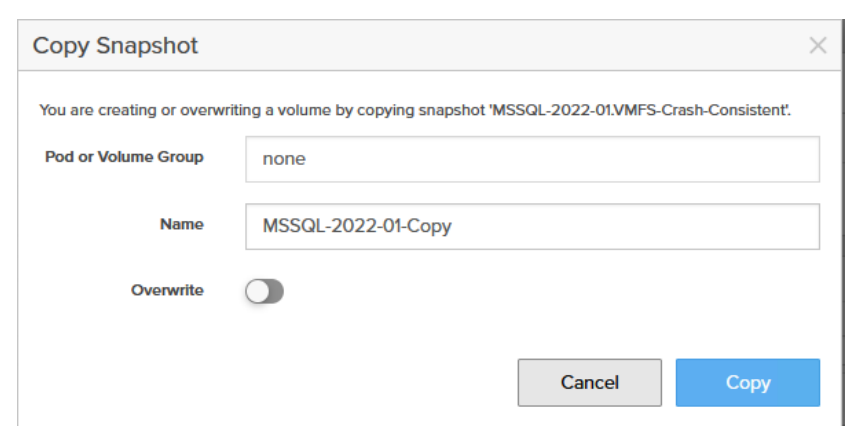

FIGURE 17 The Copy Snapshot dialog

3. Once the volume has been copied it will show up in the volume view. It can now be attached to an ESXi host. An example is shown in Figure 18.

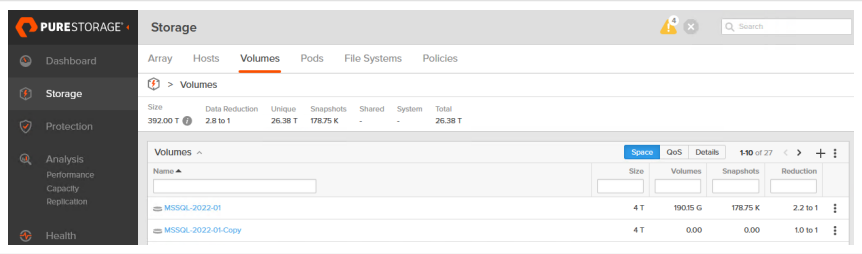

FIGURE 18 The list of volumes in the FlashArray When attaching a volume which is a clone of a VMFS datastore it will need to be re-signatured prior to accessing its contents. The process of re-signaturing is described in detail in this knowledge article.

4. After the volume has been attached to the relevant ESXi hosts, those hosts need to scan for new storage devices and VMFS volumes. After this has been done, the New Datastore wizard can be used to perform the re-signature process. The user must provide a name and select a cloned volume from the list. An example can be seen in Figure 19.

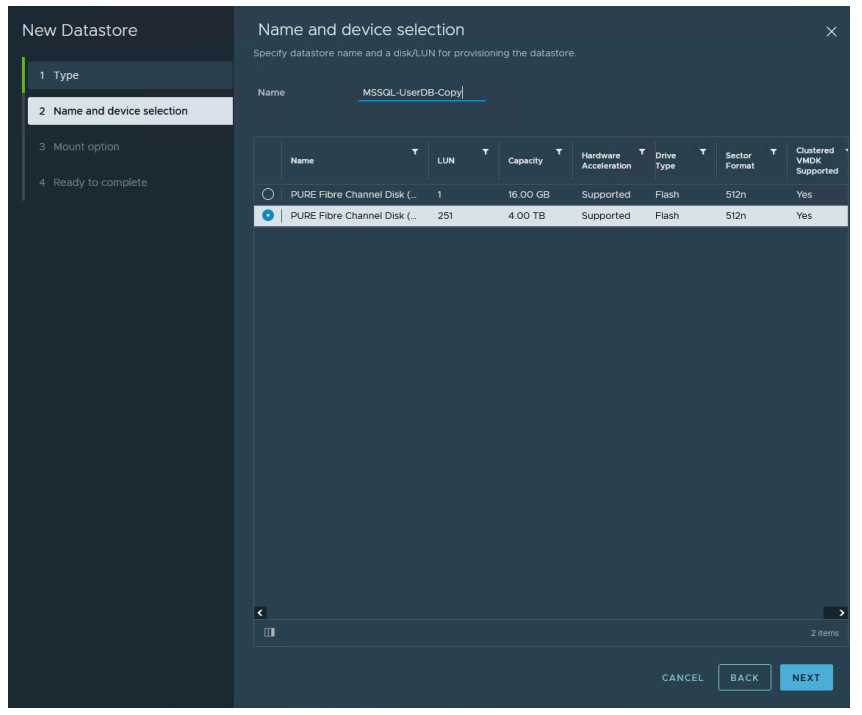

FIGURE 19 New Datastore dialog

5. Ensure that in the Mount option section Assign a new Signature is selected. Do not format the disk as this will erase all existing data. An example is shown in Figure 20.

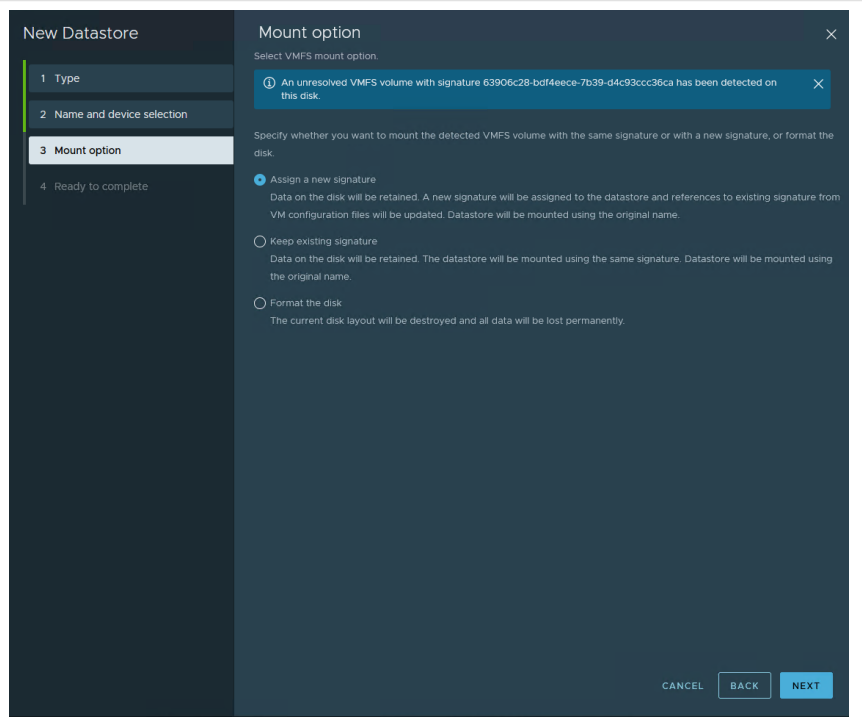

FIGURE 20 Mount option section of the New Database dialog

6. Review all of the options and select **Finish** as shown in Figure 21.

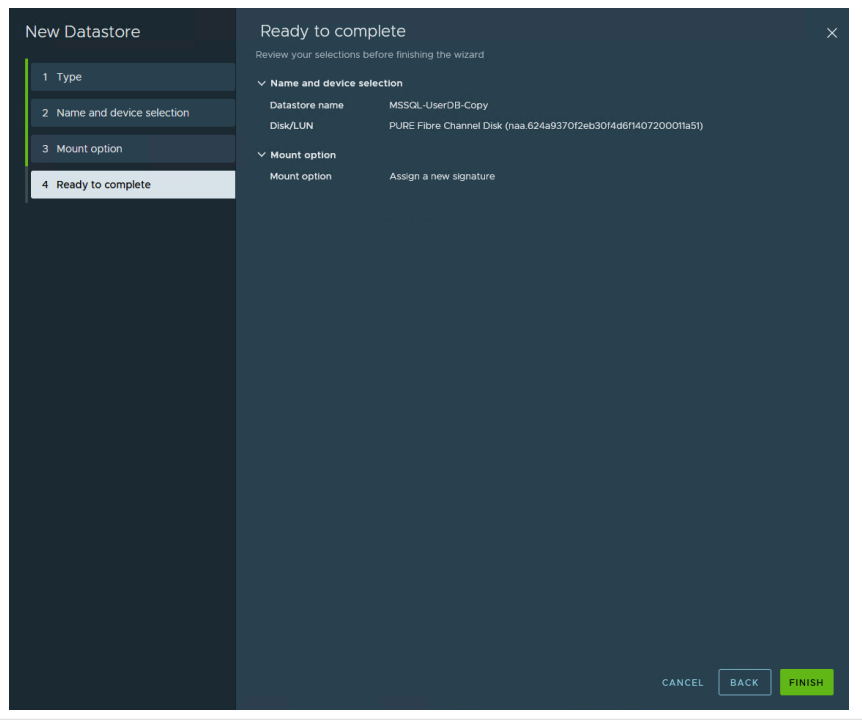

FIGURE 21 Ready to complete section of the New Datastore dialog

Once the datastore is mounted and has been res-ignatured the virtual disks contained within it can be identified as shown in Figure 22.

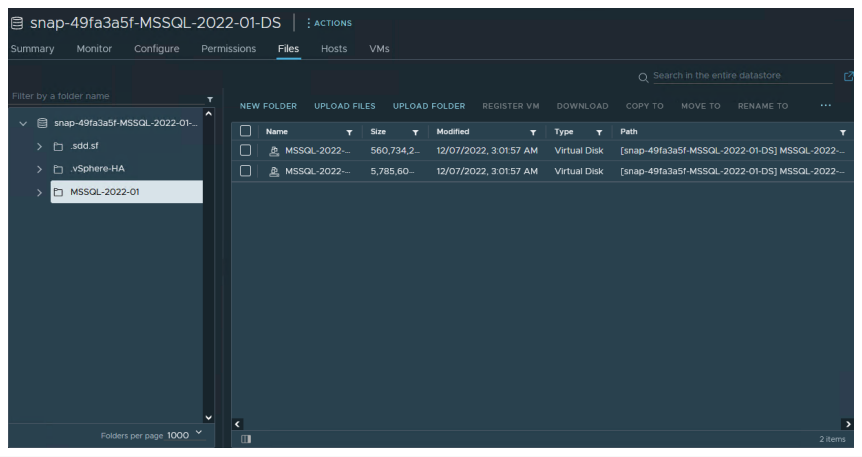

FIGURE 22 The Files view of the selected datastore

1. The virtual disks from the cloned datastore can be attached to an existing VM by editing its virtual hardware in vCenter, selecting **Add New Device,** and then selecting **Existing Hard Disk** from the drop-down menu. Figure 23 shows an example.

.

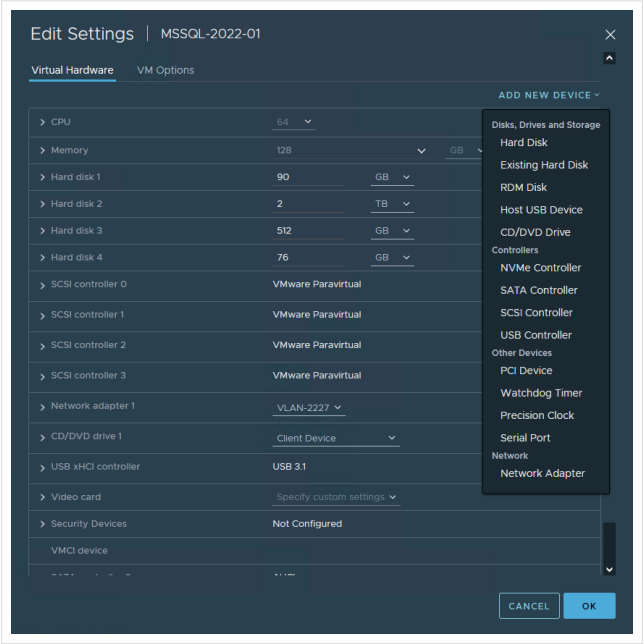

FIGURE 23 The Edit Settings dialog for a virtual machine

2. In the Select File dialog shown in Figure 24, navigate to the cloned datastore and then identify the disk to add, select it and then click **OK**.

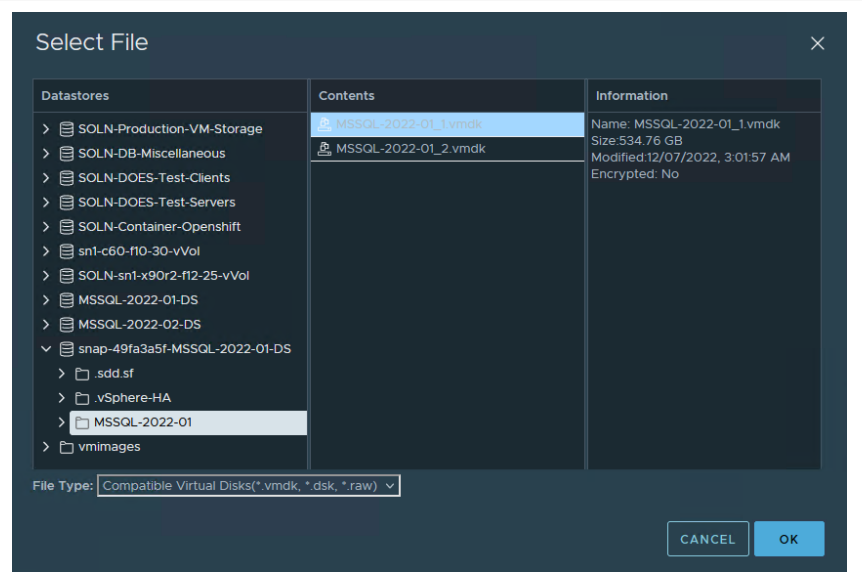

FIGURE 24 The Select File dialog with the virtual disk to add to the virtual machine selected

3. Once all of the disks have been added click **OK** in the Edit Settings dialog. An example is shown in Figure 25.

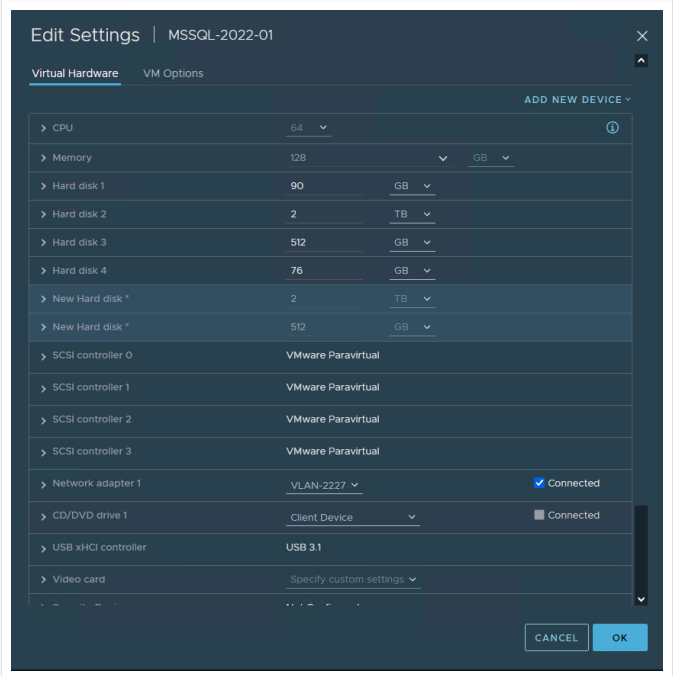

FIGURE 25 The Edit settings dialog after all of the virtual disks have been added

NOTE: *If adding the cloned virtual disks back to the same virtual machine, the user may be prompted with an error message "disks with same UUID should not be assigned to a VM". To work around this issue, follow the steps in this knowledge article to assign a new UUID to the cloned virtual disks.*

After the virtual disks have been added they will show up in Disk Management. An example is shown in Figure 26.

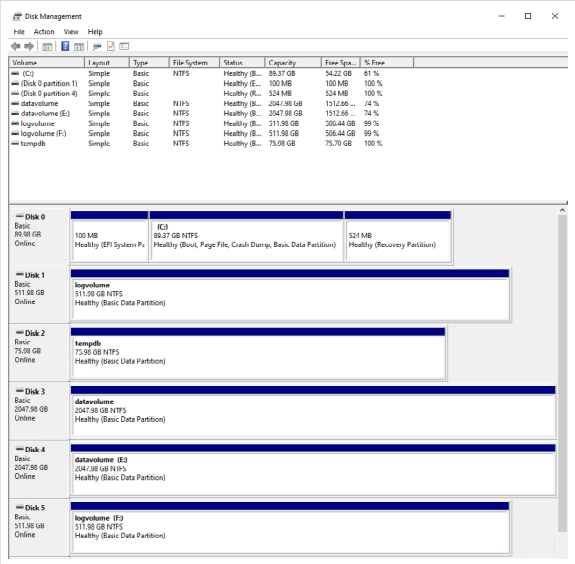

FIGURE 26 The disk management view with the cloned virtual disks being added

At this point the volumes can be provided with a drive letter or mounted, and then the databases attached following the steps set out in this Microsoft article.

Virtual Disks as Physical Raw Device Mapping Disks

Example: Creating volume snapshots of database files held in pRDM virtual disks. This example uses the same database layout as in "Creating a Volume Snapshot for a Single database with Virtual Disks on a Single VMFS Datastore". Two physical volumes are created on FlashArray and then two pRDM's are created from them and attached to a virtual machine. Figures 27 and 28 show what this configuration looks like.

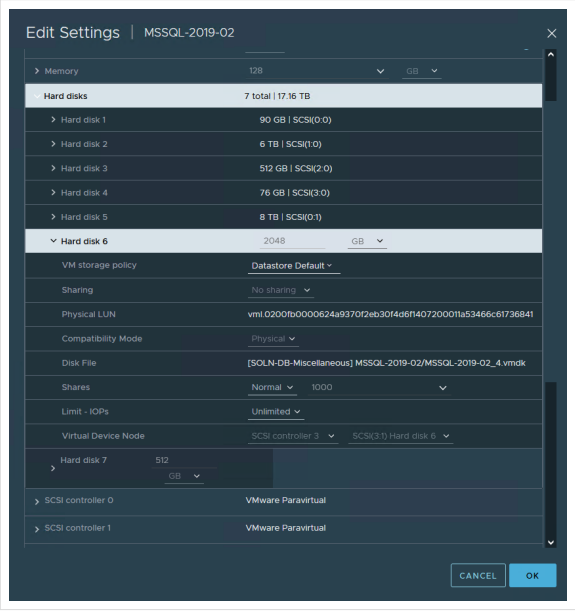

FIGURE 27 The pRDMs used for the datafiles of the UserDB database

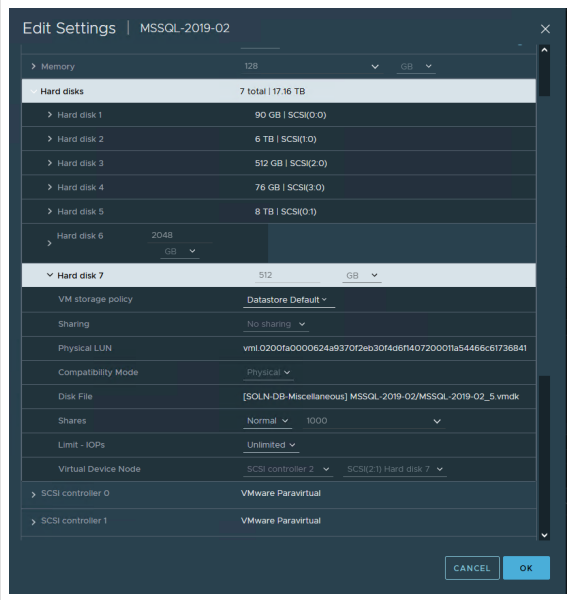

FIGURE 28 The pRDMs used for the datafiles of the UserDB database

As both of the pRDMs have been used for the database files, prior to creating a snapshot add both volumes to a protection group. The protection group ensures that all of the database files are consistent with one another at the point when the snapshot will be created.

1. Identify the section for protection group snapshots and then select the **+** to create a new protection group snapshot as shown in Figure 29.

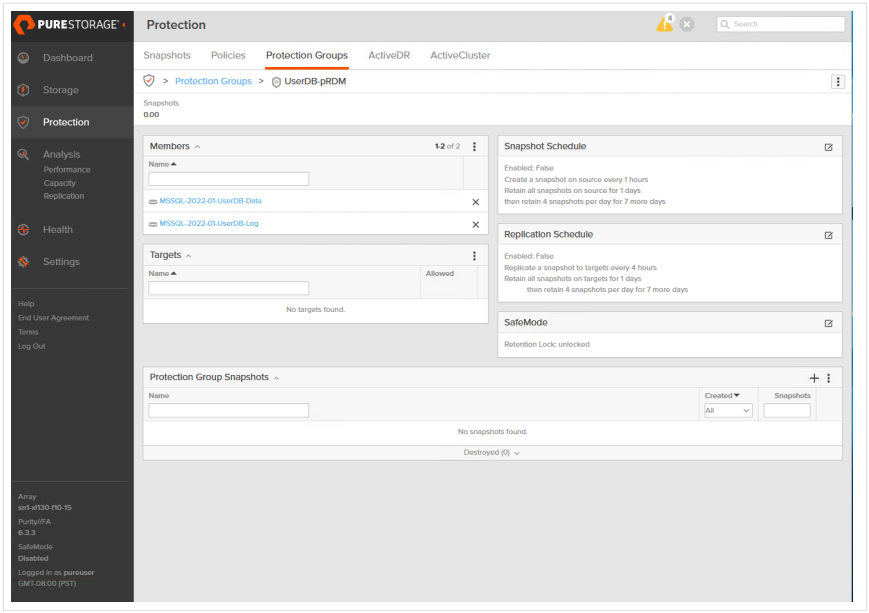

FIGURE 29 The two volumes in a protection group

NOTE: *The protection group can be asynchronously replicated to one or more targets which can be other FlashArrays, a Pure Cloud Block Store instance, or a single offload target which can include cloud storage or an NFS share. A snapshot and replication schedule can be set to customized snapshot retention.*

2. On the Create Snapshot dialog, provide a name for the protection group snapshot and then select **Create**.

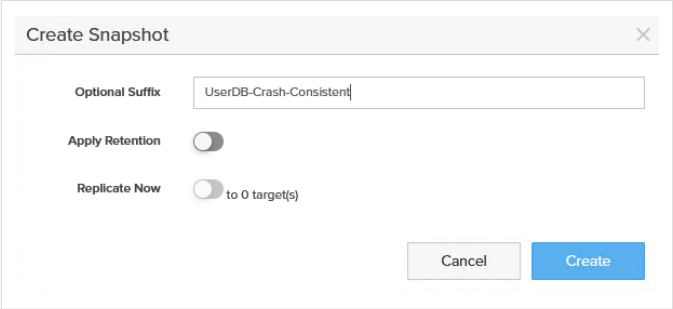

FIGURE 30 The Create Snapshot dialog

3. Once the protection group snapshot has been created it will show up in the Protection Group Snapshots list as shown in Figure 31.

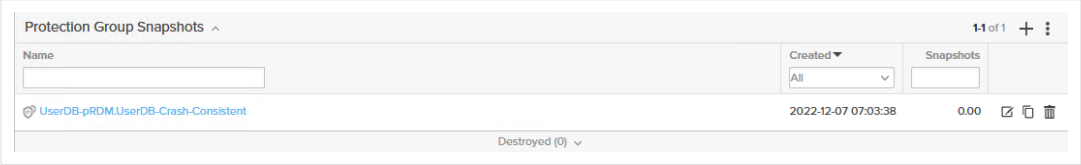

FIGURE 31 The list of snapshots for the protection group

Example: Recovering database files from a volume snapshot of a pRDM virtual disk. Using the protection group snapshot created in "Creating Volume Snapshots of Database file's Held in pRDM Virtual Disks" this example is going to recover the database from these volume snapshots.

1. Select the Protection Group Snapshot from the list of protection group snapshots to bring up the volume snapshots that were created. Using the dialog shown in Figure 32, the volumes can individually be copied or restored.

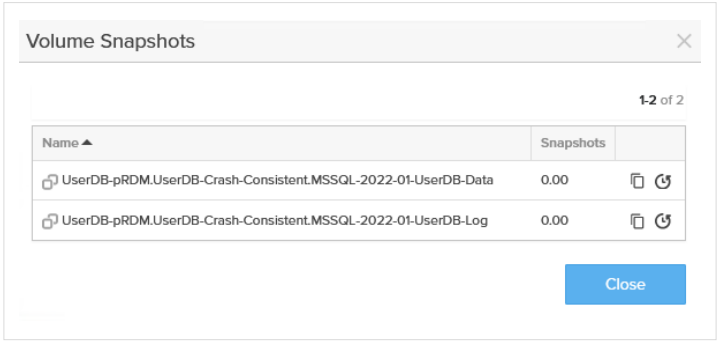

FIGURE 32 The volume snapshots included in the protection group snapshot

2. Selecting the volume snapshot to be copied will bring up a dialog to provide a name for the new volume. Once complete, select Copy. An example is shown in Figure 33.

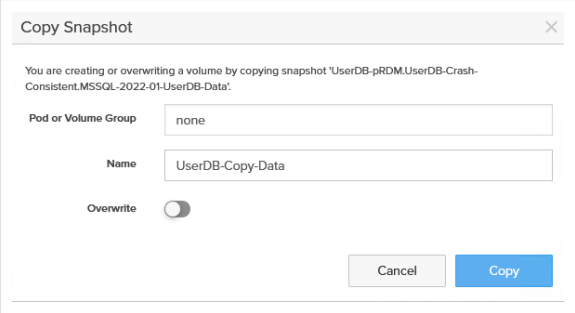

FIGURE 33 The Copy Snapshot dialog

3. Once the volumes have been copied, they can then be attached to the relevant ESXi hosts. Rescan storage on each host to recognize the new volumes.

4. Add the cloned volumes as pRDMs following the exact same process as adding new pRDMs. An example is shown in Figures 34, 35, and 36.

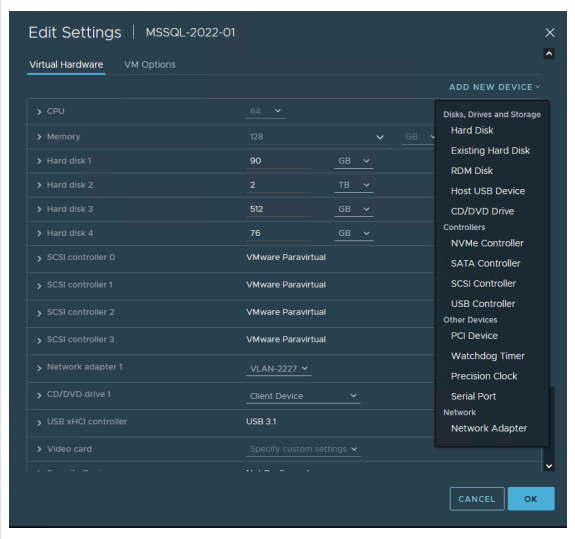

FIGURE 34 Adding cloned volumes to a VM

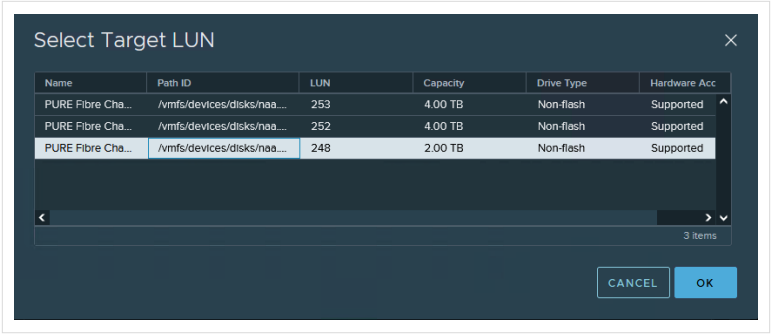

FIGURE 35 The cloned volume selected as a target LUN

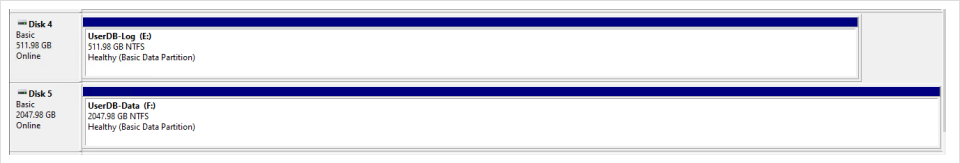

FIGURE 36 Disks in Disks Management after they have been formatted

At this point the volumes can be provided with a drive letter or mounted, and then the databases attached following the steps set out in this Microsoft article.

Virtual Disks in VMware Virtual Volumes

Example: Creating volume snapshots of database files held in VMware virtual volumes. This example uses the same database layout as in "Creating a Volume Snapshot for a Single Database with Virtual Disks on a Single VMFS Datastore". The virtual disks on which the UserDB datafiles reside are migrated with storage vMotion from the VMFS datastore to the vVol datastore. This has the effect of creating a volume group on FlashArray which contains all of the relevant volumes, two of which are the virtual disks containing the UserDB datafiles. Figure 37 shows what the volume group looks like.

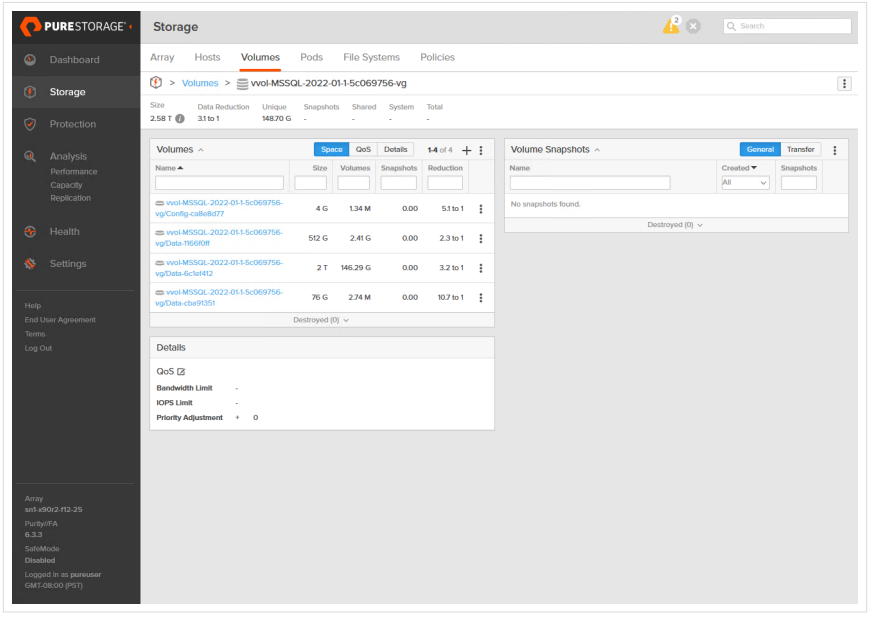

FIGURE 37 The volume group containing all of the volumes for the MSSQL-2022-01 virtual machine.

As the UserDB database spans multiple volumes to create a consistent volume snapshot, Pure uses a protection group.

1. Identify the section for protection group snapshots and then select the **+** to create a new protection group snapshot as shown in Figure 38.

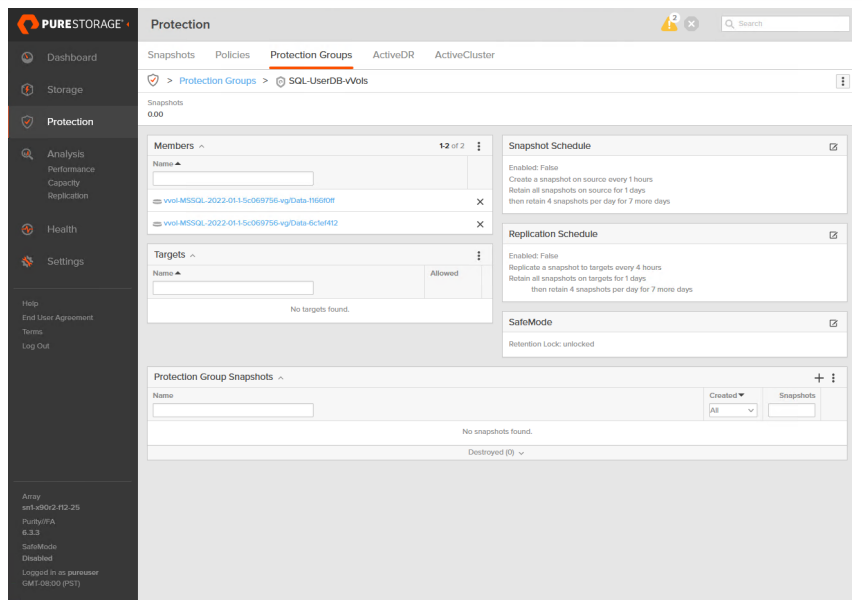

FIGURE 38 A protection group containing the volumes for the UserDB database

2. Provide a name for the protection group snapshot and then select **Create** as shown in Figure 39.

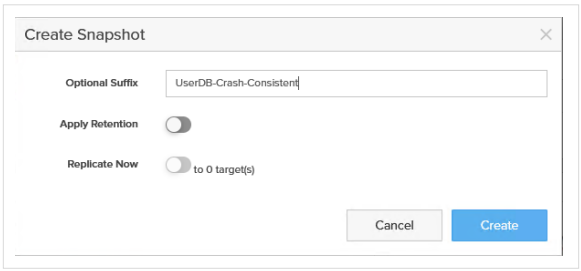

FIGURE 39 The Create Snapshot dialog for the protection group

3. Once the Protection group snapshot has been created it will show up in the Protection Group Snapshots list as shown in Figure 40.

FIGURE 40 The Protection Group Snapshots List

Example: Recovering a database from a volume snapshot created from VMware virtual volumes. Selecting the protection group snapshot from "Creating Volume Snapshots of Database file's held in VMware Virtual Volumes" the volume snapshots created as a part of it can be listed. An example is shown in Figure 41.

FIGURE 41 The volume snapshots included in the protection group snapshot

To make the contents of a virtual volume snapshot available to a virtual machine the following steps need to be followed (This is a copy process):

1. Create new virtual disk(s) for the virtual machine the snapshot will be restored to. Ensure this virtual disk is located in the vVol datastore. Match the size of the original volumes from which the snapshot was created.
2. Copy the volume snapshots and overwrite the new volumes created in step 1; ensure that the device is not online or mounted in the guest operating system.

The volumes can now be set to online in the guest operating system and provided with a drive letter or mount point.

At this point the volumes can be provided with a drive letter or mounted, and then the databases attached following the steps set out in this Microsoft article.

FlashArray Data Service: ActiveCluster High Availability

ActiveCluster is a storage volume based synchronous replication, high availability data service included in every FlashArray. This data service can be used to ensure that the volumes on which virtualized SQL Server data and log files reside remain available in the event of site or array failure.

ActiveCluster allows for organizations to enable zero (0) RPO and zero (0) RTO between two (2) FlashArrays with true Active/Active bi-directional synchronous replication with transparent failover.

ActiveCluster implementations are made up of the following core components:

- **Pure1® Cloud Mediator**: Used to determine which array will continue data services should an outage occur in the environment. An on-premises mediator virtual machine is also available for download where internet access is not available.
- **Active/Active clustered array pairs:** Utilize synchronous replication to maintain a copy of data on each array and present those as one consistent copy to hosts that are attached to either, or both, arrays.
- **Stretched storage containers (PODS):** Management containers that collect storage objects such as volumes into groups that are stretched between two arrays. Stretched storage containers also provide consistent IO continuation behavior for the storage objects within them.

This architecture can be seen in Figure 42.

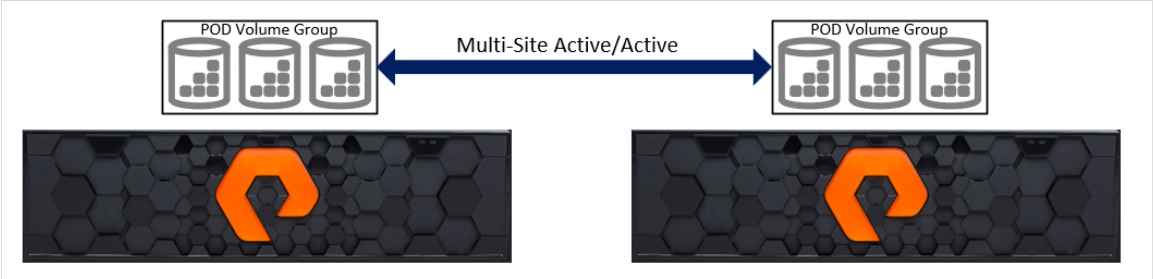

FIGURE 42 Synchronous Active/Active replication with two (2) FlashArrays

More information on using ActiveCluster in vSphere environments can be found in:

- ActiveCluster VMware vMSC(51656) article
- ActiveCluster with VMware User Guide on support.purestorage.com
- Increase SQL Server Resilience with Hybrid Cloud Pure Validated Design

Using ActiveCluster with Virtualized SQL Server Databases

ActiveCluster ensures the availability of the underlying volumes on which datastores are created and the virtual disks containing Microsoft SQL Server data/log files. The ActiveCluster functionality leveraged with vSphere Metro Stretch Clustering (vMSC) provides the ultimate in business continuity with synchronous replication and zero recovery point objective (RPO) and recovery time objectives (RTO) that SQL Server workloads require. With FlashArray there are no additional licenses to purchase for ActiveCluster and it integrates with your VMware vSphere infrastructure with partner-verified support between Pure Storage and VMware. As vMSC is transparent to virtual workloads, there are minimal changes to the management and operation tasks.

Due to the performance overhead associated with synchronous replication it is recommended that the following advisories be followed:

- Ensure that a single database has a dedicated filesystem (or systems) for its data and transaction log files. Do not share filesystems and virtual disks with the data or transaction log files from multiple databases.
- (VMFS Datastores) Ensure that the virtual disks for individual databases are contained within isolated datastores. The virtual disks used for a single datastore should contain the data and transaction log files for a single database.
  - It is possible to share datastores where multiple databases need to have the same availability level. Having the virtual disks in separate locations allows for granular control at the virtual disk level.
- For standalone Microsoft SQL Server instances), so not include the virtual disks on which tempdb's files reside in a non-uniform ActiveCluster configuration.
- For always-on failover cluster instance), include all virtual disks and associated datastores in the POD to ensure that the instance can failover or the virtual machine be migrated live to another ESXi host at another site.

Virtual Disks in Virtual Machine File System and Physical Raw Device Mapping Disks

For the configuration of ActiveCluster with pRDMs or VMFS datastores see the knowledge article Implementing vSphere Metro Storage Cluster With ActiveCluster: Configuring ActiveCluster.

Virtual Disks in VMware Virtual Volumes

Currently, Purity//FA does not support vVols objects in Pods on FlashArray devices. Additionally, VMware does not support Stretched Storage with vVols. This means that due to limitations both with Purity//FA and vSphere, vVols are not supported with ActiveCluster.

FlashArray Data Service: ActiveDR Continuous Replication

ActiveDR continuous replication is a data service included in every FlashArray. This data service can be used for virtualized SQL Server instances for a range of use cases such as disaster recovery and data mobility.

ActiveDR

ActiveDR provides continuously active replication capabilities for block storage volumes from one FlashArray to another with no impact on front end application performance.

Synchronous replication technologies such as ActiveCluster provide higher availability at the cost of requiring network latency between the two targets to be as low as possible. ActiveDR does not require the target system to acknowledge that data has been received before allowing the source to continue, thus eliminating the need for a low latency network, and allowing for continuous replication over larger distances.

The management of ActiveDR is vastly simplified making it suitable for implementation in any business scenario requiring a near zero (0) recovery point objective. Some management aspects of ActiveDR includes:

- **POD replication**: Pods are management containers for volumes using ActiveCluster or ActiveDR. Pods provide a simple management construct to organize data volumes and associated settings into groups. Once pods are linked together on separate systems via a replica link, data in that pod automatically starts replicating.
- **Continuous change tracking**: Automatically manage changes without the need to provision or monitor journal devices.
- **Single-command failover**: ActiveDR makes it simple to implement, test and manage disaster recovery. This true disaster recovery testing ensures that any runbook or orchestration steps for the entire environment stay the same during a test or in an actual failover event to minimize risk.
- **Multi-direction replication**: Configure multiple pods in different directions between two FlashArray systems.

This architecture can be seen in Figure 43,

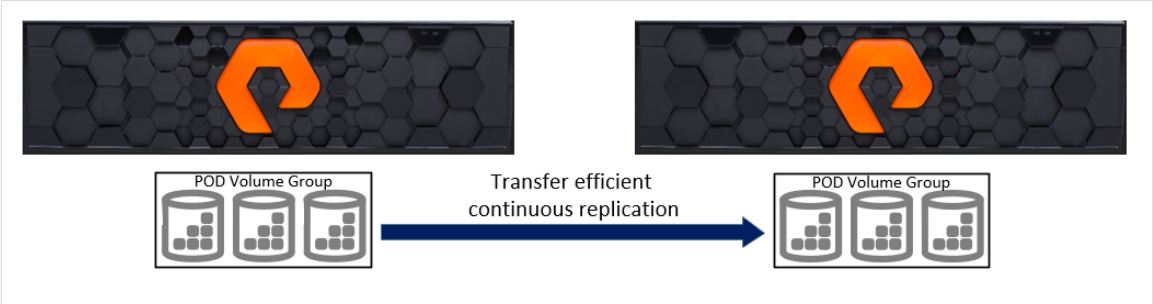

FIGURE 43 Continuous replication of volumes from a source to a target

Using ActiveDR with VMware Site Recovery Manager

VMware Site Recovery Manager automates the orchestration of failover and failback operations in VMware environments. Combined with ActiveDR it provides continuous replication to provide easy and predictable near-zero RPO. VMware datastores are placed into an ActiveDR pod and then any modifications or data operations (such as snapshots) are then applied by ActiveDR automatically to a secondary site. Site Recovery Manager protects VMFS datastores and anything contained within them using policy-based management which recovers applications in disasters.

Combining ActiveDR and Asynchronous Replication for Multiple Recovery Point Objectives

ActiveDR and Asynchronous replication can be combined to provide the best of both scenarios where near zero- and five minute RPOs can be achieved simultaneously. This is achieved by adding a protection group with a snapshot-and-replicate schedule to an existing ActiveDR Pod.

Using ActiveDR with Databases That Are Part of Virtualized SQL Server Instances

ActiveDR allows for a near-zero RPO for the datastores or volumes on which Microsoft SQL Server data/log files may reside. To ensure recovery is possible on a per-database level the following advisories should be followed:

- Ensure that single databases each have a dedicated filesystem(s) for their data and transaction log files. Do not share filesystems and virtual disks with the data or transaction log files from multiple databases.
- (VMFS Datastores) Ensure that the virtual disks for individual databases are contained within isolated datastores. The virtual disks used for a single datastore should contain the data and transaction log files for a single database.
  - It is possible to share datastores where multiple databases need to be recovered together. Having the virtual disks in separate locations allows for granular control at the virtual disk level.
- Do not include the datastores and virtual disks on which tempdb's files reside.

Virtual Disks in Virtual Machine File System and Physical Raw Disk Mapping Disks

For more information about using ActiveDR to achieve a near-zero RPO for virtualized Microsoft SQL Server databases, see the knowledge article on Enabling ActiveDR for ESXi Storage.

Virtual Disks in VMware Virtual Volumes

Currently Purity//FA does not support vVols objects in Pods on FlashArray devices. Additionally, VMware does not support Stretched Storage with vVols. This means that due to limitations both with Purity//FA and vSphere, vVols are not supported with ActiveDR.

Pure Storage Solutions for Microsoft SQL Server

Pure Storage offers several storage-based solutions for Microsoft SQL Server. These solutions can be seen in the Pure Storage Resource Center by isolating the search criteria to Microsoft Solutions. Additional solutions, knowledge articles, and how-to guides for SQL Server can be found in the Microsoft Platform Guide.

Pure Storage SSMS Extension

The Pure Storage FlashArray Management Extension for Microsoft SQL Server Management Studio (FlashArray SSMS Extension) is a database backup and restore extension for Microsoft SQL Server Management Studio (SSMS). The extension supports the creation of application consistent snapshots using the Pure Storage Volume Shadow Copy Service (VSS) hardware provider. The extension can be used from the SSMS GUI or automated using the included Pure Storage Backup SDK Windows PowerShell module. Database administrators (DBAs) can manage backup and restore operations on local and remote databases from a centralized SSMS deployment. The FlashArray SSMS Extension supports physical connections (bare metal) to the FlashArray and both vVols and pRDMs on VMware virtual machines.

Pure Validated Design: Increase SQL Server Resilience with Hybrid Cloud

Enterprises require business continuity regardless of where the data exists. This solution provides business continuity across on-premises data centers and disaster recovery with the public cloud. This guide outlines how to ensure data is available wherever and whenever needed. This solution is based on VMware vSphere 7.0U1 but does not dictate a specific bare-metal hardware implementation.

The Pure Validated Design "Increase SQL Server Resilience with Hybrid Cloud," provides design consideration and deployment best practices for the validated design using Pure Storage FlashArray//X for high-performance SQL Server with business continuity across two data centers, FlashArray//C for capacity-optimized and economical database backup, and Pure Cloud Block Store for disaster recovery with Microsoft Azure cloud.

Intra-cloud Disaster Recovery for Microsoft SQL Server in Azure

SQL Server has several built-in mechanisms for recovery in the event of a data loss failure; however, even with the most advanced capabilities, core issues such as recovery time objectives (RTO) and recovery point objectives (RPO) are still sometimes difficult to achieve. With Pure Cloud Block Store's rich data services these core issues can be mitigated or eliminated entirely depending on what functionality is utilized.

The Intra-Cloud Disaster Recovery for Microsoft SQL Server in Azure white paper focuses on implementing disaster recovery capabilities with Microsoft SQL Server deployed on Microsoft Azure infrastructure-as-a-service systems when using Pure Cloud Block Store as persistent database storage. This document provides in depth guidance on using asynchronous replication or ActiveDR to achieve different recovery point objectives for cloud-based SQL Server databases in the event of a disaster.

Pure Storage Solutions for VMware vSphere

Pure Storage offers several storage-based solutions for VMware vSphere. These solutions can be seen in the Pure Storage Resource Center, isolating the search criteria to VMware Solutions. Additional solutions, knowledge articles, and how-to guides for SQL Server can be found in the VMware Platform Guide.

Pure Storage FlashArray Plugin for the vSphere Client

The Pure Storage FlashArray Plugin for the vSphere Client offers customers the ability to configure FlashArray host groups, configure iSCSI, create, and manage VMFS datastores and configure and use Virtual Volumes.

More information on the plugin can be found on the Pure Storage Support site or the VMware Marketplace.

Pure Storage FlashArray Storage Replicator Adapter for VMware Site Recovery Manager

VMware Site Recovery Manager integration with FlashArray ensures simple and automated recovery of VMs across sites with minimal or no downtime. In addition, the solution centralizes recovery plans for thousands of virtual machines, while Pure ActiveCluster and ActiveDR deliver non-disruptive recovery testing with automated workflows for failover, migration, and failback. Pure Storage's suite of modern data protection capabilities requires no third-site mediation, no extra infrastructure, extra licenses, or additional fees, taking no more than a few minutes to set up.

More information on the adapter can be found in the Pure Storage Support Site or the VMware Marketplace.

VM Analytics in Pure1

Pure1® is a cloud-based as-a-service data management platform. Common and complex data management tasks for Pure Storage products such as FlashArray, FlashBlade, and Portworx® can be accomplished with ease. With Pure1 outcomes can be delivered in seconds instead of hours or days. Costly downtime can be eliminated by leveraging predictive analytics and responding to dynamic changes from anywhere in the world.

Pure1 provides the following benefits for modern data infrastructure management:

- **Centralized setup and monitoring:** Setting up Pure1 is easy: Login to the Pure1 portal, and the software does the rest. As soon as a system is online, Pure1 Meta is hard at work gathering analytics. Live monitoring is available within minutes and accessible from anywhere in the world.
- **Full-stack analysis**: Access critical information about the health and functioning of an entire stack, including predictive fault analysis, and alerting. Auditing for ransomware protection is also included which assists with the investigation of vulnerabilities in an environment.
- **Reporting**: Pure1 has an intuitive, built-in reporting engine that can be used to generate shareable reports on commonly requested information such as capacity, performance, or even service subscription status.

All of the above can be seen in Figure 44.

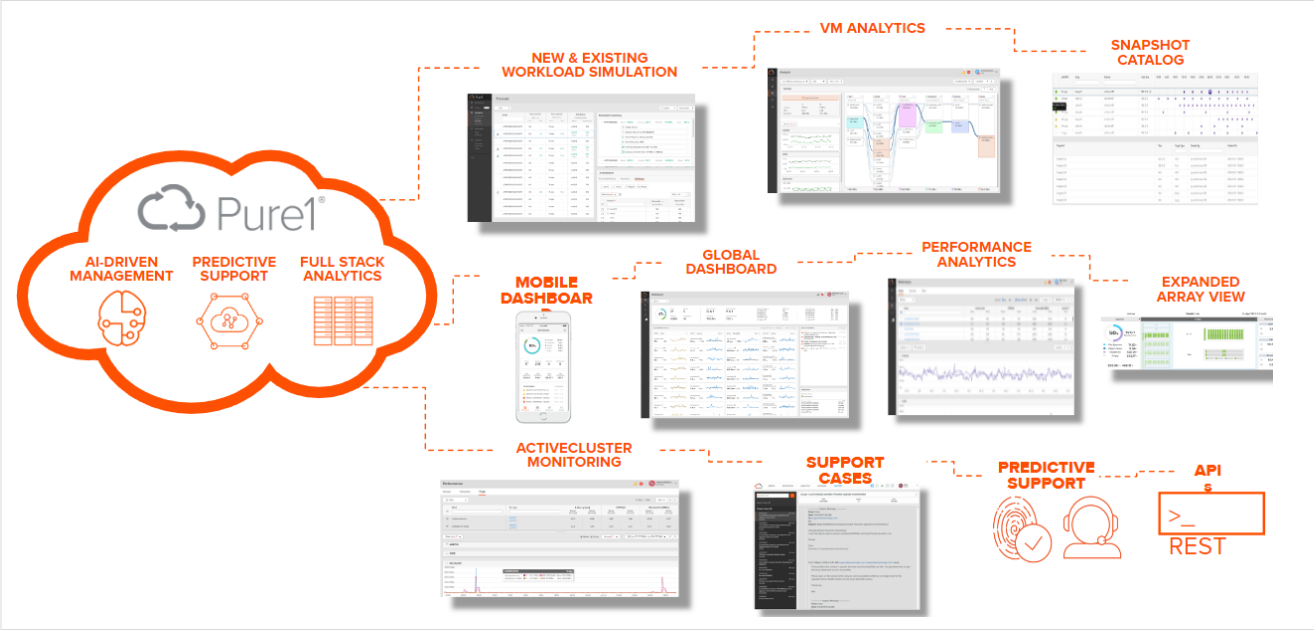

FIGURE 44 Pure1 Fleet Management

Full Stack Analytics (VM Analytics)

Pure1 extends the visibility of its deep analytics on storage infrastructure up the stack to provide performance metrics on volumes and virtual machines in a VMware environment. This visibility enables fast and efficient troubleshooting through the stack via insight into latency, bandwidth, and IOPS of the objects in the environment to pinpoint problems and solve issues quickly.

Support is provided for VMFS, vVols, NFS, and vSAN with comprehensive analytics and visibility into virtual environments.

Full Stack Infrastructure

FlashStack® is a software-defined, hybrid-cloud infrastructure solution and holistic alternative to status quo converged and hybrid-converged infrastructure offerings.

Developed jointly by Pure Storage and Cisco, FlashStack provides best-of-breed components that have been pre-tested and validated for the most popular and demanding workloads.

FlashStack is AI-managed, discreetly provisioned, and can be consumed as a service with no up-front costs. It scales without disruption as your business grows and provides a powerful, reliable platform for your cloud, business-critical applications, DevOps, and modern analytics—right out of the box.

Modern data infrastructure solutions have become smarter, simpler, and more efficient than ever before. They're virtual machine-aware and hybrid cloud-ready, while retaining the predictability and efficiency advantages of dedicated compute and storage tiers. None of this new breed of hybrid cloud solutions are as simple to deploy and operate, or as easily scalable as FlashStack. Because of this, FlashStack has become the leader in modern infrastructure, outgrowing the server market by almost 3:1 since 2020.

FlashStack is a holistic alternative to disk-based CI and HCI offerings. Software-defined from top to bottom, and intelligently managed and provisioned via Intersight, its cloud-native management layer, FlashStack is delivered in partnership with Cisco a nd VMware from best of breed components that have been rigorously tested and well documented with Cisco Validated Designs.

To provide a greener, more compliant data center, FlashStack has been re-designed from the ground-up to be 80% more efficient with reduced energy consumption, improved cooling, and a far denser architecture for a dramatically reduced data center footprint.

FlashStack deploys with the simplicity of an appliance, but scales—without disruption—as your business grows, providing a powerful, secure, and reliable platform for your cloud, business-critical applications, DevOps, and modern analytics—right out of the box.

Validated Designs for VMware

The following validated designs can be used with FlashStack to implement Microsoft SQL Server on VMware:

- Server Virtualization (VSI) Design and Deployment Guides
- FlashStack for Microsoft SQL Server 2019 with VMware vSphere using Fibre Channel Protocol

Conclusion

Pure Storage provides a high-performance, reliable environment to run SQL Server workloads on VMware vSphere. This solution provides a proven, integrated platform for scaling SQL Server and providing robust cloning, replication, and disaster recovery capabilities in any size environment.

Organizations can use the suggestions and best practices in this reference architecture to plan their SQL Server deployment while optimizing the various components within a combined VMware vSphere and Pure Storage FlashArray environment.
Additional Resources

- Find more guidance at Pure Storage Support.
- Discover FlashArray//C, FlashArray//X, and FlashArray//XL.
- Virtualize your most critical applications with confidence with Pure Storage and VMware.

PS2372-01 01/2023