Welcome!



Welcome to Software-Defined Storage Concepts provided by NDG and VMware IT Academy! It is our pleasure to introduce you to the basics of software-defined storage (also known as “SDS”). Before starting, we would like to take the opportunity to explain the scope and objective of this micro-course.

Simply put, software-defined storage is an idea whose time has come. Software-defined storage manages data storage resources and services independent of the underlying physical devices. It replaces static, inefficient hardware with dynamic, agile, automated storage solutions.

You may have heard about software-defined storage and wanted to know more but felt that the learning material available to you was too advanced or too technical. No doubt about it, SDS is a complex topic, but we have designed this course to be an introduction for beginner learners with limited or no previous knowledge of software-defined storage. Software-Defined Storage Concepts will lead you topic-by-topic to build your understanding of…

* what is software-defined storage
* the main components involved in software-defined storage
* VMware’s role in the software-defined storage industry
* and how knowledge of SDS can lead to great career opportunities

The best part about learning software-defined storage is that it is not that difficult to get started. NDG has developed this course in partnership with VMware, a leader in software-defined storage innovation and technology, to introduce you to real-life SDS software that is currently being used by companies and individuals worldwide.

## If you wish to continue learning about software-defined storage beyond an introductory level, we recommend you consider the VCA (VMware Certified Associate) Digital Business Transformation (VCA-DBT) video-based course or the IT Solutions for Digital Businesses course offered by the VMware IT Academy. You can also find additional courses on storage in the VMware Learning Zone.

## Course Objectives



By the end of this course, you should be able to meet the following objectives:

* learn key concepts related to storage in the data center
* understand software-defined storage concepts
* describe the architecture, technical characteristics, and benefits of vSAN
* compare the functionalities and performance of vSAN with other traditional storage options
* be exposed to basic information about the Hyper-Converged Infrastructure

Course Overview



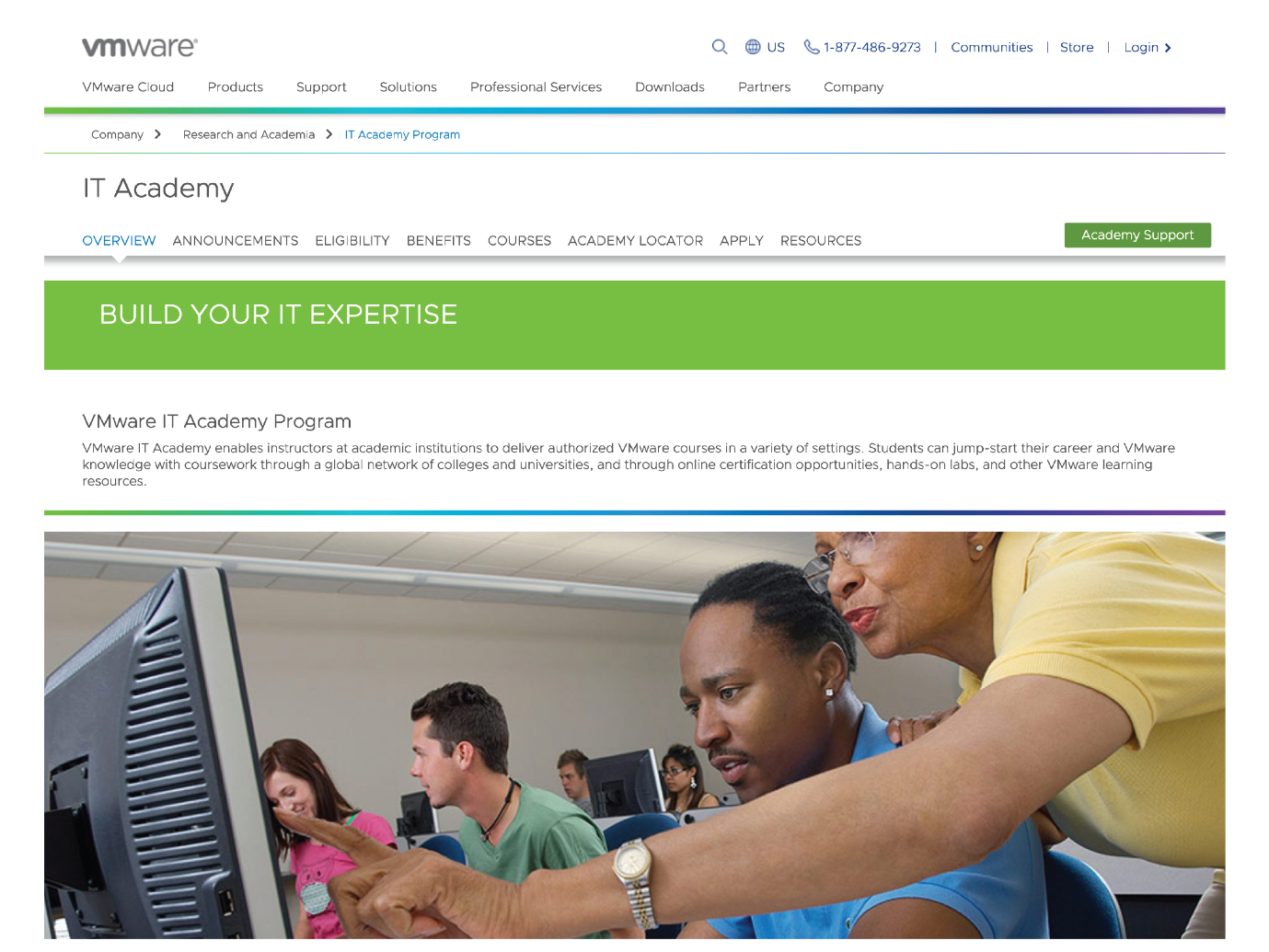
The Software-Defined Storage Concepts micro-course consists of seven modules:

* Module 1: Welcome!
* Module 2: Storage Concepts in the Data Center
* Module 3: Introduction to Software-Defined Storage
* Module 4: Software-Defined Storage Model
* Module 5: Hyper-Converged Infrastructure
* Module 6: Hyper-Converged Storage vSAN
* Module 7: Where To Go From Here?

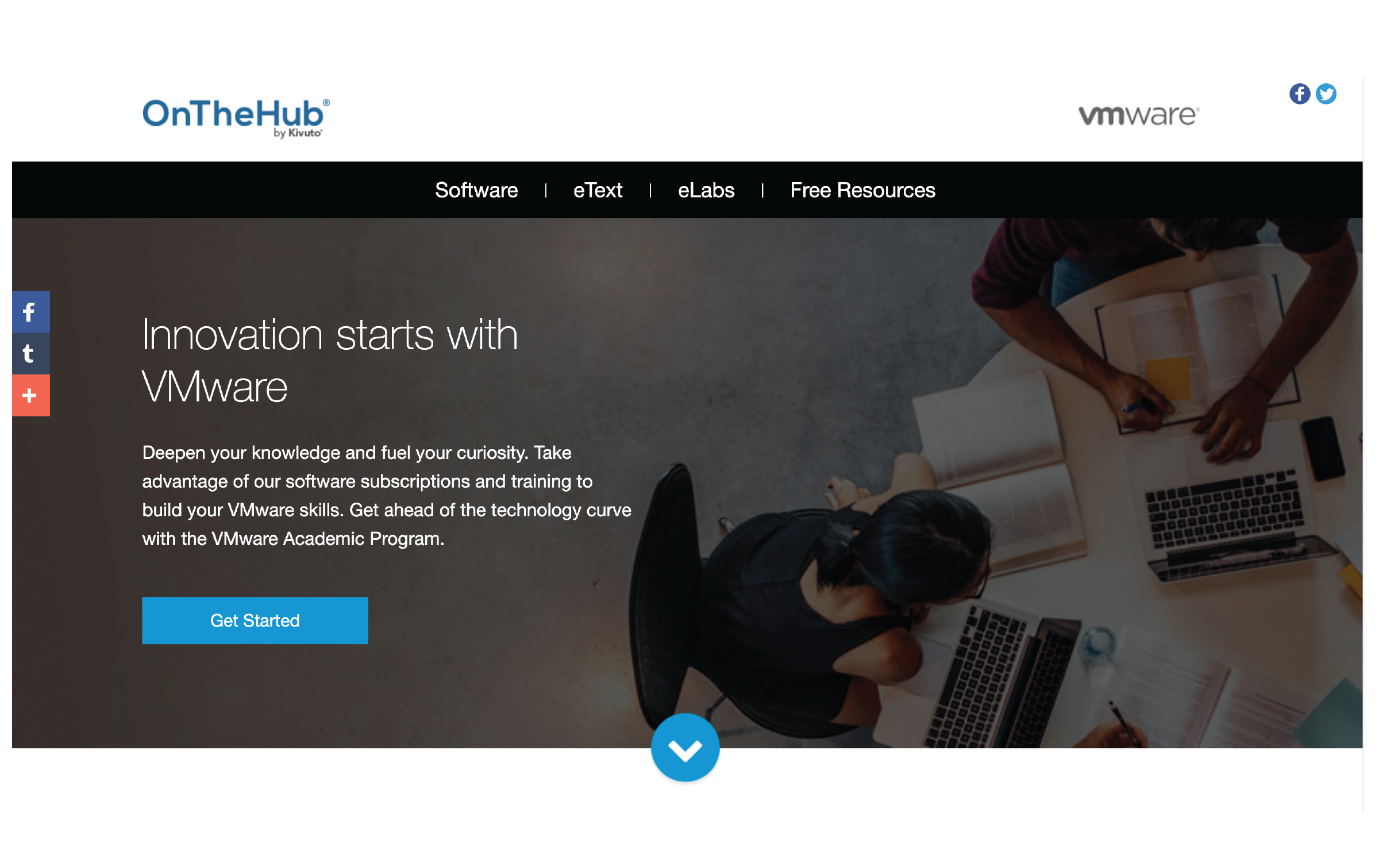
## Additional resources available via VMware

NDG partnered with the VMware IT Academy to develop this course. Academic institutions and instructors interested in helping learners develop additional knowledge may want to consider the benefits of participating in the VMware IT Academy and the VMware Software Licensing Program.

The VMware IT Academy is a global network of approved educational institutions and non-profits providing students access to VMware content and technologies to develop industry knowledge and skills. To learn more about the VMware IT Academy, visit [www.vmware.com/go/academy.](https://www.vmware.com/company/research/it-academy.html)



The VMware Software Licensing Program provides many VMware software products to help academic institutions teach and conduct research. To learn more about the VMware Software Licensing Program, visit [http://www.onthehub.com/vmware/.](https://onthehub.com/vmware/)



VMware IT Academy has arranged for learners that complete this course to receive a digital badge to share on social media that documents their knowledge. After completing this course, you can submit a request for the digital badge.

## Storage Concepts in the Data Center

We all understand the importance of data storage in our personal lives. Imagine having a phone that could only store five songs at a time. Every time you wanted to download a new song, you’d have to create space by deleting one of your five songs! How inconvenient would that be, and how much time would be wasted?

Now imagine the impact that inadequate storage would have on a data center. Bear in mind that 2.5 billion bytes of data are currently produced around the world every single day – the “structured data” of spreadsheets and databases, and the “unstructured data” that is our audio files, video files, images, messages, text files, and presentations. It’s been estimated that by 2025 there’ll be over 40 billion devices connected to the internet, generating nearly 80 zettabytes of data. A zettabyte is 1,000,000,000,000,000,000,000 bytes, or one trillion gigabytes, in case you were wondering! We’ll have to wait and see if that estimate comes to pass, but with the amount of data stored around the world doubling every two years, the need for storage that can handle this volume of data can certainly only grow.



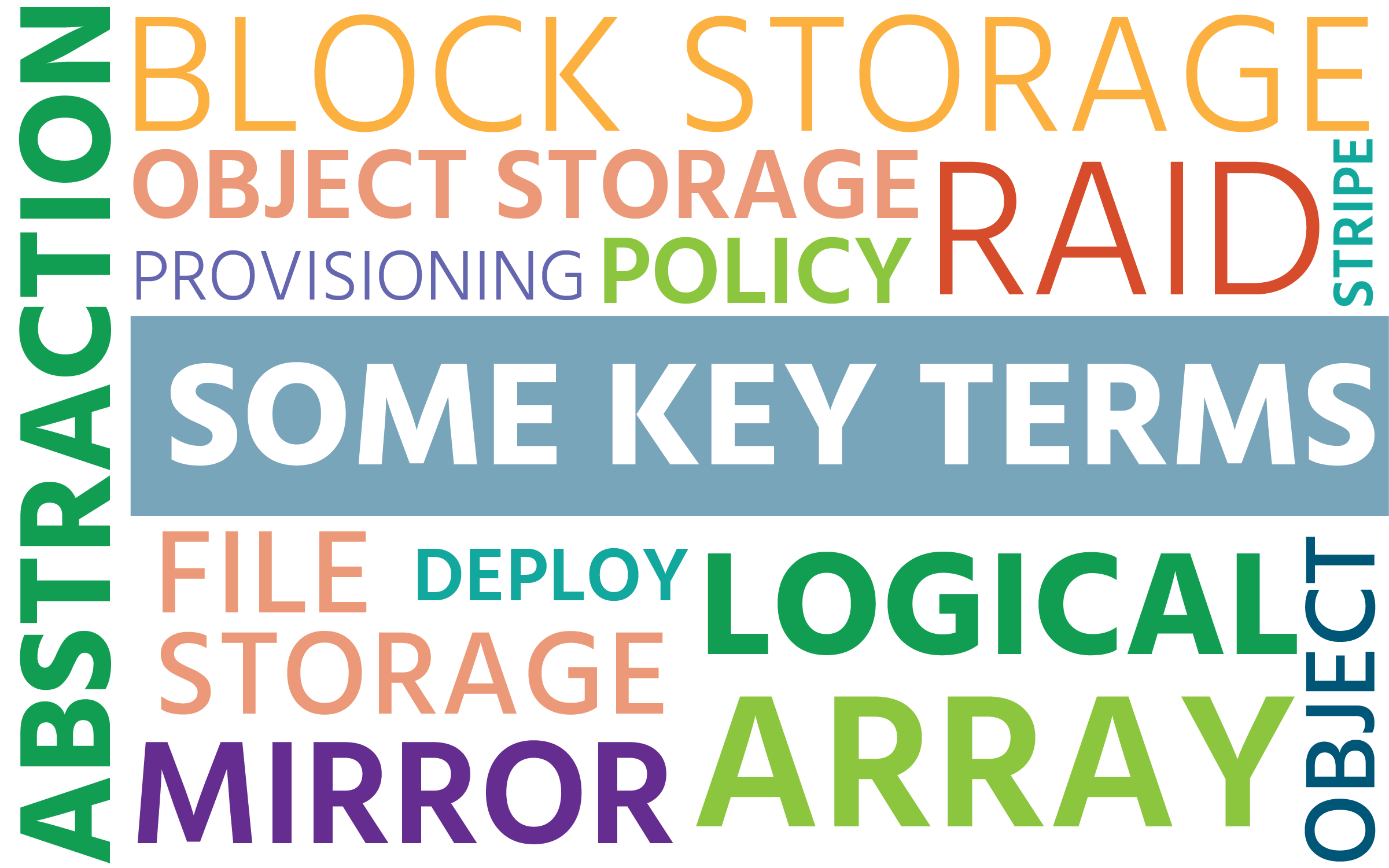
Demand for storage already outstrips supply. When we consider the fact that data centers exist solely to store and process data, it’s not difficult to see why storage might be viewed as the most important “layer” of a data center’s infrastructure. Data center storage shortages mean application failures and service disruptions. (Most modern data centers from a high-level approach have 3 “layers”: a management layer, a virtual layer, and a physical layer. The physical layer is made up of 3 layers: a network layer, a compute layer (i.e. computing resources) and a storage layer.)

Many data centers contain devices built for storage only. This is because server memory – random access memory (RAM) and hard drives – can only accommodate a certain amount of data depending upon the model. RAM is a kind of “short-term” memory that’s used to hold data that needs to be accessed, read, and written speedily by the central processing unit (CPU). Hard drive capacity is still in the double-digit terabytes range – far lower than the three-figure terabytes range of some storage-only devices (which enters the petabyte range when the devices are clustered together). (One petabyte is 1,024 terabytes or 1,048,576 gigabytes in binary.)

These storage-only devices are typically connected to the server or to the network the servers are on. Apart from their larger capacity, another benefit of storage-only hardware is their ability to deliver data to users quickly and efficiently.

A data center’s storage must deliver high “availability” of storage. “Availability” refers to the expectation that a storage device is running rather than experiencing downtime. If you lost a flash drive containing all your favorite photos and videos, it would be incredibly inconvenient. Imagine, then, if a data center storage device went offline: the businesses and enterprises that relied on that storage would suffer – financially and reputationally. Data center storage, therefore, clearly needs to be far more robust and fail-safe than personal storage.

The method used to improve the availability of storage is called “redundancy”. Redundancy is the duplication of critical components of a system to provide a back-up solution. It can be implemented in such a way that the system will create a copy of data, saved in another location, that can be accessed in the event that the original location becomes corrupted or breaks down. As you can probably guess, redundant storage is critical for data centers.



You may not be familiar with some of the words that are used in the industry. Here’s an explanation of some of the key ones, with others explained throughout the course:

**abstraction**(noun) – in a complex system or piece of software, abstraction is focusing on the most relevant details and hiding what can be ignored

**array** (noun) – data storage made up of multiple storage devices and cache memory (that’s temporary memory for fast data access)

**block storage** or **block-level storage** (noun) – data is saved in huge fixed-sized volumes called “blocks”; each block is treated as an individual storage device, has a unique identifier, and has its own file system (file systems will be discussed in general in section 2.4)

**deploy** (verb) – to install, test, and run hardware or software in a live environment

**file storage** or **file-level storage** (noun) – data is saved in files and folders in a hierarchical system of directories and sub-directories; in order to be accessed, the storage drives need to be configured with the Network File System (NFS – discussed in section 2.5) if it’s a Unix or Linux system, or Server Message Block (SMB) if it’s a Microsoft Windows system

**logical** (adjective) – not physical

**mirror** (verb) – to make an exact copy of data from one storage device drive to another storage device in real-time, in order to prevent data loss in the event of a disk failure; also known as “RAID 1” - RAID is defined below

**object** (noun) – with vSAN (discussed in sections 2.7 and 6-6.5), an object is a virtual machine disk file (VMDK), a snapshot (a copy of a VMDK taken at a specific point in time), or the virtual machine home folder

**object storage** or **object-based storage** (noun) – data is bundled together with its metadata (more information about the data, e.g., date created/modified, size, author) and a unique identifier to form an “object”

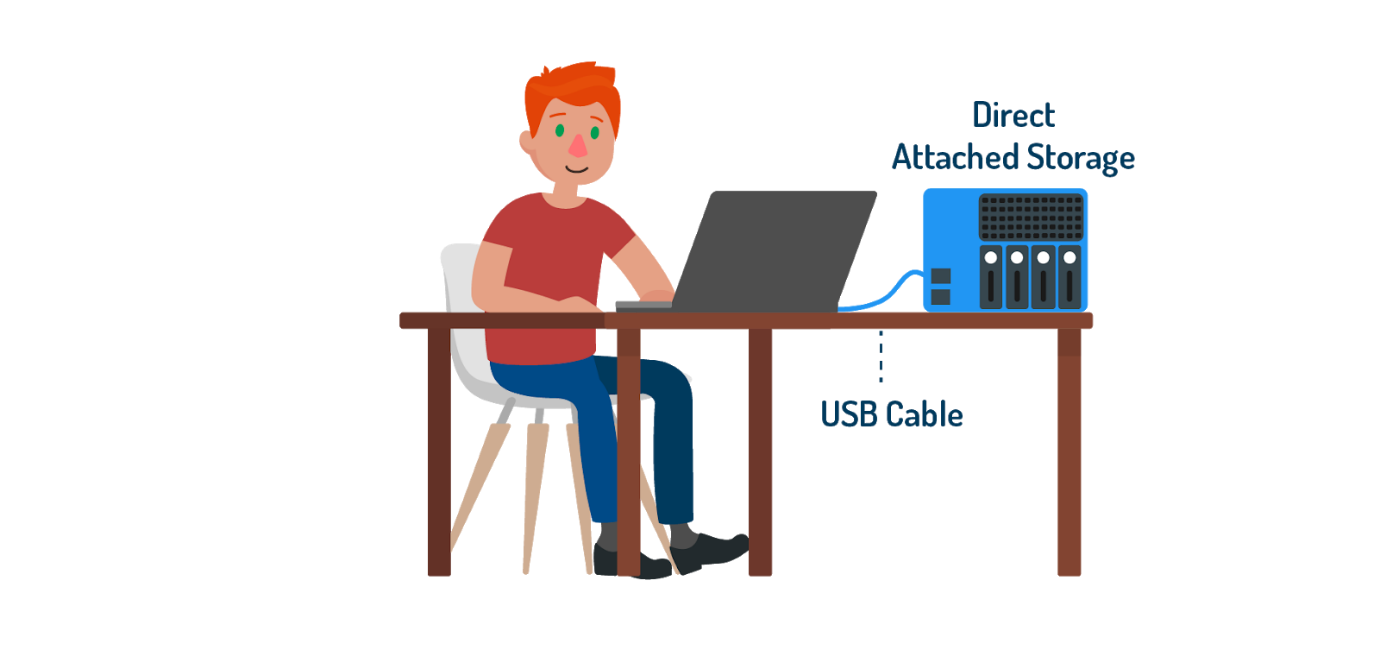
**policy** (noun) – a set of rules about the storage requirements of virtual machines and the applications that they run

**provisioning** (verb) – setting-up and making available IT resources, and managing them

**RAID** (noun) – a “**r**edundant **a**rray of **i**ndependent **d**isks” is storage that is made up of multiple separate hard drives; the same data is stored across different disks using a variety of methods, known as “RAID levels”; mirroring (see above) and striping (see below) are two of these methods

**stripe** (verb) – to divide a piece of data into equally-sized units which are then spread across multiple storage devices; no copies of the data are made; also known as “RAID 0”

Local Disk Storage



In the last section, we imagined having a phone that could only store five songs and needing to delete songs in order to be able to upload new ones. What if we attached that phone to an external storage device that could store a terabyte’s worth of data? That would be over 70,000 songs! Or perhaps we would give up on the phone as a media device altogether, and store all our songs on the hard drive of a PC or laptop. Both of these options (the external storage device attached to the phone and the hard drive of a PC/laptop) are examples of “local disk storage”, or “direct-attached storage” (DAS). The storage device is *local* to (i.e., *physically attached to*) a personal device or to a server in a 1:1 relationship. (Remember: a server is a computer or computer program that provides services to other computers. Servers are typically more powerful and more robust than ordinary computers and usually won’t have a monitor, mouse, or keyboard.) In this 1:1 relationship, data is not accessible to other devices or servers.

The following are types of local disk storage:

1. **Hard Disk Drives (HDDs)**



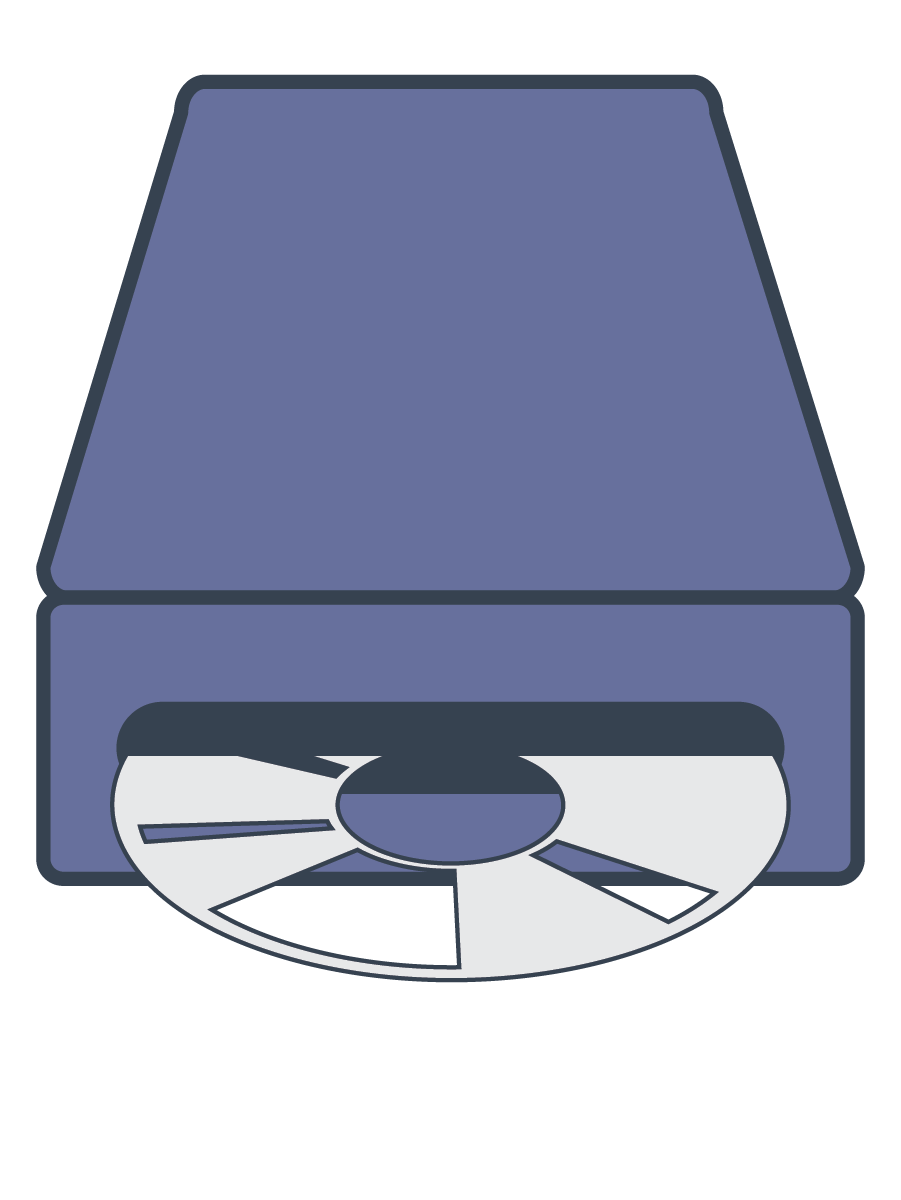
Usually referred to as “hard drives”, these are found in many personal computers and store data in binary form. The actual storage happens on a circular “platter” of magnetic material which is divided into billions of individual areas. These individual areas are organized into circular “tracks”, and the tracks are organized into “sectors”. A platter will spin at up to 10,000 revolutions per minute, and as it does so, two tiny magnets pass back and forth over its top surface and its bottom surface, either magnetizing an individual area to store a value of 1, or demagnetizing it to store a value of 0. This type of memory is referred to as “non-volatile” - it remains even when your storage device is turned off.

1. **Solid State Drives (SSDs)**



These also provide “non-volatile memory” but use transistors (devices that regulate the flow of electrical signals) rather than magnetized media to store data. They have no moving parts. Instead, transistors are stacked in a grid of rows and columns. Data is stored in electrical charges: a chain of transistors that conducts current has a value of 1; a chain that doesn't conduct current has a value of 0. With no mechanical parts, SSDs are faster, more durable, and more efficient than HDDs. 2TB SSDs are widely-sold but are significantly more expensive than their HDD equivalents.

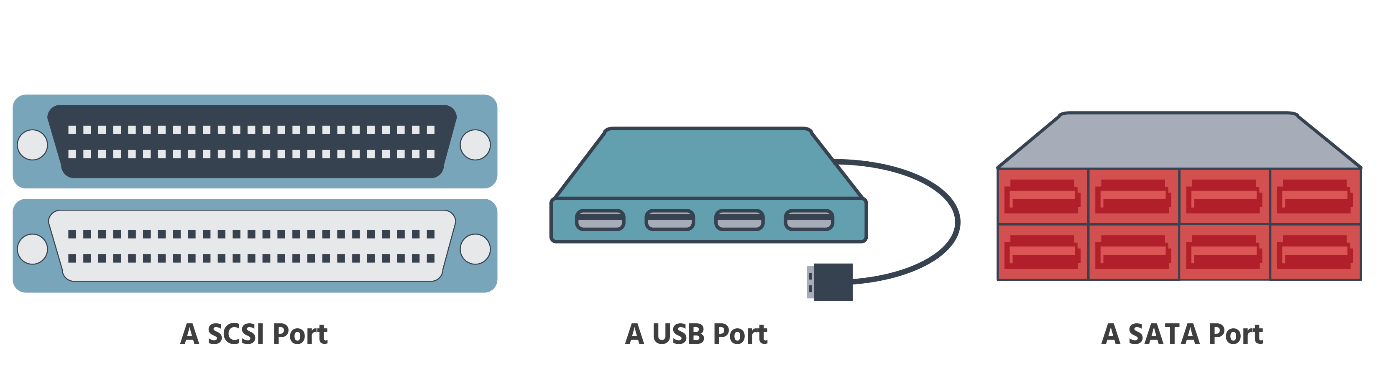
1. **Optical Disk Drives (ODDs)**



These use a laser to read data from or write data to the underside of a spinning disk. Data is encoded as binary “pits” (0 means off) and “lands” (1 means on) on the reflective surface of the disk. A lens focuses the laser beam and photodiodes (which convert light into electric current) sense the light reflected from the disk and regulate the laser. ODDs can store much less data than HDDs and SSDs: a CD can hold about 700MB, most DVDs can hold between 4.7GB (single-layer) and 8.5GB (dual-layer), and Blu-rays between 25GB (single-layer) and 50GB (dual-layer). However, ODDs are convenient because they are inexpensive and highly portable.

Local disk storage is a good choice for individuals and small businesses that require a set amount of data storage and has the added benefit of being very simple to use. Additional drives can be added when they’re needed.

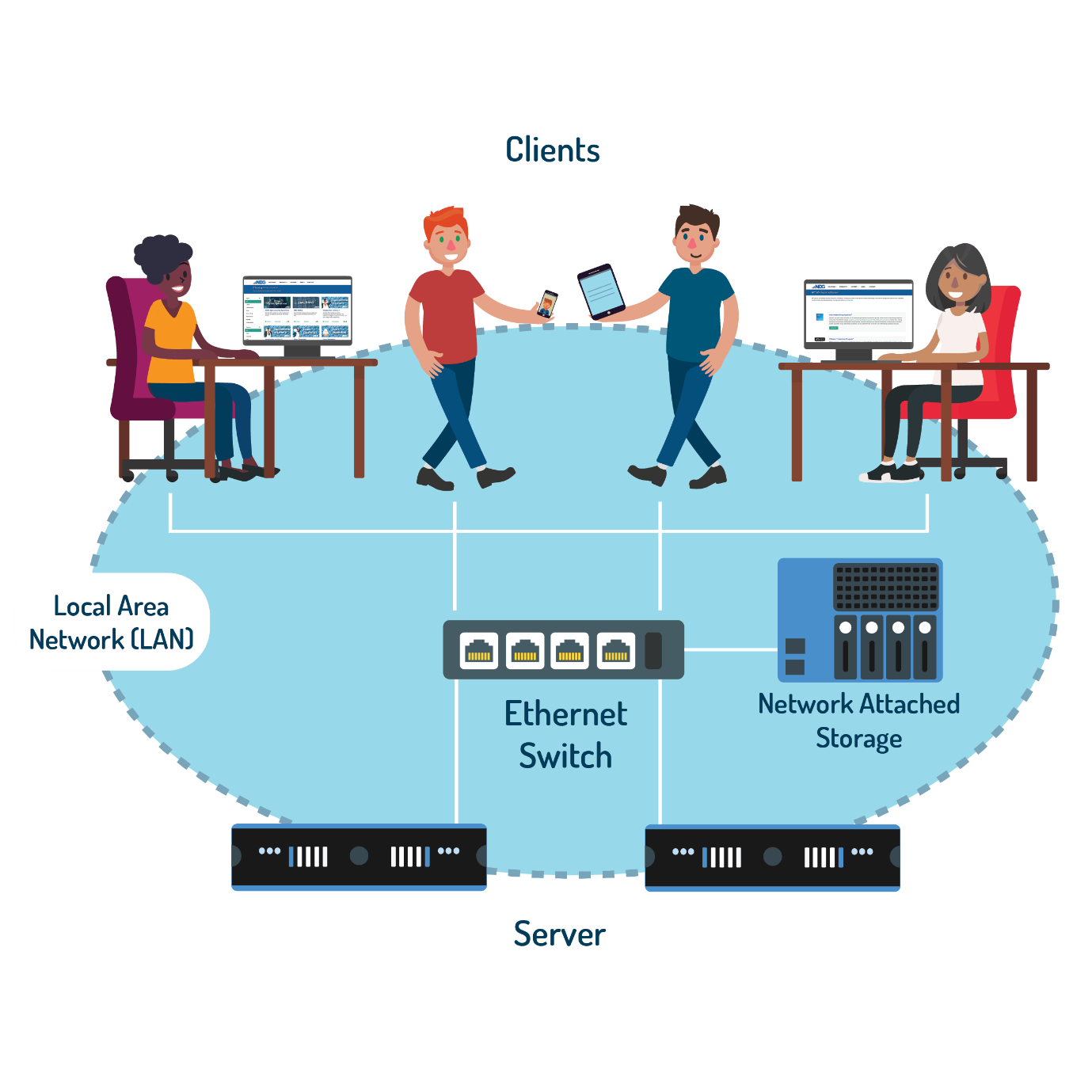
Local disk storage communicates with whatever it’s attached to using a language known as a “protocol”. There are a number of different protocols used in this type of storage, each with a different port (i.e., connection-point); for example:



The protocols most commonly-used with local disk storage are SCSI (Small Computer System Interface), SAS (Serial Attached SCSI), SATA (Serial Advanced Technology Attachment), USB (Universal Serial Bus), and FC (Fibre Channel).

Because different local storage devices use different protocols, a server may need an adapter to be able to communicate with one. This hardware (known as a “host bus adapter” or HBA) is plugged-in to a server and adapts the server to the storage device’s protocol. So if, for example, a SCSI-protocol HBA is attached to the server, and the HBA is then connected to a SCSI local storage device with the right cables, it will be able to transmit data from the server to the storage and vice versa.

## Network-Attached Storage



What if you personally wrote one of the 70,000 songs on your 1TB local storage disk and want to share it with your family? But with their busy schedules, sitting everyone down at the same time to listen to your masterpiece isn’t practical, and the file’s too large to email or to send via WhatsApp. (It’s an hour-long symphony!)

In this scenario, the 1:1 relationship of your local disk storage to your server limits your ability to share the data from that storage. It’s too local. “Network-attached storage” (NAS) makes sharing data possible. It’s file-level storage that’s attached to a “local area network” (LAN) via a standard ethernet connection. Everyone authorized to be on the network has access to everything that’s on the NAS device, which provides a centralized location for storage and backup. So, your song is there for everyone to listen to whenever they’re free. NAS has similarities with cloud storage services such as OneDrive and Dropbox, but with NAS your data is stored on a physical device (not in the cloud) that’s owned by you (not by a third party) and over which you, therefore, have full control. For these reasons NAS is sometimes called a “personal cloud”. Cloud storage is essentially a large NAS managed by a vendor who gives you access to it on payment of a subscription fee.



In a business setting, NAS is similar to a “private cloud”, giving employees the ability to collaborate in real-time, with round-the-clock remote access to company data from any device.

Because NAS units are dedicated to serving files, they don’t have keyboards or monitors, and use a stripped-down operating system fine-tuned for data storage and file sharing. Some come with one or more hard drives already installed; with others, you can purchase the hard drives separately according to your specific needs. Storage capacity can be increased by adding more hard drives, which can be done without shutting down the network. A NAS device with more than one hard drive can be used for data backup: data is duplicated on “redundant” drives (i.e., drives that are not being used). The term “RAID” relates to this type of backup.

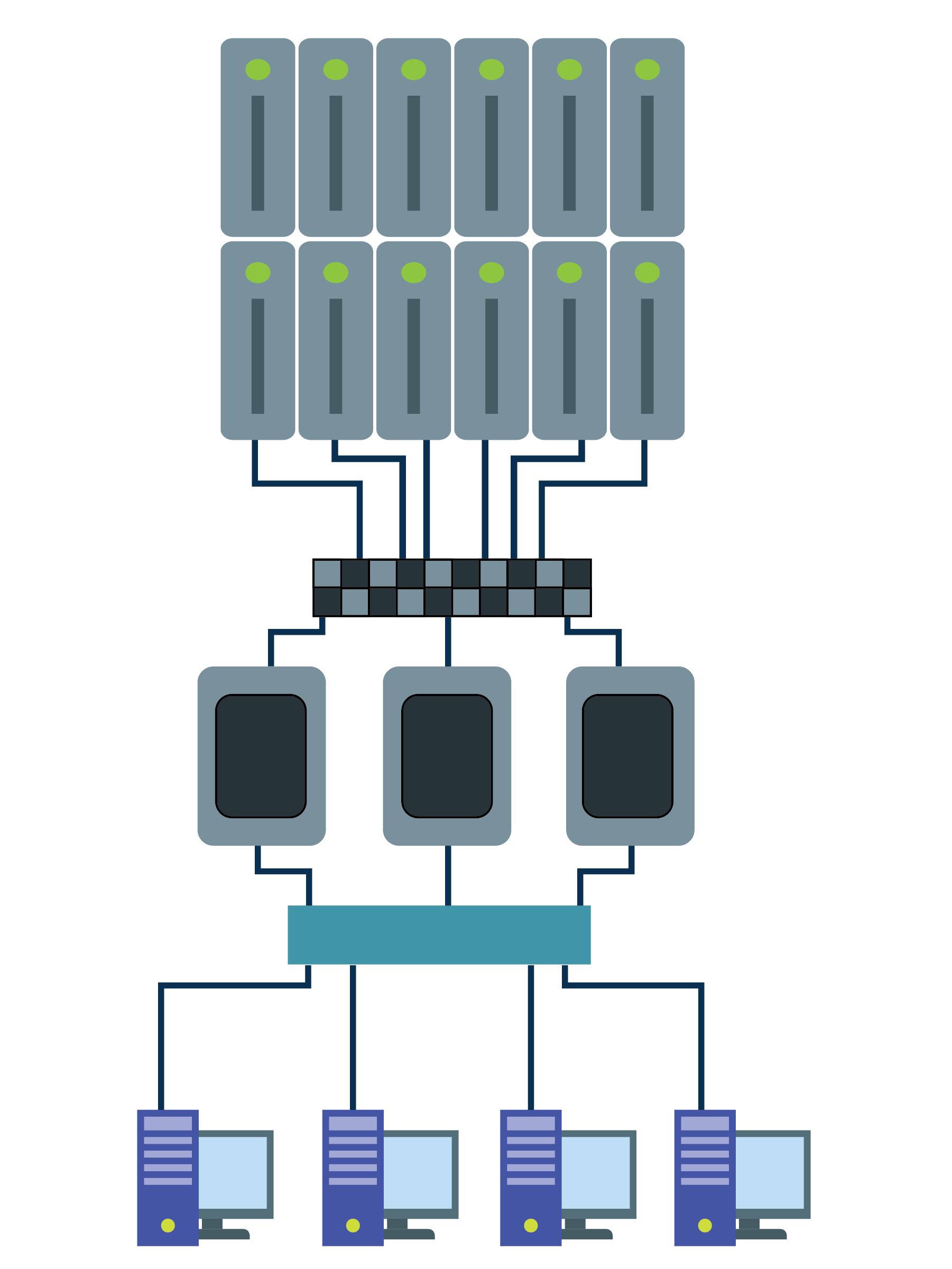
A NAS device will have its own IP address, and, in addition to storing data, will provide a file system by which the data is organized. As with local disk storage, protocols are used in NAS to enable communication between devices. NAS usually uses the Transmission Control Protocol/Internet Protocol (TCP/IP) to transfer data over the internet: TCP bundles data into “packets” and ensures that they are reliably sent; IP standardizes the way data is addressed and routed. The “HTTP” or “HTTPS” that you usually see at the beginning of web addresses in a browser are part of TCP/IP.

The file system provided and managed by the NAS device also needs to follow a network protocol to be accessed. Commonly-used file system protocols used on NAS devices are: NFS (Network File System), which is a frequently used on Linux and Unix systems; SMB (Server Message Blocks), which is mainly used with Microsoft Windows systems and is sometimes referred to as CIFS/SMB as it developed from the Common Internet File System; and AFP (Apple Filing Protocol), which is used with Apple devices. We will speak more about NFS a few sections from here.

## Storage Area Network (SAN)

NAS was perfect for sharing your symphony with your family because it enabled the sharing of your data across a local network. (NAS also tends to be associated with unstructured data such as audio and video files.) It’s also a good storage solution for small to medium-sized businesses (SMBs) wanting to share company data across their LAN (whether they occupy just one office or an entire building).

But how can large-scale enterprises – often spread over several physical sites – share their vast amounts of corporate data (hard drives, disk arrays, and tape libraries) in a way that’s efficient and cost-effective? A “Storage Area Network” (SAN) provides a means for them to centralize their storage management. A SAN is a dedicated high-speed data transfer network that provides block-level access to storage. It gives multiple servers access to a network of storage devices. To avoid confusion with NAS (which is also storage connected to a network), a good way to differentiate between the two is to remember that NAS is storage that connects to a network and SAN is a network of storage that servers connect to. It operates alongside a LAN and is typically associated with structured data – spreadsheets and databases, for example.



In a SAN, the wide range of storage devices in an enterprise (its hard drives, disk arrays, and tape libraries) is consolidated into a virtual storage unit that is accessed via appropriately-configured servers. Although SAN provides block-level storage, file-level access is available via file systems created by the servers’ operating systems.

SAN storage does not need to be in the same location as the servers. They actually appear to the servers’ operating systems to be locally-attached storage devices. So, when a computer wants to access data on a SAN, it doesn’t see or need to navigate the separate components of which the SAN made, it just sees a centralized pool of storage that it can access.

A SAN has 3 components: a “host layer” made up of servers to which other computers connect to request files; a “fabric layer” containing the equipment and cabling needed to create the storage network; and a “storage layer” containing the actual storage devices that hold the data.

SAN uses a number of protocols to enable the communication between the servers and the storage. The Fibre Channel protocol (FC) uses fiber optic cabling (rather than Ethernet) to provide very high-speed access to data. Fiber Channel over Ethernet (FCoE) allows the FC protocol to be used over ethernet and is a cheaper alternative. Internet Small Computer Systems Interface (iSCSI) carries SCSI commands over Internet Protocol (IP) networks and is a good option for SMBs. ATA over Ethernet (AoE) is a simplified protocol that is used to build less complicated and more economical networks. InfiniBand (IB) also offers very high-speed access to data and is most often used for communication between super-computers.

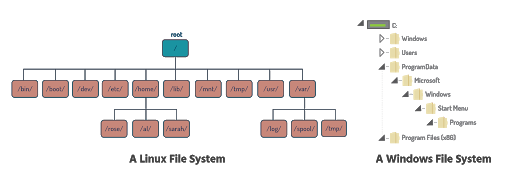
SANs provide enterprises with a range of benefits. Extra storage devices can be added for immediate “scalability” (i.e., the ability to grow in a controlled and stable way). Because SANs work alongside LANs, LAN data traffic does not interfere with their data flow, preventing “network bottlenecks” and reducing the risk of LANs being overloaded. SANs separate storage from servers, allowing the computing resources of those servers to be used for other purposes. A SAN will typically enable the replication (i.e., the continuous copying) of data to another site, which in turn enables business continuity in the event of a failure anywhere in the system. With the centralization of large-scale storage management that a SAN provides, an enterprise has a clearer overview of its storage processes and can, therefore, adopt a more consistent approach to its disaster recovery and security.

Many organizations now, in fact, combine SAN and NAS into a single system (referred to as “unified storage” or “multiprotocol storage”) that uses both block- and file-level protocols.

## Virtual Machine File System (VMFS)

In our discussion of NAS, we briefly mentioned data being organized by a “file system”. Without some method of managing files, the data in a storage device would be an unintelligible mass, with no indication of where one set of data ended and the next began. You can imagine how confusing it would be if the 70,000 songs on your 1tb hard drive were all lumped together, without names and track information. It’s only when data is separated into individual pieces that it can be identified and retrieved.

These separate pieces of data are grouped into files, which are in turn grouped into directories (which may contain subdirectories – directories within directories). This hierarchical arrangement allows users to navigate the different levels of a file system as required. File systems provide “metadata” for each file (e.g., file name, timestamps, permissions) and indexes that let the operating system know what files are on a drive and where they are.



There are many types of file systems, including file systems for disks, for databases, and for networks. Examples of file systems that you may have heard of are NTFS ( supported by Windows), ext (e.g., ext2, ext3, and ext4) and ReiserFS (both supported by Linux), HFS+ and APFS (both supported by Apple’s macOS), and UDF and ISO 9660 for optical disks. On a hard disk the file system is created when the disk is initialized (if it’s new) or formatted (whether new or not).

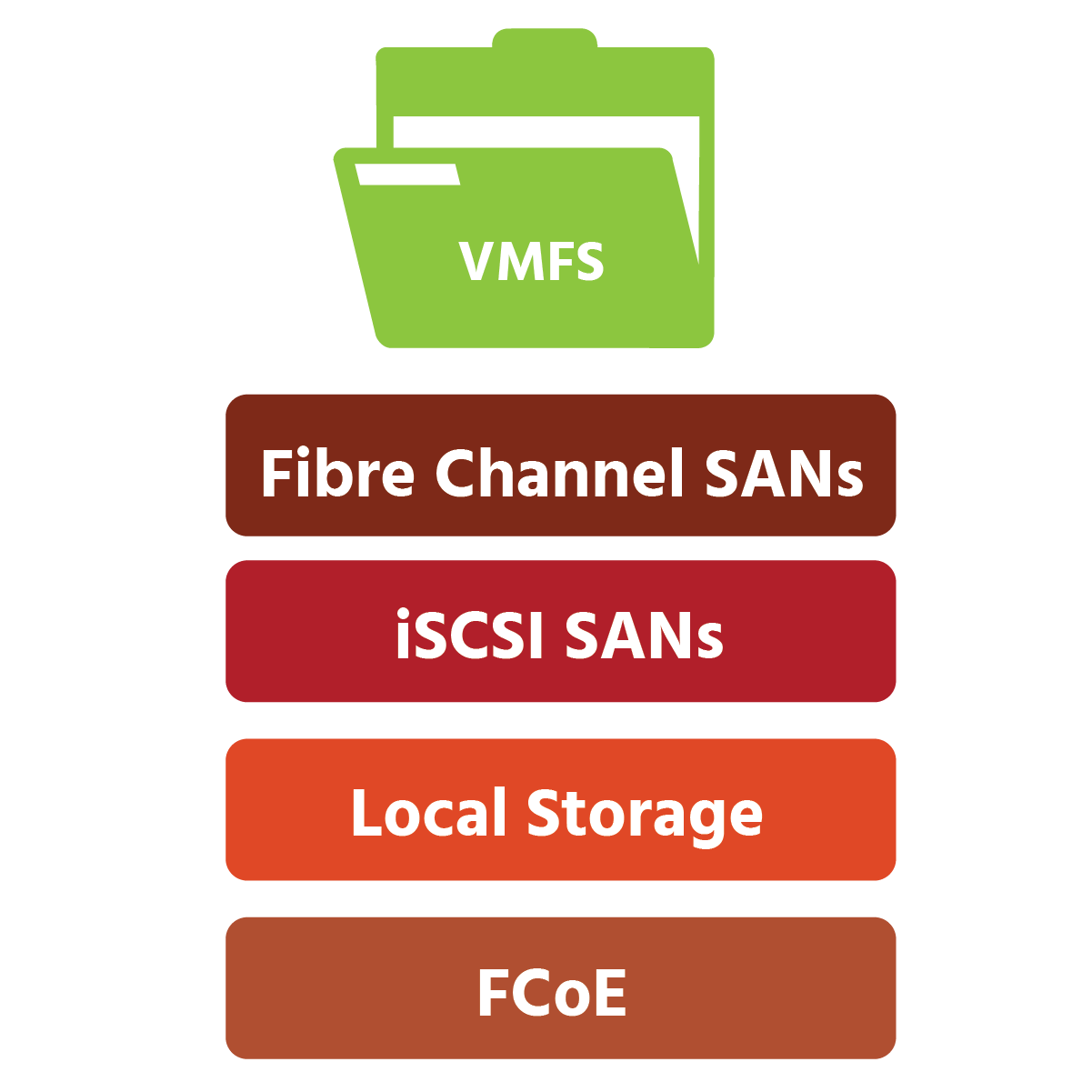
Virtualized storage is designed to give virtual machines (VMs) the storage they need to host their operating systems and applications. In a virtualized environment, the virtual disks of a virtual machine are stored on a “datastore”. (In the virtualization process, a partition is created on a physical drive to form a space known as a “Logical Unit Number” or LUN; one or more LUNs make up a “volume”; a number of volumes make up a datastore.) A datastore typically needs to have a file system installed on it in order for it to be accessed. With VMware's cloud computing virtualization platform, vSphere, datastores are set up with one of two file system formats: the “Virtual Machine File System” (VMFS) or the “Network File System” (NFS). NFS will be discussed in the next section.

VMware pioneered virtualization technologies and developed their own file system, VMFS, to work with them. Conventional file systems allow only one server to have read/write access to a given file at a given time. By contrast, VMFS allows multiple servers to have read and write access to the same storage resources at the same time. This is because VMFS is a high-performance clustered file system (CFS), “cluster” meaning that it is simultaneously mounted (made accessible) on multiple servers.

VMware's virtualization technology allows VMFS to establish and allocate virtualized storage for virtual machines and servers. VMFS also enables multiple VMs to share a single file system. In a simple configuration, the disks of virtual machines are stored as files within VMFS. For VMFS to understand requested file operations from a SCSI device, the virtualization layer translates the SCSI commands issued from a guest operating system.

A virtual machine ‘sees’ a virtual disk that’s stored on a VMFS datastore as a mounted SCSI device. The operating system running inside the VM does not ‘see’ VMFS – it ‘sees’ its own file system instead.

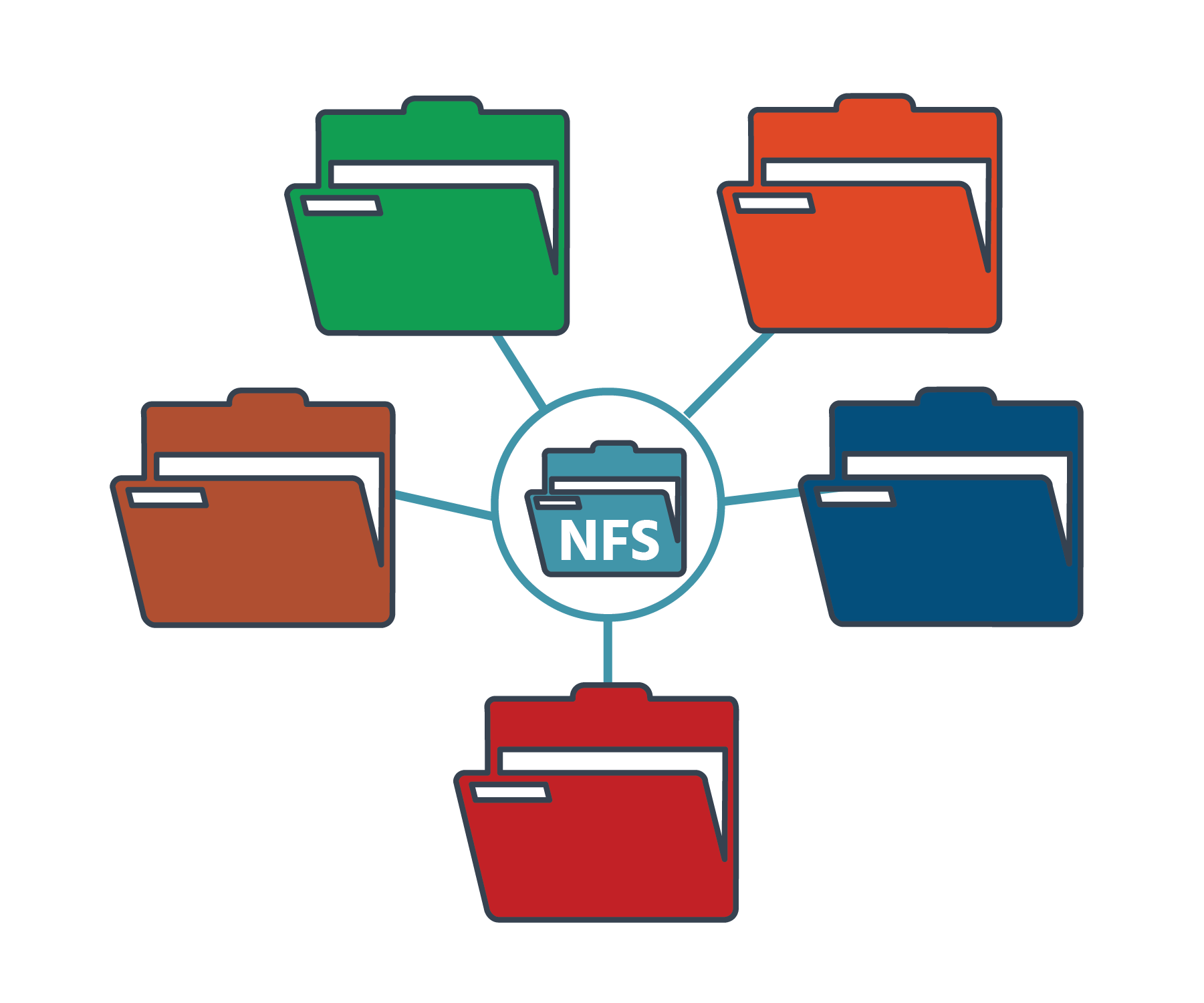
A VMFS datastore can be linked to a single SAN LUN or stretch over multiple SAN LUNs. You can even expand a datastore while virtual machines are running on it, either by growing the datastore or by adding a new area of storage. The VMFS datastore can be extended over 32 physical storage areas of the same storage type. (VMFS can be used on a wide variety of block storage devices, including Fibre Channel SANs, iSCSI SANs, local storage, and Fibre Channel over Ethernet.)



With multiple ESXi servers in a cluster, virtual machines are powered on, the VMFS locking mechanism feature ensures that the VMs are not powered on by all the ESXi servers in a cluster. Multiple ESXi hosts can access the same VMFS datastore in a shared storage environment, thus the reason for on-disk locking. With VMware High Availability, however, the on-disk locking mechanism is disabled so that the ESXi hosts in the cluster can failover successfully in an attempt to power on the virtual machine on another working physical server. As with all hypervisors, ESXi (short for “Elastic Sky X Integrated”) creates and runs VMs. It’s a “Type 1” hypervisor, which means that it doesn’t rely on its host’s operating system, but runs directly on the host’s hardware, containing within itself a basic operating system (OS) that includes the kernel. (The kernel is the core component of an OS, managing system resources.) VMware HA (or “High Availability”) pools VMs and the hosts they reside on into a cluster. Hosts in the cluster are monitored, and in the event of a failure, the VMs on a failed host are restarted on different hosts. VMware HA allows companies to provide high availability to any application running in a VM.

Each virtual machine is encapsulated in a small set of files in a single directory. Disaster recovery is therefore greatly simplified; the entire VM can be remotely mirrored and easily recovered in the event of a disaster.

## Network File System (NFS)



The Network File System (NFS) is an IP-based file-sharing protocol that is used by NAS systems to allow multiple remote systems to connect to a shared file system. NFS uses file-level data access and the target (or destination) NAS device controls the storage device.

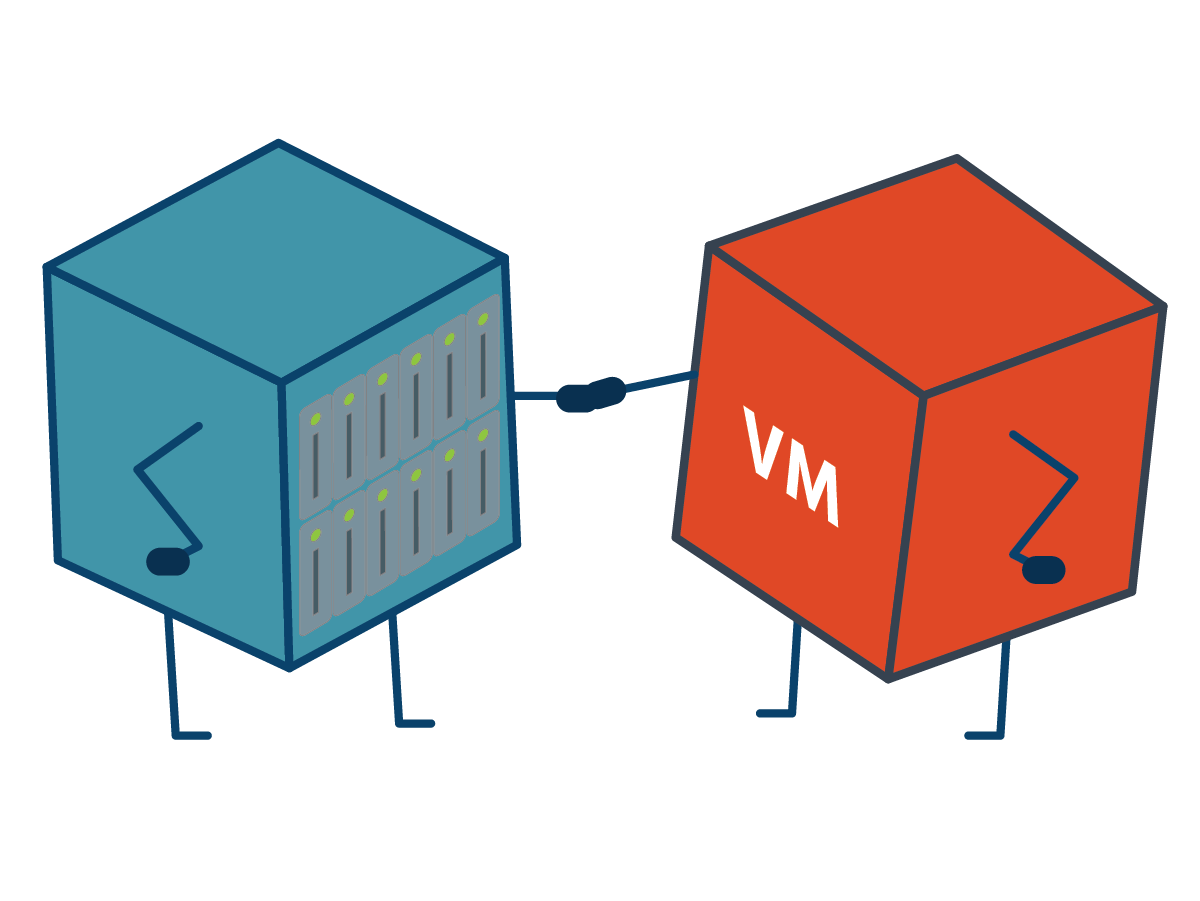
It is a “distributed” file system protocol, which means that its data is stored on a server and that it uses a network protocol to share – and restrict – access to the file system, using the same coding that is used for local files. NFS has three components: a client, a server, and the protocol that enables the two to communicate. It allows a user on Computer 1 (the “client”) to access File A on Computer 2 over a network in much the same way they would access a file on Computer 1’s own hard drive. To Computer 1, NFS is ‘invisible’; while NFS is keeping track of where files are located and transporting data, Computer 1 ‘sees’ a file system that looks local. The naming convention used by NFS makes it impossible to tell from File A’s name that it’s being accessed remotely. And if File A is being modified on Computer 2, those modifications will be seen on Computer 1. Similarly, if the file is moved to a different server, the user of Computer 1 won’t know it.

While NFS is OS- (operating system) independent, it is typically used on Unix systems, macOS and Linux. It is also hardware-, network architecture-, and transport protocol-independent, and works well in both small and large environments.

Distributed file systems such as NFS offer many benefits. Storing data on servers frees up the resources of client machines. Distributing data (e.g., important administrative data) to multiple clients is easier, and it is even possible to transfer files between computers that use different operating systems (i.e., between computers in “heterogeneous environments”). For example, you could use a Windows NFS file server in a Linux environment to give Linux client computers access to files on that Windows NFS server. And rather than every user in an organization having their home directory on their individual network machine, all the home directories could be set up on an NFS server and then accessed as appropriate by everyone on the network.

NFS is the only NAS protocol supported by VMware’s vSphere, which supports NFS version 3 over TCP/IP. (Note that you cannot initialize or format a NAS target from a remote server.) VMware vSphere virtualizes and aggregates (i.e., gathers) the underlying physical hardware resources across multiple systems and provides pools of virtual resources to the data center. It’s made up of infrastructure services for compute, storage, and networking (namely, vCompute, vStorage, and vNetwork), application services (including VMware HA discussed in the last section), and a single point of control for the data center (VMware vCenter Server) which provides essential data center services such as access control, performance monitoring and configuration. Users can access the VMware vSphere data center through the vSphere Client. The home screen of the vSphere Client is a system dashboard that aggregates data from different sources in the environment together in a single, unified view.

Virtual Volumes (VVOLS)



Until recently, storage administrators have been somewhat limited in the flexibility given to them by their datastores. All the virtual machines linked to a particular LUN would have to be treated in the same way – they would all have to be backed up at the same time, for example, or all archived in the same way, even when they served very different purposes. Storage management was ‘LUN-centric’. “Virtual Volumes” (or VVOLs) makes it ‘VM-centric’ for the first time, by enabling administrators to treat each VM individually. With data services, stats, and reporting now available for each VM, management and troubleshooting are faster and more detailed than ever before.

VVOLs encapsulate (i.e., contain within themselves) virtual disks and other virtual machine files. They are located directly (“natively”) on a physical storage array without using a file system.

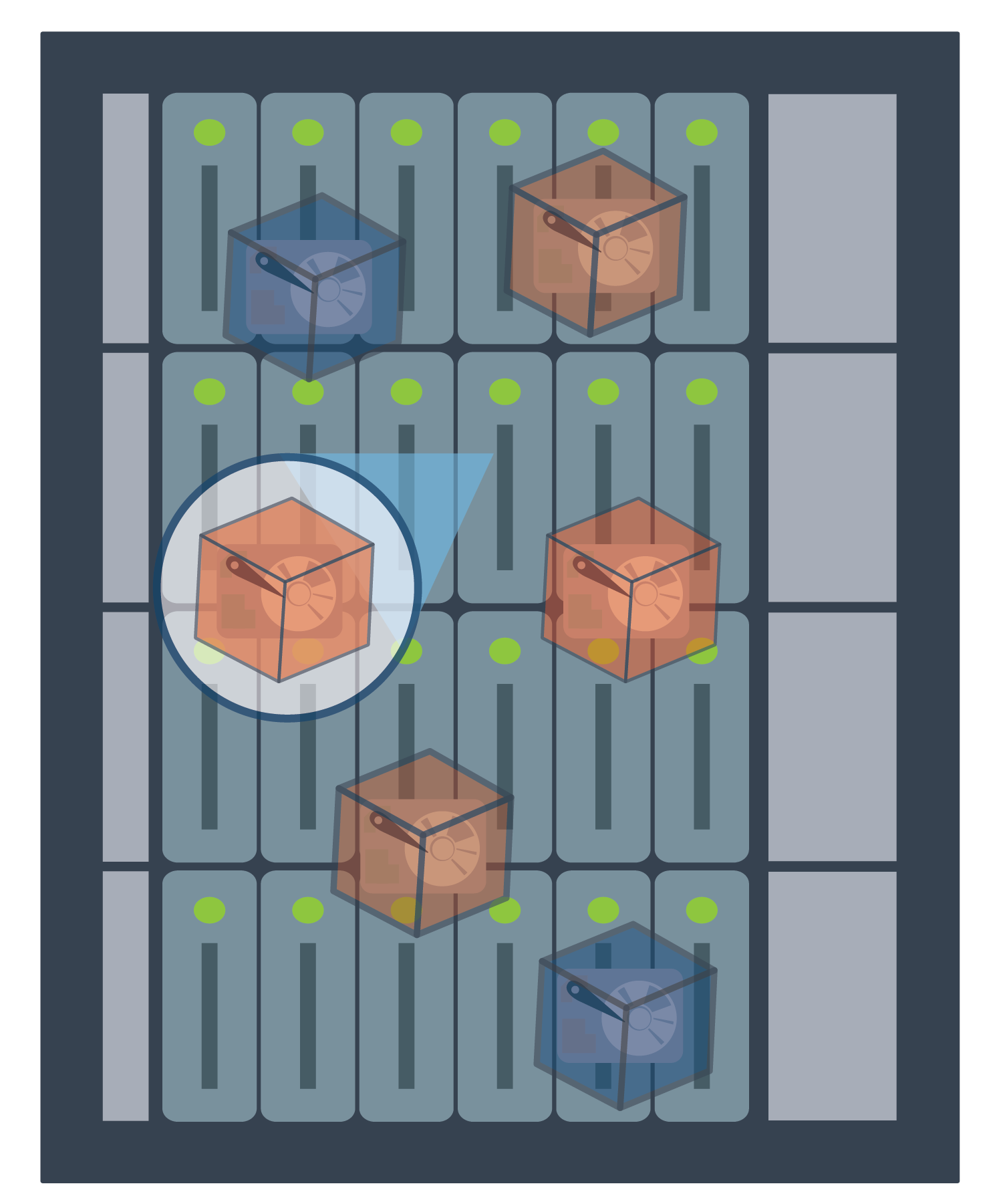
VVOLs can broadly be classified into five types:

* **Config-VVOLs** contain VMX (the primary configuration file), NVRAM (the file that stores the state of the virtual machine’s BIOS; the BIOS controls the booting process), and log files
* **Data-VVOLs** contain data related to VMDKs (virtual disk drives which store the contents of the virtual machine’s storage device) and delta files (a snapshot file – snapshots being copies of VMDKs taken at a specific point in time)
* **Mem-VVOLs** contain data related to memory snapshots (which are more detailed snapshots)
* **Swap-VVOLs** contain information about swap (memory) files
* **Other-VVOLs** are a generic type of VVOL containing files relating to particular vSphere features.

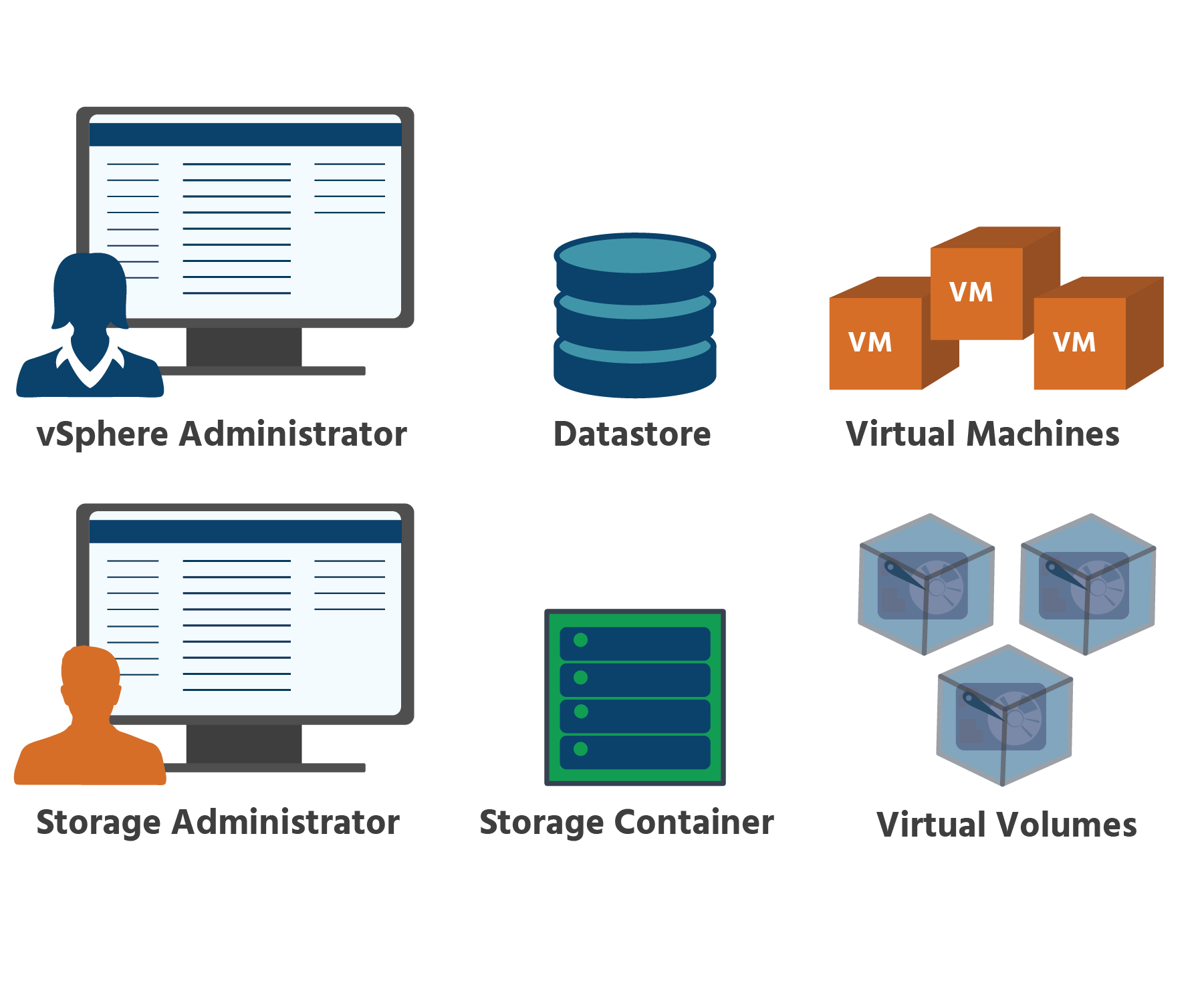
When a virtual machine is created or cloned, or a snapshot is taken of one, a VVOL is automatically created. Every VM will have a config-VVOL. When a VM is powered-on, a swap-VVOL will be created. (It will be deleted when the machine is powered off, and the array will reallocate the space.) As you add virtual disks, each one will have a data-VVOL. Additional VVOLs can be created for other virtual machine components and for items derived from virtual disks, such as clones, snapshots, and replicas.

ESXi hypervisors (which create and run the virtual machines) do not have direct access to virtual volumes. They use “protocol endpoints” to communicate with VVOLs and the virtual disk files that they encapsulate. Each virtual volume is bound to a specific protocol endpoint, but each protocol endpoint can connect to hundreds or thousands of virtual volumes.

For management purposes, VVOLS are grouped into “storage containers”, which are pools of raw storage capacity provided by the storage arrays to the VVOLs. Storage administrators set VVOLs up and can control their size and number. The ability to adjust the size of VVOLs is another example of the flexibility they bring, as LUNs have a fixed size. At least one storage container per array is required, and its size will depend on the physical storage capacity of the array. A single container can be simultaneously accessed through multiple protocol endpoints. Multiple containers can be created depending, again, on the storage array capacity. Containers can’t, however, go across multiple storage arrays. Each storage container serves as a virtual volume store, and virtual volumes are allocated out of the storage container capacity.



Storage containers are not visible to a vSphere administrator. In the vSphere Client they are represented by familiar datastores:



Although they might appear as two different entities, in reality, a virtual datastore is nothing but a storage container in disguise. (However, vSphere needs to preserve the concept of a datastore for virtual volumes because several vSphere features use datastores as the main unit of storage.) A vSphere administrator will use the datastore wizard to link (or “map”) storage containers to virtual datastores. The virtual datastore that they create will correspond directly to a specific storage container and will represent that container in vCenter Server and the vSphere Client. From the perspective of the vSphere administrator, the virtual datastore is similar to any other datastore and is used to hold virtual machine files.

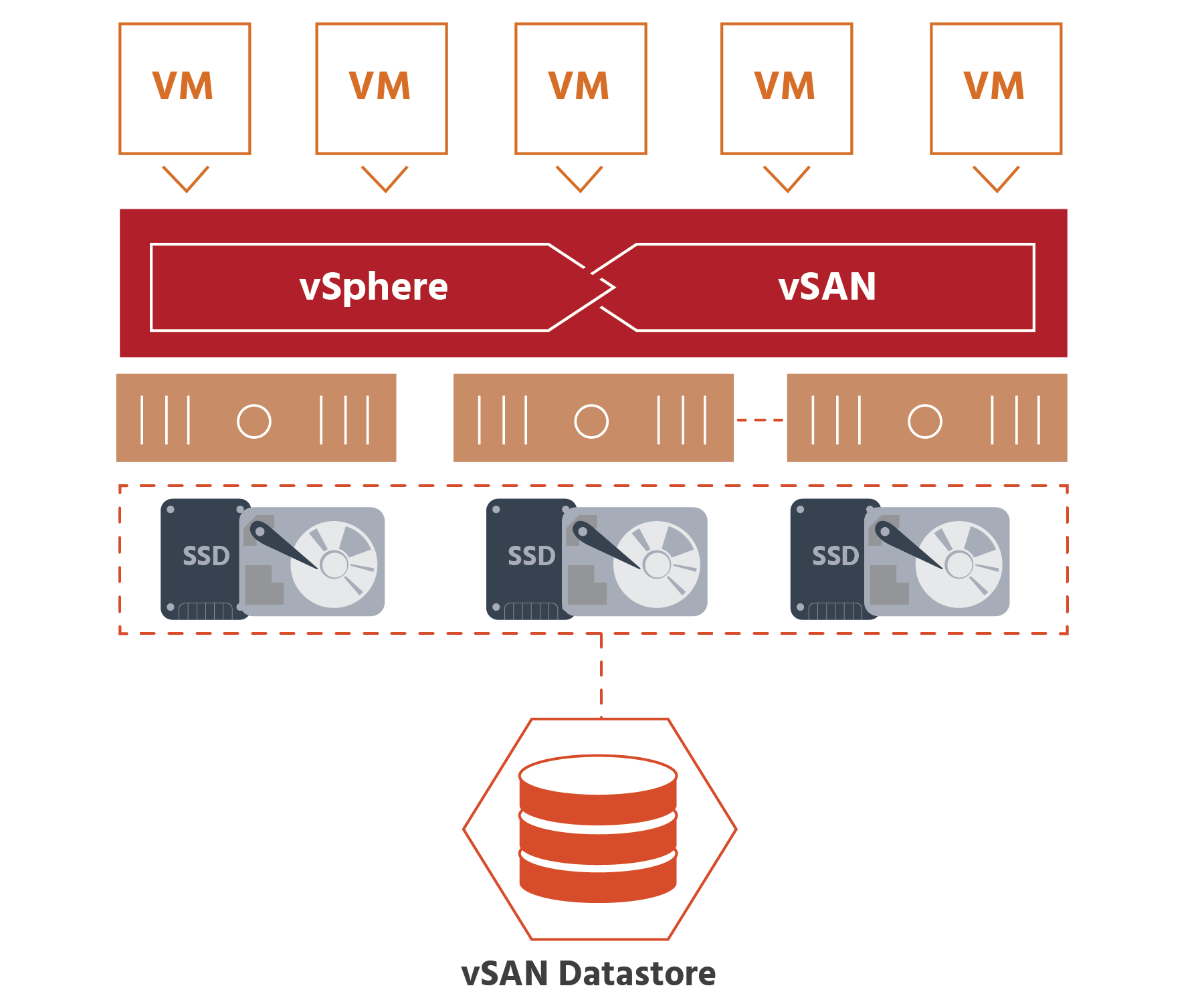
VVOLs use “storage providers” (also known as “VASA providers”) to manage all aspects of their storage. A storage provider delivers information from the underlying storage container. The storage container capabilities appear in vCenter Server and the vSphere Client. Then, in turn, the storage provider communicates virtual machine storage requirements. VASA (which stands for “vStorage APIs for Storage Awareness”) is a set of application program interfaces (APIs) that enables vSphere vCenter to recognize the capabilities of storage arrays. (We’ll be looking at APIs in section 5.4.)

The virtual volume is an industry-wide initiative supported by all the major storage vendors.

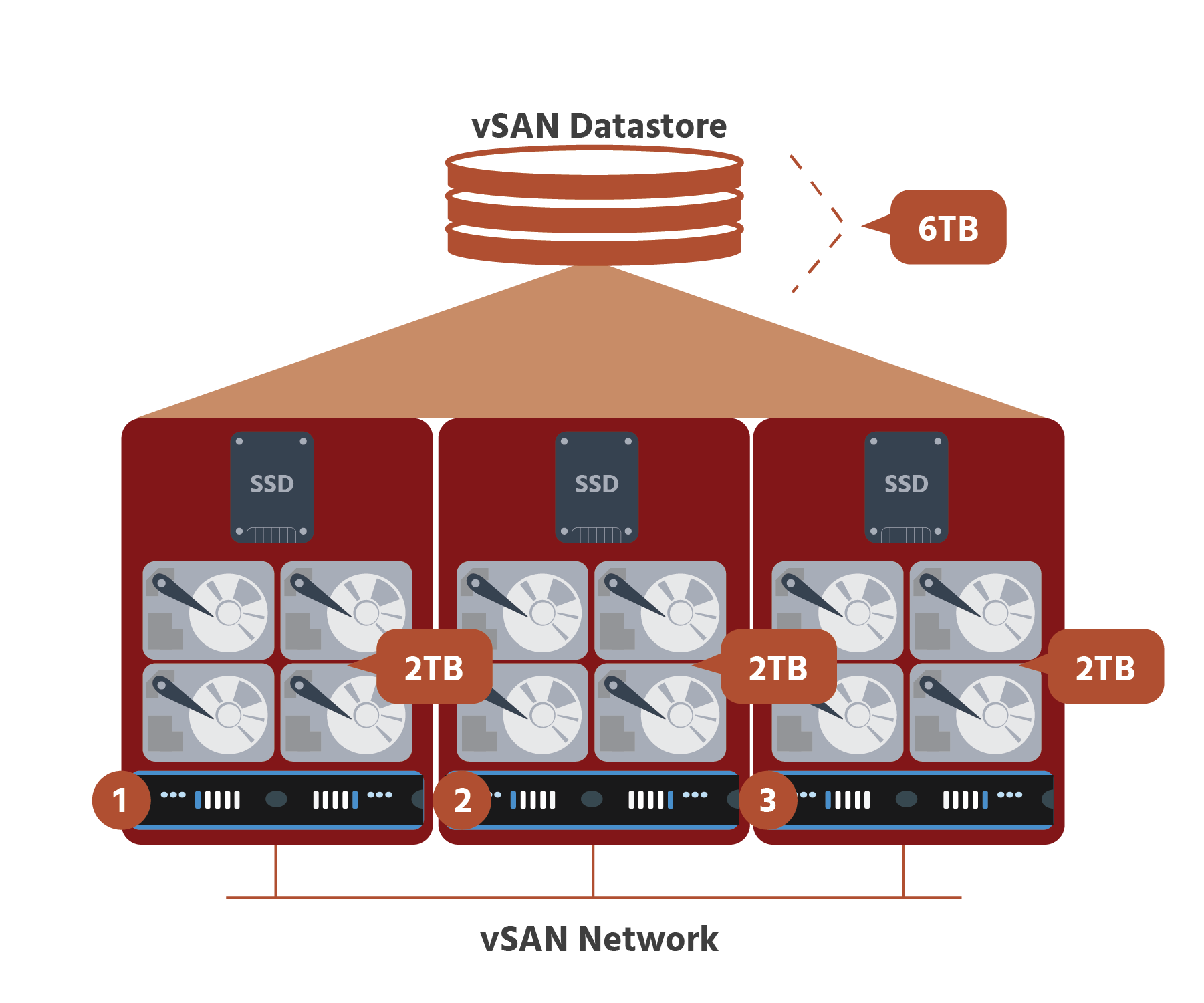
## vSAN

IT operations are a balancing act. Administrators have to maintain efficiency, manage more and more complex storage infrastructure, and reduce costs – all at the same time. They are constantly looking for ways to simplify their operations and make their IT infrastructure more efficient.

vSAN is a virtualized storage-area network that is being used today by storage administrators around the world for just these reasons. It’s a layer of software included in the ESXi hypervisor that eliminates the cost and complexity of traditional storage by virtualizing the physical storage resources of ESXi hosts and turning them into a pool of storage called the vSAN datastore. This pool is shared across all hosts in a vSAN cluster (a cluster is a group of computers working together as a single system.) Virtual routing and switching are used to network this clustered storage. As a result, administrators no longer need to set up and maintain separate physical arrays, storage networking hardware, and cabling.



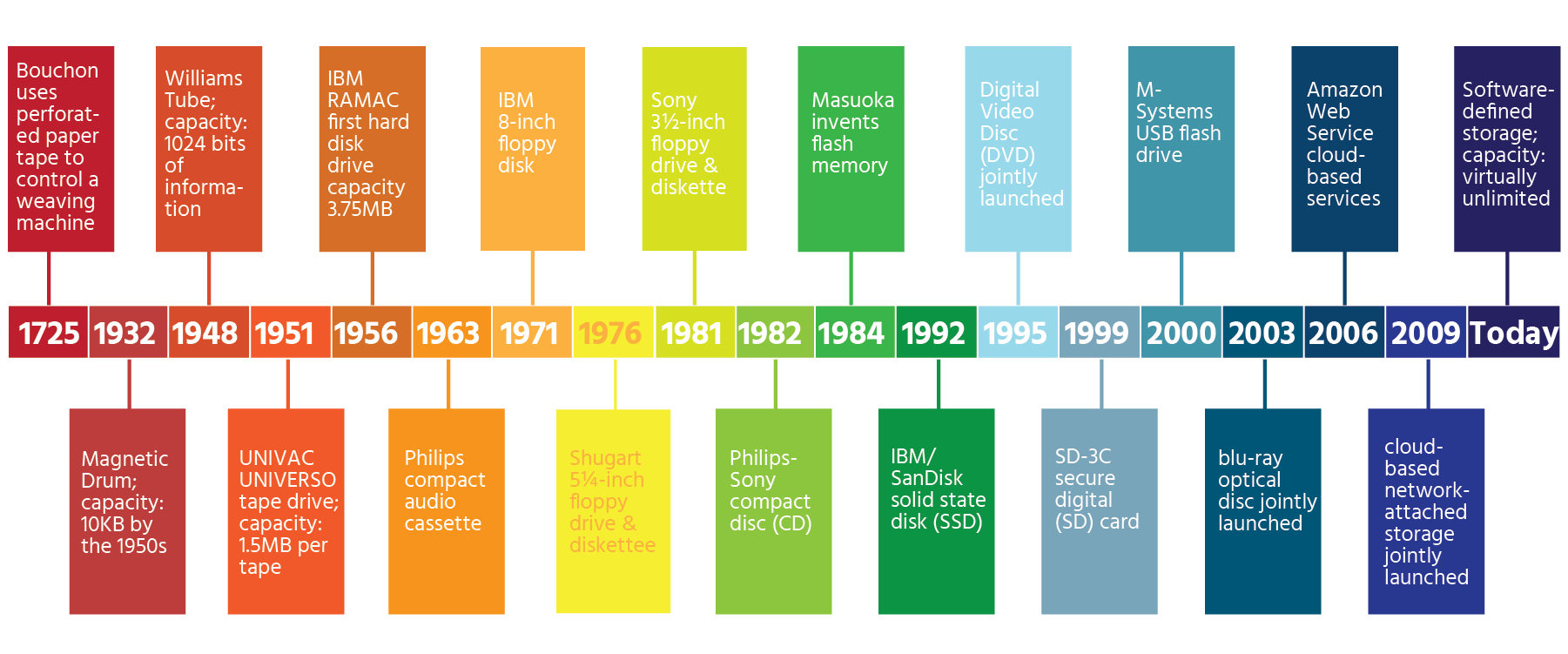
The vSAN datastore is object-based storage that is created using disk groups from the hosts in a cluster. (A vSAN disk group is a logical container grouping multiple disks together on a host.) The disk groups of all ESXi hosts in a vSAN cluster are combined to create a vSAN datastore. The size of a vSAN datastore is governed by the number of disk groups per ESXi host and the number of ESXi hosts in the cluster. For example, if each host has one 2-terabyte disk group and there are three hosts in the cluster, the raw capacity of the vSAN datastore is 6 terabytes. (Note: vSAN requires at least one flash-based storage device per disk group. VMware recommends that whenever possible, all the storage on a vSAN disk group should be flash-based for optimal performance potential.)



The vSAN datastore is used to store virtual machine files, including the virtual machine disks (VMDKs). The vSAN datastore is only accessible by hosts that are part of a cluster that is enabled by vSAN. The vSAN datastore cannot be presented to external hosts, but VMware uses an iSCSI service to create iSCSI LUNs and storage devices to present vSAN storage to external resources. This enables vSAN to provide protections to non-vSAN systems. Remote datastores LUNs and NFS can be presented to hosts that are part of the vSAN cluster.

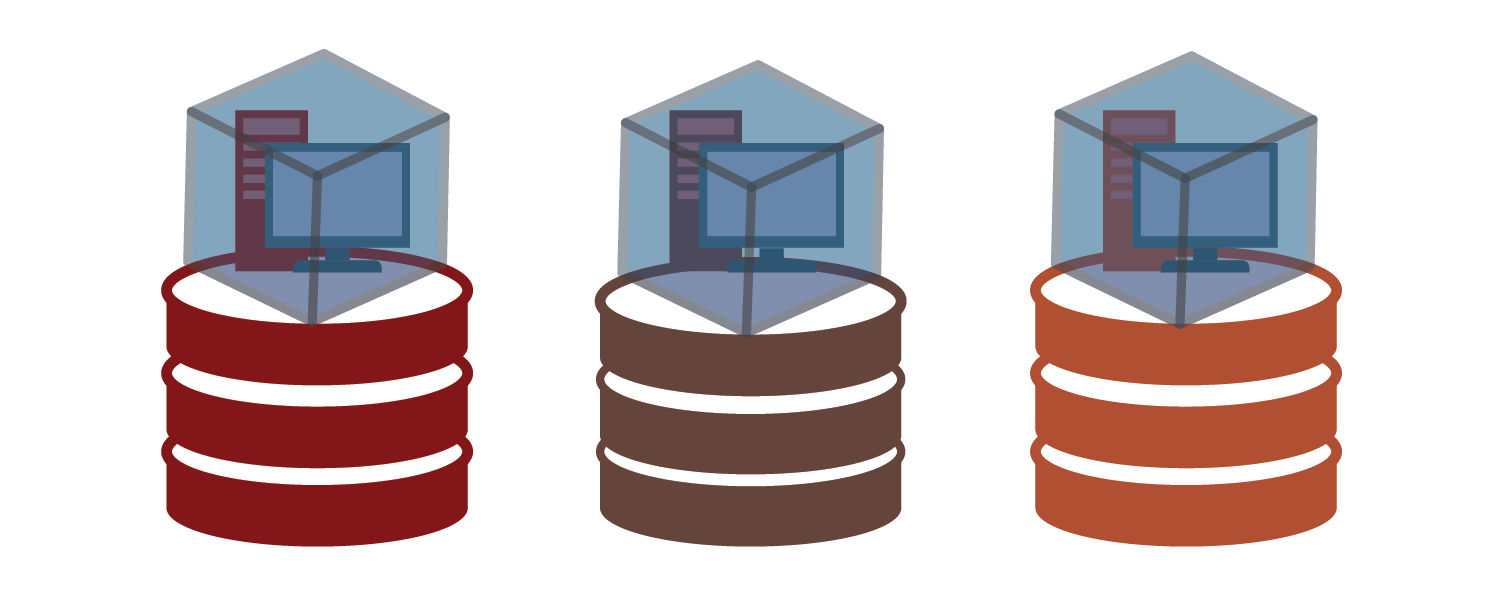
## Introduction to Software-Defined Storage

Virtualization has brought many benefits to the IT landscape over the years – cost and space savings, reduced energy consumption and better disaster recovery, to name just a few. Software-defined storage (SDS) aims to bring similarly positive transformation to the world of storage by enabling a more efficient and flexible approach to storage in virtual environments. But before we delve into software-defined storage, we thought you might enjoy a brief tour of some of the milestones in the history of data storage – to see how far we’ve come!

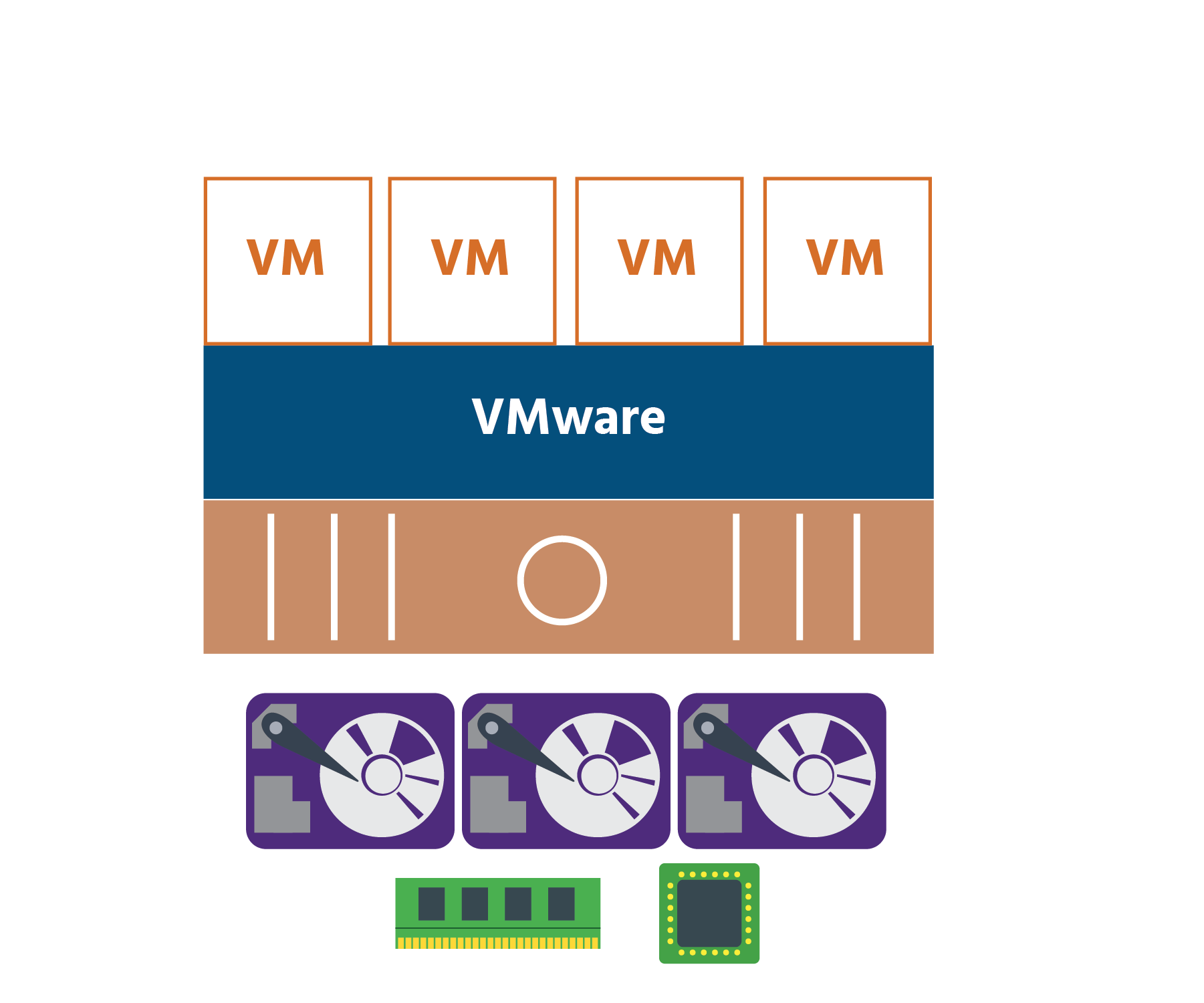


At the heart of this transformation is the hypervisor – the software that creates and runs virtual machines. As the layer between applications and available resources, the hypervisor can balance all IT resources such as compute, memory, storage, and networking that are needed by an application.

Virtual machines are encapsulated in sets of files and/or objects to store their operating system, program files, and other data that is associated with their activities.

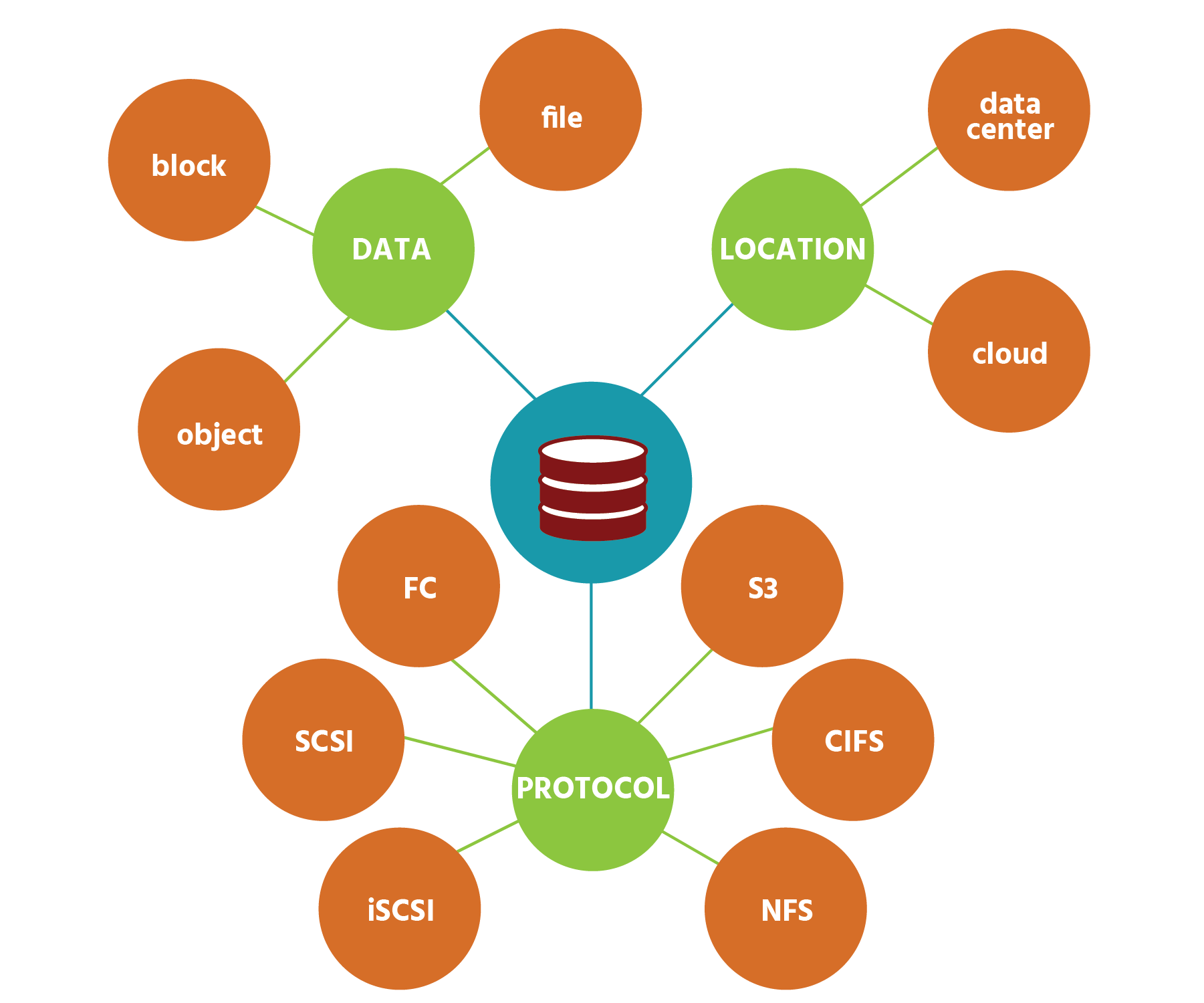


The operating system of a physical server directly accesses local or network-based storage devices. Storage devices are statically and individually mapped and configured for each host. The operating system of a virtual machine interacts with installed hardware through the hypervisor. The hypervisor provides storage resources dynamically to virtual machines as needed to support the operation of the virtual machines.



Using the hypervisor, virtual machines can operate with a degree of independence from the underlying physical hardware. For example, virtual disks can be moved from one type of storage system to another without affecting the virtual machine’s functions.

## What is Software-Defined Storage?



In one sense, it’s easy enough to know what software-defined storage (SDS) is because the storage industry itself provides us with a definition. The Storage Networking Industry Association ([www.snia.org](https://content.netdevgroup.com/contents/sds-concepts/A7bi6oYHax/)), the global trade association, describes SDS as “Virtualized storage with a service management interface”. (An “interface” is a boundary through which separate systems interact with one another.) It is important to note that the terms storage virtualization and software-defined storage are not the same and that storage virtualization is only a piece of the whole SDS stack. SNIA goes on to say that SDS “must include:

* “Automation – Simplified management that reduces the cost of maintaining the storage infrastructure.”
* “Standard Interfaces – APIs for the management, provisioning, and maintenance of storage devices and services.” (An API is a set of tools that allows one set of software components to interact with another – we will be discussing these later in the course.)
* “Virtualized Data Path – Block, File and/or Object interfaces that support applications written to these interfaces.” (A data path is a route taken by blocks, files and objects to and from storage devices.)
* “Scalability – Seamless ability to scale (the storage infrastructure without disruption to the specified availability or performance.” (To scale is to increase/decrease the size of your storage.)
* “Transparency – The ability for storage consumers to monitor and manage their own storage consumption against available resources and costs.”

But to more fully understand the purpose of SDS, it’s necessary to understand at least some of the limitations of the storage technologies that existed before it. Limitations such as the inflexibility and high cost of proprietary hardware – hardware manufactured by one vendor that cannot be used with hardware manufactured by another vendor. An organization using such hardware that needs a new feature may need to have the feature custom-made – at more expense. And increasing the capacity of proprietary hardware by adding more disks is, of course, costly. Another limitation is how complicated storage technologies have been. RAID levels, striping, mirroring. Continuous data protection, asynchronous replication, data deduplication. LUNs, volumes, datastores. Storage administrators have had to know all this (and much, much more) while keeping up-to-date with the latest developments in the field.

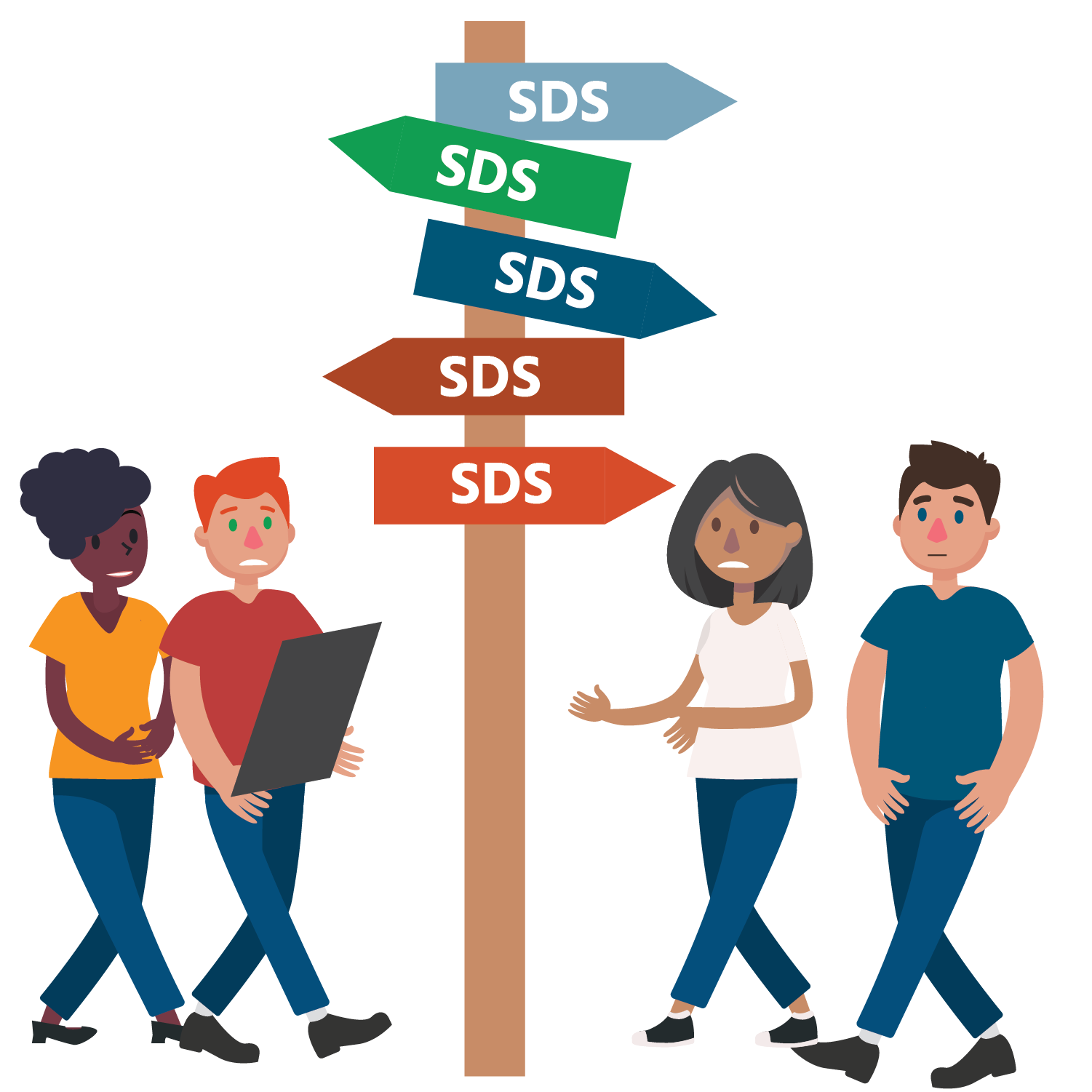
Against this backdrop, it’s easy to see the advantages of software-defined storage. SDS separates storage software from the hardware it’s managing. Storage is no longer locked into proprietary hardware because SDS works with “commodity” hardware – that’s any industry-standard or X86 system. Organizations can, therefore, buy lower-cost hardware that genuinely works for them. If they want to upgrade the software independently of the hardware, they can. SDS helps manage all the different types of storage device, by adding a software layer that controls all of input and output operations from the different storage devices.

VMware’s SDS technologies are simple – yet intelligent. Whereas in traditional storage, applications request data from storage devices, now SDS responds to those requests, locating the data and retrieving it. And while administrators are busy doing other things, SDS can monitor how often data is being accessed and move it to optimal storage media.

Because SDS technologies work at the VM level, storage services can be tailored to precise requirements and adjusted as needed for each individual application – without affecting neighboring applications. Administrators are in complete control of what storage services (and therefore costs) are consumed by each application. They can precisely match application demand and supply at the exact time the resources are needed. Storage services become fluid – a little more for this application now, a little less for that one later.

With the rise of the virtualized data center, software-defined storage brings storage technology into the modern era.

Types of Software-Defined Storage

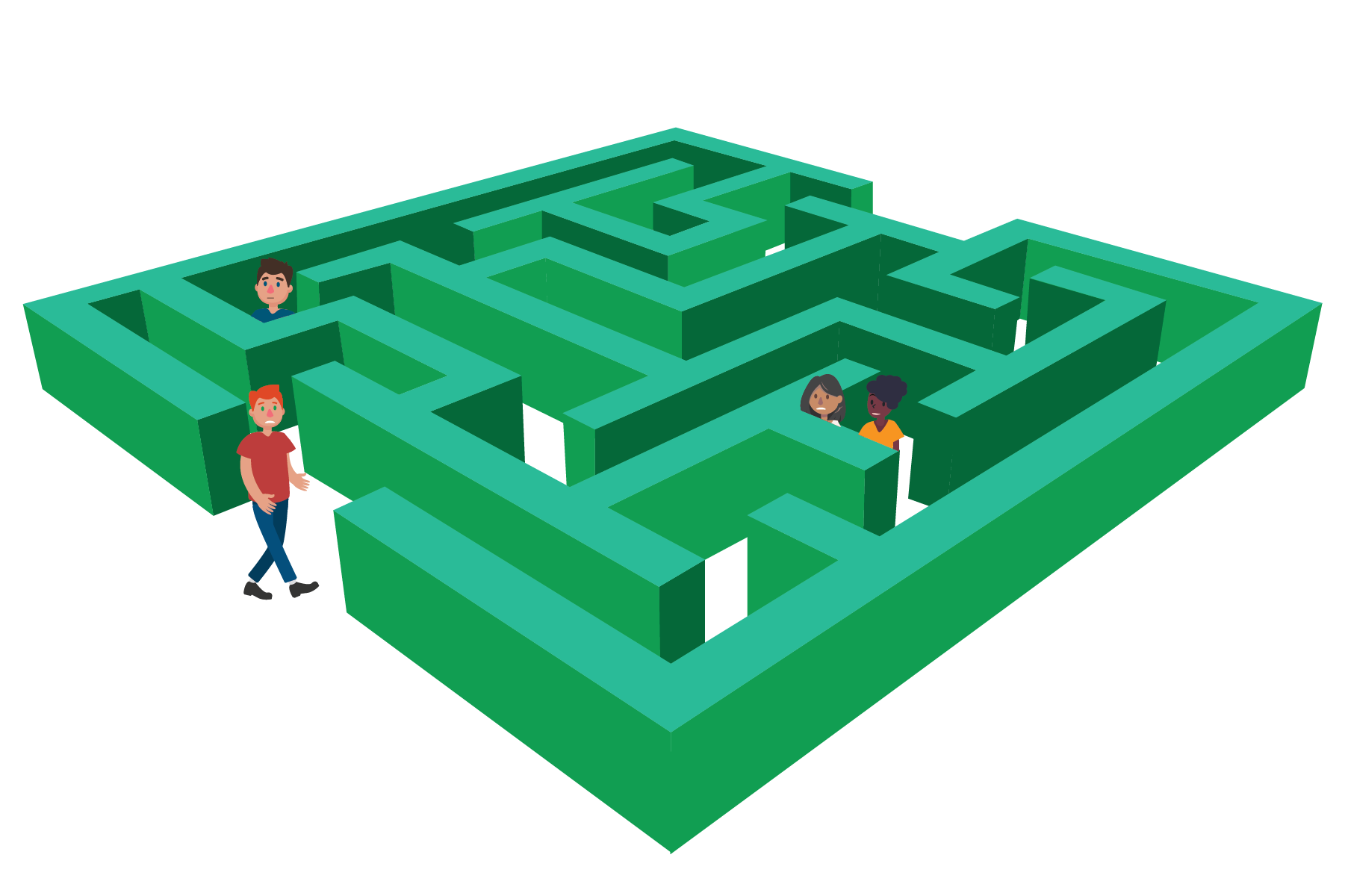


There’s a problem with software-defined storage. It’s a good problem to have, but it’s a problem nevertheless. And it’s this: SDS offers administrators *so many potential benefits* that getting to know them all can be difficult! Beyond just giving access to and storing data, as well as protecting it and moving it when necessary, SDS can control the entire storage process, monitoring it, assessing it, and managing it.

Add to that its ability to...

* work with file-, block- and object-based storage
* be hypervisor-based or container-based
* be software-only, or software plus hardware, or software plus hardware plus storage
* be part of a hyper-converged infrastructure (where virtualized computing resources, networking, and storage are bundled together into one package)

...and the “problem” only deepens!



There *is* a solution to this “problem”, though, and it’s two-fold. First, administrators need to have (or gain as quickly as they can) a clear and detailed understanding of what their current storage needs are and what their future needs might be. Asking themselves key questions will be vital in this: questions, for example, about the amount of storage that is currently being used and the amount that is currently not being used; about how often data is being accessed and, following on from this, whether the current types (or tiers) of storage are the most useful and cost-effective; about the pace at which storage needs are growing and the speed with which data can currently be retrieved; about the level and type of protection needed; and about the relationship of their on-site storage to their storage in the cloud.

Secondly, with the answers to these questions, administrators can then simplify all of SDS’ potential benefits into its three areas of strength: its ability to *manage* what data is being used and accessed, when and from where, producing cost-saving and efficiency-improving reports that provide accurate insights into current and future best practice; its ability to protect through backup, data recovery, appropriate access rights, sanitization (removing sensitive data from documents), and other services; and its ability to function as storage *infrastructure*, providing all the components that storage needs including copying, compression, encryption, along with more sophisticated features such as moving data to the most appropriate tier of storage and even adjusting the behavior of storage media.

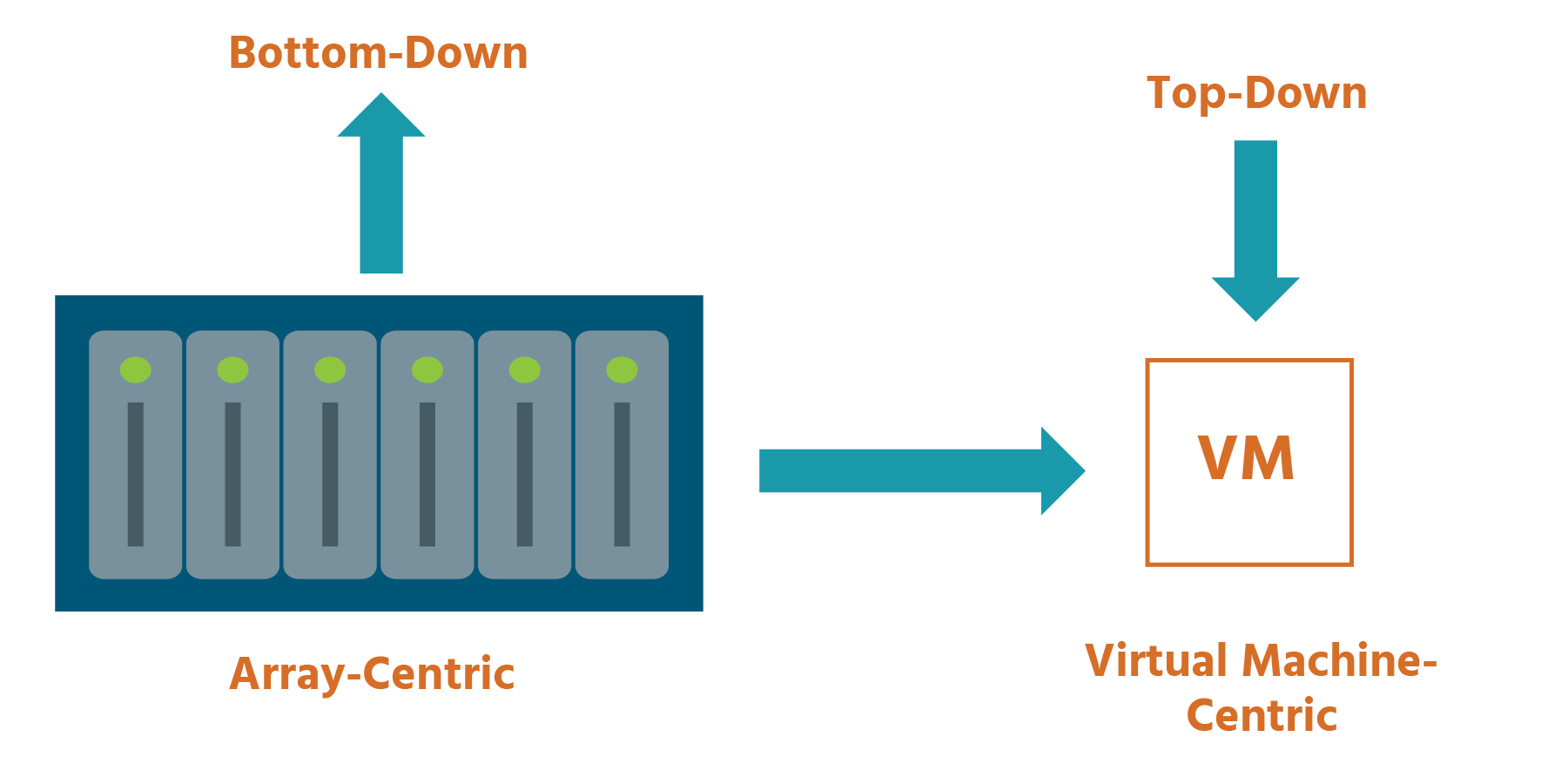
With detailed knowledge of their systems and a clear overview of software-defined storage’s three areas of strength, administrators should be able to permanently archive the “problem” of SDS’ many benefits.

There is a range of SDS types that exist, to name a few:

* Hypervisor-based: a storage hypervisor that helps manage multiple pools of storage; an example of this would be VMware’s vSAN technology
* Hyper-Converged Infrastructure Package: this option packages compute, storage, networking, and virtualization in the same hardware
* Container-based: software-defined storage that is specifically built for container environments like Docker

## Software-Defined Storage Model

VMware’s software-defined storage model literally turns the world (of storage) upside down. It introduces a 180° shift from the ‘bottom-up’ array-centric approach of existing storage options to a ‘top-down’ virtual machine-centric model:



It puts the application and its requirements at the top of the hierarchy, with storage responding to changes in application requirements.

This is in sharp contrast to the current approach, which usually requires configuring static pools of storage resources, and hoping that there’s a match between what applications need and the services that have been provided for them. By comparison, software-defined storage uses application policies (or pre-set rules) to create a “just-in-time” model for storage services. Storage assets and capabilities aren’t configured and assigned to specific applications until they’re needed. And if the policy changes, the storage environment dynamically and automatically changes with it.

These storages services (capacity, performance, protection, encryption, replication, etc., as needed by an application) are ‘made to order’ from available resources, rather than being selected from a static pool of static services. As a result, storage services are precisely-aligned to application requirements.

This shift in approach enables the construction of a “converged” model for IT operations, one where traditionally individual technology disciplines merge to provide dynamic application services. The hypervisor makes this possible. It is positioned between the physical server and the VMs that run on it, and manages the underlying storage infrastructure, balancing the needs of each VM and the applications that each VM runs.

This model offers many benefits. As there are far fewer steps (due to automation), they’re usually done by far fewer people, increasing efficiency. There’s no waste: applications get just what they need (performance, capacity, protection, etc.) and no more. Underlying infrastructure can change, but the higher-level automation processes won’t need to. And storage can be built on lower-cost, industry-standard hardware, as opposed to expensive proprietary hardware.

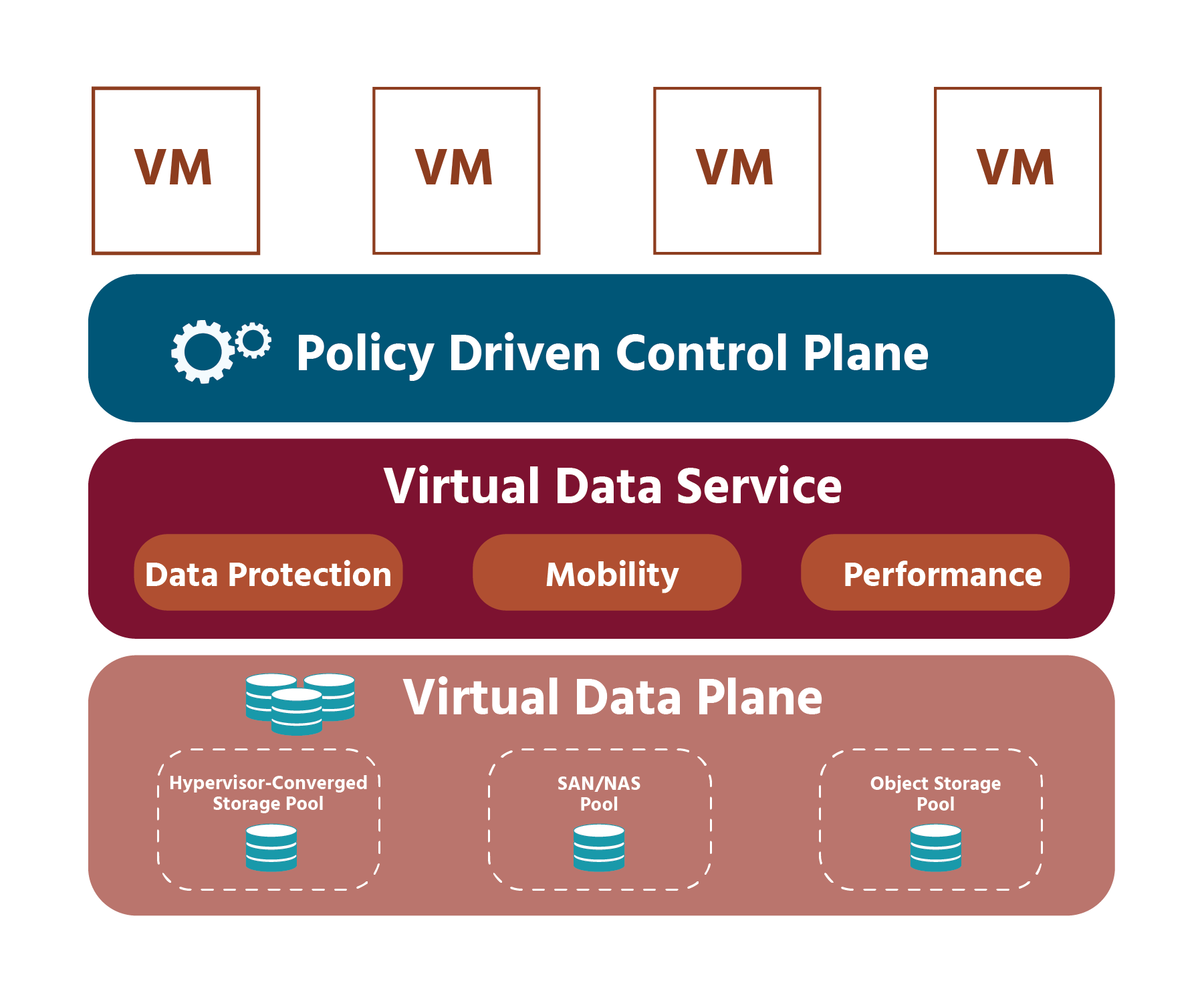
Not all organizations will adopt software-defined storage at the same rate, or in the same way. One useful way of distinguishing different IT organizations is by their preferred automation model, a good indication of their progression towards a software-defined environment:

| **USER INTERFACES AND WIZARDS** | **CUSTOM AUTOMATION** | **HOMOGENEOUS CLOUD** | **HETEROGENEOUS CLOUD** |
| --- | --- | --- | --- |
|  |  |  |  |
| Modestly-sized environments | Larger environments | Focus is on moving to cloud, but at a reasonable pace | Focus is on cloud-style automation and scale |
| Use vendor-supplied tools to configure, monitor, and run infrastructure | Use vendor-supplied user interfaces and APIs for extensive low-level automation of regular tasks (scripting) | Usually have a single-purpose cloud | Cloud is default platform for all applications |
| Focused on virtualization, not “cloud” | Focused on virtualization, not “cloud” | 10-15% of VMs running in a cloud “sandbox” (i.e., separated for security reasons) | 80-90% of apps running on self-service platform |
| No interest in additional automation, so unlikely to fully embrace SDS | No management or automation framework | “Custom Automation” for core apps and mission-critical tasks | Strong DevOps team |
|  | Aspire to cloud, where policy can replace ad hoc scripts | Will value the ability to create application-centric policies | Will greatly benefit from SDS in all of its aspects |
|  |  | Most aspire to heterogeneous cloud |  |

## Virtual Data Plane

The word “plane” has a number of uses in the English language, but it is used in virtualization to refer to a specific level or layer in an organization’s IT architecture.

The virtual data plane stores data for later retrieval, and applies data services such as compression (reducing the size of files), replication (continuous copying of data), caching (storing temporarily for quick access), snapshots (images of a system at a particular point in time), deduplication (eliminating duplicate copies) and availability (the proportion of time that a system is working properly).



While data services are provided by a physical array or implemented in software, the virtual data plane abstracts the services and presents them to the policy-driven control plane (which we will speak about in the next section) for use. The virtual data plane also applies whatever policy has been selected to the objects in the virtual datastore.

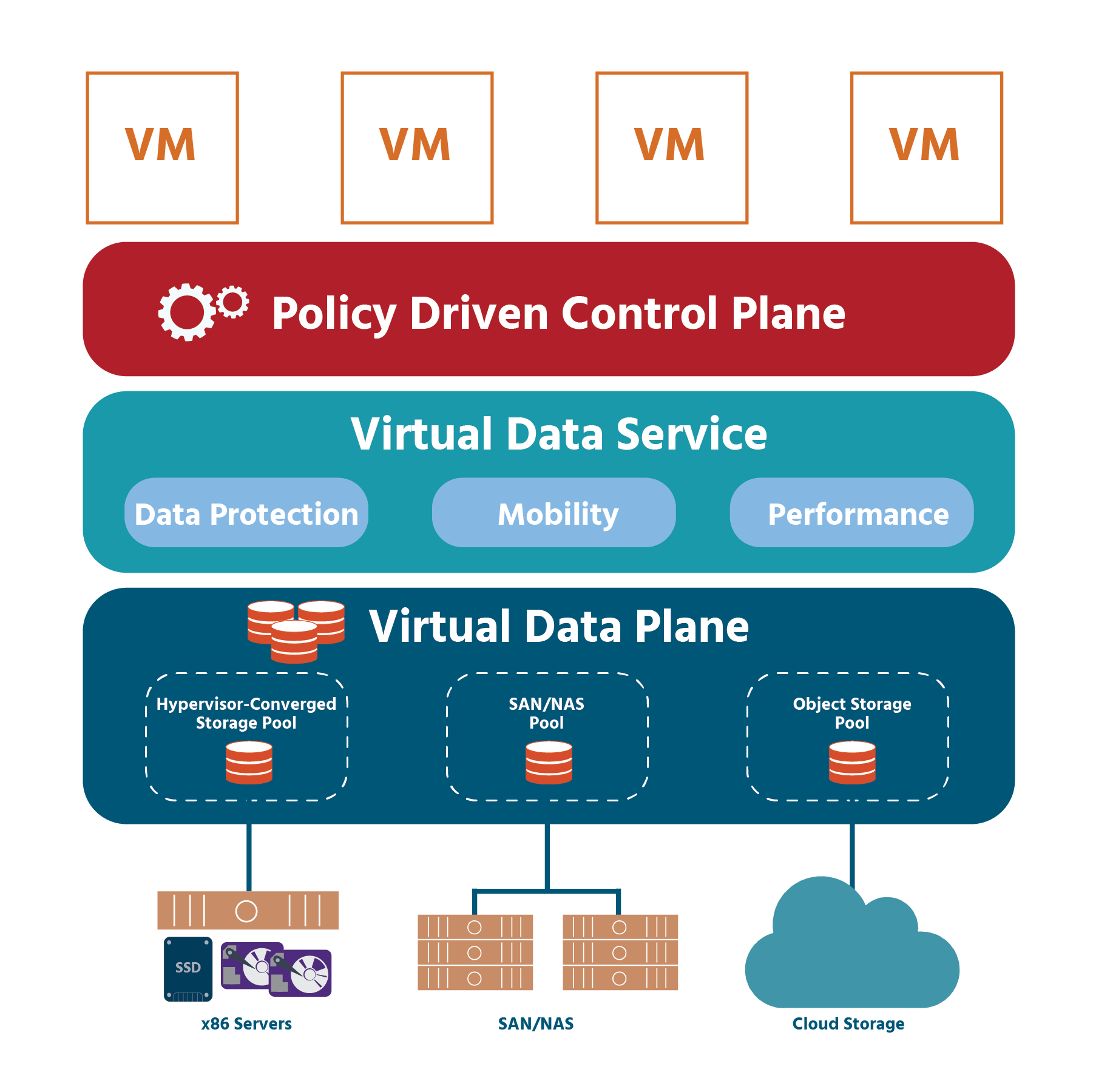
In today's model, the data plane operates on rigid infrastructure-centric entities such as LUNs or storage volumes. In the VMware SDS model, the data plane is virtualized by abstracting the physical hardware resources and aggregating them into logical pools of capacity. These logical pools of capacity are known as virtual datastores, and they can be flexibly used and managed.

By making a virtual disk the fundamental unit of management for all storage operations in the virtual datastores, exact combinations of resources and data services can be configured and independently-controlled for each virtual machine.

The virtual data plane in the VMware model is delivered through vSAN for X86 (i.e., industry-standard, “off-the-shelf”) hyper-converged storage, and through vSphere virtual volumes for external storage SAN and NAS.

## Policy-Driven Control Plane

In the VMware software-defined storage model, there is a bridge between the infrastructure and applications called the control plane, which helps dictate standardized management and automation between different tiers of storage. A virtual machine is the primary unit of management that is thought of as a convenient container for an application or application component. Policies are associated with application containers (VMs) and carried out by the control plane.



Through SDS, storage classes of service (which are types of storage classes designed for different use cases) become logical entities that can be completely controlled by software and carried out through policies. Policy-driven automation simplifies provisioning on a large scale, enables dynamic control over individual service levels for each virtual machine, and ensures that all legal requirements (referred to as “compliance”) are met throughout the life-cycle of the application (from its beginnings as a concept to its last use).

Aspects of the policy-driven control plane (“perspectives”) are visible to specific roles in a typical IT environment: application administrator, storage administrator, data protection administrator, business operations, and so on. This wider access wouldn’t be possible in a typical storage environment, where each type of storage array has its own separate management tools that can’t create multiple perspectives for different team members.

The policy-driven control plane can be programmed through public application programming interfaces (also known as “API”. APIs will be discussed in section 5.4). The APIs are used to control policies through scripting and cloud automation tools which in turn enable self-service consumption of storage for application “tenants”. (Where a software application is used by a group of people, each of those people is called a “tenant”.) VMware’s storage policy-based management (SPBM) enables the policy-driven control plane. SPBM makes possible management over external storage such as SAN and NAS through vSphere virtual volumes and over X86 storage through vSAN. We will be looking at SPBM in more detail shortly.

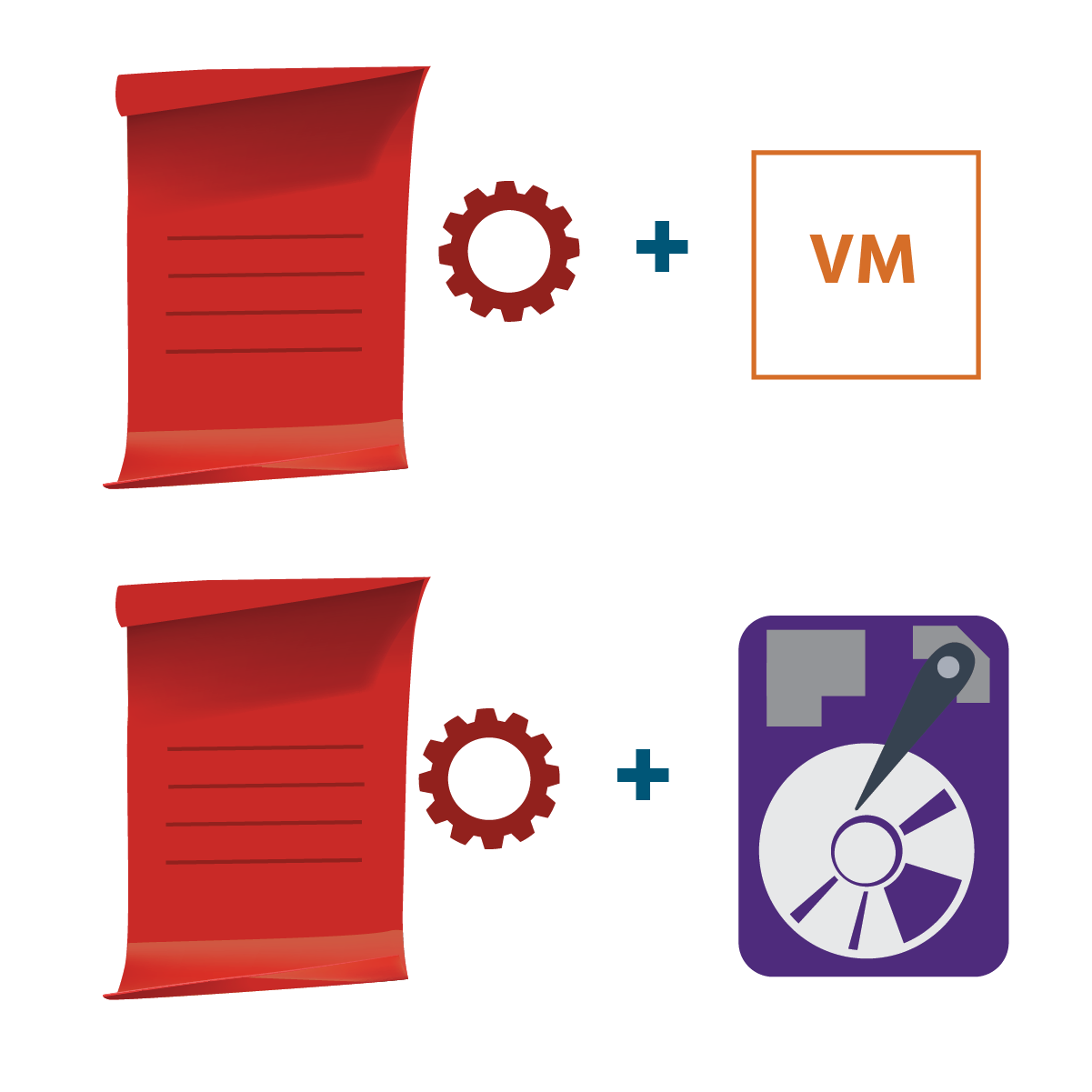
We spoke briefly about “converged” infrastructure (where traditionally individual technology disciplines merge) in section 4, and we’ll be looking at “hyper-converged infrastructure” (HCI) in section 5. For now, it would be helpful for you to know that HCI brings innovation to the management layer, allowing application requirements (not hardware resources) to be the priority in storage decisions.

## Storage Policy-based Management

As a storage administrator in an organization that uses a conventional storage model, you might find that your work involves more guess-work and over- and under-provisioning than you’d like. Say, for example, you’d like to make some applications available. You start by guessing what those applications will need in terms of storage - how much capacity will they require, and what about performance and protection? Next, you purchase hardware. Then you allocate the hardware into various predefined resource pools. Only after that, are the applications ready for use. But what if one of the applications has more needs than a particular resource pool can provide? You find yourself having to spend more time moving the application to a resource pool that can meet its needs.

This is just one of the many scenarios thrown up by the traditional storage model.

Storage policies, and the automation that a hyper-converged infrastructure gives you access to, eliminate this inefficiency. With VMware vSphere storage policy-based management (SPBM), policies can be set for any data service. SPBM automates the provisioning and monitoring of the services based on these policies, identifying available datastores that precisely match each application’s needs. When needs change, SPBM automatically adjusts storage resources to meet the new requirements. You can apply a policy either to a virtual machine or to individual disks.



**Note**

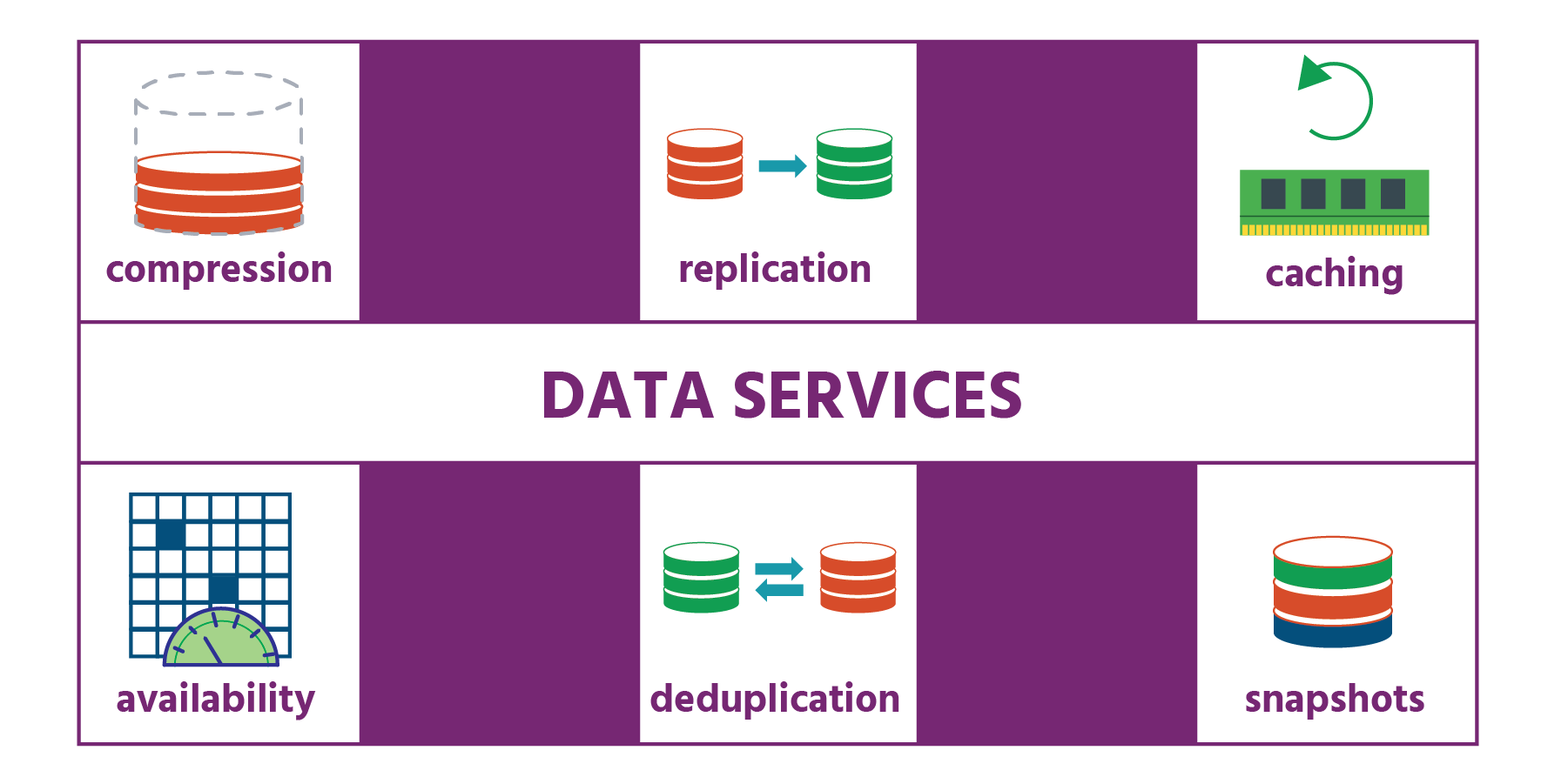
The SPBM is performed from the vCenter server.

The vSAN default storage policy is compatible only with vSAN datastores. If you don’t select any other vSAN policy, and set the virtual machine storage policy field on a selected storage page to “none”, the default storage policy will be applied to all virtual machine objects. If the policy that was assigned during virtual machine provisioning does not include rules specific to vSAN, the default storage policy is applied. Because the policy is compatible with any vSAN datastore in the vCenter server, you can move your virtual machine objects provisioned with the default storage policy to any vSAN datastore managed by the vCenter server in the vSphere environment.

If you have the “StorageProfileView” privilege, you can edit the default storage policy using the esxcli command. However, VMware highly recommends that you do not edit the settings of the default storage policy. Although you cannot edit the name and description of the default storage policy or the vSAN storage provider specification, all other parameters are editable. (**Note** that when you edit the default policy, you should repeat the process on every host in the vSAN cluster.)

You can clone the default policy and use it as a template to create a user-defined storage policy. You cannot delete the default policy, but if a user-based policy is created and made the default policy, then the original default policy can be deleted.

## Virtual Data Services



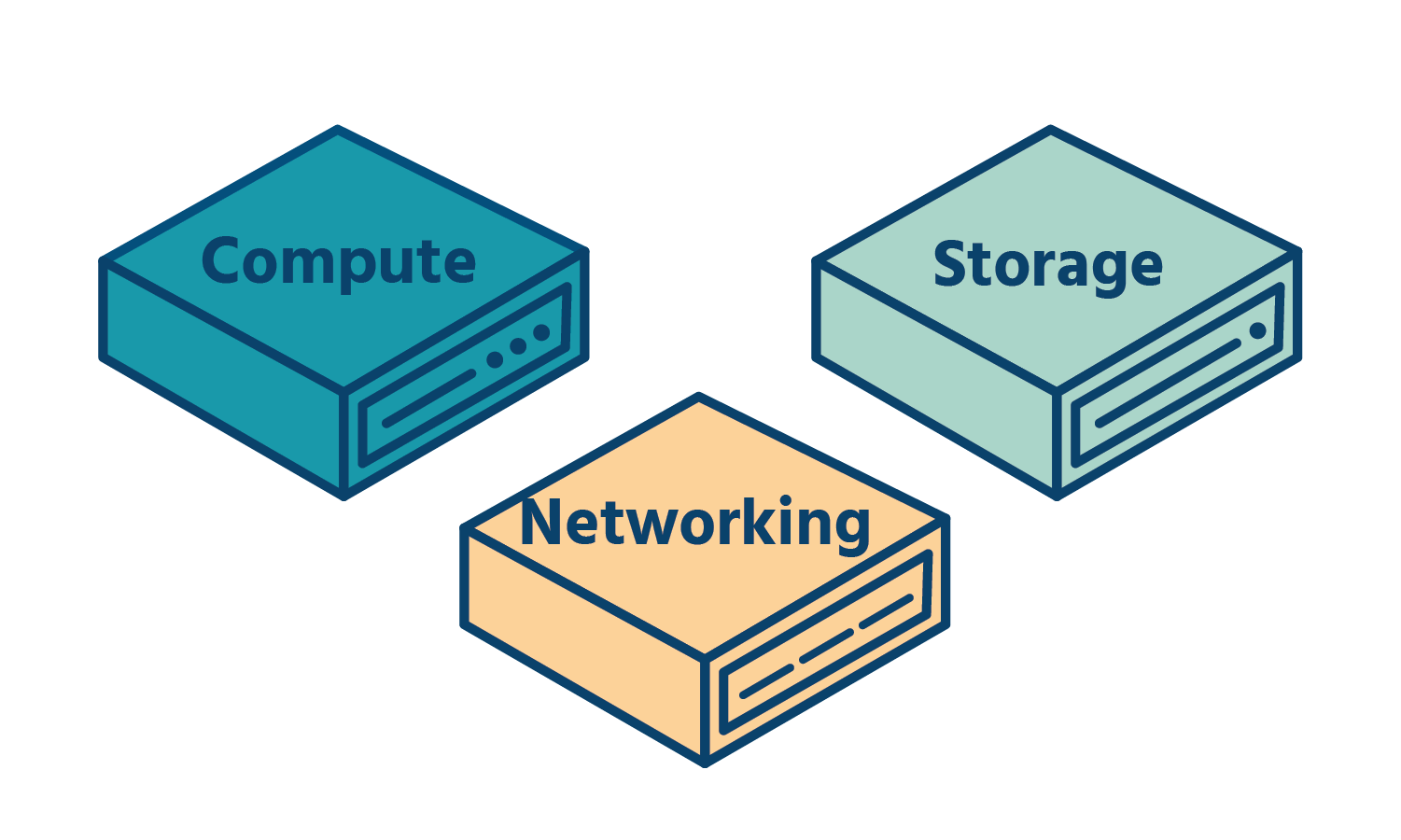
“Data services” is a catch-all phrase for all the interesting things that storage does: snapshots, cloning, replication, deduplication, caching, tiering, encryption, archiving, compliance, searching, and more. Data services represent much of the “intelligence” found in storage arrays today. Virtual data services are those that are completely abstracted from the underlying storage medium and are applied to precise application boundaries.

In the VMware SDS model, data services are composed and applied along with precise application container (i.e., virtual machine) boundaries — no more, and no less. Exact combinations of data services can be dynamically created and changed as needed. The implementation of these data services may be done by the storage array (as is the norm today), or by a newer class of data services that operate entirely in software.

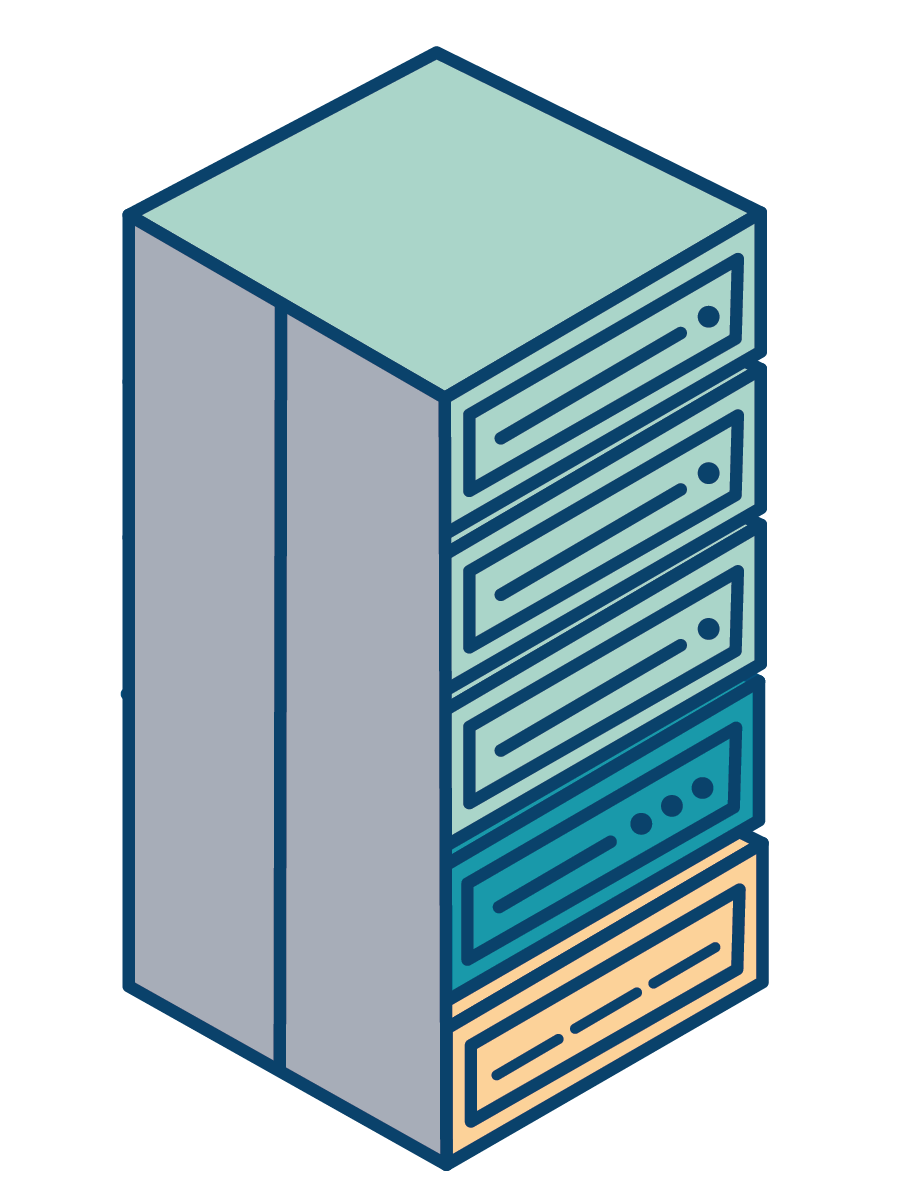
For storage services implemented in software, the resources used to provide storage services (compute, memory, network, disk, flash) are also managed by the control plane, providing the opportunity for dynamic sizing of resources used for storage services.

## Hyper-Converged Infrastructure

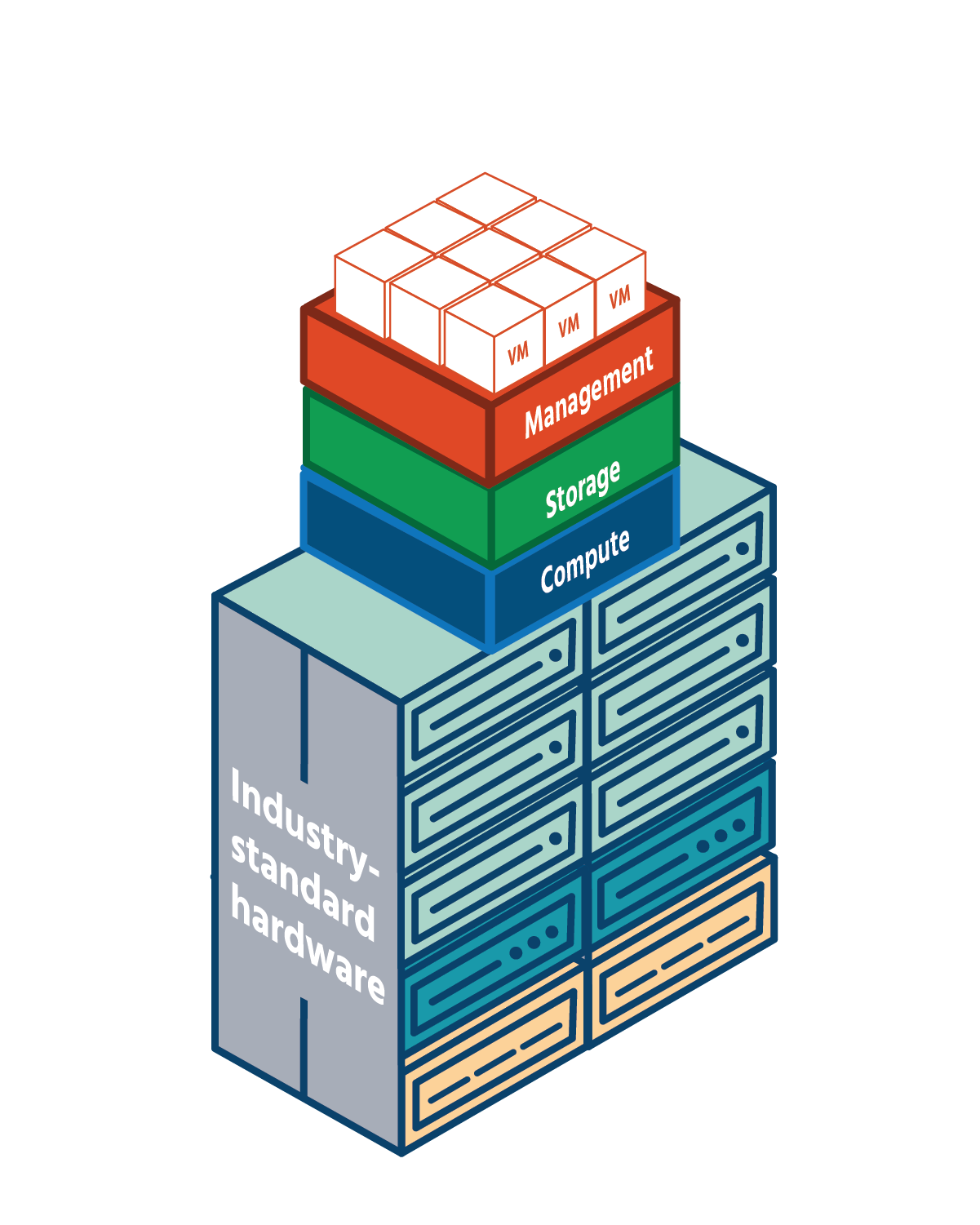
First, there was storage management that relied on fixed lists of service requirements (which, once implemented, could not be changed), proprietary hardware, management software from multiple vendors, and specialist staff. It was expensive, wasteful, and overly-complex.



Then came converged infrastructure which brought compute, storage, management, and networking together, integrating the infrastructure’s overall management and eliminating the physical boundaries. But it often involved separate systems and management tools, and preconfigured hardware that could not be adjusted to suit new needs.



Now, there is hyper-converged infrastructure (HCI). With HCI, compute, storage, networking, and management are integrated and all run as software – on the hypervisor – on a virtualized physical infrastructure of standard (i.e., not proprietary) servers, with common management tools. Efficiency is improved and delivery times are reduced.



Shared storage that behaves just as storage provided by SAN or NAS devices does, is created by the ESXi server which pools together direct-attached storage devices (not external storage systems, as with storage virtualization) from across the cluster.

A hyper-converged infrastructure can be achieved in one of two ways. Most commonly, third‐party storage software is run in virtual machines that sit on top of a hypervisor. While this method does achieve hyper-convergence, it results in the use of too many resources, as well as reduced performance and less than optimal integration with the current environment.

The second, more innovative, method – which is the method used by VMware – is to build the storage software into the hypervisor itself. Whereas in the first method, convergence takes place on the hypervisor, in this second method, convergence takes place inside the hypervisor.



Because of the increased performance, decreased complexity, and reduced cost that HCI enables, demand for it continues to grow, especially as developments in technology make it a more and more attractive option. VMware-powered hyper-converged infrastructures help organizations transition seamlessly and better position themselves for the future.

## Benefits of an HCI Model



It doesn’t take much to fully-realize the many benefits of a hyper-converged infrastructure. Start by unlocking the general benefits of virtualization (speed, savings, and simplicity, to name just three) with an industry-leading hypervisor. Many of the Fortune 100 companies – and more than half-a-million other companies! - use VMware virtualization technologies. Continue with hyper-converged storage, ideally with your storage software built into the hypervisor. Continue on with a single set of management tools (preferably with an interface that you’re already familiar with) for compute, network, and storage. And finish with the industry-standard (i.e., not proprietary) hardware you already have or know. You’re all set! The following are just some of the benefits you’ll enjoy with HCI.

You’ll use fewer resources. This applies especially if you use the built-in method of hyper-converged storage, as virtual CPUs won’t be needed for each of the VMs that you’re not using for your storage software.

You’ll have better performance and lower data-processing and -transmission times (known in the industry as “latencies”).

Your system will be simpler, more streamlined. With one integrated software stack for compute, networking, and storage, you’ll have far fewer physical pieces of hardware to deal with. Add automation provided by policy templates, and you have a system that’s less complex and easier to run.

HCI will save you money. Using your existing x86 (industry-standard) infrastructure means no new purchases will be needed, and no time will be lost learning how to use new hardware. Even if you do have to purchase hardware (which would be “capital expenditure” or CapEx), it can be off-the-shelf, so it will still be much cheaper than buying proprietary hardware and maintaining it (“operational expenditure” or OpEx). HCI also saves you money by improving the efficiency of your storage: there’s no more guesswork about what storage services applications will require, and therefore no more (or vastly reduced) waste. And with automation, you need less staff – another cost-saving.

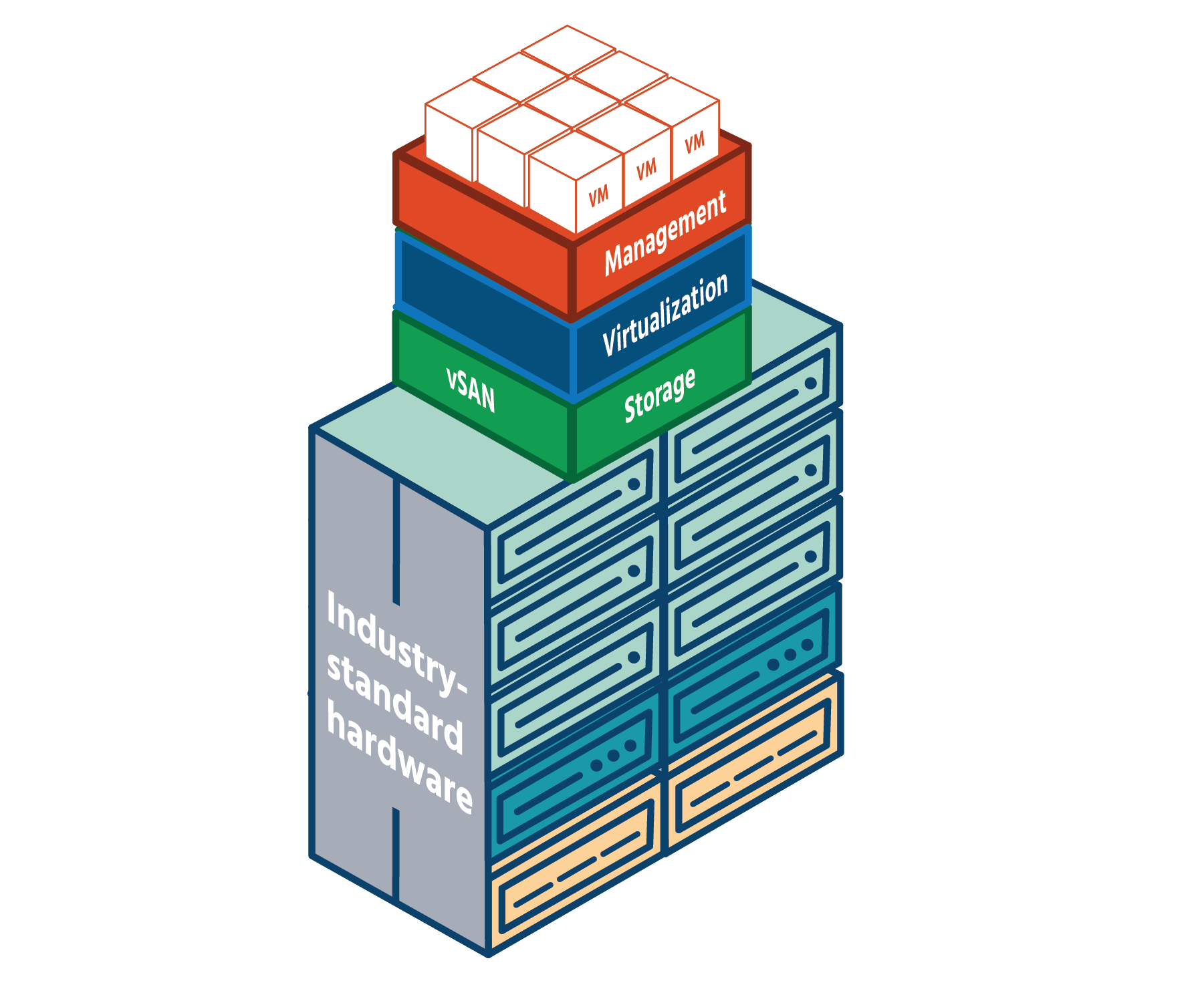
Security is another area where HCI will bring unique benefits to your system. Modern HCI provides native (i.e., built-in) software-based security, removing the need for separate, specialized hardware. And this built-in security meets the latest legal requirements for many industries and sectors.

With the industry-leading hypervisor, HCI gives you “five nines” availability – in other words, your system is up-and-running 99.999% of the time. With this high availability, the financial and reputational costs of system downtime are avoided.

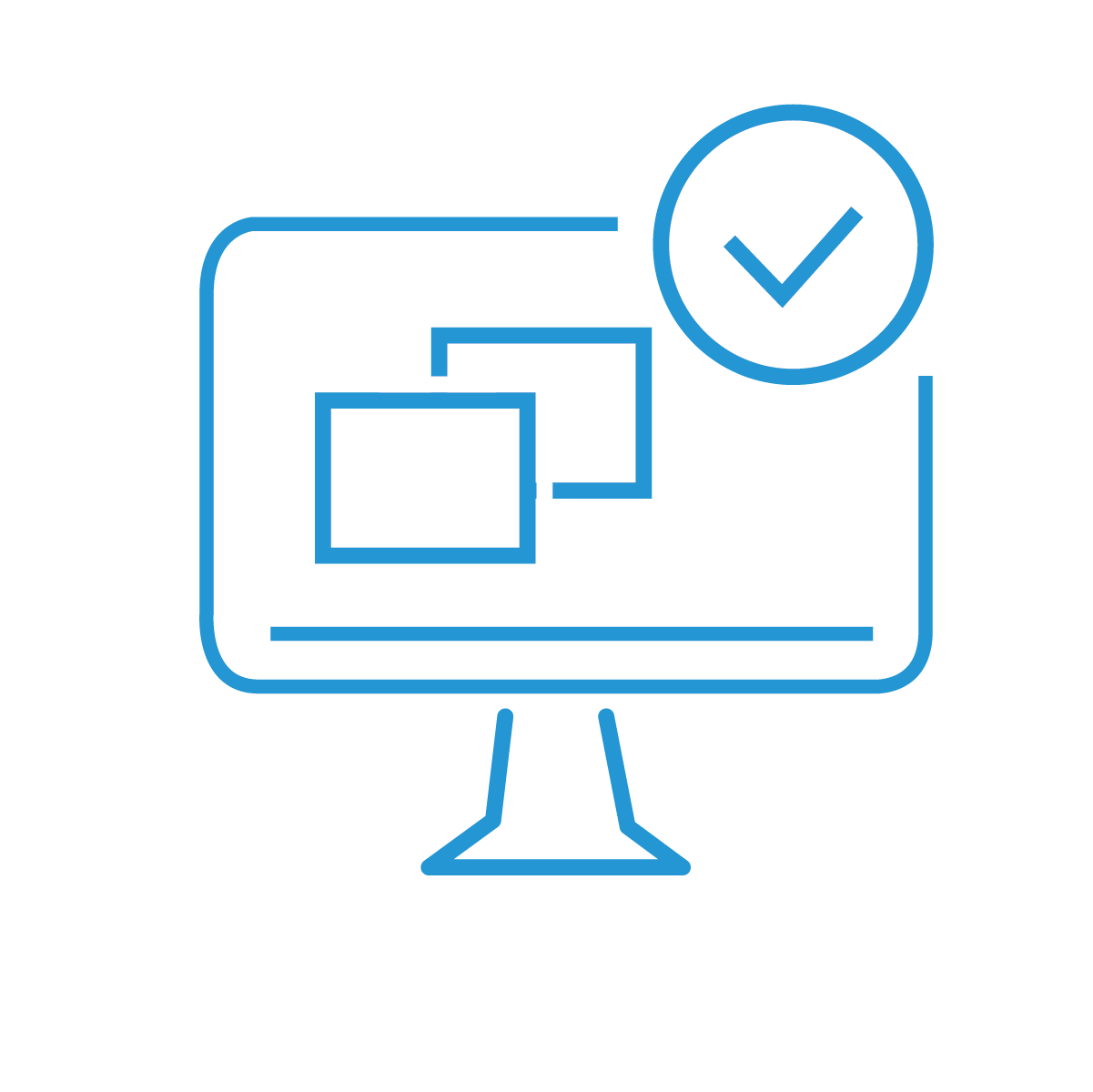
HCI adds agility to organizations. Using industry-standard hardware makes it easier to buy more (“scaling out”) or upgrade what you already have (e.g., by adding more memory or processing power - “scaling out”). This, plus the ability to change policies with a few clicks, enables organizations to respond rapidly and with minimal fuss to changing requirements and a changing business environment.

This agility is much needed with the new generation of applications, the requirements of which quickly change. VMware’s HCI technologies already integrate seamlessly with some of the most popular of these new applications – for containers and the cloud, for example – meaning its customers are well-placed for the future.

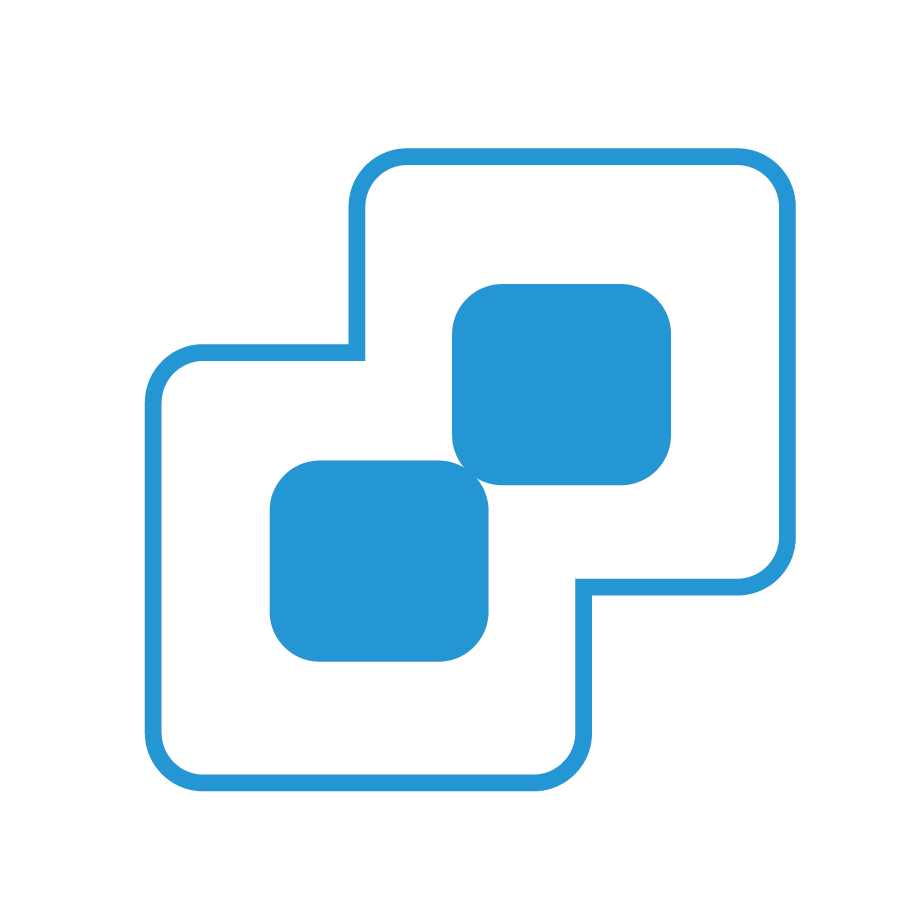
## Software Stack Components



VMware’s hyper‐converged infrastructure software stack is made up of three industry-leading solutions:



**VMware vCenter Server** is an advanced and unified server management software that provides a centralized platform for controlling your VMware vSphere environments, allowing you to automate and deliver a virtual infrastructure – all from a single console.



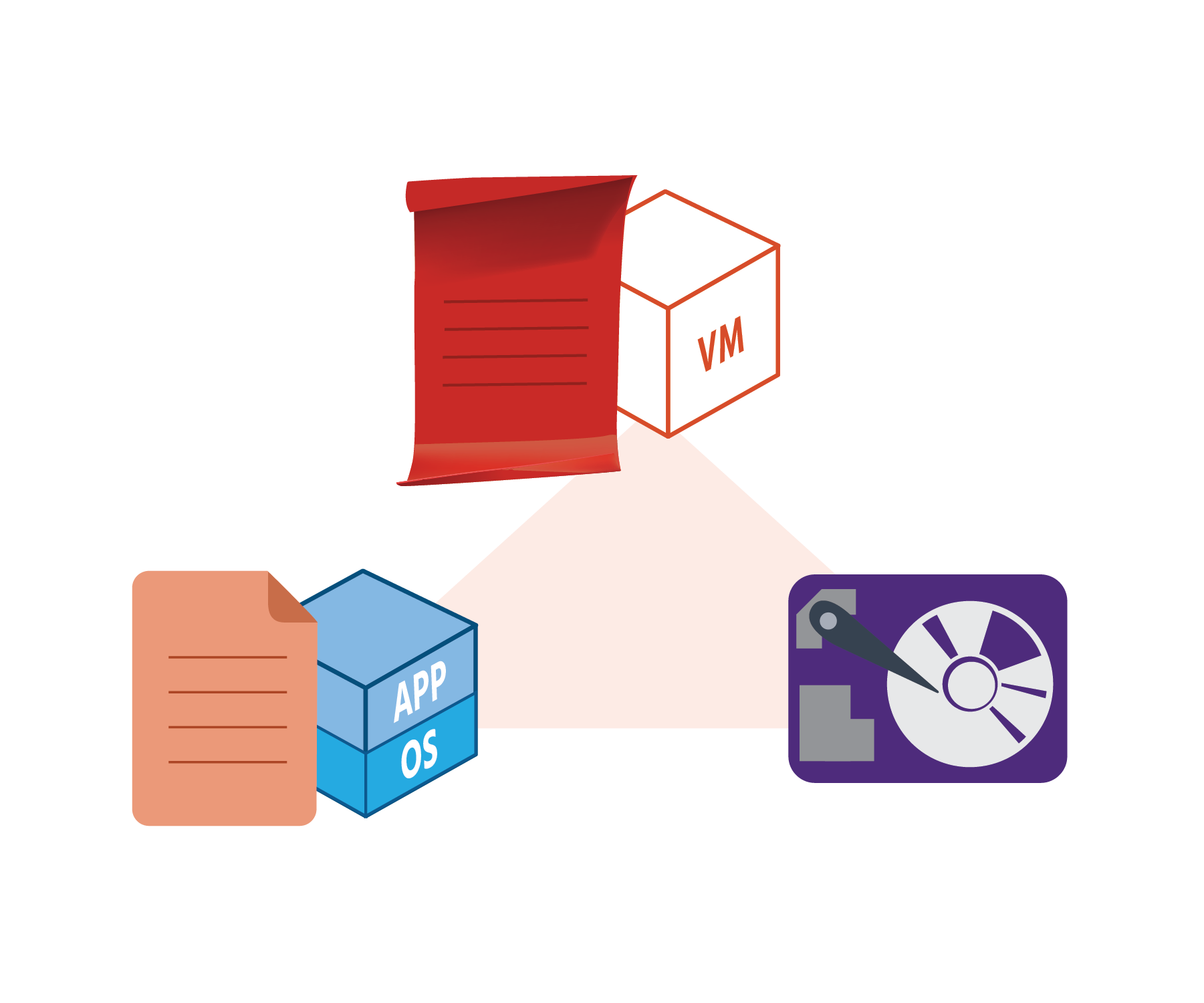
**VMware vSphere** is the world’s leading server virtualization software and the heart of a modern software-defined data center (SDDC). It helps hundreds of thousands of organizations across the globe run, manage, connect, and secure their applications in a common operating environment across clouds. New advanced security capabilities, fully integrated into the hypervisor and powered by machine learning, provide better protection and response times for security incidences.



**VMware vSAN** is the only vSphere-embedded, flash-optimized storage for virtual machines and containers. In combination with vSphere, it allows you to manage compute and storage with a single platform. It seamlessly joins all storage devices across a vSphere cluster into a shared data pool. Using commodity x86 server components, vSAN-powered hyper-converged infrastructure lowers storage costs 40% or more compared to traditional server and storage architectures. There’s no need to deploy or maintain separate arrays and storage networking hardware with VMware vSAN. Its policy-based management removes the burden of provisioning and modifying numerous LUNs and data services. It provides the simplest path from server virtualization to hyper-converged infrastructure.

## Storage Policies Management

Virtual machine storage policies are a set of rules that define how vSAN stores the files for the VM. Storage policies capture the storage characteristics that are required by the virtual machine home files and virtual disks to run various applications within the virtual machine.



Storage policies are created before the deployment of virtual machines using the VM Storage Policies interface of the vSphere Client. When you create a storage policy, you define placement and data service rules. The rules are the basic elements of the VM storage policy. Each individual rule is a statement that describes a single requirement for virtual machine storage and data services. Within the policy, the rules are grouped in collections of rules or rule sets. The storage policy can be applied during any phase of a VM’s cycle. If a VM is cloned or migrated, the policy can be changed to apply a new one - otherwise, the previously-applied policy (if there is one) is applied. During the application of the policy, the SPBM mechanism will list which datastores are compatible and which ones are not.

A VM storage policy can include one or several reusable and interchangeable building blocks, called storage policy components. Each component describes a particular data service to be provided for the virtual machine. You can define the policy components in advance and associate them with multiple VM storage policies. Storage policies can be changed whenever required and chosen based on storage capabilities and requirements.

A policy is applied to a VM and tells the datastore how to store an object. The object is created across ESXi hosts and disk groups in a manner that satisfies the policy.

## Application Programming Interfaces



If you've ever downloaded a movie, or posted a message on social media, or booked travel online, chances are you've probably been using APIs – application programming interfaces. They work behind the scenes to make our digital experiences possible – in just a few clicks.

To visualize how APIs work, imagine you're in a restaurant looking at the menu. A waiter comes and takes your order to the kitchen, and returns soon afterward with your meal. The menu offers you a selection from which to choose. The waiter takes your request to the kitchen and then comes back with what you chose.



In an actual app (a movie-streaming app, for example), the menu would be the user interface (UI) from which you choose your movie. The waiter would be the API that takes your movie-order. And the kitchen would be the database(s) or website(s) to which the API-waiter goes to get your movie and bring it back to you. The API is the software intermediary that allows you (one party) to interact with the movie-streaming service (the other party).

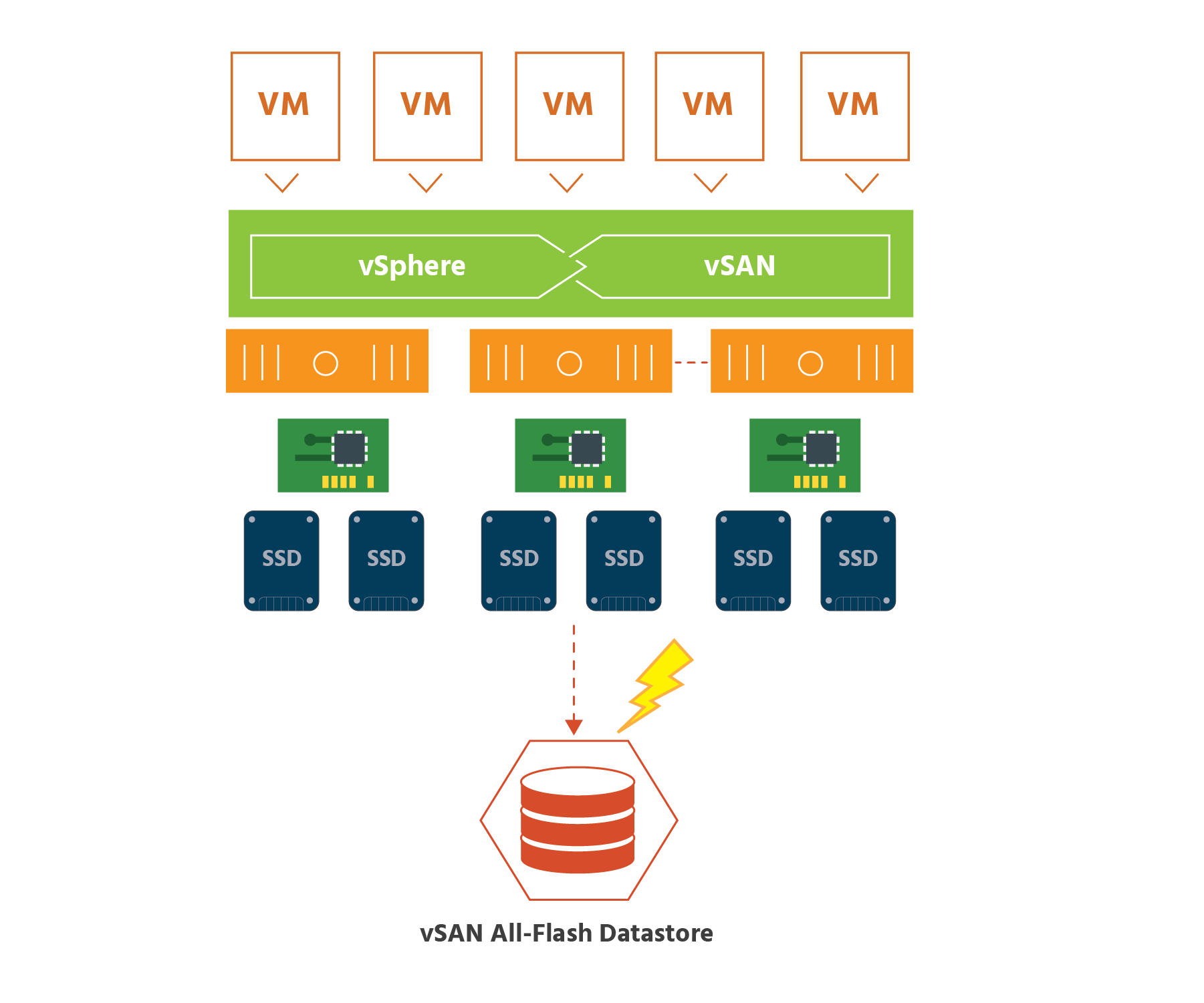
Your choice (of meal, movie, etc.) is drawn from the "assets" of the organization you're connecting with. In the case of the restaurant, the different menu items would be their assets; in the case of the movie-streaming service, the different movies would be their assets. Now in actual fact, it's only really developers who have direct contact with APIs. APIs give them access to assets (such as software components or pieces of code that have been made freely available by their owner-organizations) which they can use to create new software or apps without having to start from scratch. The APIs allow the new software that's being created (one party) to interact with the assets of the owner-organizations (the other party).

There are public (or "open") APIs (as mentioned briefly in section 4.2) and private APIs. With public APIs, proprietary software or a proprietary web service is freely shared with users outside the owner-organizations. Private APIs are used internally within a single owner-organization (and any other organization or team to whom they chose to give access).

In a hyper-converged infrastructure environment, the policy-driven control plane uses APIs to provide seamless integration between data center management tools. APIs allow maximum flexibility between various layers for servers and storage devices to communicate effectively. This allows for faster HCI deployments, as opposed to deploying manually, which can take more time. Only critical functions are displayed to the developer, which helps abstract the layers of virtualization. Not only do developers benefit from APIs but also system administrators due to having the ability to use scripts to administer systems dynamically. This helps cut downtime for management compared to a graphical management dashboard where tasks are more than likely done manually. Requesting information from systems can also be done faster through APIs, by only displaying information that was asked by an administrator.

## Hyper-Converged Storage vSAN

There are two types of vSAN clusters. An “all-flash” vSAN cluster provides extremely high performance and is made up of devices such as solid-state drives (SSDs) and Peripheral Component Interconnect Express (PCI-E) devices that only use “flash memory” - storage with no moving parts. All-flash vSAN clusters provide up to 150,000 input/output operations per second, per host, (IOPS being the unit that’s used to measure the performance of storage devices and networks) for applications and services requiring extreme performance.



The other type of vSAN cluster, a “hybrid” vSAN cluster, combines server-attached flash storage devices used for caching (i.e., temporary data-storing) with magnetic drives used for capacity to create a single storage pool called a “vSAN datastore”:



A hybrid vSAN can provide as much as 40,000 IOPs per host.

vSAN is a hypervisor-converged storage solution for virtual machines. As we learned in section 2.7, it virtualizes the physical storage resources of VMware ESXi hosts and turns them into a pool of storage. It then uses the pooled storage of the cluster to allocate storage resources to virtual machines based on their policies.

vSAN also simplifies storage configuration and virtual machine provisioning.

vSAN is an enterprise-class (i.e., designed to meet the needs of larger organizations) storage solution for any virtualized application, including business-critical workloads. Its seamless integration with vSphere and the entire VMware stack makes it the ideal storage platform for virtual machines.

## vSAN Overview

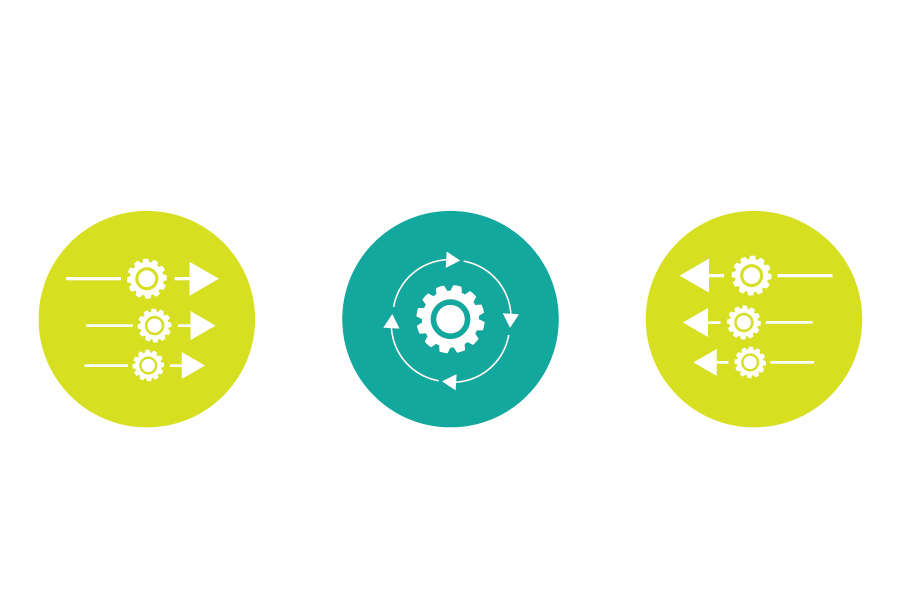


vSAN is VMware’s industry-leading hyper-converged SDS, providing enterprise-class storage to nearly two-thirds of organizations that have adopted hyper-converged infrastructure.

vSAN pools together a number of server-attached storage devices (SSDs, HDDs, and PCIs, for example) to form a shared datastore that is extremely reliable with no single point of failure, meaning that it is able to continue working even if a fault occurs somewhere in the system. It uses industry-standard, x86-based servers and at the same time is compatible with the newest developments in flash technology. It provides advanced data services – including software-based encryption – that virtual environments require.

As mentioned in section 6, vSAN can be configured as either all-flash or hybrid. It fits perfectly with the rest of the VMware software stack, allowing users to use a single, familiar set of management tools for compute, storage, and networking. vSAN is storage for virtual machines made simple.

## Attributes of vSAN



VMware vSAN supports storage administrators with easy, step-by-step guidance on how to create vSAN clusters. The guidance covers every aspect of the initial configuration as well as the steps necessary for adding additional hosts to a cluster. Once a vSAN cluster is up and running, it can easily be added to if more capacity and performance are needed - “scaled up” by adding new drives to existing hosts, or “scaled out” by adding new hosts, both without disruption. Adding new hosts would be adding more storage devices to the datastore.

VMware’s innovative approach to hyper-converged storage – building vSAN inside the ESXi hypervisor (as opposed to placing it in a VM on top of the hypervisor) – makes management simpler and removes the need for dedicated hardware and complicated networking and the risks they can carry.

Everything about vSAN has been designed to streamline storage operations and use the newest developments in flash technology to the greatest advantage. This means fast response times for applications and more virtual machines per physical host for administrators. A dedicated vSAN engineering team (“ReadyLabs”) partners with hardware vendors to do risk assessments, create tests and debug issues to ensure that hardware can be certified to run vSAN. And as we’ve mentioned a number of times, that hardware only needs to be industry-standard (not proprietary), giving users great flexibility of choice.

Deduplication (getting rid of unnecessary copies) and compression (making files smaller by removing unneeded data) massively decrease the amount of storage that needs to be used.

vSAN brings increased efficiency to updates, which are an important part of day-to-day operations in a datacenter and can be time-consuming. The VMware Update Manager (VUM) centralizes all updates in a single location and checks for issues after updates.

As mentioned earlier in the course, storage policy-based management allows administrators to create policies (for example, about storage availability levels or consumption levels) to precisely control storage services. These policies can be for individual VMs or for the individual objects inside them. As an organization’s requirements change, these policies can easily be modified with just a few clicks and no downtime.

We discussed APIs a few sections ago, and vSAN works extremely well with public APIs, as well as with software development kits (SDKs). A software development kit is a collection of tools used to create applications for a specific operating system, platform, or device. There is an Android SDK and an iPhone SDK, for example. Many companies provide free SDKs for download from their websites, in the hope that developers will create applications for their products. SDKs also allow such companies to gather usage data.

vSAN also supports advanced automation and scripting.

With the cost of data breaches rising, protecting data in all its forms (whether “in use”, “in transit” or “at rest”) becomes increasingly important. vSAN Encryption is the industry's first native (i.e., built right into the hypervisor software) HCI encryption solution. With a couple of clicks, vSAN Encryption can be enabled or disabled for all items on the vSAN datastore, with no additional steps. It can run on any vSAN-certified drives and devices. And there’s no need to use specialized and expensive self-encrypting drives (SEDs), as there is with the other HCI solutions that offer encryption.

vSAN uses VMware vSphere Replication to “replicate” data – i.e., to continuously copy data from one server to another (in another location, if needed). The continuous nature of replication means that replicated data is kept up-to-date and is synchronized with its source. A vSAN datastore can either contain the source virtual machines from which the original data comes, or be the target (or destination) for replicated data. In the event of a disaster, this replicated data can be recovered so that businesses can continue with minimal disruption.

vSAN includes a snapshot-based feature that is SPBM-driven (Storage Policy Based Management) to capture the entire state of a virtual machine at the time you take the snapshot - the state of all the VM's disks, the contents of its memory, and its settings for virtual machine storage replication. When you revert to a snapshot, you return all these items to the state they were in at the time that you took the snapshot. So, if you want to save the current state of the virtual machine, so that you can revert to that state if you make a mistake, use vSAN to take a snapshot.

If you want to make a copy of a virtual machine for separate use, use vSAN to create a linked clone. A linked clone is a copy of an existing virtual machine. The existing virtual machine is called the “parent” of the clone. When the cloning operation is complete, the clone is a separate VM with its own MAC address and ID — though it may share virtual disks with the “parent” VM. Changes made to a clone do not affect the “parent” VM, and changes made to the “parent” VM do not reflect in a clone.

vSAN also has a “Quality of Service” (QoS) feature that uses a policy setting to limit the number of input/output operations per second (IOPS) an object can consume. (We will discuss objects in section 6.4.) The term “noisy neighbor” is often used to describe a VM or virtual disk (a “workload”) that monopolizes available I/O or other resources, which can negatively affect other workloads or the performance of the vSAN datastore as a whole. If an IOPS limit is proving to be too restrictive, either the existing policy can be modified or a new policy be created with a different IOPS limit. Once the new or updated policy has been assigned to the VM, it will take effect just moments after the change is made.

## Cache Layer and Capacity Layer

vSAN architecture consists of two layers (or tiers). The first is a “cache layer” of devices that are used for read caching and write buffering. Read caching is temporarily storing data in order to be able to more quickly fulfill future requests to read that data. Write buffering, which takes place within the CPU, is holding data that’s being written from the cache to memory, so that the cache can focus on read requests.

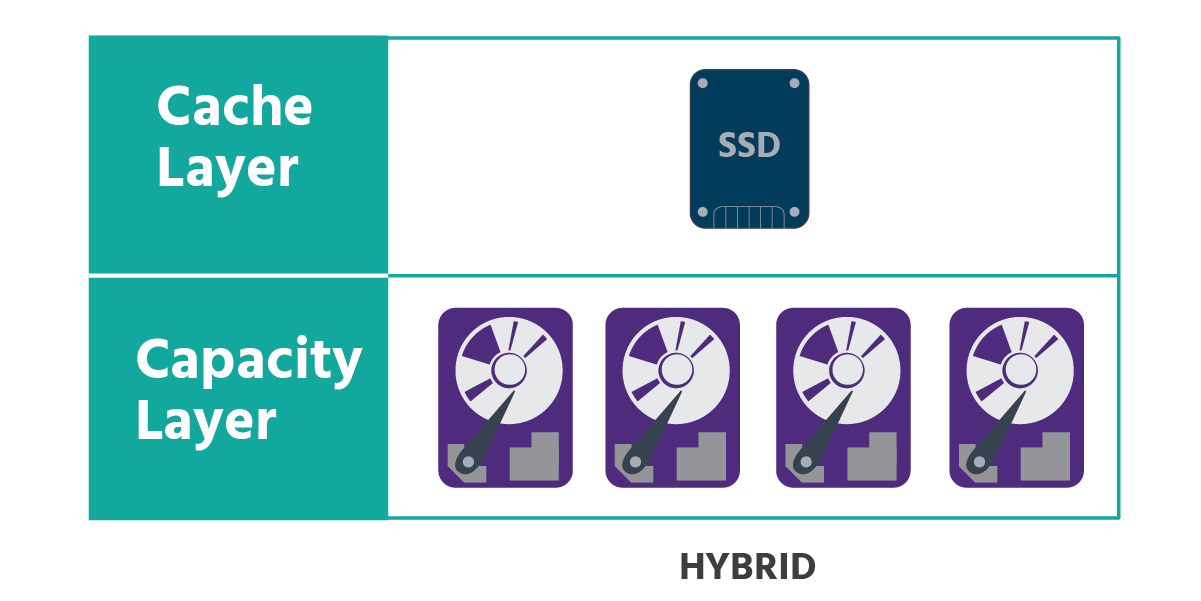
The cache layer only deals with “hot” data - data that’s frequently-accessed.

The second layer is a “capacity layer” of magnetic devices that are used for persistent storage (i.e., storage that retains data after it has been switched off). The capacity layer deals with “cold” data - data that’s less-frequently accessed.

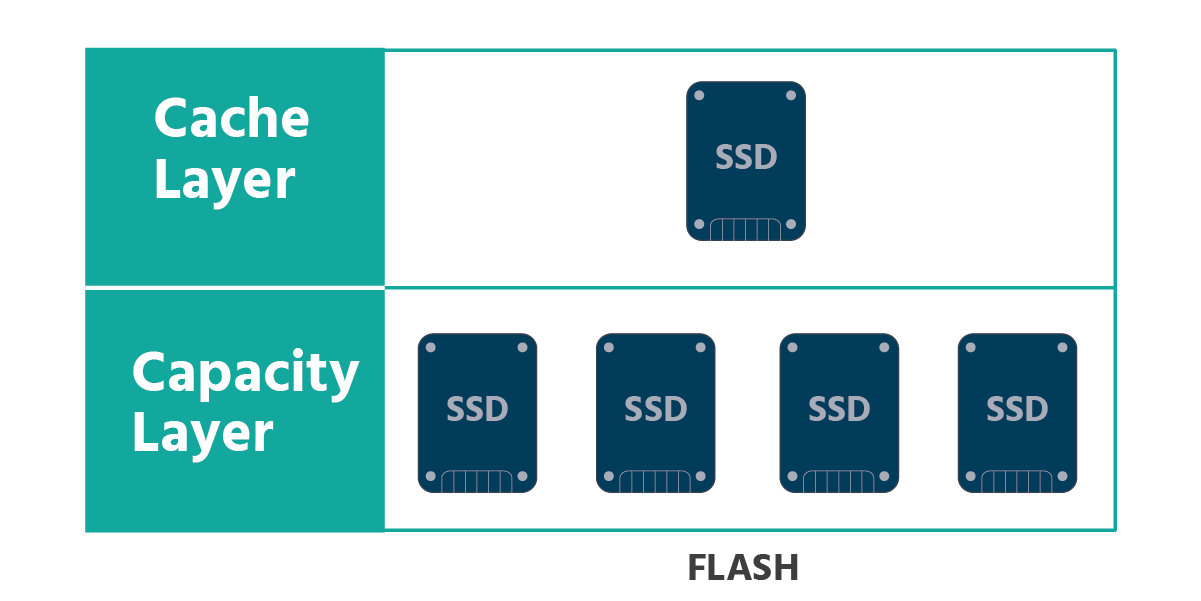
vSAN uses disk groups to manage the relationship between capacity (magnetic) devices and their cache tier. Each host that contributes storage in a vSAN cluster will contain at least 1 disk group. Disk groups contain at most 1 cache device and between 1 to 7 capacity devices. At most a vSAN host can have 5 disk groups, each containing up to 7 capacity devices, resulting in a maximum of 35 capacity devices for each host.

Whether vSAN is configured as hybrid (combining magnetic devices with flash devices) or as all-flash, the cache device must be a flash (SSD or PCIe) device.

In a hybrid configuration, the cache device is utilized by vSAN as both a read cache (70%) and a write buffer (30%). This dramatically improves performance.



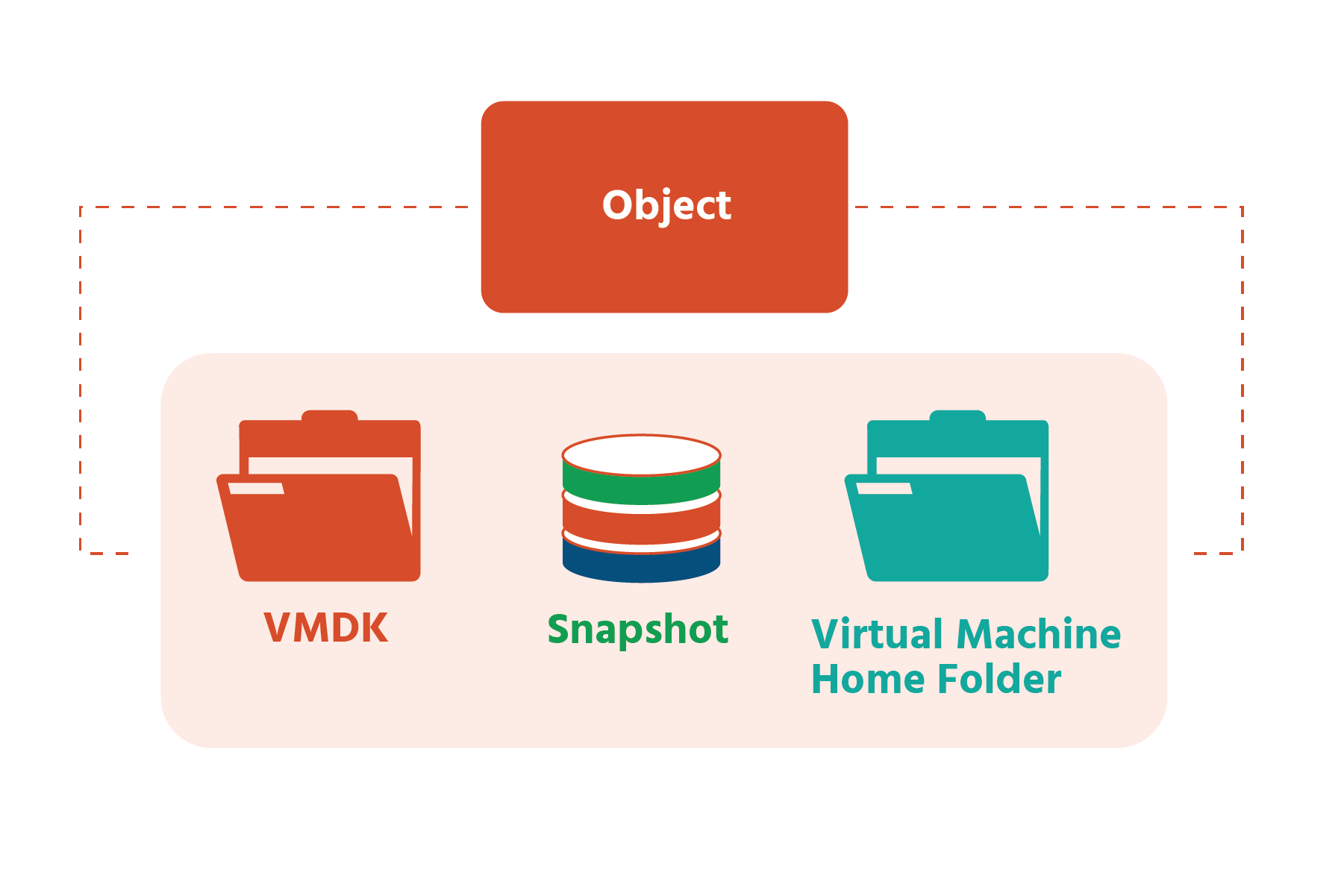
In an all-flash configuration, 100% of the cache device is used as a write buffer.



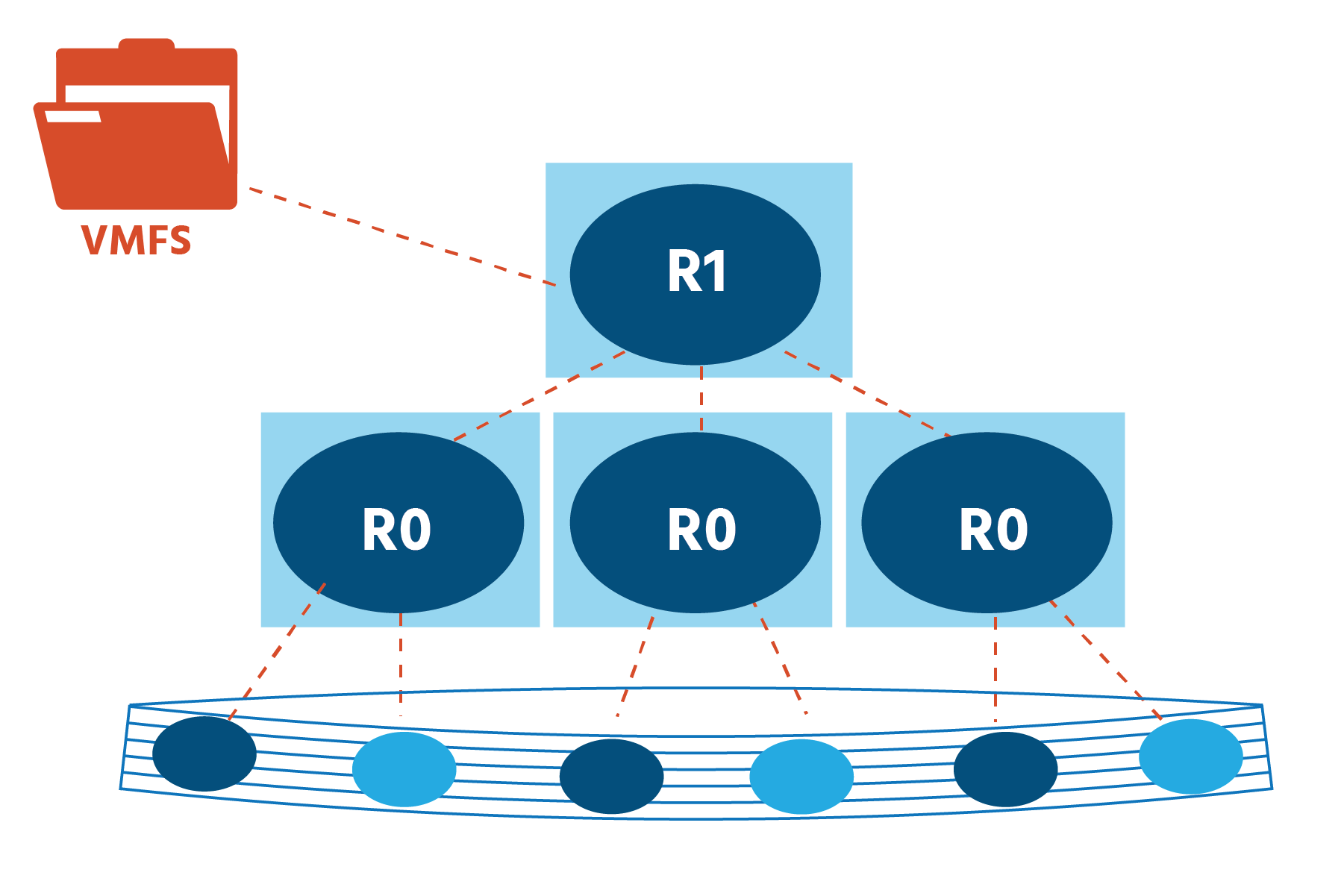
## Object and Component Layout

Virtual machines have five different types of objects: VM Home, Virtual Machine Disk (VMDK), VM Swap, snapshot, and memory. VM Home stores all the virtual machine configuration files. VMDK is the virtual machine disk file. VM Swap is created when virtual machines are powered on to reduce the amount of memory the host needs to reserve for VM operations. In addition to the memory a host needs to give to a VM for the VM’s consumption, the host must have some memory allocated to the process that controls the VM. This is what VM Swap applies to. By default, the swap file is stored in the same location as the virtual machine’s configuration file. Snapshots are copies of VMDKs taken at a specific point in time. Memory (or “vmem”) backs up the guest (i.e., virtual machine) main memory on the file system of the host (i.e., the physical hardware running the virtual machine). This file exists only when the virtual machine is running.

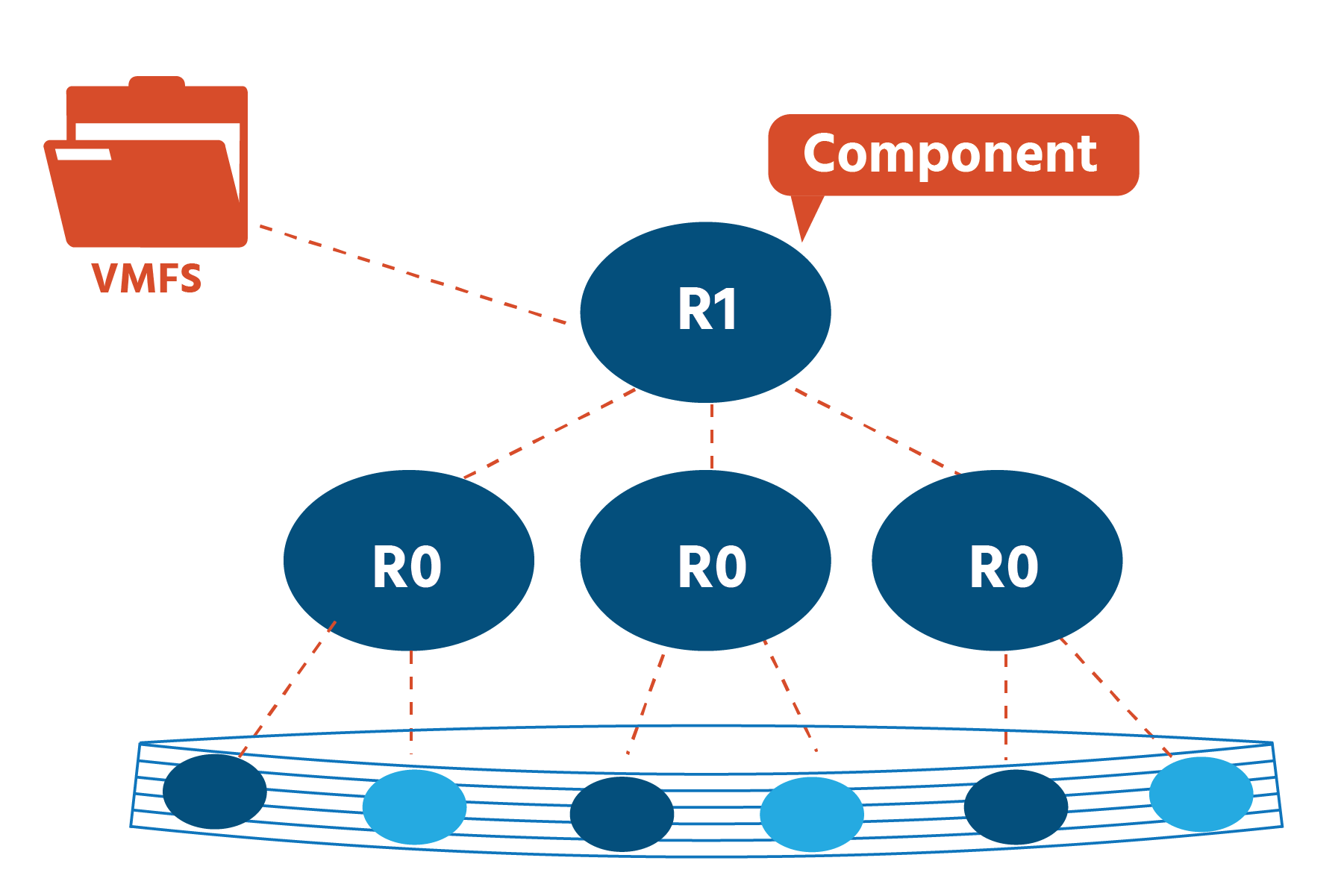
When a virtual machine is provisioned on a vSAN datastore, vSAN creates an object for each virtual disk. These objects are logical (i.e., not physical) volumes with data and metadata and are mirrored across the entire cluster. (See section 2 for a reminder of what mirroring is.) Mirroring takes place by default. vSAN can also use “erasure coding” (fragmenting data and storing it in different places - a.k.a. RAID 5 and 6) to provide more space efficiency, though this comes at the cost of an increase in CPU usage.



A vSAN cluster can store and manage tens of thousands of objects, and each object has its own “object tree”:

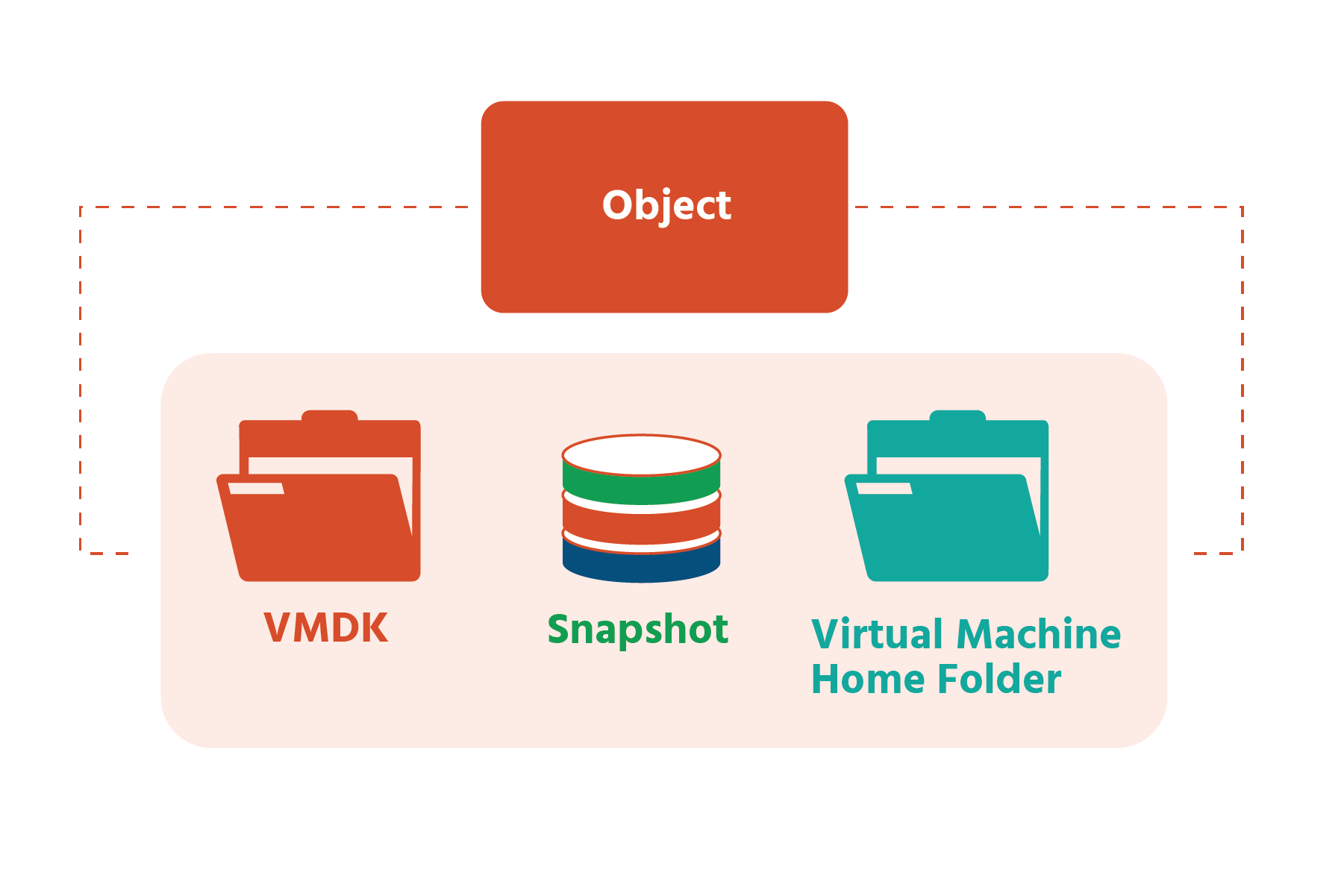


The objects are, in turn, striped into components. (See section 2 for a reminder of what it means to stripe data.) Each object consists of one or more components. The number of components that make up an object depends primarily on a couple of things: the size of the objects and the storage policy assigned to the object. The maximum size of a component is 255GB. If an object is larger than 255GB, it is split up into multiple components. In addition, how a policy is configured can increase the number of components an object is split into. These components are distributed across the hosts in a vSphere cluster, each component being a "leaf" of the “object tree":

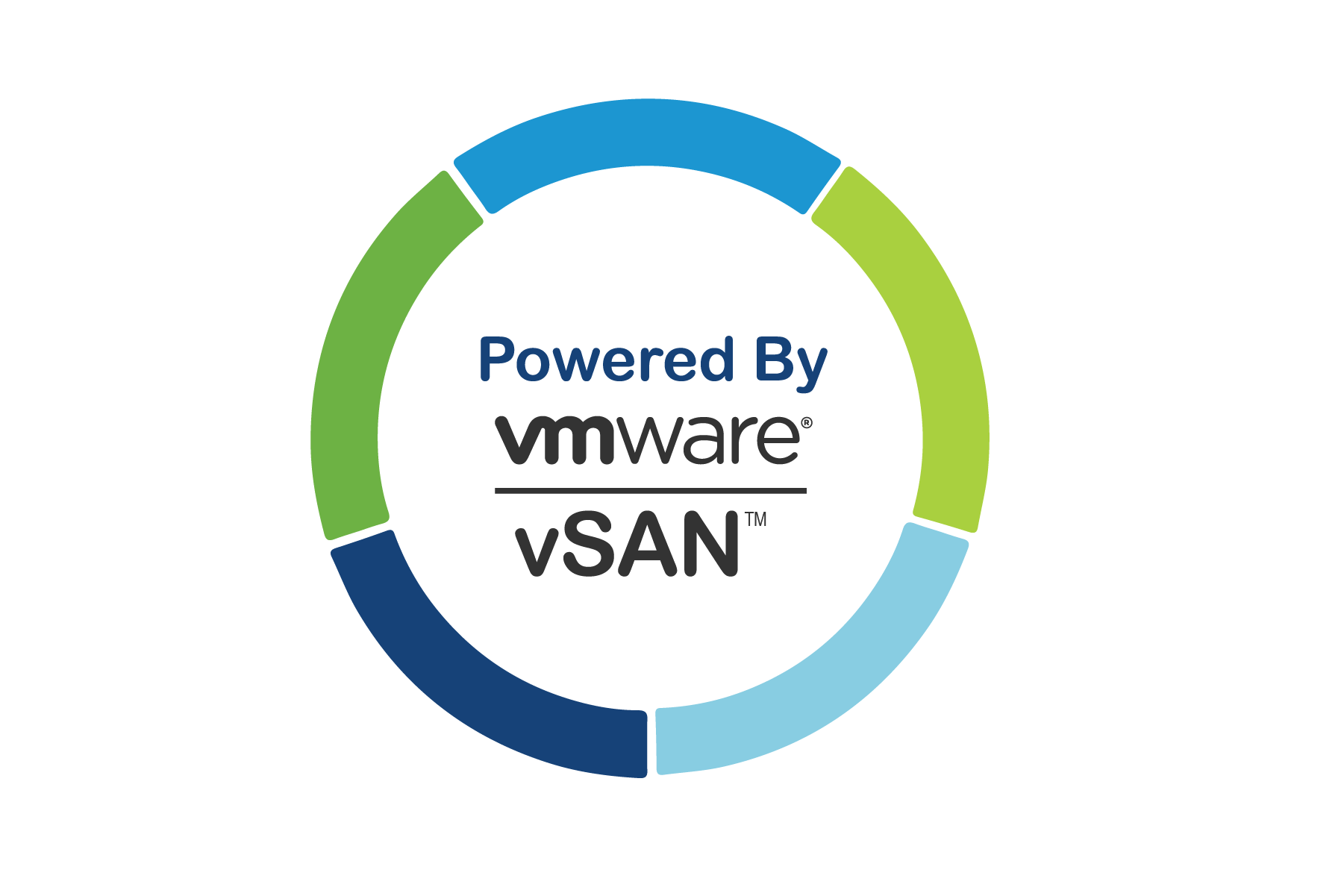


Objects and components are distributed and stored in distinctive combinations of disk groups within the vSAN datastore. This distribution forms the layout of the objects and components.

The VM Home directory object is formatted with VMFS so that a virtual machine's configuration files can be stored on it. "Replicas" are copies of objects and components, and the number of replicas determines how much storage is available. Disk performance, on the other hand, is determined by the number of devices across which each replica is distributed. This number is known as the "stripe width". Objects and components can reside on different hosts:



Benefits of a vSAN Environment



It’s our hope that having learned about the attributes of vSAN in sections 2.7 and 6.2, and about the hallmarks of hyper-converged storage in sections 5 and 6, you’ll already have a sense of the benefits of a vSAN environment:

* how vSAN is configured, provisioned, and managed; because vSAN is seamlessly embedded in the hypervisor, installation and configuration are done with a few clicks from the vSphere Client
* its flexibility and agility in meeting changes in demand at the drop of a hat (in terms of application requirements as well as easy scaling up/out of hosts)
* the cost savings that come from being able to use inexpensive industry-standard x86 hardware
* the higher performance and lower latencies, fewer resources and greater security, streamlined operations and “five nines” availability the HCI brings.

Bearing all this in mind, and the power that integrating storage with the hypervisor places at administrators’ fingertips, virtual machines have their ideal storage in hyper‐converged storage.