

Cloud Infrastructure Suite Overview

Tichomir Tenev

October 30, 2012

Confidential

vmware®

© 2011 VMware Inc. All rights reserved

Cloud Infrastructure Suite (CIS) – Purpose and Goals

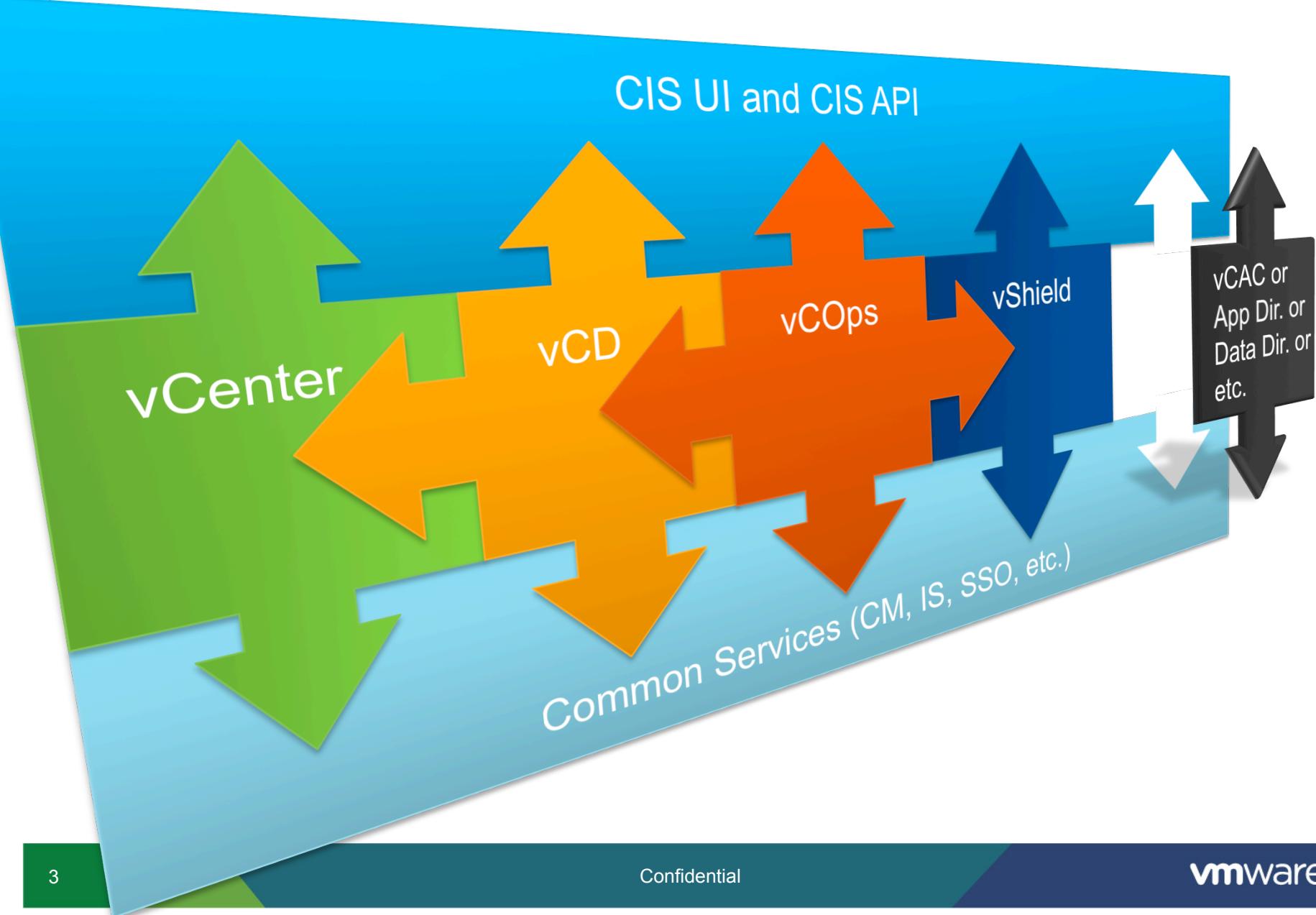
- **Business purpose:**

- Deliver the Software Defined Datacenter on physical Infrastructure
- State-of-the-art scale and hybrid cloud enablement with low opex
- Platform for next-gen apps, self-service offerings and datacenter automation

- **Architecture goals:**

- Unity of concepts and function
- Unified UI, API and Deployment
- Modularity and extensibility
- Scalable design

CIS Composition



Outline: Functionality, Unification and Scale



Functionality

What the system does



Resource Management Stack

Infrastructure Authority

Binding of service request to resources based upon business policy along with associated automation to deliver.

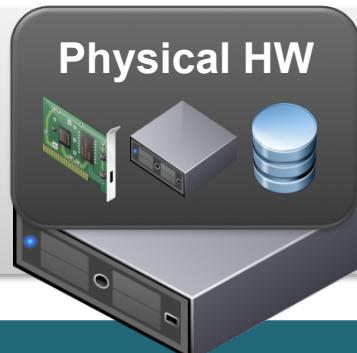
Delegation of resources

Resource allocation to users or tenants. Quota enforcement.

Aggregation of resources

Resource pooling, network stretching, elasticity. Enables adding capacity without a priori identifying the consumer.

Managed resources



Resource Management Stack – Where CIS fits

Infrastructure Authority

Other Management Products
(vCAC, vCC, etc.)

Delegation of resources

Aggregation of resources

Cloud Infrastructure Suite
(Core Functionality)

Managed resources



Resource Management Stack – New with CIS

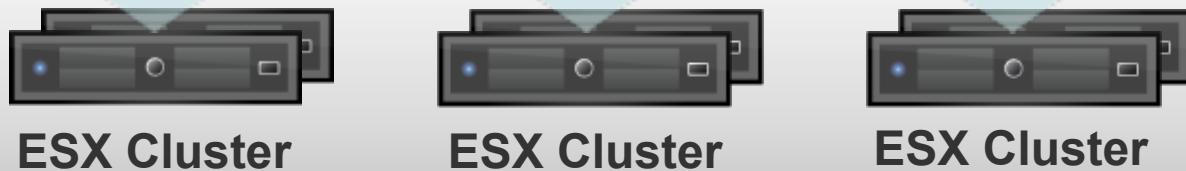
Delegation
of resources



Aggregation
of resources



Managed
resources



Resource Management Stack – Future Vision

Infra.
Auth.

Dele-
gation

Aggre-
gation

Managed
resources

Other Management Products
(vCAC, vCC, etc.)

Cloud
Infrastructure
Suite
2014

Hetero-
geneous

???

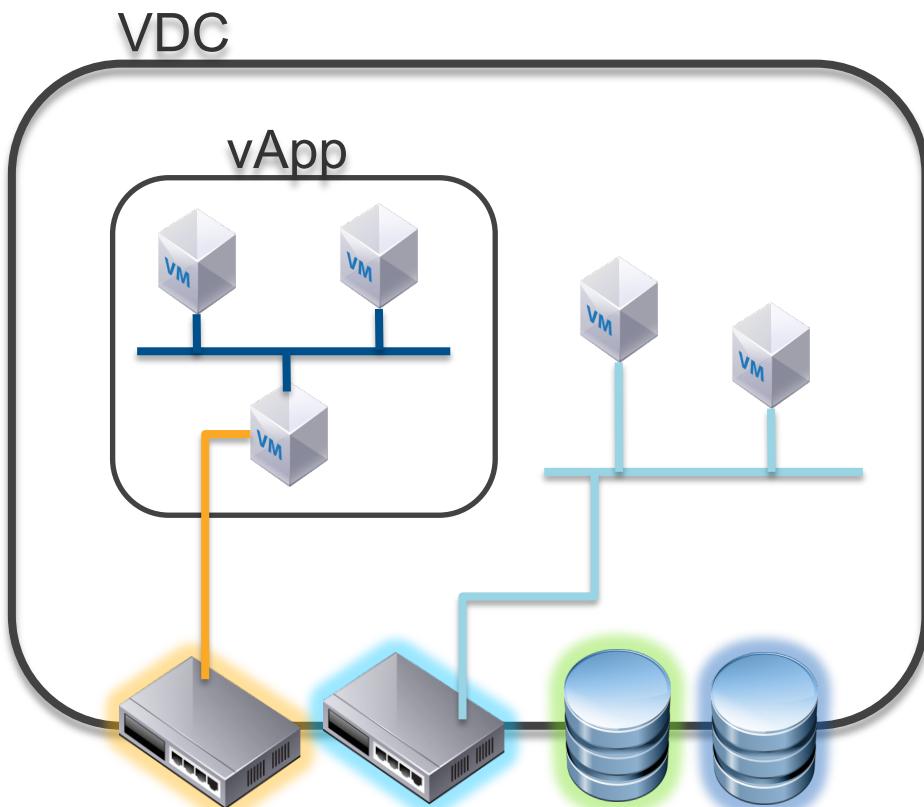
Hybrid Cloud

2015?

2016?

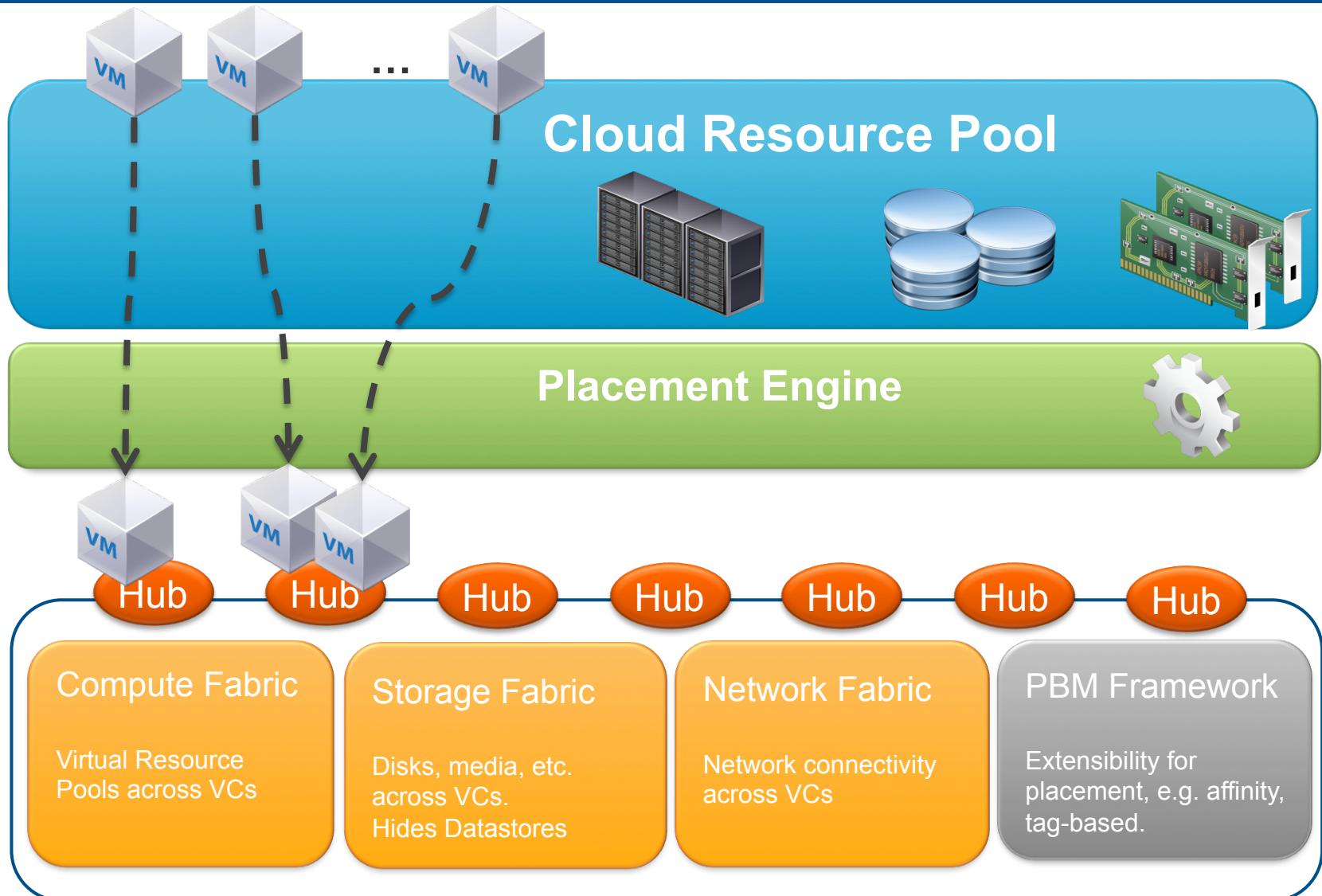


Virtual Data Center



- Environment for provisioning Cloud workloads
- Hides underlying physical resources. Takes SLAs as input.
- A container for “promised” resources. Tallies total/used.
- Leverages the Cloud Resource Pool to maintain SLAs for the various resource types

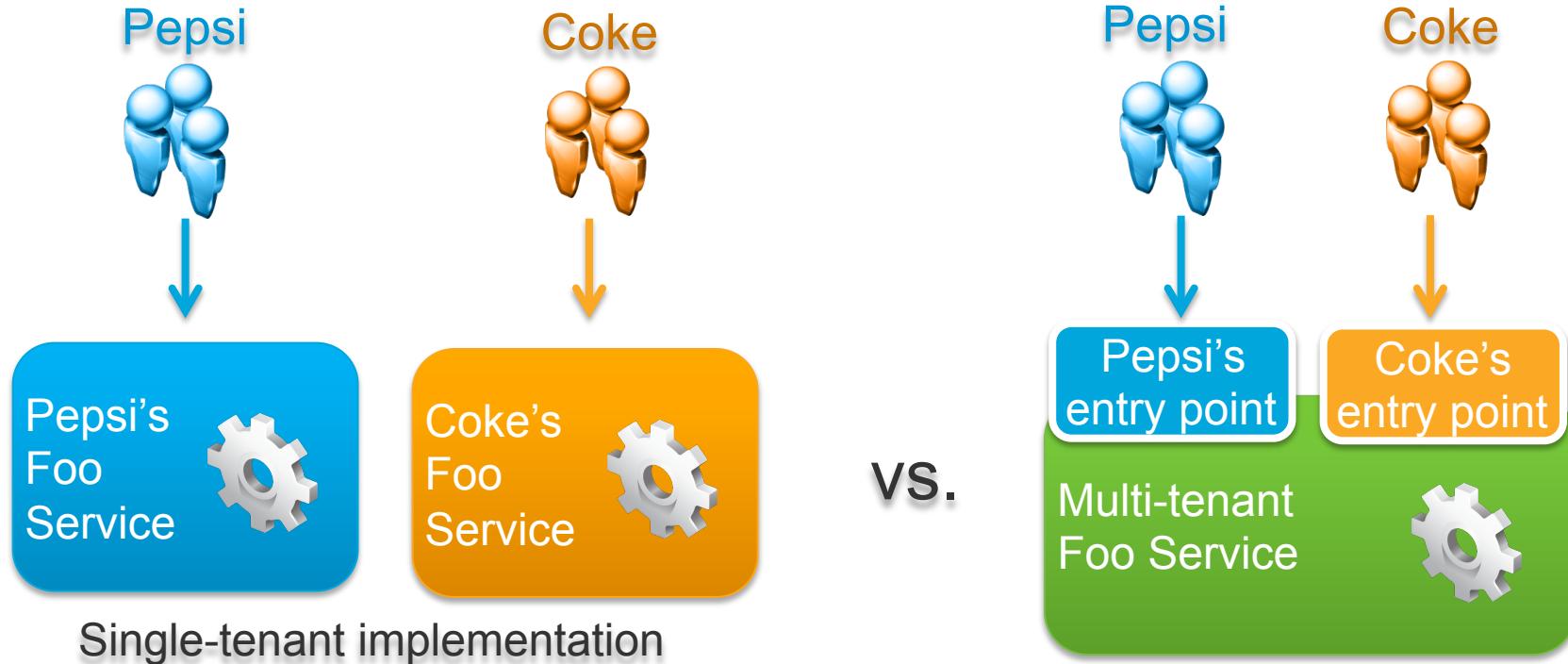
How Cloud Resource Pools Work



Hubs, Fabrics and Placement

- **A fabric...**
 - Manages storage, networking or compute resources across a site (multiple VCs)
 - Is responsible for provisioning workloads relative to its type of resource
 - Hides the underlying topology. Only tells which workload can go to which hub
- **A hub...**
 - Defines a logical location at which workloads can be placed. In 2014 a hub is a VC cluster or a stand-alone host
 - Recognized by all fabrics
- **PBM Framework: behaves as a fabric. Used for extensibility**
- **Placement Engine**
 - Negotiates placement (Workload → Hub map) that meets all workload requirements for all fabrics.
 - If needed, does re-placement of workloads in response to SLA violation

Multi-tenancy



- Artifact of the implementation and independent of functionality
- Not about cleverly designed permissions but a kind of service virtualization
- In CIS, the entity corresponding to tenant is called “Org”

Unification (aka Suitification)

How the many parts form a whole



Unification Principles

- Unity in concepts, UI, API and Deployment
- Common services: common obligations and common benefits
- Freedom in all else: e.g. components' choice of technology, architecture and release methodology.
- The same principles foster modularity and extensibility.

Unity of Concepts

- **VM.** Unity of VM identity across all components.
- **Tenant and user.** Unity of tenant and user identities all components.
- **Site.** Common site notion as maintained by the *Component Manager*
- **Common Authentication model and protocols.** VMware SSO.
- **Common queriable inventory of entities.** *Inventory Service*.
- **Common Authorization model: {principle, role, entity} tuples.**
Authorization Service.
- **Common operational model. Events, tasks, alerts, stats, logs.**
Event Service, Task Service, Stats & Alerts Service, Logging Service

CIS UI – Overview

- **The one UI shipped standard with CIS**
- **Built on Serenity fwk (aka NGC). Merge of the vCenter and VCD UIs**
- **Currently built in Flex; overtime to be transitioned to HTML5**
- **Personas (who it is for)**
 - Central IT: Buys, installs and operates the suite. Enterprise or Cloud SP admin
 - Developer: Builds solutions for CIS
- **Roles (group of functionality for one or more personas)**
 - Cloud Admin: Manages physical infrastructure. Creates CRPs, Storage Classes, Provider Networks, Orgs and VDCs.
 - Org Admin: Creates VMs, vApps, Networks within an Org. User administration within an Org. Owner of the IaaS account.
 - End-user: Operates VMs. A subset of the Org Admin role.

CIS UI – Design

- **Coud Admin Role: Always done by the Central IT persona.**
 - The CIS UI will be the primary client for all this functionality and offers task-oriented style UI.
 - Also covers the Cloud Provider scenario. Designed for large scale management.
- **Org Admin & End User roles:**
 - Continue to offer classic vCenter provisioning.
 - Full support for all platform features.
 - Power-user interface for the central IT personal as well as for developers
 - Self-service portals and other solutions provide alternative UIs for other personas.

2013 – Home

VMware vSphere Web Client - Microsoft Internet Explorer

http://vmware/vSphereWebClient

File Edit View Favorites Tools Help

Favorites Suggested Sites Bugzilla ESS-TechOps KB HelpZilla VMware Wiki VMweb Home Web Slice Gallery

VMware vSphere Web Client

Home Getting Started Home Inventories vCenter Hosts and Clusters VMs and Templates Storage Networking vCenter Orchestrator Ping Networks Cubic Storage ACME Blades Monitoring Task Console Event Console Host Profiles VM Storage Profiles Customizations Administration Roles Licensing Solutions

Extensions

Local intranet 100%

2014 – Home

VMware vSphere Web Client - Microsoft Internet Explorer

http://vmware/vSphereWebClient

File Edit View Favorites Tools Help

Favorites Suggested Sites Bugzilla ESS-TechOps KB HelpZilla VM

VMware vSphere Web Client

Cloud is off
Can turn Cloud on

vmware vSphere Web Client

Home Getting Started Home Workflow

Inventories

- Cloud (highlighted with a red box)
- vCenter
- Hosts and Clusters
- VMs and Templates
- Storage
- Networking

vCenter Orchestrator Ping Networks Cubic Storage ACME Blades

Monitoring

- Task Console
- Event Console
- Host Profiles
- VM Storage Profiles
- Customizations

Administration

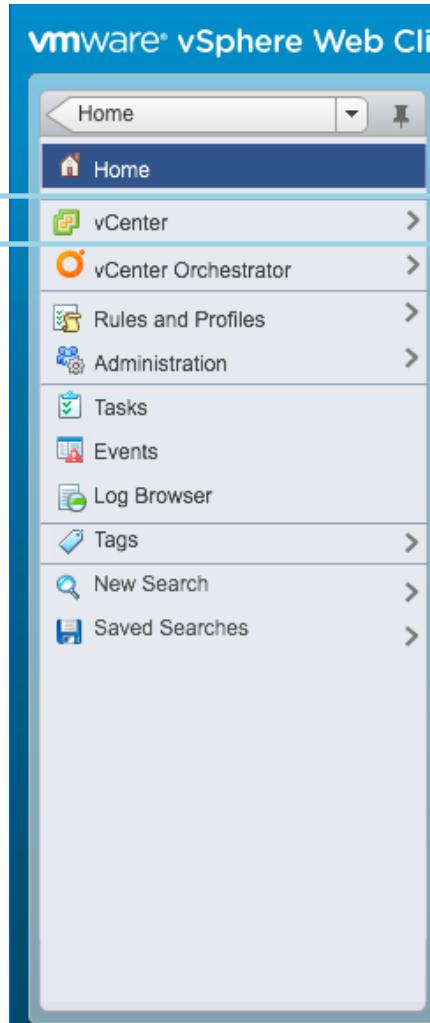
- Roles
- Licensing
- Solutions

Local intranet 100%

The screenshot shows the VMware vSphere Web Client interface in Microsoft Internet Explorer. The main navigation bar includes File, Edit, View, Favorites, Tools, and Help. Below the bar are favorite links for Bugzilla, ESS-TechOps KB, HelpZilla, and VM. The title bar reads "VMware vSphere Web Client - Microsoft Internet Explorer" and the address bar shows "http://vmware/vSphereWebClient". The left sidebar has a "Home" link selected, followed by "vCenter", "vCenter Orchestrator", "Rules and Profiles", "Administration", "Tasks", "Events", "Log Browser", "Tags", "New Search", and "Saved Searches". The main content area is titled "Home" and contains sections for "Inventories", "Monitoring", and "Administration". In the "Inventories" section, the "Cloud" icon is highlighted with a red box. A red callout box with the text "Cloud is off" and "Can turn Cloud on" points to this icon. Other icons include vCenter, Hosts and Clusters, VMs and Templates, Storage, Networking, vCenter Orchestrator, Ping Networks, Cubic Storage, ACME Blades, Task Console, Event Console, Host Profiles, VM Storage Profiles, Customizations, Roles, Licensing, and Solutions. The bottom status bar shows "Local intranet" and "100%".

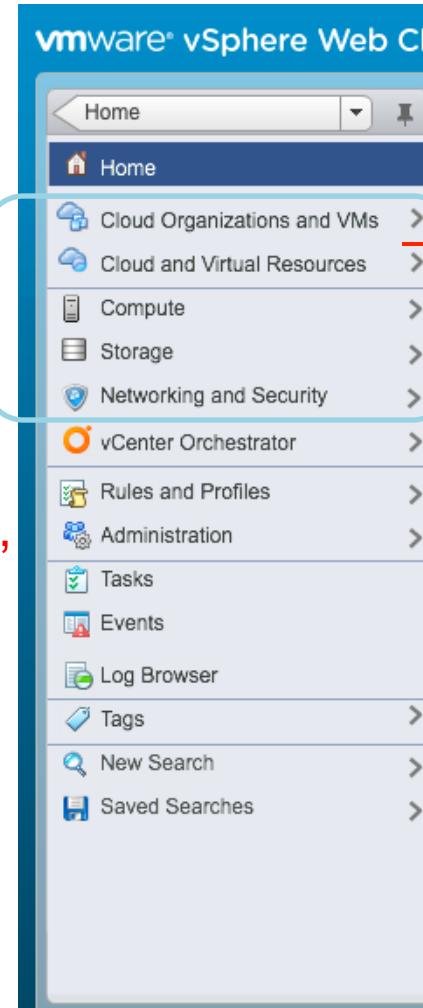
Before and After Cloud Is On

Cloud Off



vCenter
“Fans Out”

Cloud On



VCD and
vCenter Objects
Seamlessly
Combined

2014 – Home – Cloud On

VMware vSphere Web Client - Microsoft Internet Explorer

http://vmware/vSphereWebClient

File Edit View Favorites Tools Help

Favorites Suggested Sites Bugzilla ESS-TechOps KB HelpZilla VMware Wiki VMweb Home Web Slice Gallery

VMware vSphere Web Client

Home Getting Started Home Workflow

Inventories

- Cloud Organizations and VMs
- Cloud and Virtual Resources
- Compute
- Storage
- Networking and Security
- vCenter Orchestrator
- Rules and Profiles
- Administration
- Tasks
- Events
- Log Browser
- Tags
- New Search
- Saved Searches

Cloud Organizations and VMs Cloud and Virtual Resources Compute Storage Networking and Security Inventory Trees

vCenter Orchestrator Ping Networks Cubic Storage ACME Blades

Monitoring

- Task Console
- Event Console
- Host Profiles
- VM Storage Profiles
- Customizations

Administration

- Roles
- Licensing
- Solutions

Local intranet 100%

2014 – Cloud and Virtual Resources

VMware vSphere Web Client - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Favorites Suggested Sites Bugzilla ESS-TechOps KB HelpZilla VMware Wiki VMweb Home Web Slice Gallery

VMware vSphere Web Client

Home Cloud Resources Home

Cloud and Virtual Resources

- Cloud Resources Home
- Virtual Resources Home
- Cloud Inventory Lists
 - Cloud Resource Pools 1 >
 - External Networks 1 >
- Virtual Inventory Lists
 - vCenter Servers 2 >
 - Datacenters 4 >
 - Folders 8 >
 - Clusters 8 >
 - Resource Pools 5 >
 - Datastore Clusters 2 >
 - Distributed Switches 2 >
- Virtual Inventory Trees
 - Hosts and Clusters >
 - Storage >
 - Networking >

Cloud Resources Manager
Version 6.0
Check for updates

CPU 15 GHz
3 GHz USED 12 GHz FREE

Memory 12 GB
4 GB USED 8 GB FREE

Storage 700 GB
600 GB USED 100 GB FREE

USED FREE CAPACITY

Getting Started Actions

- Create new cloud resource pool
- Create new organization
- Create new external network

Actions

- Add capacity to a cloud resource pool
- Allocate more resource to an organization
- Add a catalog to an organization

VCD objects

VC objects

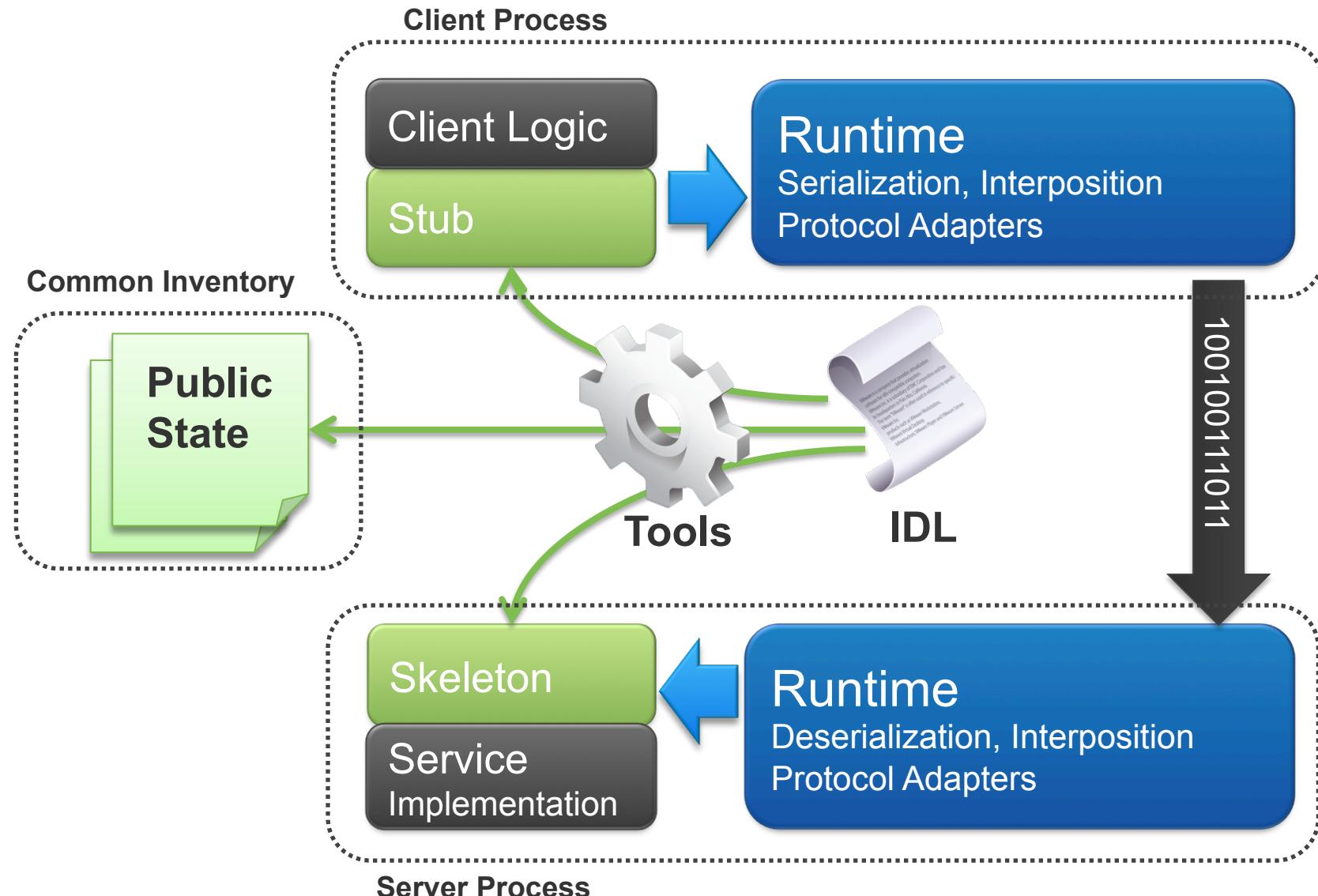
Local intranet 100%

The screenshot shows the VMware vSphere Web Client interface for managing cloud resources. The left sidebar lists various inventory categories. The main pane displays the 'Cloud Resources Manager' version 6.0. It features resource usage summary cards for CPU, Memory, and Storage, each with a blue 'USED' bar and a grey 'FREE' bar. Below these are 'Getting Started Actions' and 'Actions' sections, both containing links to perform tasks like creating new pools or adding catalogs. Red arrows and text annotations highlight specific sections: one arrow points to the 'Actions' section, another to the 'Getting Started Actions' section, and red text labels 'VCD objects' and 'VC objects' are placed over these sections respectively.

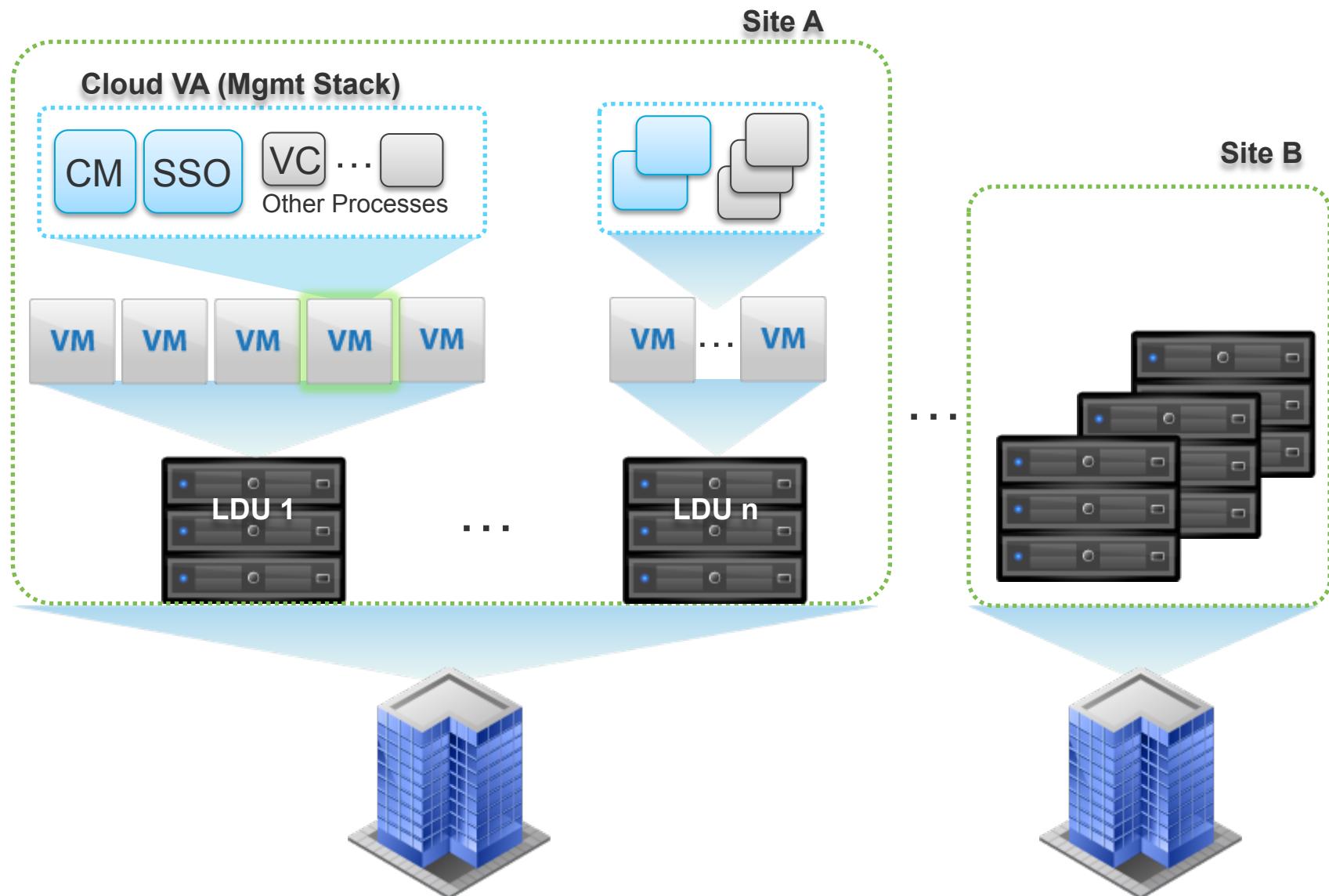
CIS API – Overview

- **IDL, runtime libraries, tools, and an API Endpoint component**
- **VMODL2 language (the IDL)**
 - The one-stop place for defining interfaces and queriable public state.
 - Resources-oriented (not OO), modular and extensible.
 - Improved versioning, extensibility and modularity.
 - Intended for use by 3rd parties
 - Evolved from classic VMODL to allow for smoother transition.
- **Runtime.**
 - Enables modularity through dynamic type loading
 - Integration with common services: Authorization, Tasks, Inventory Service.
 - Supports REST.
- **Tools: Parser not based on Java APT. Usable by 3rd parties**
- **API Endpoint: Single entry point for external API requests**

CIS API – IDL, Runtime and Tools



Deployment Vehicles and Core Components



Deployment Model

- **Deployment Components:** vCenter, SSO, IS, CM, etc.
 - Each has one or more execution nodes.
 - An execution node maybe its own process or part of a container process, e.g. TC server.
- **Cloud VA:** One or more VMs that run the deployment components
- **Logical Deployment Unit (LDU):** Hardware + Cloud VA; also runs user VMs.
- **Cloud VA and LDUs are deployment artifacts**
 - Vehicles for orderly deployment
 - In general, component design should be agnostic of these vehicles. Thus, deployment and scale are independently designed.
- **Note:** VC Linked mode is supplanted by the federation of component nodes, which apply to all components and not just VC

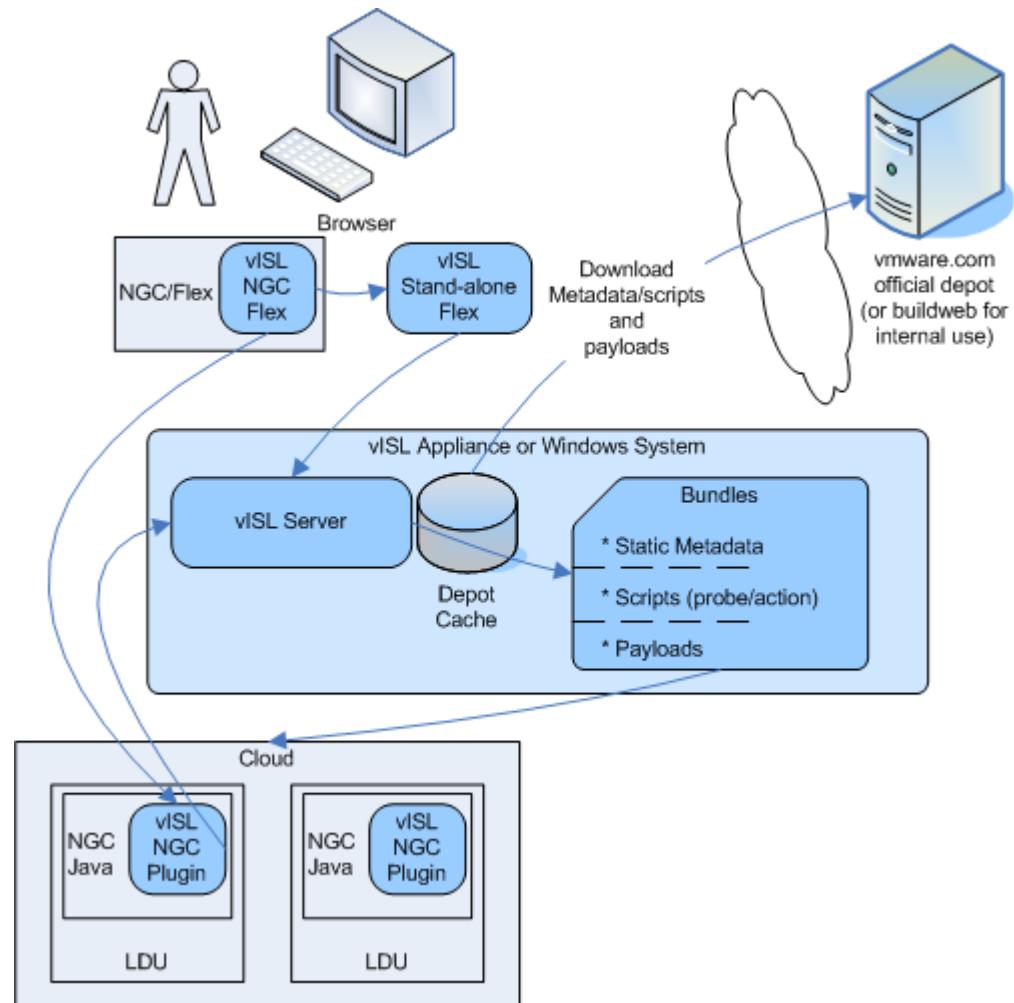
Deployment Framework and Lifecycle

■ VMware Infrastructure Software Lifecycle (ViSL) Framework

- Understands dependences and creates plans
- Modular: per-component bundles
- Windows and Cloud VA form factors

■ Life-cycle operations

- Install/Upgrade/Patch/Uninstall
- Backup/Restore
- LDU join/merge/split



Common Services – Obligations and Benefits

Service	Description	Components must...	Components benefit by...
VMware SSO	Identity management	Use for user/group authentication	Leverage existing authN mechanism
Component Manager	Components and services discovery	Register themselves	Lookup other components
Authorization Service	Setup and enforce Authorization policies	Use for access control	Leverage existing authZ mechanism
Inventory Service	Publish/query the state of the system	Publish inventory state.	Complex queries on the state of CIS
Task Service	Publish/query/archive tasks	Publish tasks	Query tasks, monitor progress
Key-value Store	Distributed and scalable store for private data	Use for site-wide and global state	Shared implementation

Also, 10+ more. For a complete list see:

https://wiki.eng.vmware.com/CIS/APOR#Common_Services

vCenter and VCD Modularity

- **Unification within the suite fosters modularity**
- **vCenter modularity effort**
 - DRS, HA, Licensing, Compatibility Checker become separate components outside of VPXD.
 - Legacy access is achieved through an API Proxy; pretends VC is still one box.
 - Stats is moving out of VC to VCOps
 - Authorization is moving to a common Authorization service.
- **VCD modularity effort**
 - Aggregate compute, storage and networking functionality moves to the respective fabrics.
 - VCD sheds its own inventory cache in favor of Inventory Service.
 - VCD authorization moves to the common Authorization service
 - ...

Scale

Where the system lives



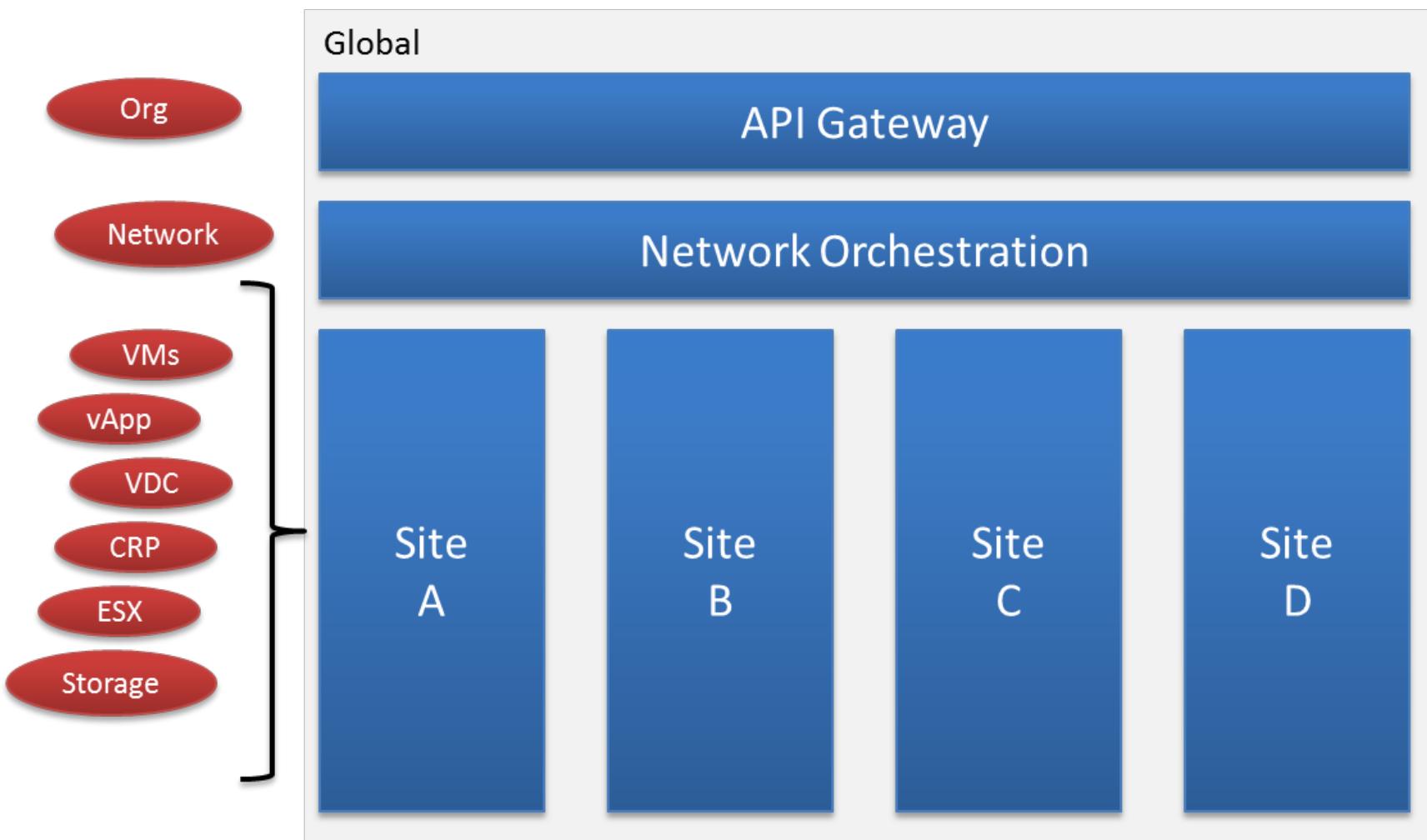
Scalability and Availability – Definitions

- **Entities**: Things the system knows about and can manipulate
- **Node**: Deployment component, e.g. Linux process inside a VM
- **Service**: Functional component; implements operations on entities
- **Availability**: How well the system can cope with node failures or network partitions.
- **Linear scale**: Increasing system resources yields proportionate increase in system's capacity to handle workload.

Scalability and Availability – Principles

- **No single point of failure**
- **No bottlenecks**
- **Update operations should be local**
- **Trade off between consistency and scale and between consistency and availability**
 - Local transactions
 - Eventual consistency
 - Bounded entities
- **End design is a combination of various trade-offs and not an uniform solution.**

The Site Concept – In Context



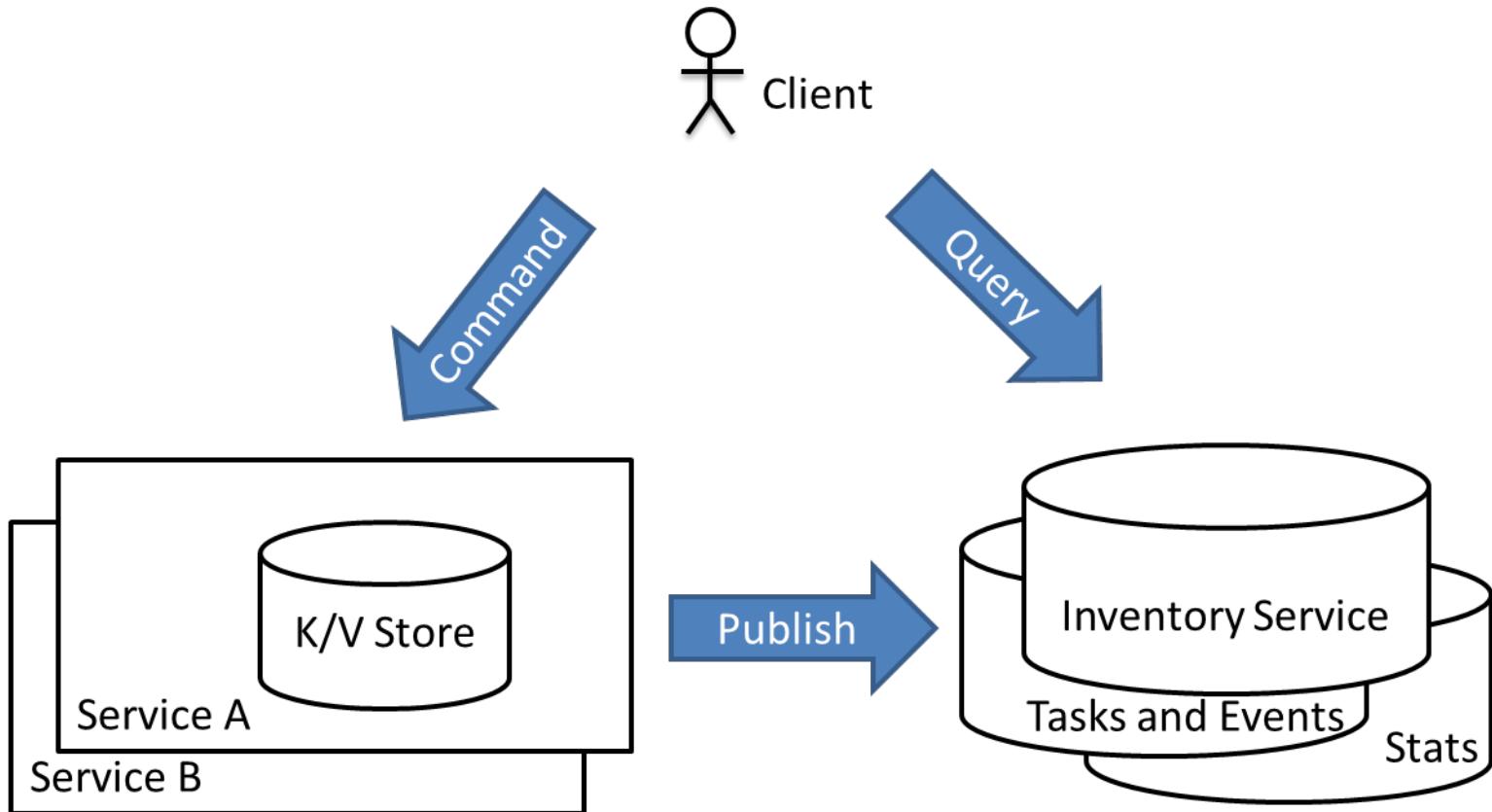
The Site Concept – Definition

- Dedicated hardware, high-bandwidth/low-latency, network partitions are short-lived and rare
- Administrative domain: administered independent of other sites.
- Fault domain: failure of one site does not affect other sites
- Operation domain: Operations must take place within a site
- Deployment domain: A service is deployed to a site
- May federate with other sites, but is managed independently
- Visible construct to all users in the system

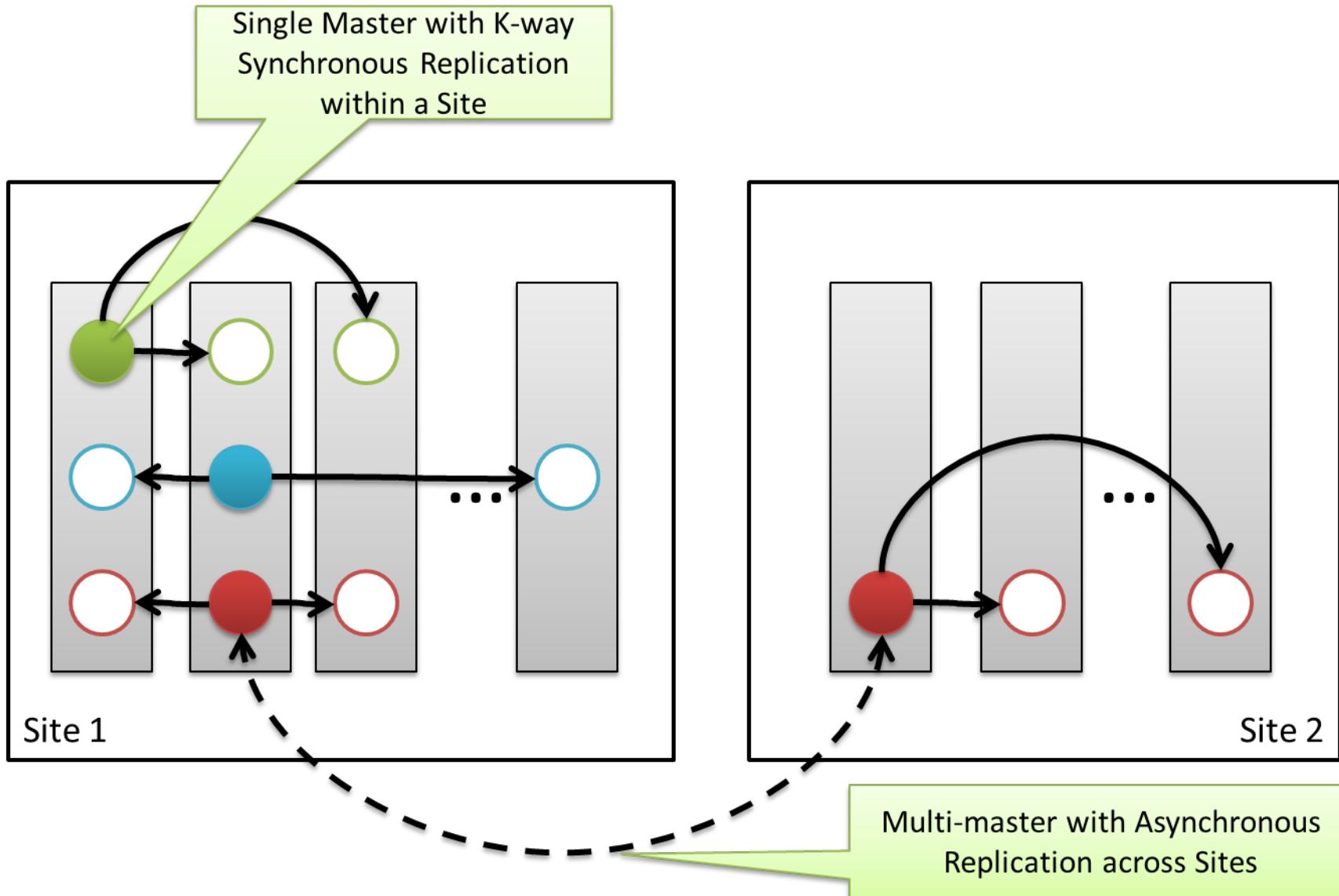
Basic Approach

- **Confine Consistency to within an Entity or a Node**
- **Activities with Idempotent Operations instead of Dist. Transactions**
- **Higher degree of consistency within a site**
 - Single-master with K-way synchronous replication
 - Upside: Simpler to write and test code
 - Downside: low-latency assumption limits geographical growth.
 - Downside: synchronous replication assumption means lower network partition tolerance.
- **Greater geographical scale and availability across sites**
 - Multi-master with async replication between sites; single master per-site
 - Upside: Greater availability and geographical scale.
 - Downside: Complex implementation logic to deal with conflicting updates.
 - Sites are visible to clients of the system.

Command Query Responsibility Separation



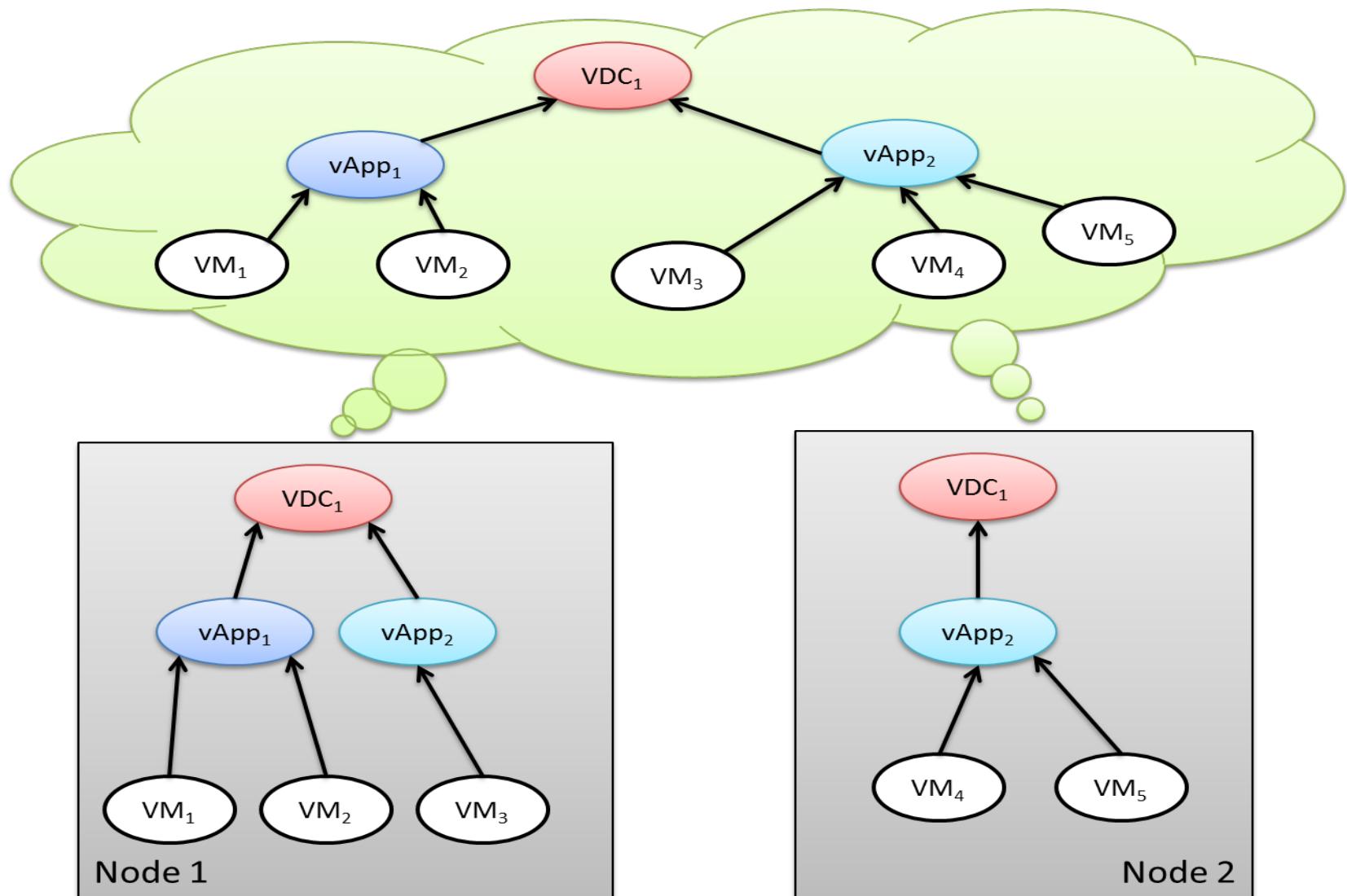
Key-Value Store (KV Store) – Overview



KV Store – Stylized API

- **KV_putSite(key, [column], value, version)**
KV_getSite(key, [column])->(value, version)
 - Synchronously replicates Values K-ways where K is a system-wide configurable constant that does not grow with the number of nodes. If the synchronous replication to a quorum number of nodes fails, then the operation will fail. (R=W=K)
- **KV_putGlobal(key, column, value)**
KV_getGlobal(key, column)->value
 - Data is replicated asynchronously to all other Sites. Due to async. replication, concurrent writes will have to be merged by the application level.
- Any node can service put/get irrespective of master, slave or neither
- Notifications for Key-value changes

Inventory Service (IS) Publishing and Query – Overview

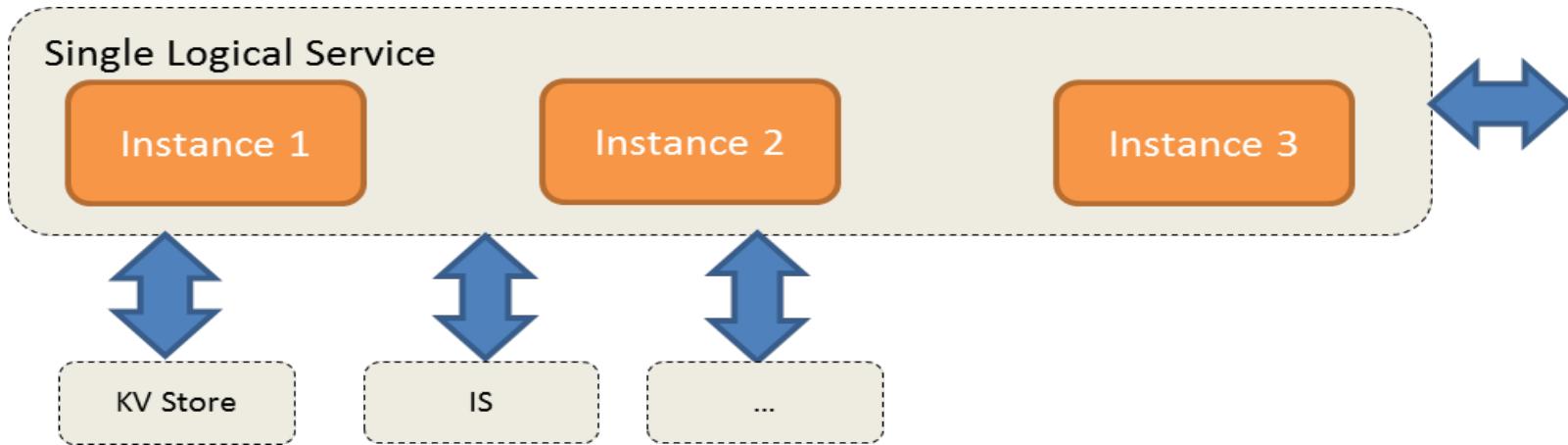


Inventory Service – Semantics

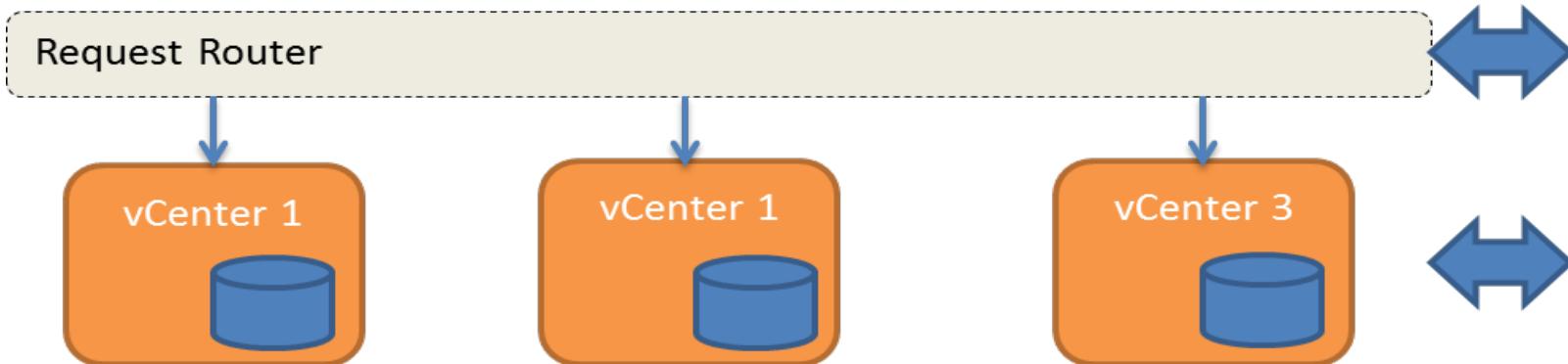
- Single logical service that can store and query large graphs and can scale horizontally within a site.
- Neither the publisher nor the query client should be aware of how the data is partitioned.
- Stylized API
 - IS_Publish(resource-id, [consistency-group-id], document-data)
 - IS_Query(query_spec)->Result
- Eventual consistency
 - Data published to IS may not immediately appear in query results
 - Querying twice may return slightly older result than the first query
 - Rule of thumb: publish to IS for others and not for yourself
- Task/query synchronization: for specific entities counters the effect of data fluctuation

Patterns for Service Scalability and Availability

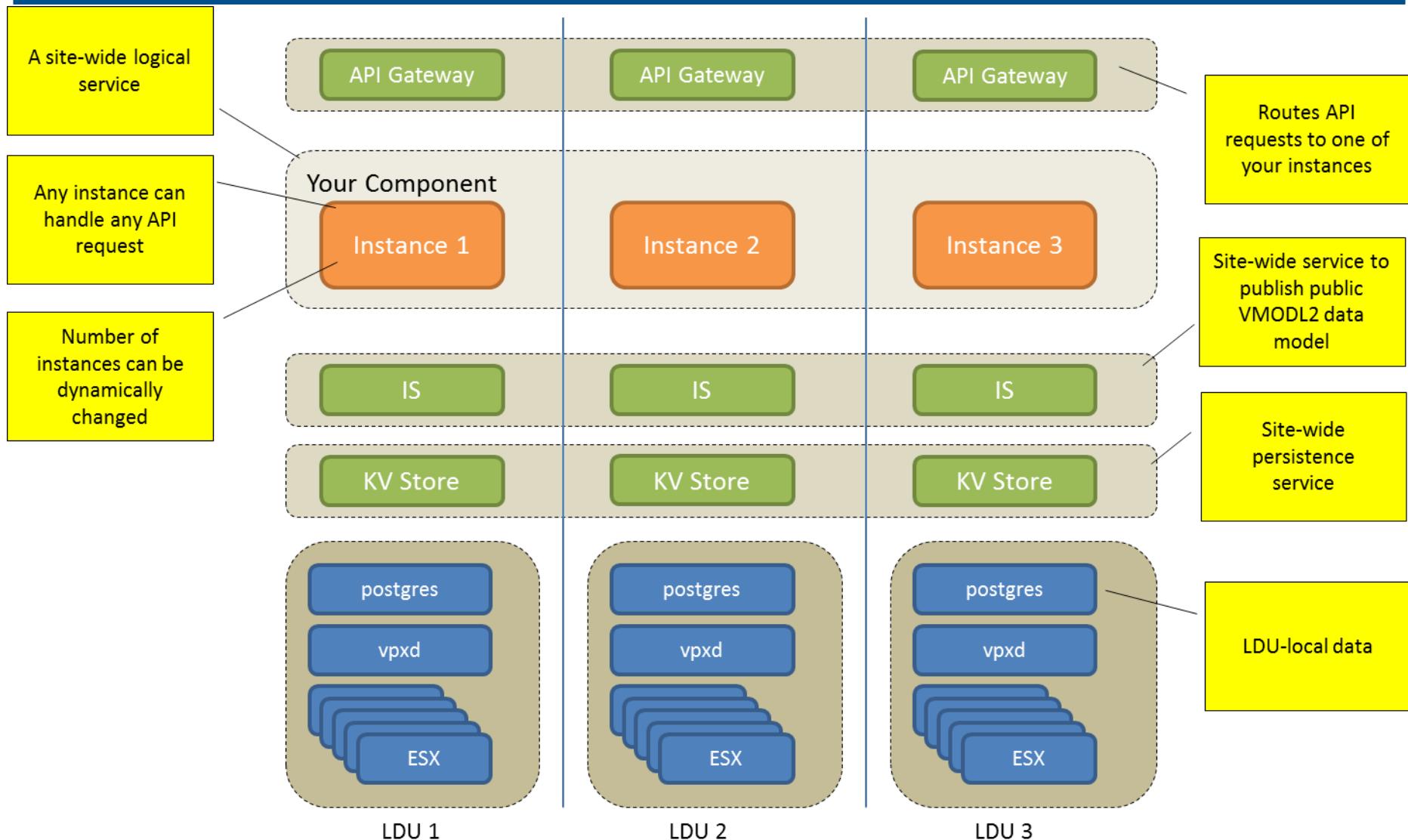
- Decouple state from execution: Single logical service per site
 - Examples: VCD, Storage & Network Fabrics...



- Compartamentalize execution by state: Node consistent pattern



The role of the Logical Deployment Units



Not all components within an LDU are shown above. This presentation is focusing on the use of the API gateway, IS, and KV store

For Further Reading

refer to the architectural plan of record:

<https://wiki.eng.vmware.com/CIS/APOR>



Targeting the Future: Cloud Management Platform

Vertical, Horizontal, and Scale

