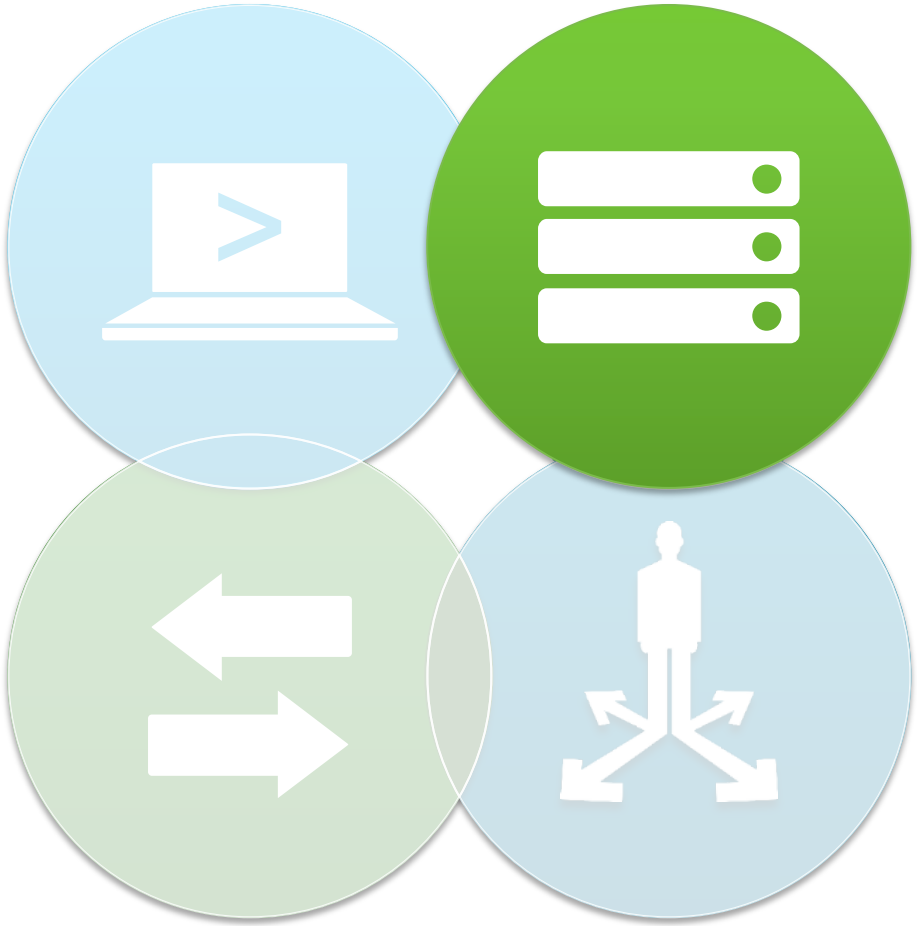
vSphere Virtual Volumes

Technical Walkthrough

Rawlinson Rivera, VCDX
Principal Architect | Storage & Availability

vmware®

The Software-Defined Data Center



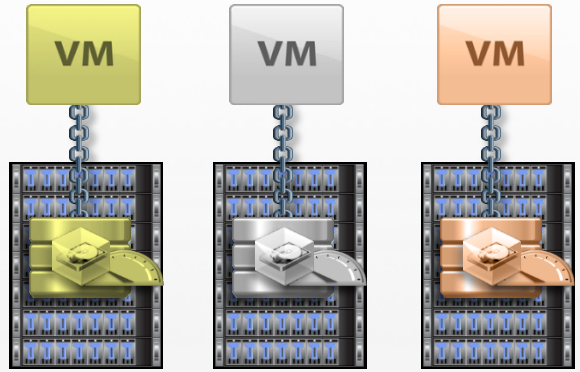
Transform storage by aligning it with app demands

Traditional Storage Architectures Can't Keep Up

Specialized Costly HW

- Not commodity
- Low utilization
- Overprovisioning

Device-centric Silos

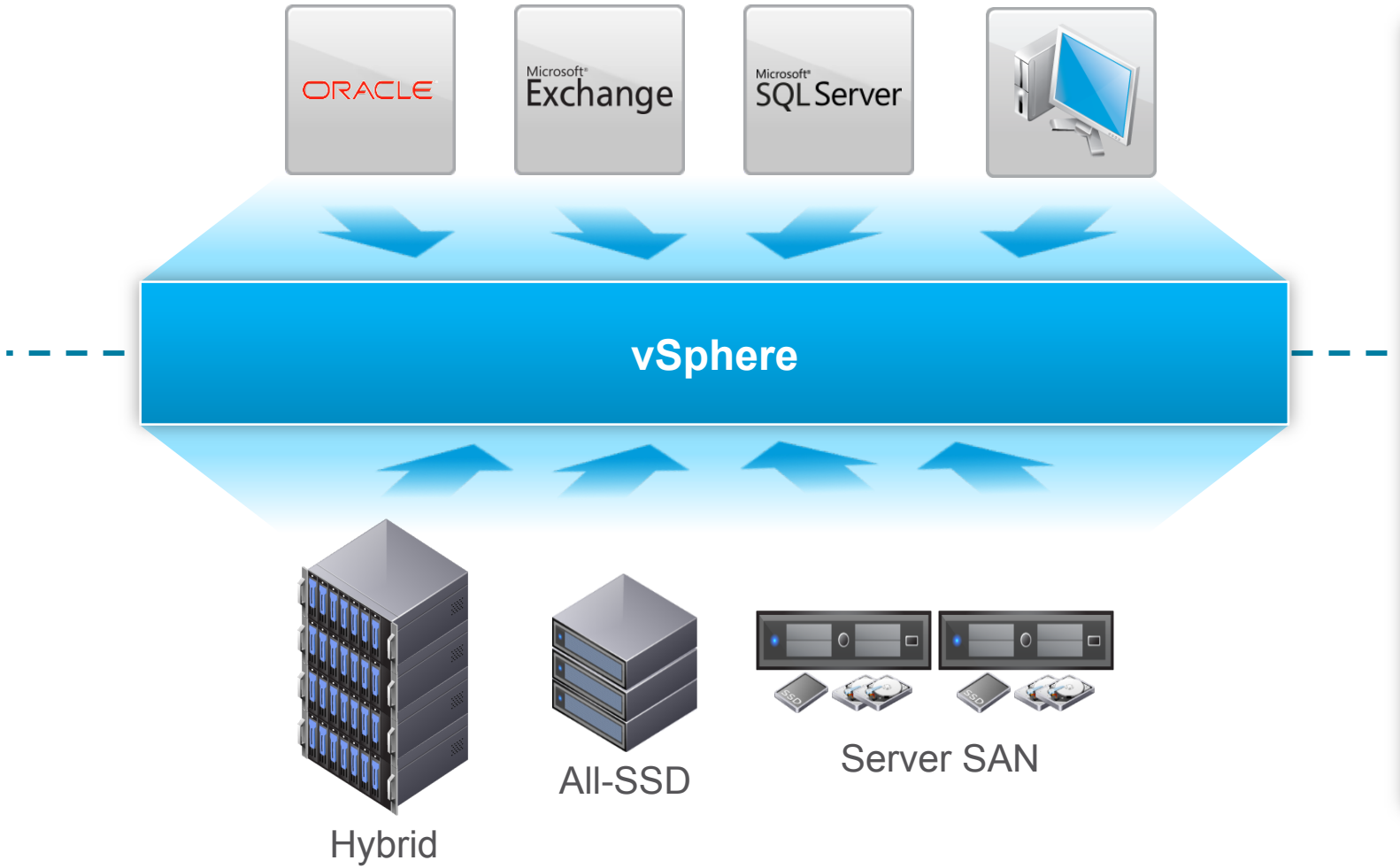


- Static classes of service
- Rigid provisioning
- Lack of granular control
- Frequent data migrations

Complex Processes

- Time consuming processes
- Lack of automation
- Slow reaction to request

The Hypervisor Enables App-Centric Storage Automation

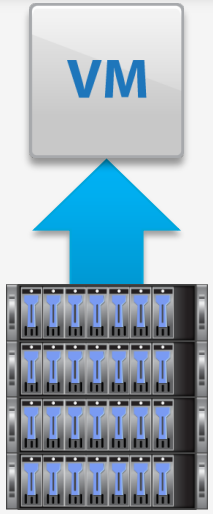


The hypervisor is the right place to drive automation:

- Knows the needs of all apps in real time
- Sits directly in the I/O path
- Has global view of all underlying storage systems
- Can configure storage dynamically
- Hardware agnostic

App-centric Automation Drives Agility and QoS

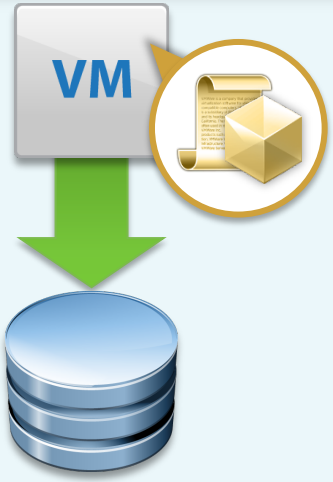
Today's Infrastructure-centric Model



- Static pre-allocation of shared storage container (LUN)
- Data services tied to storage container
- Vendor specific management

✖ Long provisioning cycles
✖ Overprovisioning of resources
✖ Management complexity
✖ Frequent data migrations

App-centric Automation



- Dynamic delivery of storage service levels when needed
- Fine control of data services at the VM level
- Common management across heterogeneous devices

✓ Rapid provisioning
✓ No overprovisioning of resources
✓ QoS automation
✓ Simple change management

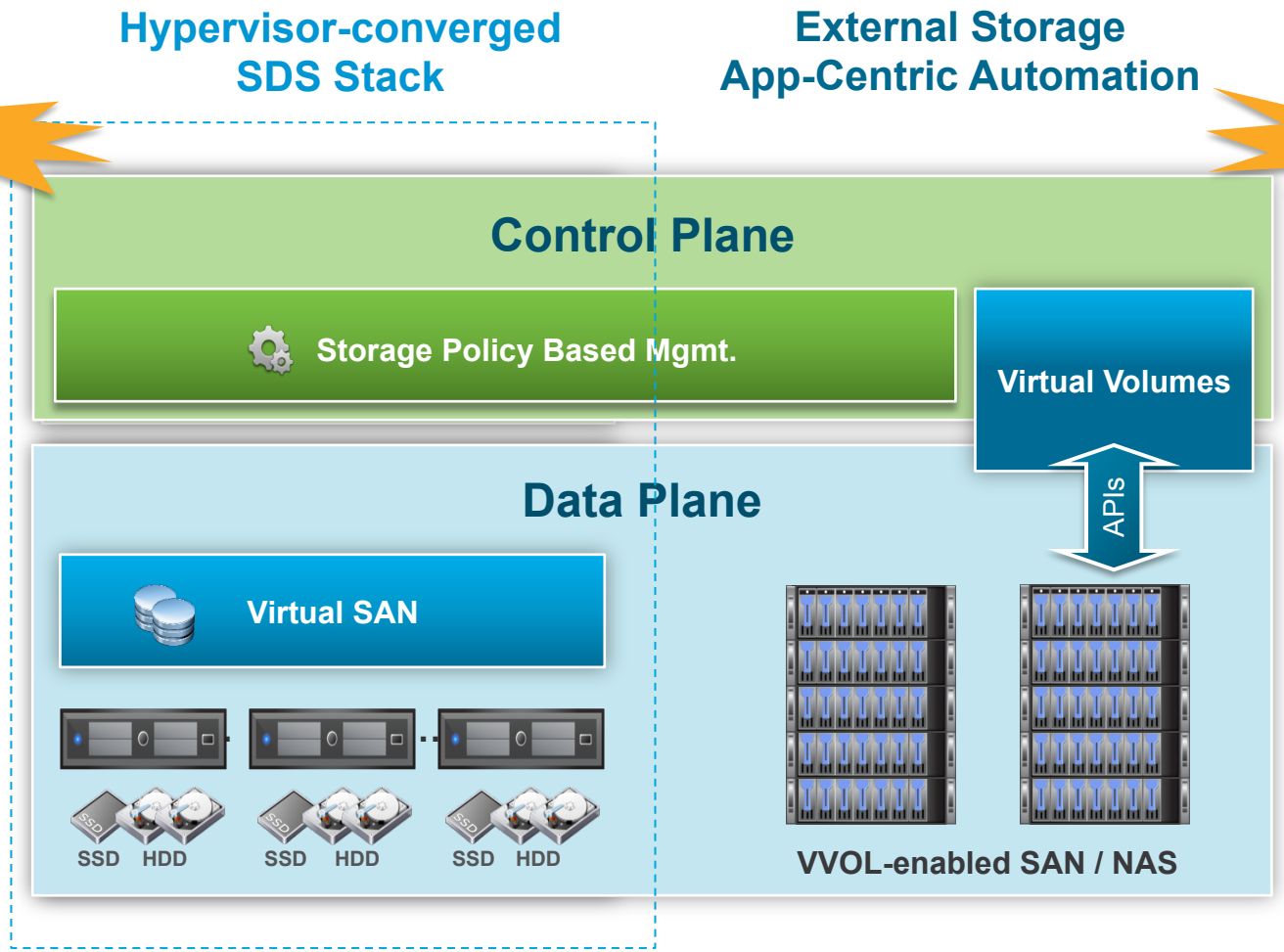
What's New In The vSphere 6.0 Release

Virtual SAN 6.0

NEW

Radically Simple Hypervisor-Converged Storage for VMs

- ✓ All-Flash architecture
- ✓ 2x greater scalability
- ✓ 4x greater with All-Flash; 2x performance with Hybrid
- ✓ Virtual SAN Snapshots and Clones



vSphere Virtual Volumes

NEW

Management & Integration Framework for External Storage

- ✓ Virtualizes SAN/NAS devices
- ✓ Uses native array capabilities
- ✓ VM-level operations
- ✓ Included with vSphere

vSphere Virtual Volumes

A More Efficient Operational Model For External Storage

vmware®

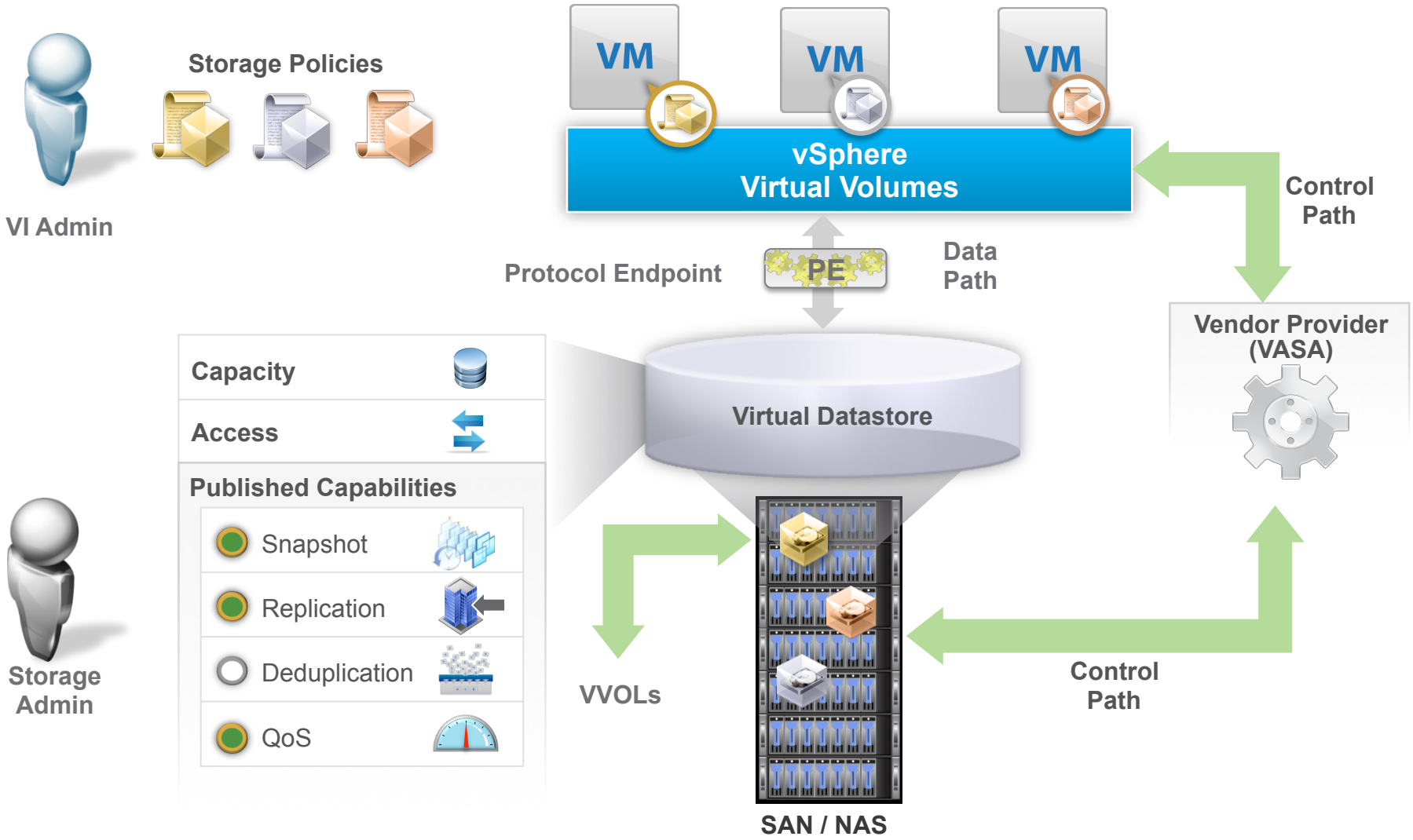
VMware vSphere Virtual Volumes

Integration Framework for VM-Aware Storage

Overview

- Virtual disks are natively represented on arrays
- Enables VM granular storage operations using array-based data services
- Extends vSphere Storage Policy-Based Management to the storage ecosystem
- Supports existing storage I/O protocols (FC, iSCSI, NFS)
- Based on T10 industry standards
- Industry-wide initiative supported by major storage vendors
- Included with vSphere

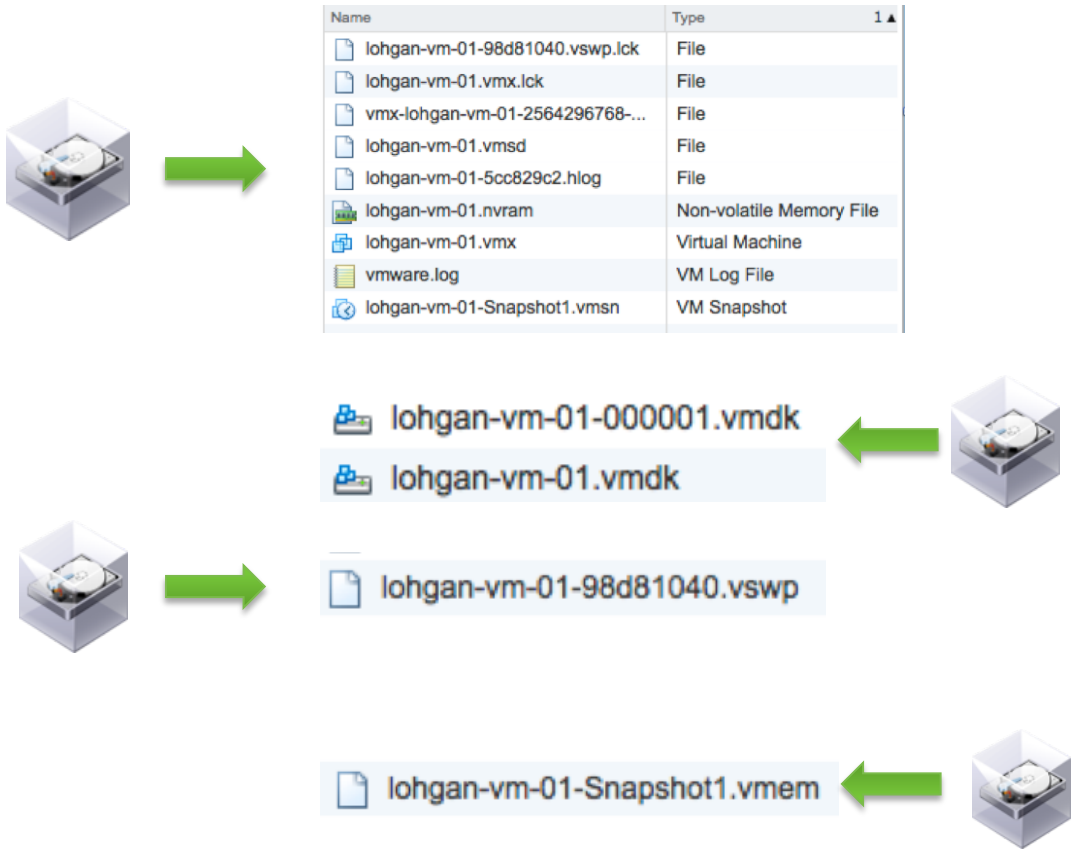
vSphere Virtual Volumes Architecture



vSphere Virtual Volumes

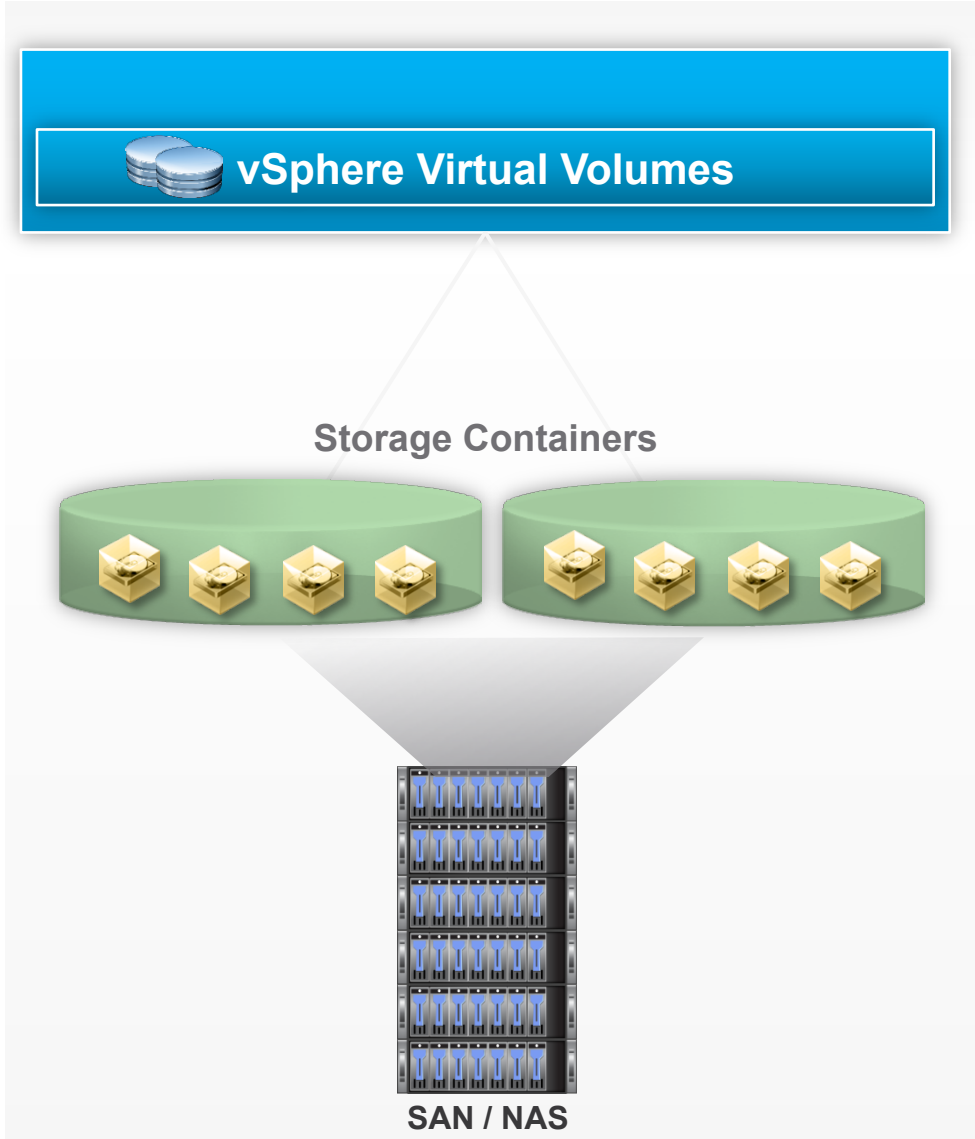
- **Virtual Volumes**
  - Virtual machine objects stored natively on the array.
  - No Filesystem on-disk formatting required
- There are **five** different types of recognized Virtual Volumes:
  - **CONFIG** – vmx, logs, nvram, log files, etc
  - **DATA** – VMDKs
  - **MEM** – Snapshots
  - **SWAP** – Swap files
  - **Other** – vSphere solution specific type

vSphere Web Client View



vmware

Storage Container



Storage Containers

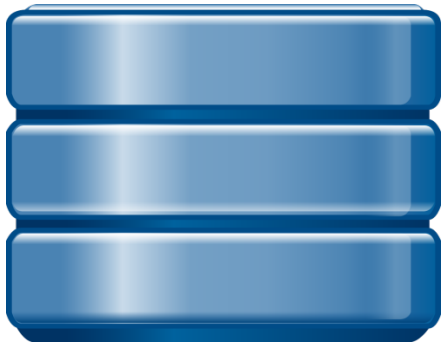
- Logical storage constructs for grouping of virtual volumes.
- Typically defined and setup by storage administrators on the array in order to define:
  - Storage capacity allocations and restrictions
- Capacity is based on physical storage capacity
- Logically partition or isolate VMs with diverse storage needs and requirement
  - Storage policy settings based on data service capabilities
- Minimum one storage container per array
- Maximum depends on the array

vmware

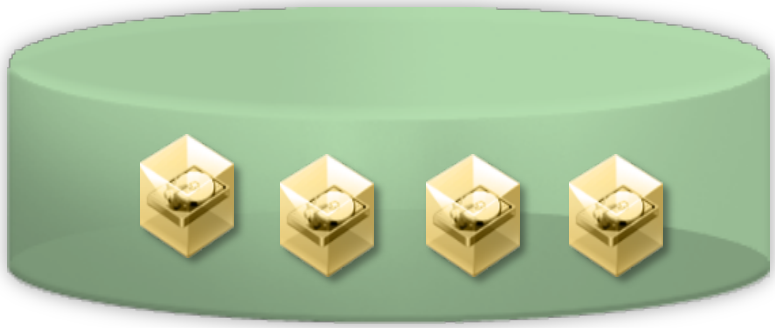
Storage Container

Do I still need to create Datastores?

Managed in similar fashion in vSphere, with additional goodness



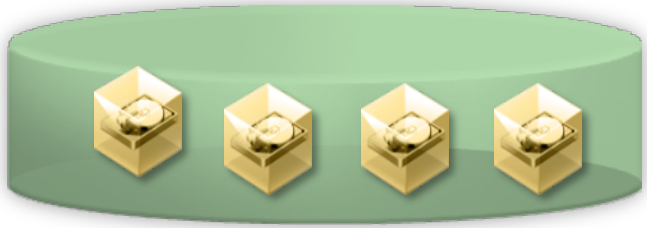
vSphere Datastore



Storage Container

Differences between Storage Containers and LUNs

Storage Containers



- Size based on array capacity
- Max number of SCs depend only on the array ability
- Size of SC can be extended
- Can distinguish heterogeneous capabilities for different VMs (Virtual Volumes) provisioned in that SC

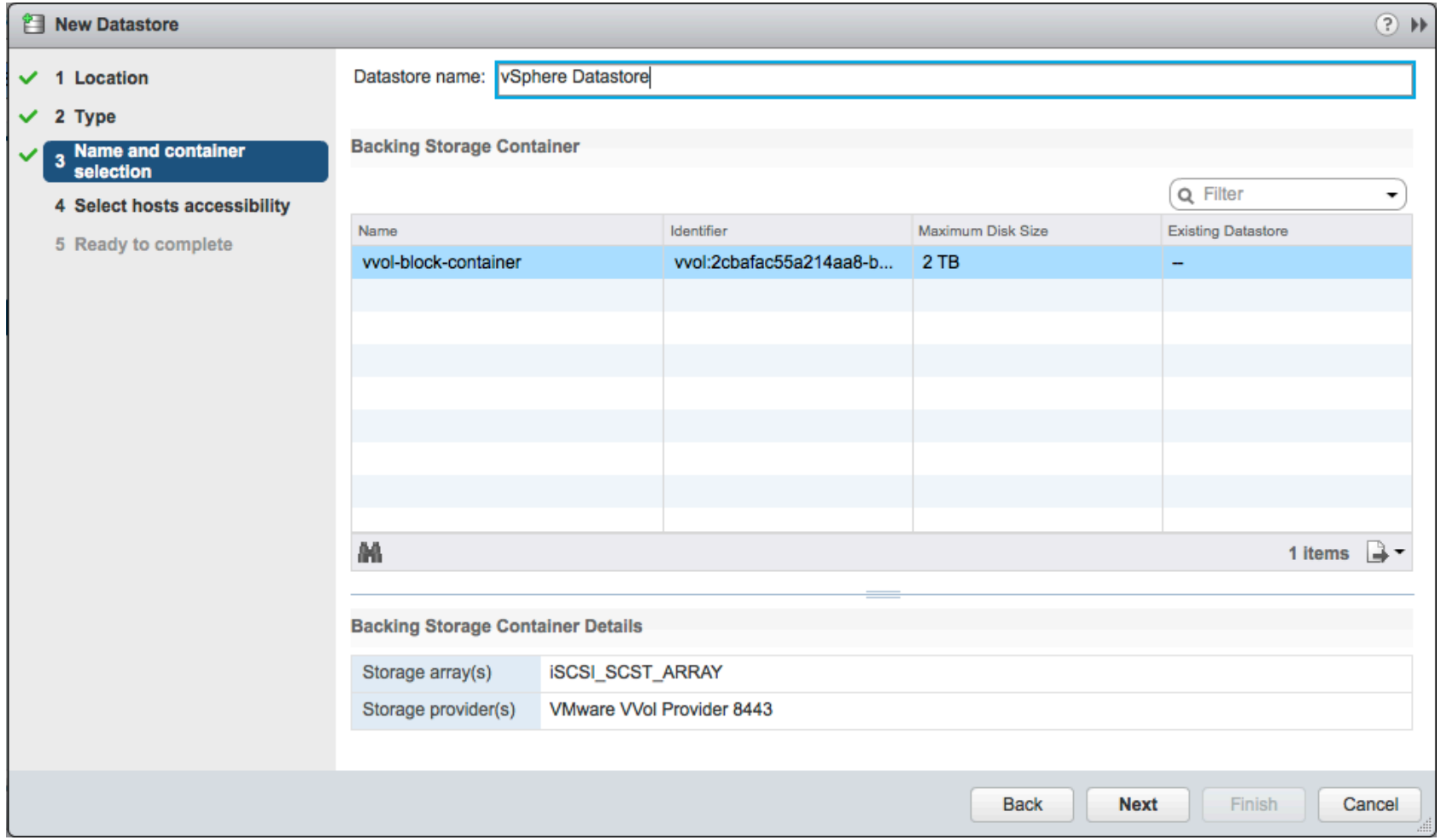
LUN

- Fixed size mandates more number of LUNs
- Needs a FileSystem
- Can only apply homogeneous capability on all VMs (VMDKs) provisioned in that LUN.
- Managed by In-band FileSystem commands

Storage Container (SC)

- What do the Admins see?
- How are the storage containers setup?
- What does the vSphere Admins see?
- Why are we still creating datastores in this new model?

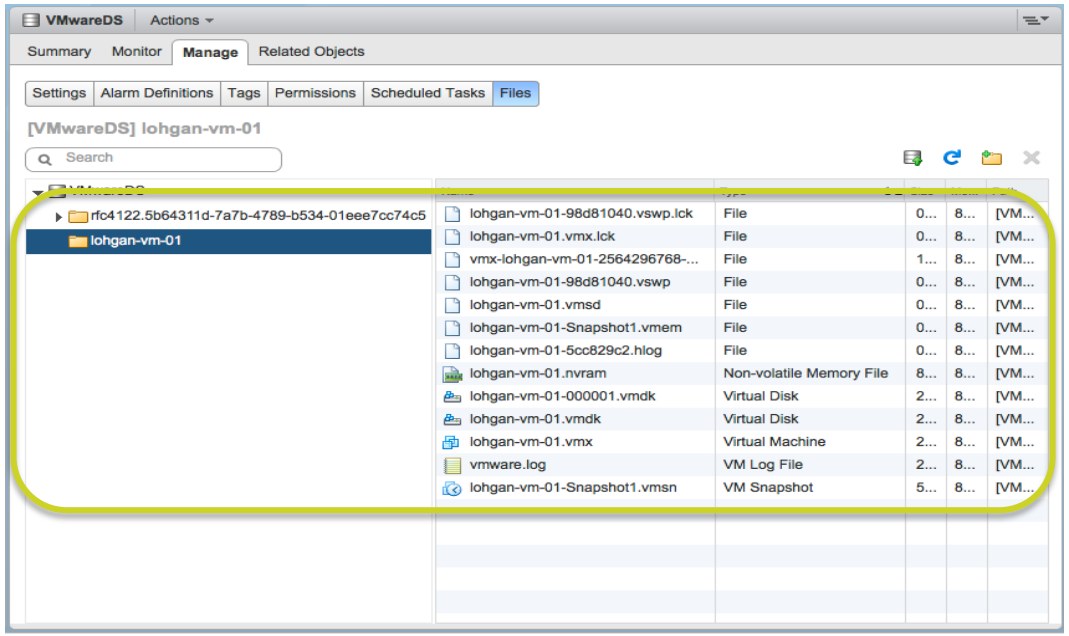
Storage Container (SC)



vmware

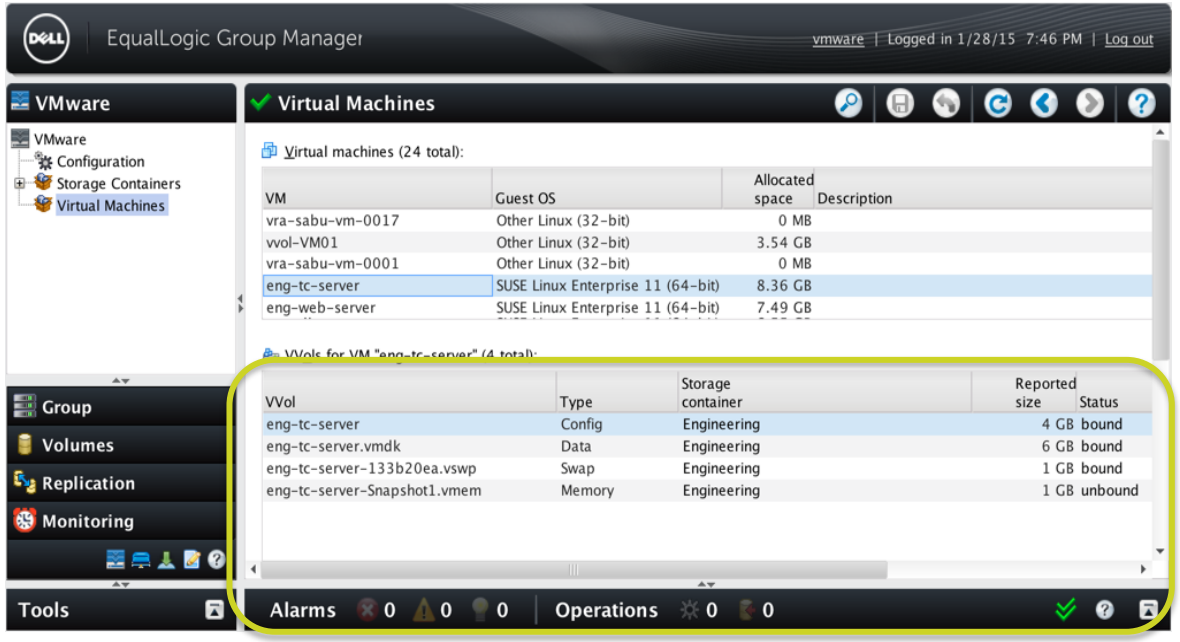
Virtual Volumes

vSphere Admin View



VM objects view from a datastore

Storage Admin View



VM objects view from a storage container on an array

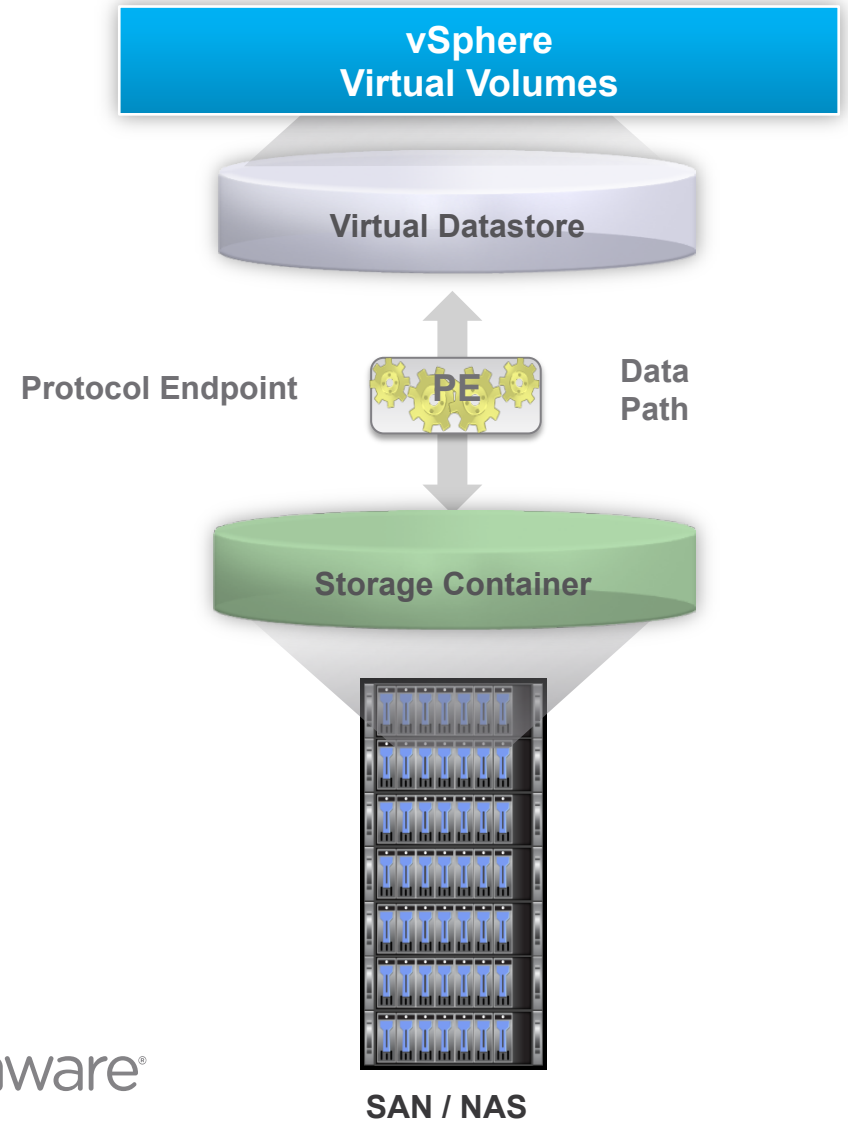
vmware

Data Plane

vSphere Virtual Volumes

vmware®

Protocol Endpoints



Protocol Endpoints

- Access points that enables communication between ESXi hosts and storage array systems.
  - Part of the physical storage fabric
  - Created by Storage administrators

Scope of Protocol Endpoints

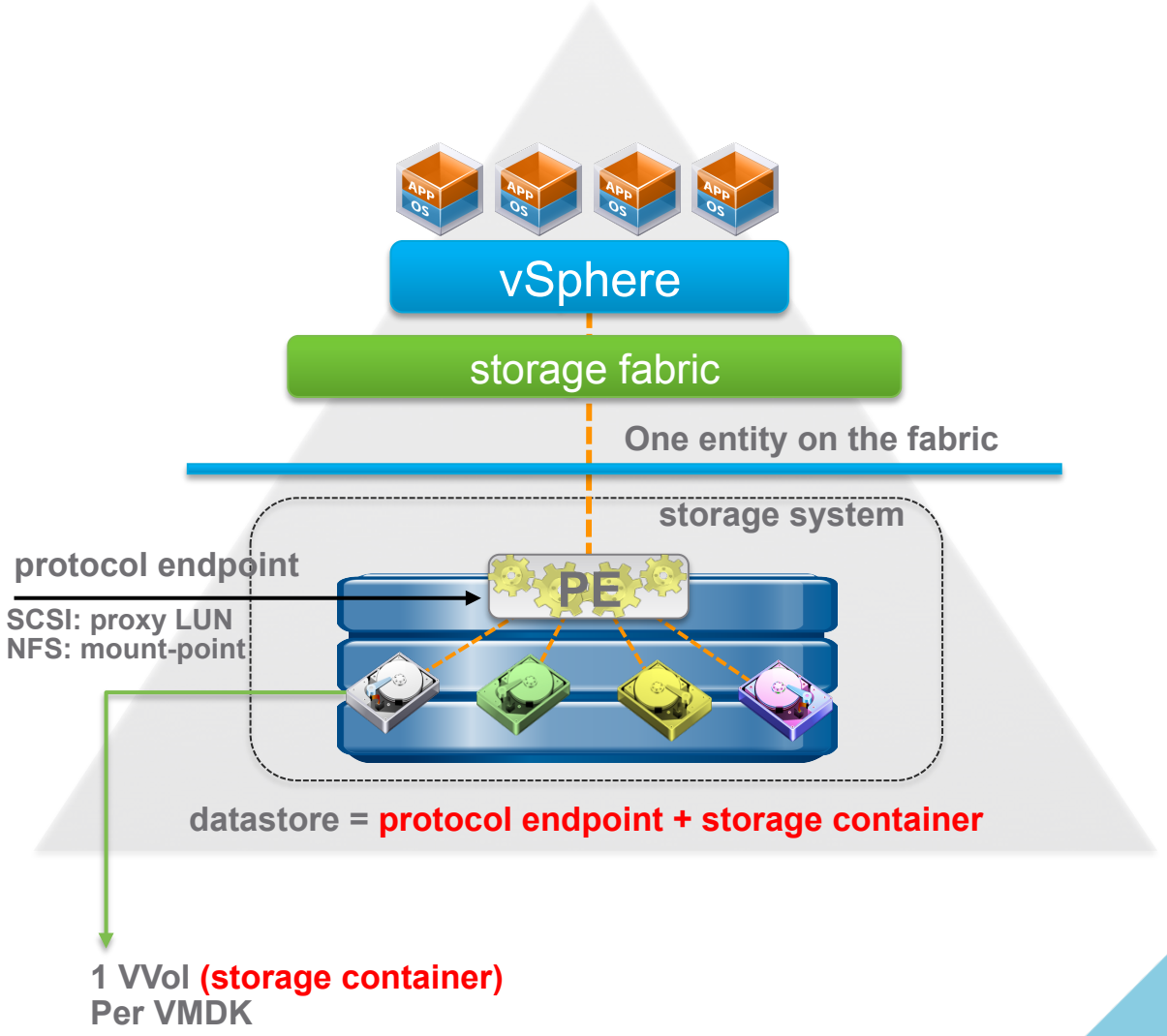
- Compatible with all SAN and NAS Protocols:
  - iSCSI
  - NFS v3
  - FC
  - FCoE
- A Protocol Endpoint can support any one of the protocols at a given time
- Existing multi-path policies and NFS topology requirements can be applied to the PE

Why Protocol Endpoints?

vmware

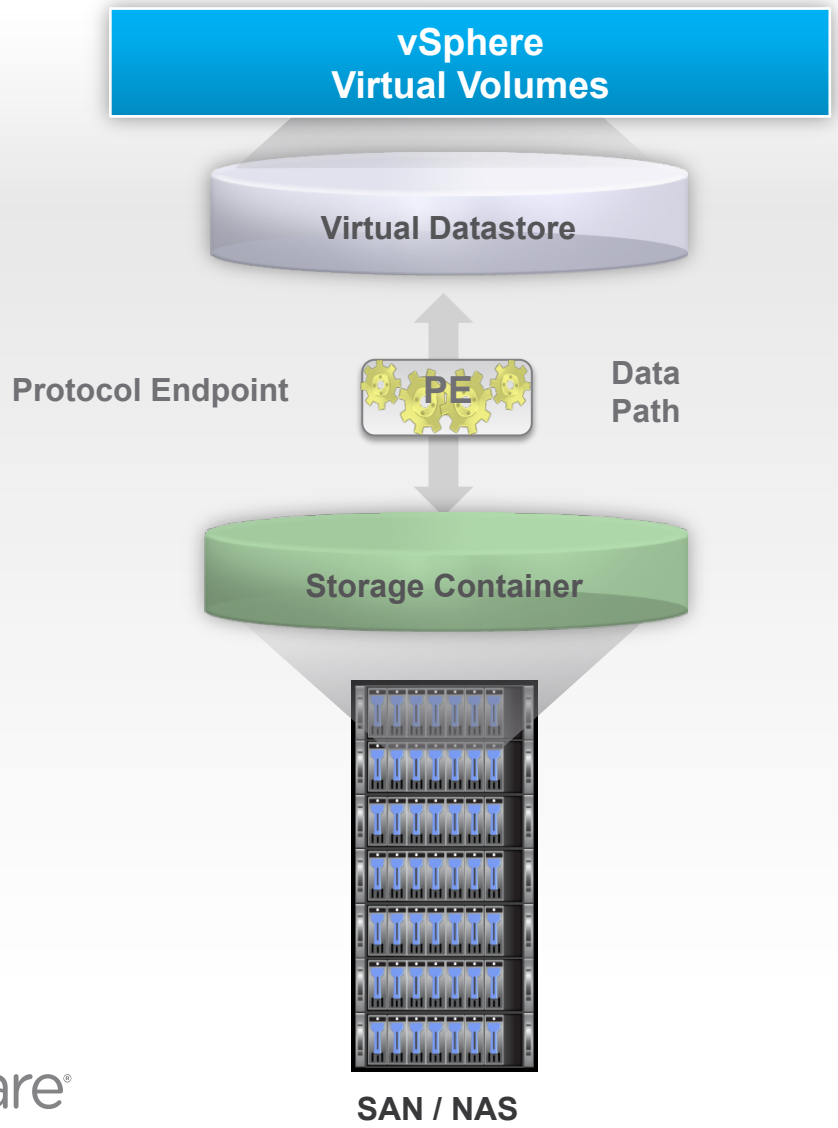
Protocol Endpoints

- **Today**, there are different types of logical management constructs to store VMDKs/objects:
  - NFS Mount Points
  - IP or block based datastores
- **Datastores serve two purposes today:**
  - Endpoints – receive SCSI or NFS reads, write commands
  - Storage Container – for large number of VMs metadata and data files
- **Differences between Protocol Endpoints and Datastores:**
  - PEs no longer stores VMDKs but it only becomes the access point.
  - Now you wont need as many datastores or mount point as before
- Certain offloading operation will be done via VASA and other will be done using the standard protocol commands



vmware

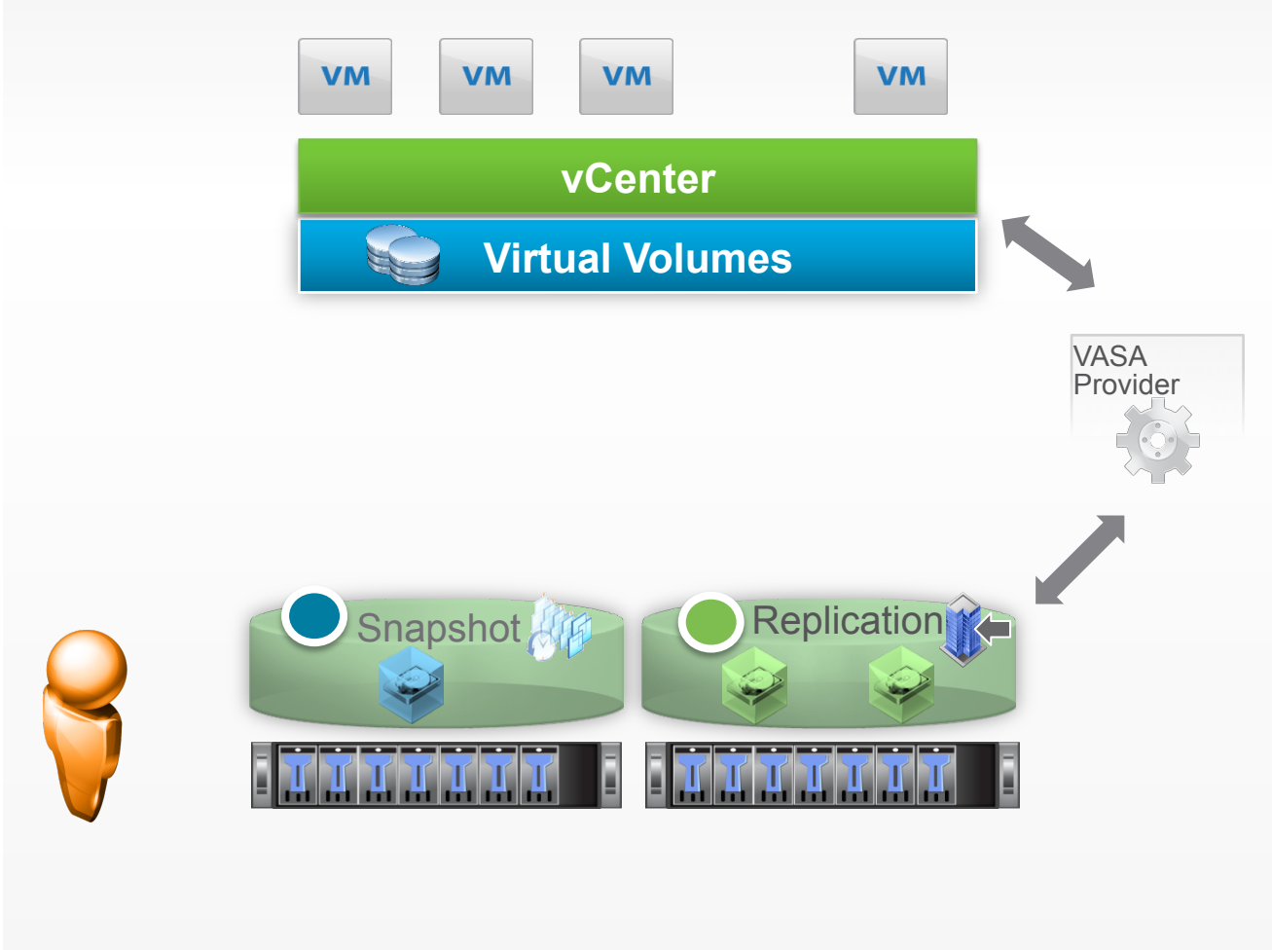
Discovery Procedures – Protocol Endpoint



Protocol Endpoint discovery process

- SCSI PEs are discovered during an ESX rescan
- NFS PEs are maintained as IP addresses or file paths
- ESX will identify PE and maintain all discovered PEs in a database.

Discovery Procedures – Storage Container



Storage Container Discovery Process

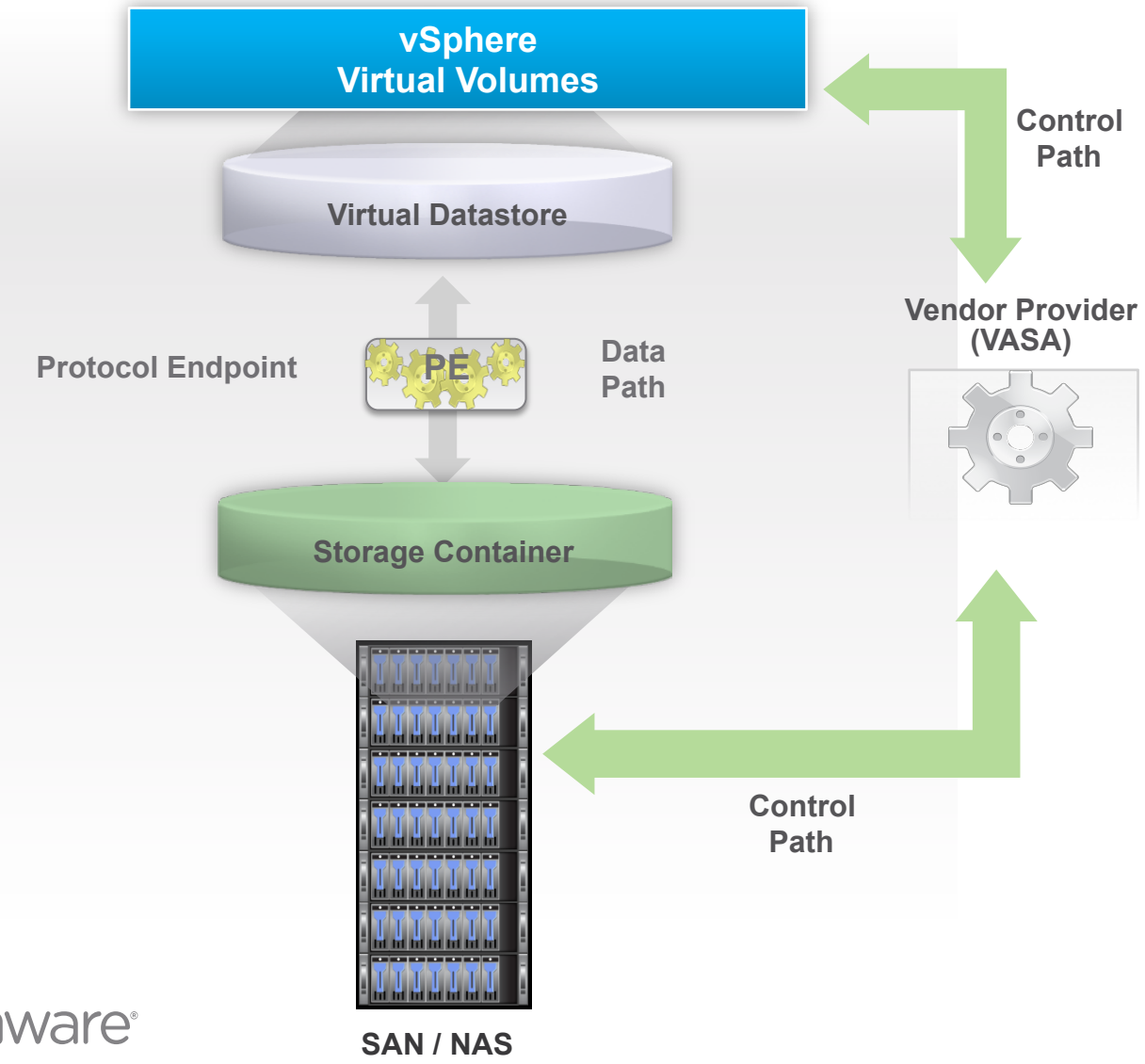
- Storage admin sets up Storage Container with desired capacity
- Desired Capabilities are applied to the Storage Containers
- VASA Provider discovers Storage Container and reports to vCenter
- Any new VMs that are created will subsequently be provisioned in the Storage Container

Management Plane

vSphere Virtual Volumes

vmware®

VASA Provider (VP)



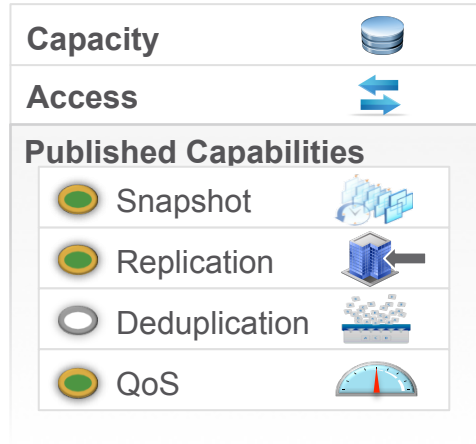
- Software component developed by Storage Array Vendors
- ESX and vCenter Server connect to VASA Provider
- Provides Storage awareness services
- Single VASA Provider can manager multiple arrays
- Supports VASA APIs exported by ESX
- VASA Provider can be implemented within the array's management server or firmware
- Responsible for creating Virtual Volumes

vmware

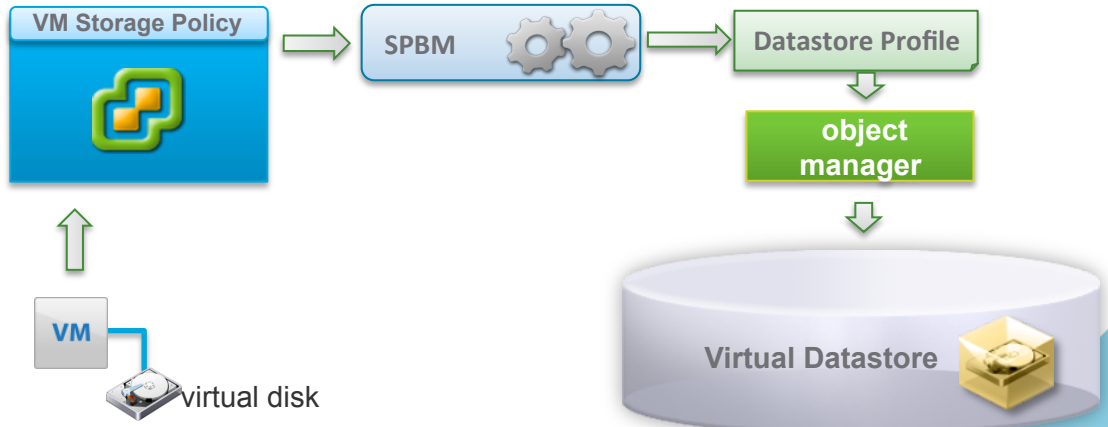
Storage Capabilities and VM Storage Policies

- **Storage Capabilities** – are array based features and data services specifications that capture storage requirements that can be satisfied by a storage arrays advertised as capabilities.
- Storage capabilities define what an array can offer to storage containers as opposed to what the VM requires.
- Arrays Storage Capabilities are advertises to vSphere through the Vendor Provider and VASA APIs
- In vSphere Storage Capabilities are consumed via VM Storage Policy constructs.
- **VM Storage Policies** is a component of the vSphere Storage Policy-based management framework (SPBM)

Storage Capabilities for Storage Array

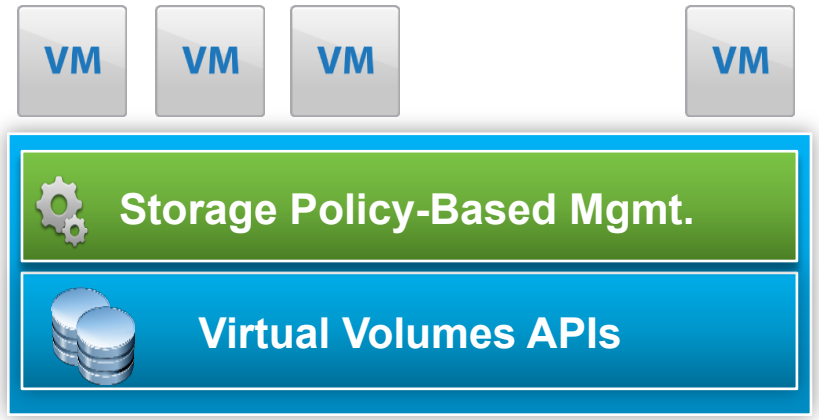


vSphere VM Storage Policy Management Framework

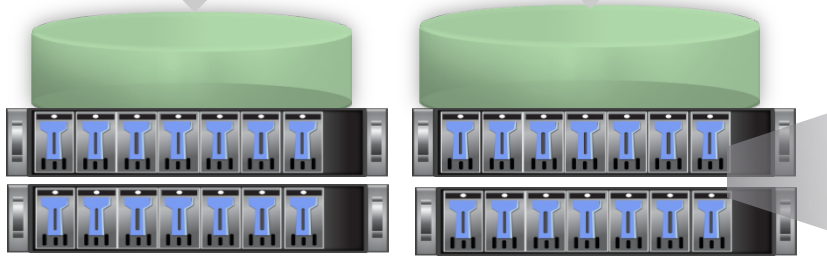


vmware

Storage Policy Based Management (SPBM) – Array Capabilities



Storage admin

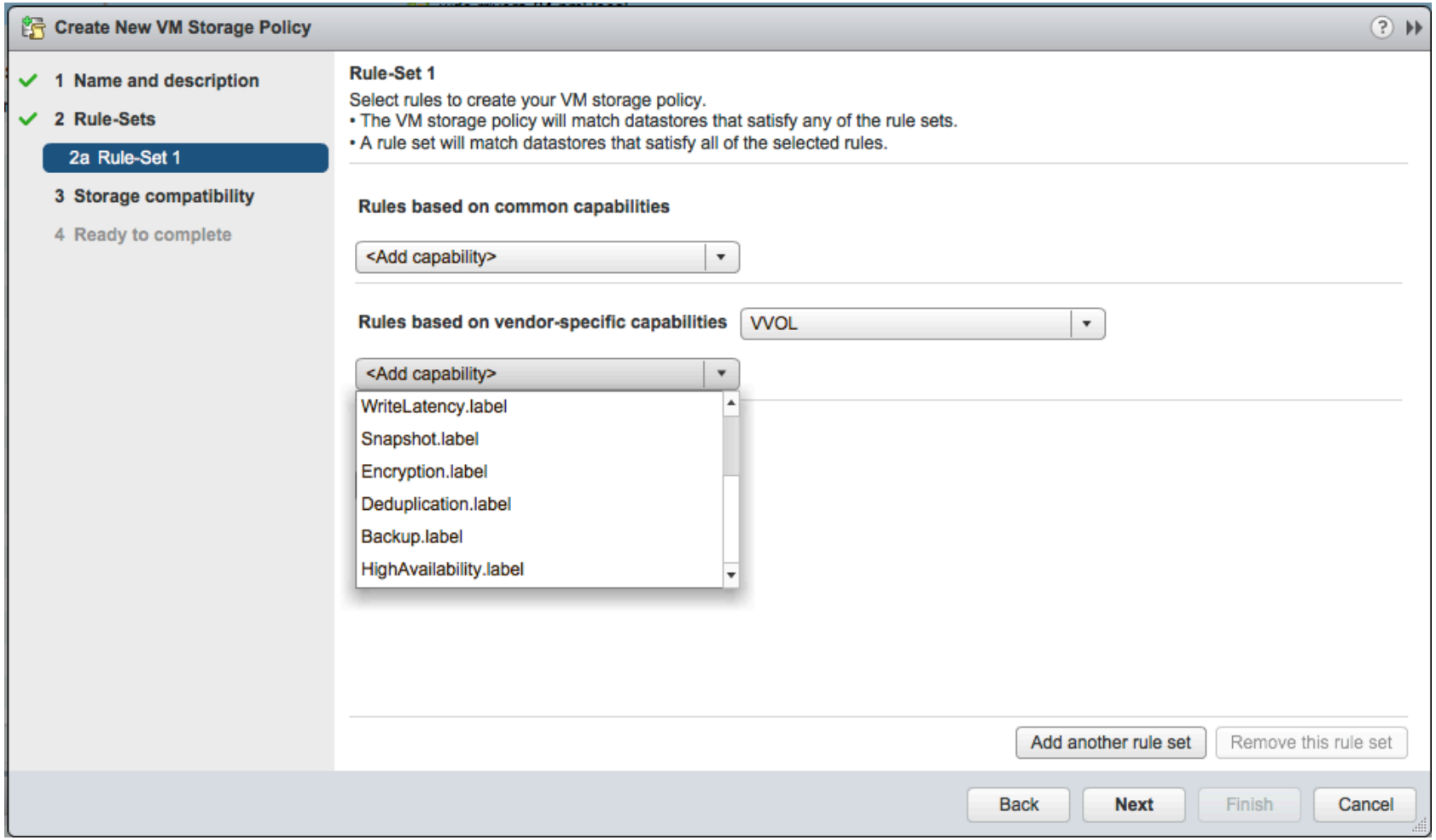


Sample Default Profile for (6090a058-cd89-ffe3-87763007db37] - capabilities

Disk Types	**Publish Capabilities** • Array based features and data services • Defines what an Array can offer • Advertised to ESX through VASA APIs
Disk Encryption	
Dedupe	
Replication	
Snapshot	

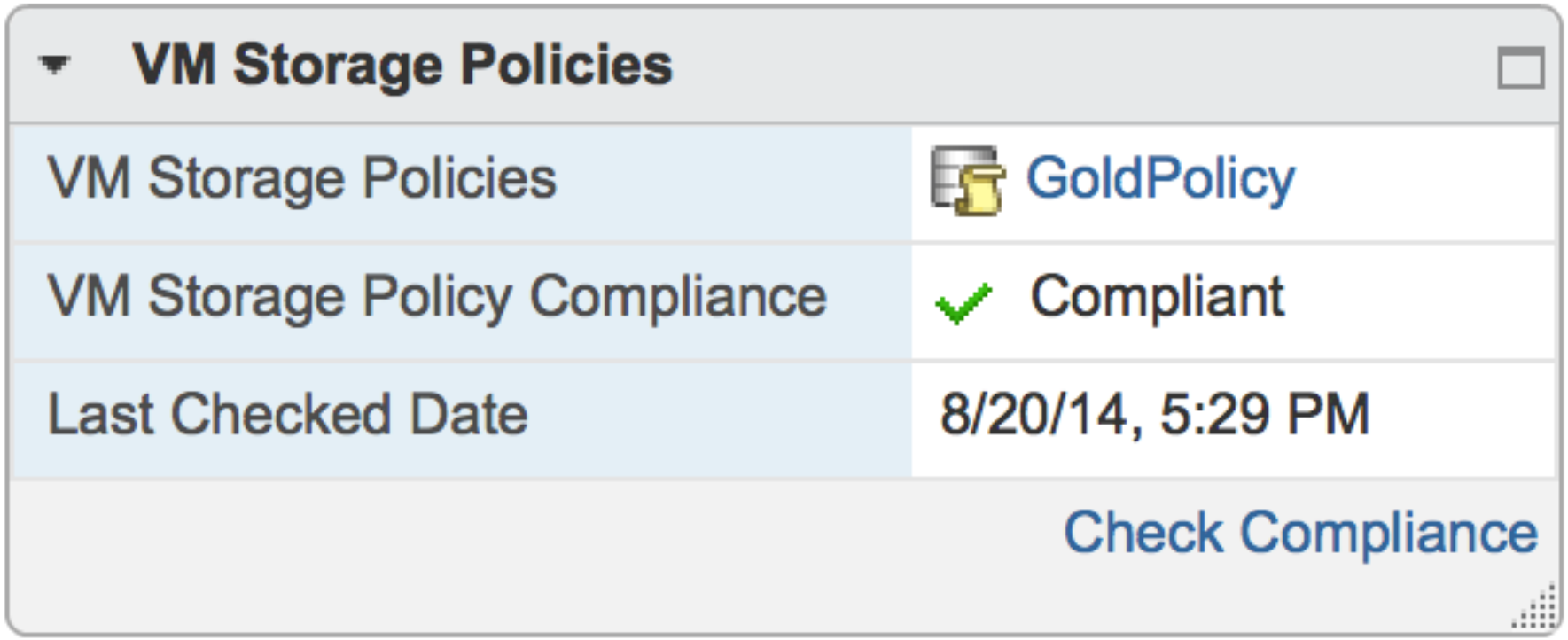
vmware

Storage Policy Based Management (SPBM) – VM Policies



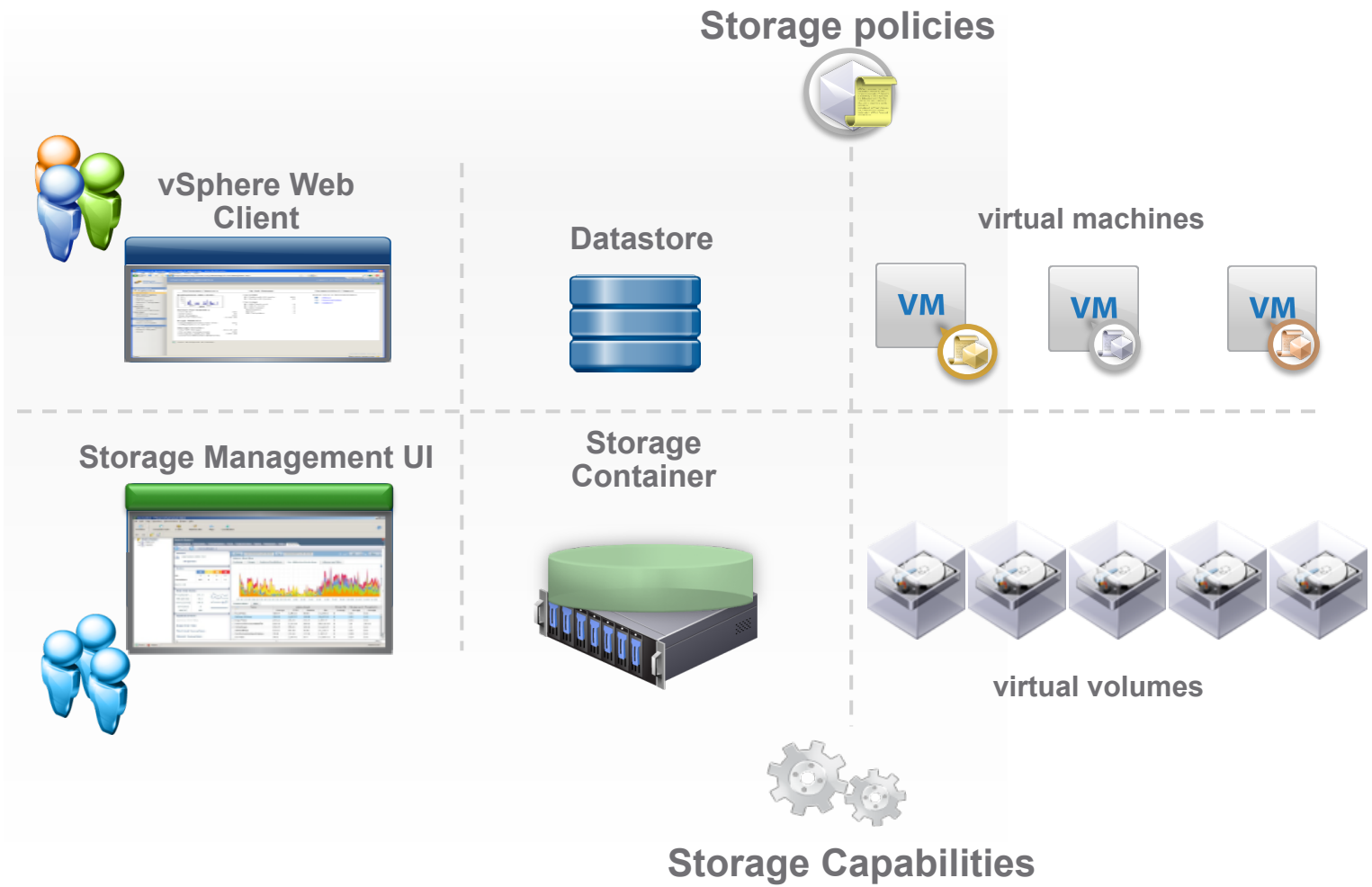
vmware

Storage Policy Based Management (SPBM)



Management Workflow

- What do the Admins see?
- How are the storage containers setup?
- What does the vSphere Admins see?
- Why are we still creating datastores in this new model?



Provision and Lifecycle Workflows

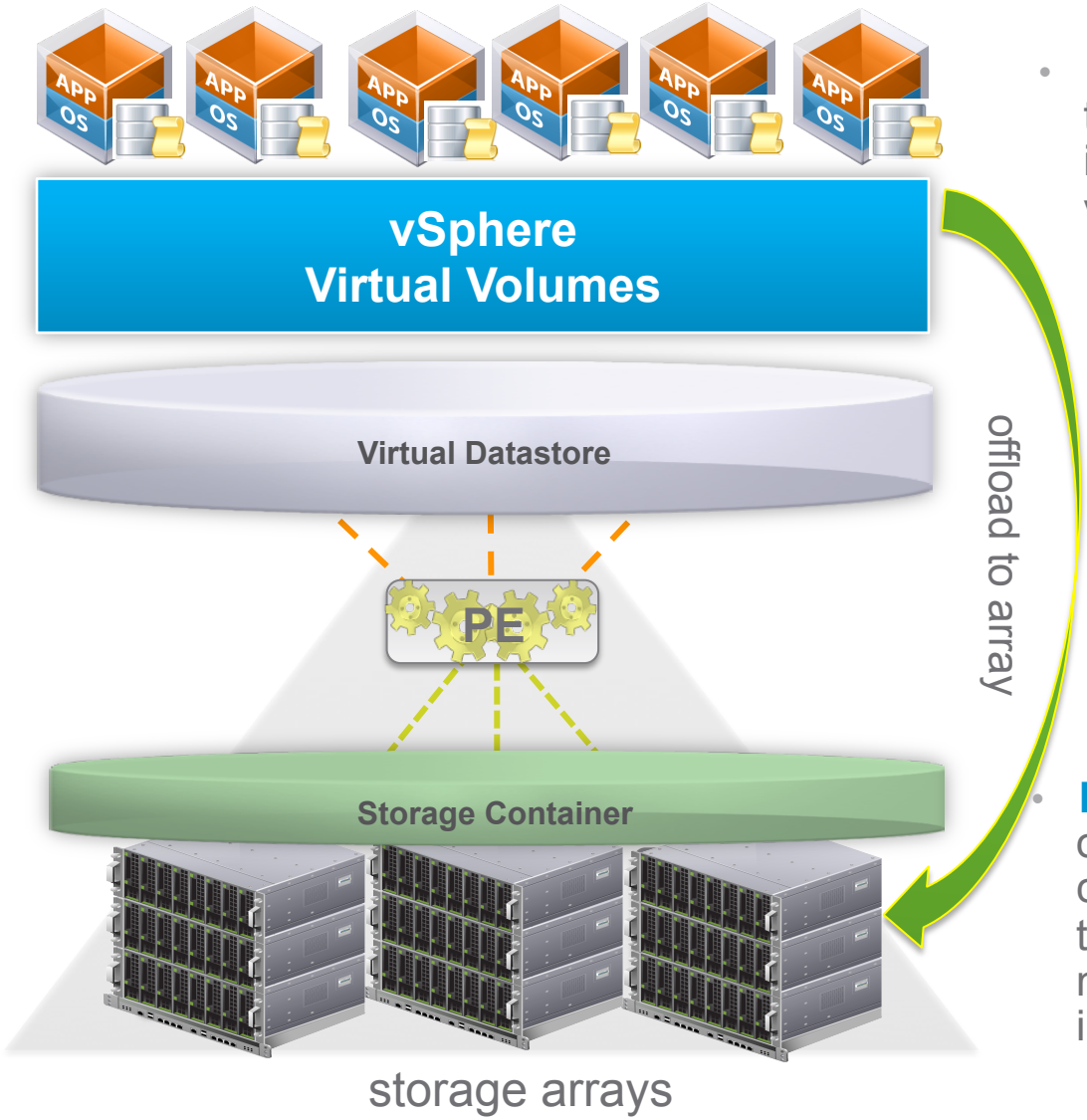
Binds

vmware

Provisioning Workflow

vSphere Admin

1. Create Virtual Machines
2. Assign a VM Storage Policy
3. Choose a suitable Datastore



Under the Covers

- **Provisioning** operations are translated into VASA API calls in order to create the individual virtual volumes.

Under the Covers

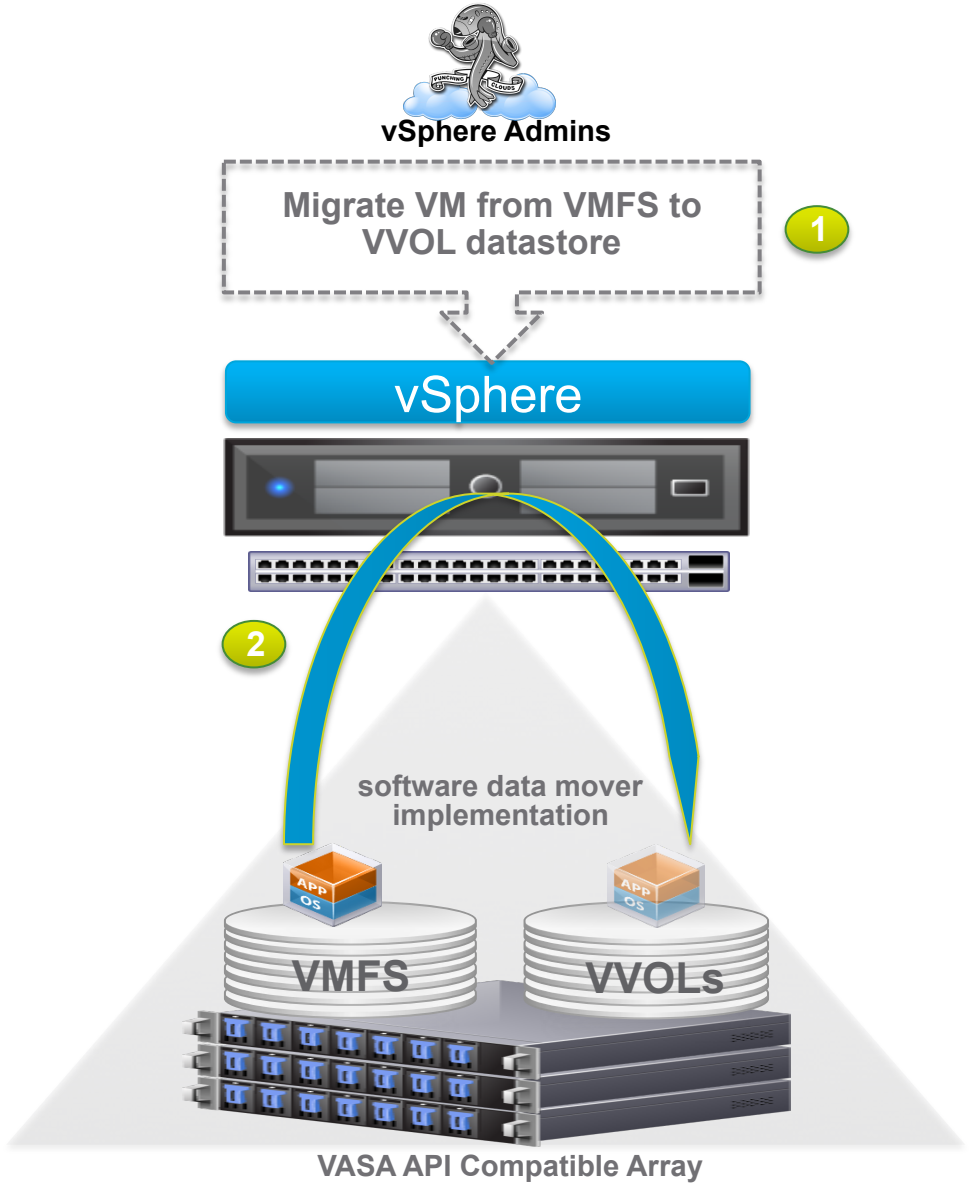
- **Provisioning** operations are offloaded to the array for the creation of virtual volumes on the storage container that match the capabilities defined in the VM Storage Policies

vmware

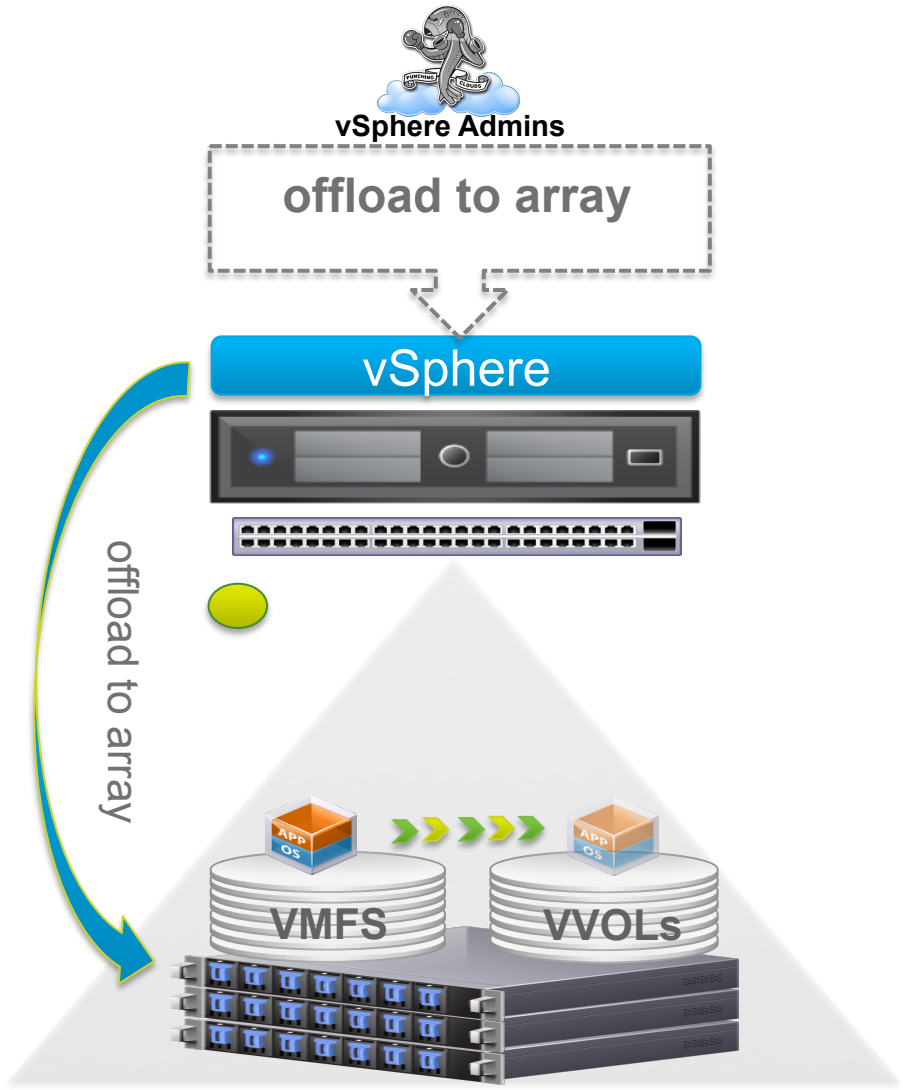
Operation Scenarios

Offload, Migrations, Snapshots

Migration Scenario: with VAAI vs Virtual Volumes



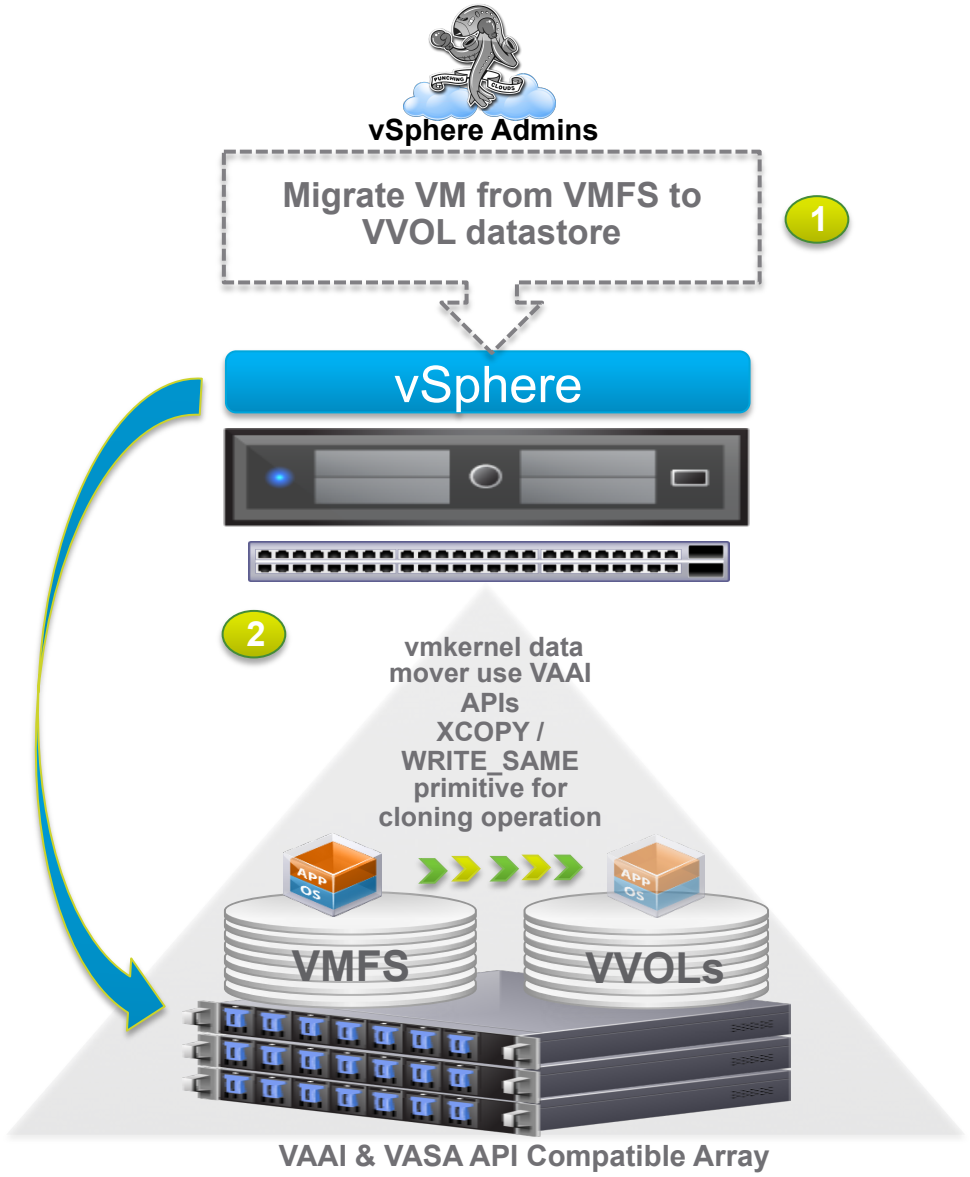
vSphere Virtual Volumes Offload Operations



Virtual Machine Operation Offloaded

- Virtual Machine provisioning
- Virtual Machine deletes
- Virtual Machine full clones
- Virtual Machine Linked Clones
- Virtual Machine Snapshots
- Storage vMotion (Powered off VMs)

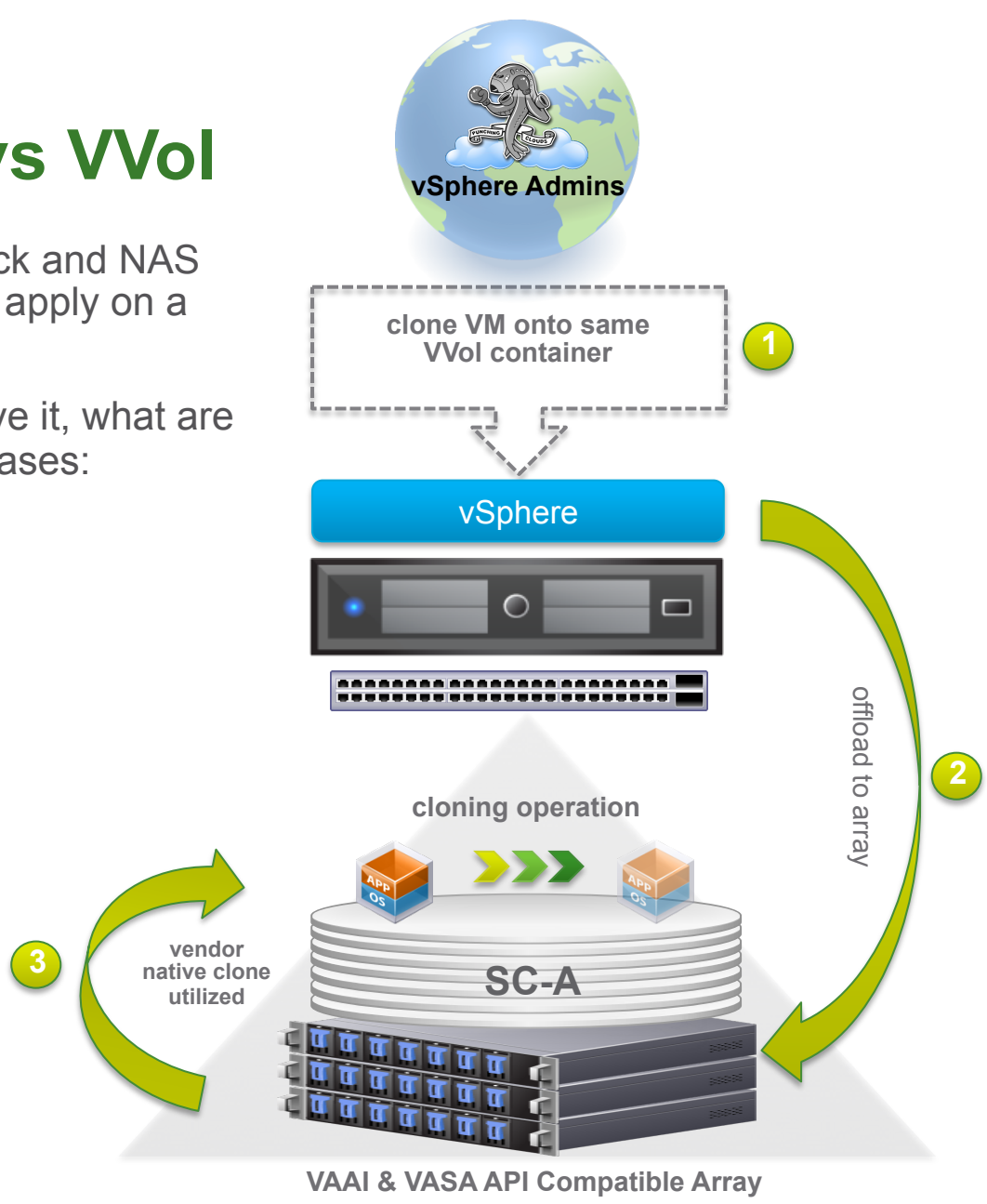
Migration Scenario: with VAAI vs Virtual Volumes

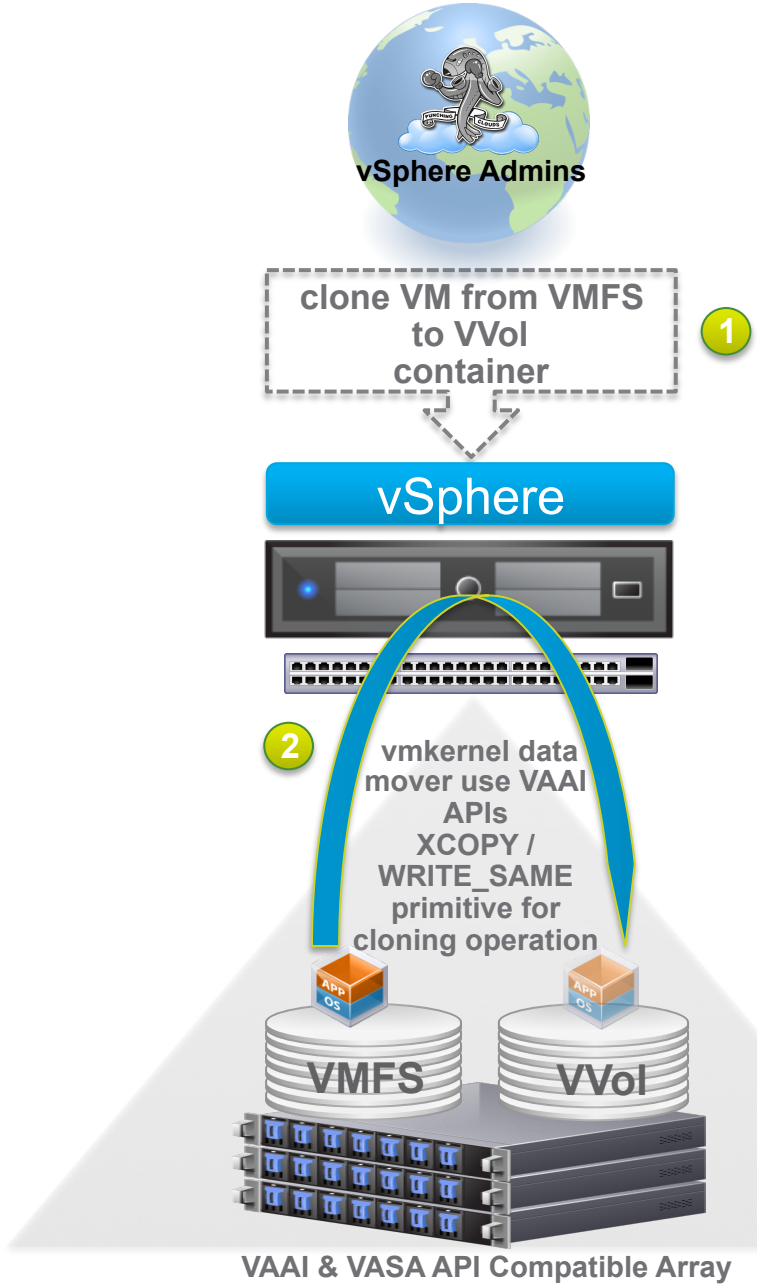


vmware

VAAI vs VVol

- VAAI Block and NAS how they apply on a VVol
- If you have it, what are the use cases:





vmware

VAAI vs VVOLs

1 VM operation from VVOLs container to different VVOLs container

vSphere

2

offload to array

cloning operation

3

SC-A

SC-B

vendor native clone utilized with VASA primitives

Fully VAAI & VASA APIs Compatible Array

always attempt default operation with VASA API primitives

1 default

fail back 2

If default operations fails VAAI API primitives are used

1 VM operation from VVOLs container to different VVOLs container

vSphere

3

2

offload to array

vmkernel data mover uses VAAI primitives for cloning operation

SC-A

SC-B

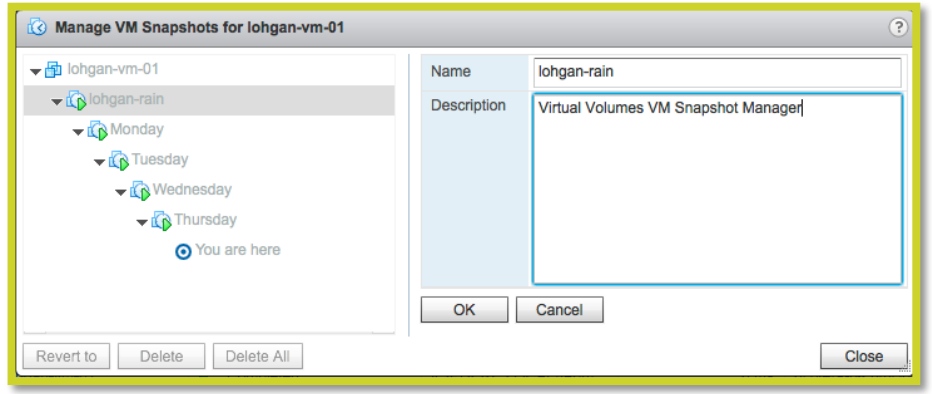
Fully VAAI & VASA APIs Compatible Array

vmware

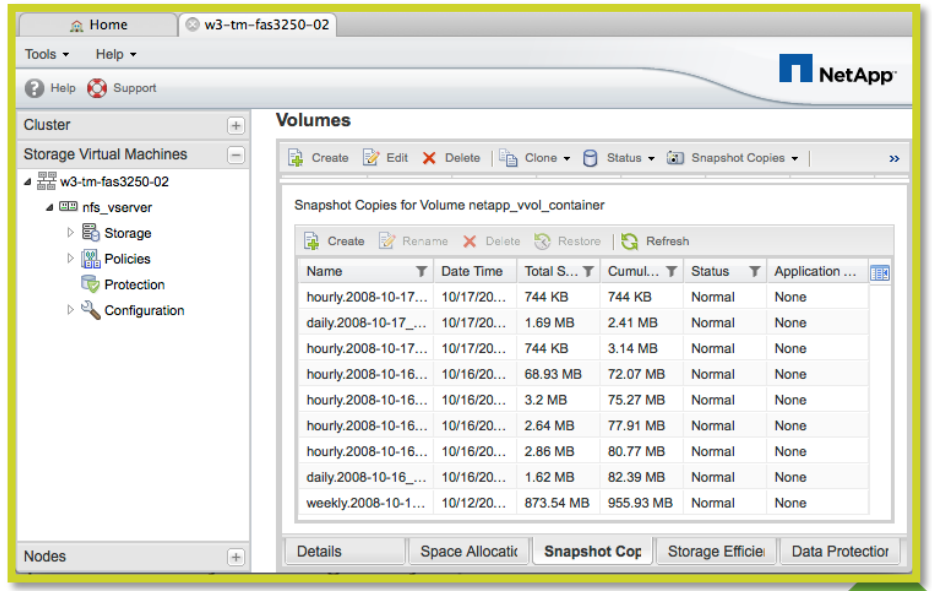
Snapshots

- Snapshots are a point in time copy on write image of a Virtual Volume with a different ID from the original.
- Virtual Volumes snapshots are useful in the contexts of creating:
  - a quiesced copy for backup or archival purposes, creating a test and rollback environment for applications, instantly provisioning application images, and so on.
- Two type of snapshots supported:
  - Managed Snapshot – Managed by ESX.
    - A maximum of 32 vSphere managed snapshot are supported for linked clones of an individual VM
  - Unmanaged Snapshot – Manage by the storage array.
    - Maximum snapshot dictated by the storage array

Managed Snapshot - vSphere

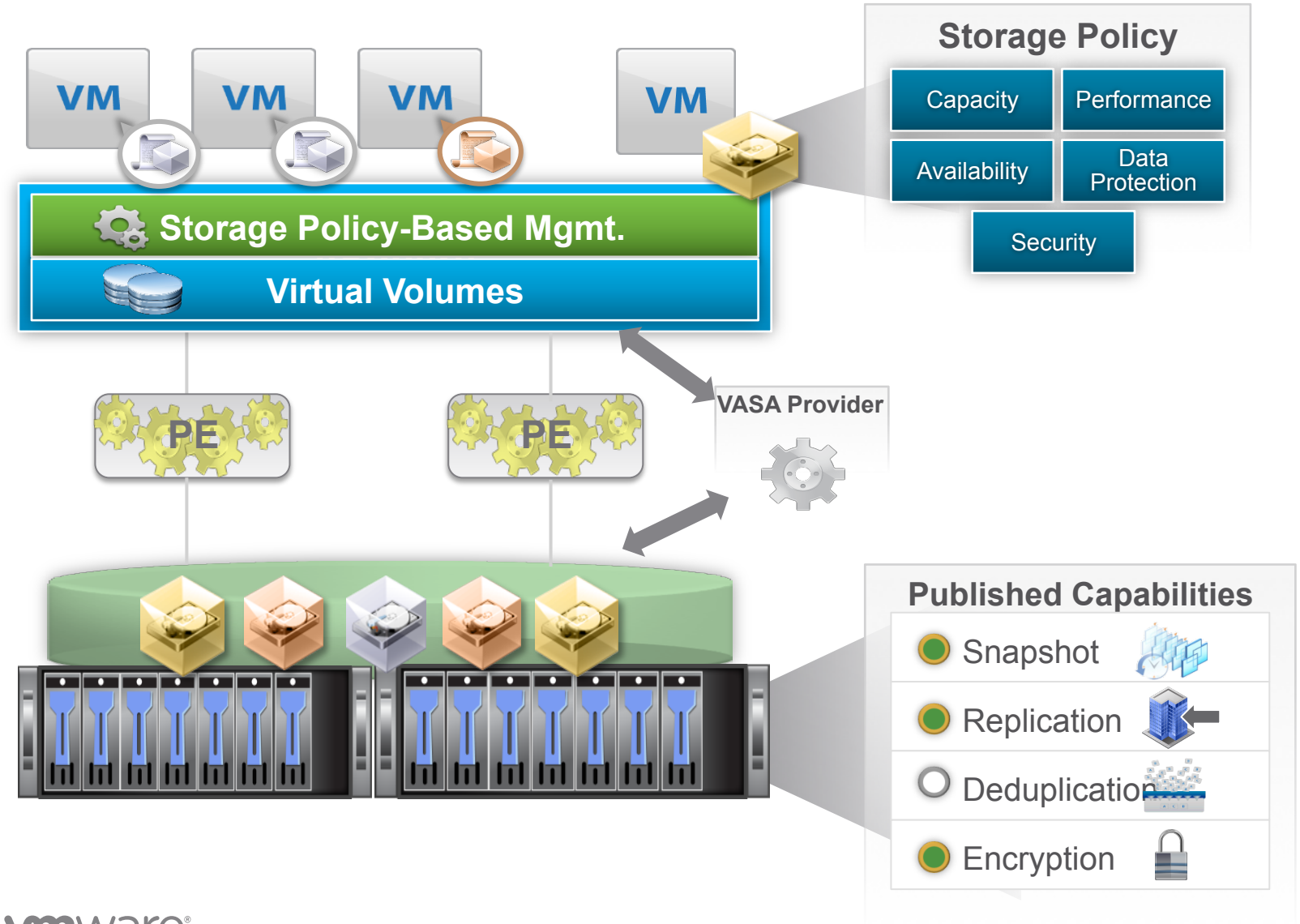


Unmanaged Snapshot - Array



vmware

vSphere Virtual Volumes Supported Features



Supported vSphere Features

- SPBM
- Thin Provisioning
- Linked Clones
- Native Snapshots
- Protocols: NFS3, iSCSI, FC, FCoE
- View Storage Accelerator (CBRC)
- vMotion
- SvMotion
- DRS
- XvMotion
- vSphere SDK (VC APIs)
- VDPA/VDP
- View
- vRealize Operations
- vRealize Automation
- Stateless / Host Profiles

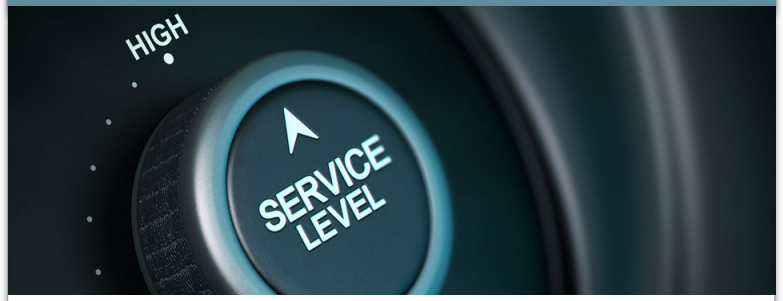
The Benefits of vSphere Virtual Volumes

A More Efficient Operational Model For External Storage

Simplifies Storage Operations

- Eliminate inefficient handoffs between VI and Storage Admin
- Faster storage provisioning through automation
- Simplified change management through flexible consumption
- Self-service provisioning via cloud automation tools.

Simplifies Delivery of Service Levels



- Leverage native array-based capabilities
- Fine control at the VM level
- Dynamic configuration on the fly
- Ensure compliance through policy enforcement using automation

Improves Resource Utilization

- Increase capacity utilization.
- Eliminate overprovisioning
- Reduce management overhead

vSphere Virtual Volumes Is An Industry-wide Initiative












And Many More…

THANK YOU

Rawlinson Rivera
@PunchingClouds