

vmware®



# **vinNOVATION**

THIS IS HOW TOMORROW IS BUILT

**VMware Developer Cloud**





# Make the Clouds Invisible

## BUSINESS OUTCOMES



Make the private cloud easy to operate



Enable developers



Manage across clouds.  
On-premises & cloud based delivery

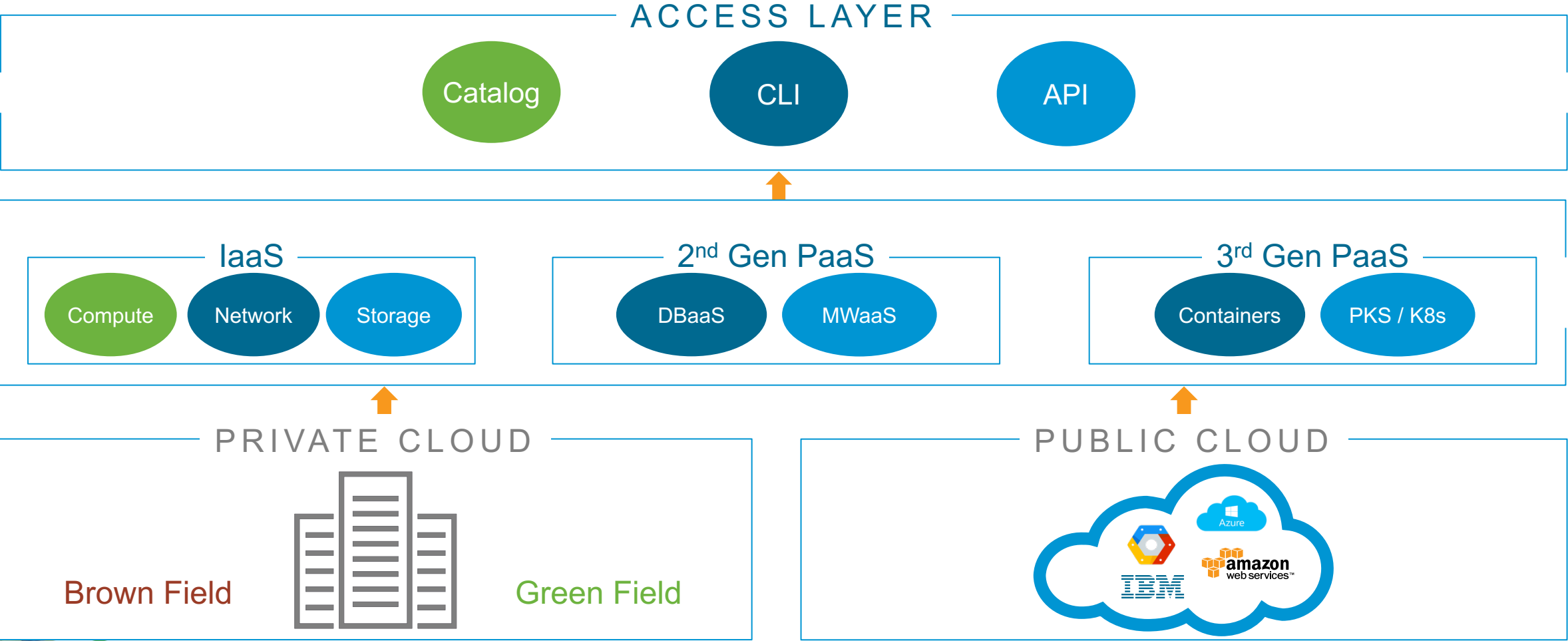
Easy to  
Operate

Built for  
Developers

Cross-cloud  
& SaaS

## AREAS OF INVESTMENT

# A Simplified View Of Your Cloud Journey End State





## VMware Cloud

Run, Manage, Connect, Secure Any App on Any Cloud to Any Device



### Existing Apps

Reduce Costs • Security • Reliability • Control

### Cloud Native Apps

Time to market • Innovation • Scale • Differentiation

## VMware Cloud Services



VISIBILITY



OPERATIONS



AUTOMATION



SECURITY



GOVERNANCE

## Cloud Management Platform



## VMware Cross-Cloud Architecture



## Public Cloud IaaS



vmware  
Cloud Foundation

amazon  
webservices™  
VMware Cloud on AWS

IBM Cloud  
for VMware

vmware  
Cloud Providers

OVH.com

amazon  
webservices™

Microsoft  
Azure

IBM Cloud



Google Cloud Platform

virtustream

## Consistent Infrastructure

VM Infrastructure • Container Infrastructure

## Consistent Operations

Management and Operations • Across Clouds

vmware

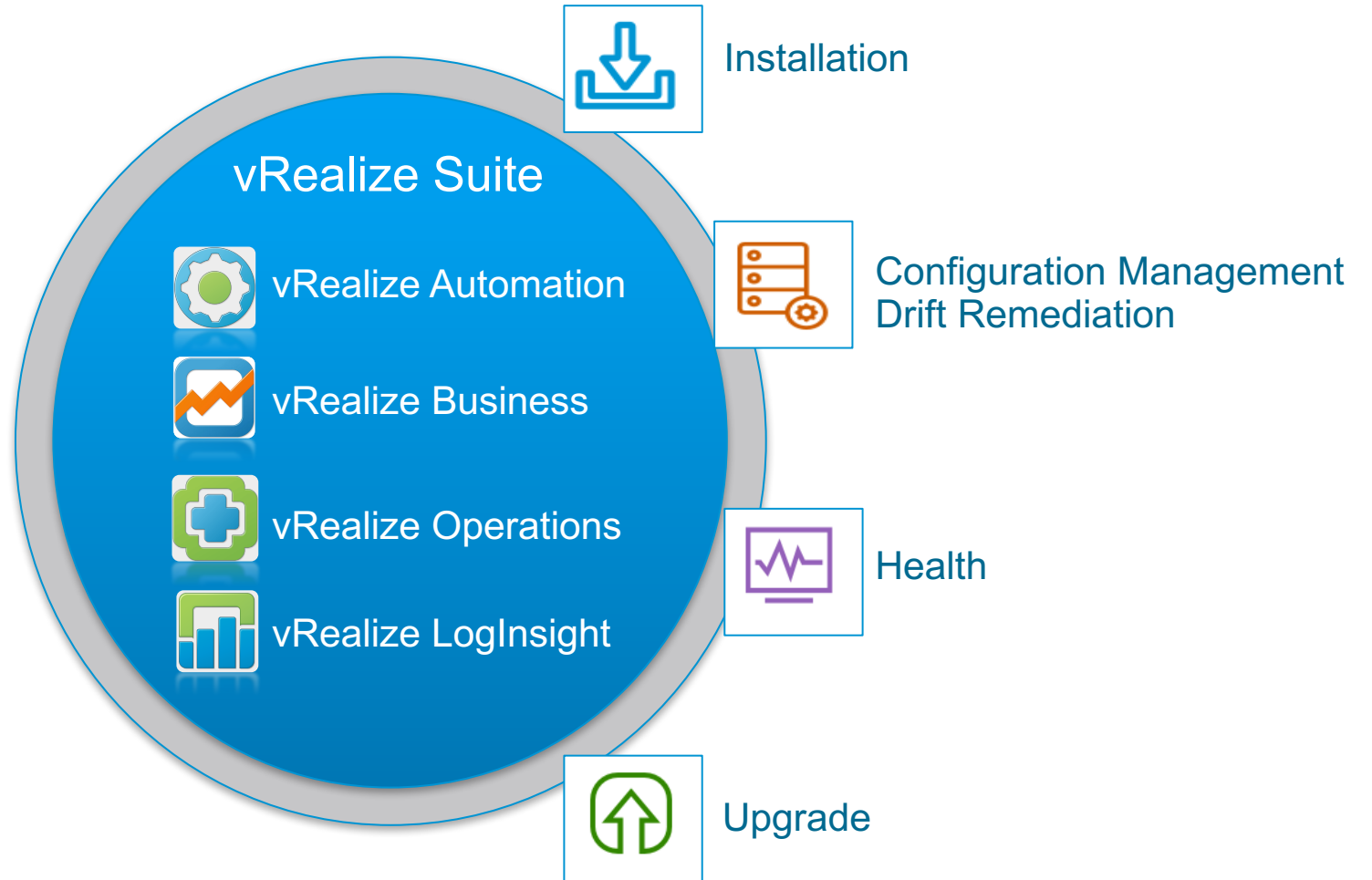


# vRealize Suite Lifecycle Manager

*Automated Lifecycle Management and Operations (Day 0 to Day 2)*

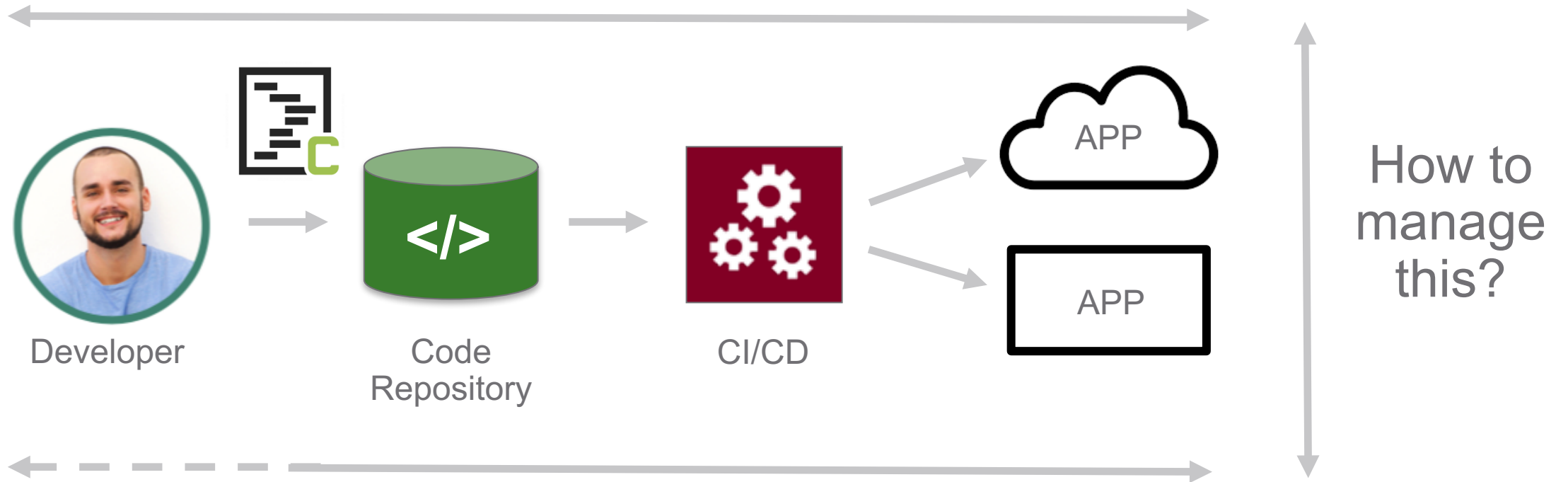


VMware vRealize Suite  
Lifecycle Manager



# DevOps From an Infrastructure Perspective

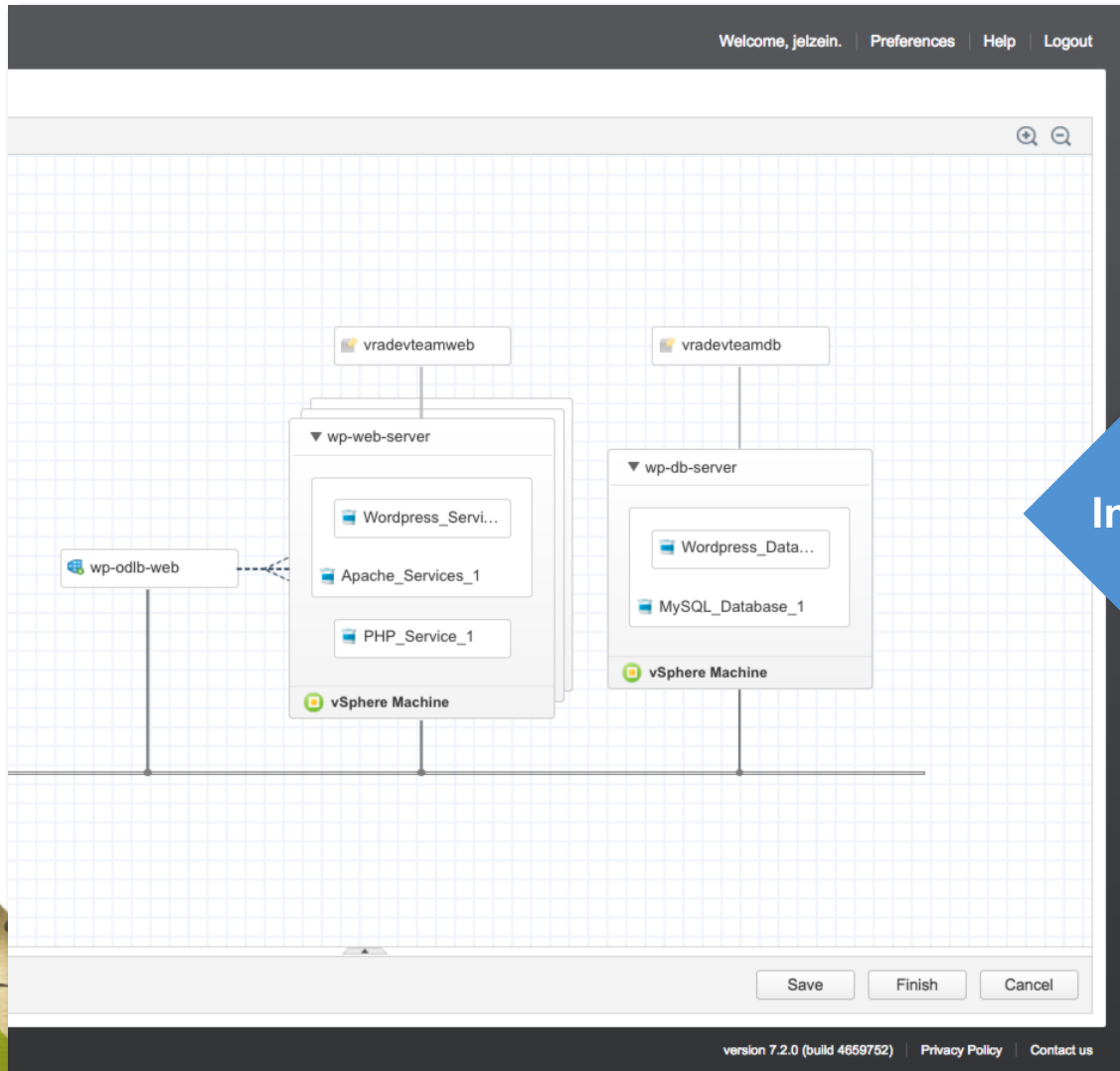
## DevOps in a Nutshell



Which infrastructure to use?

# Infrastructure as Code

Export | Import | Share



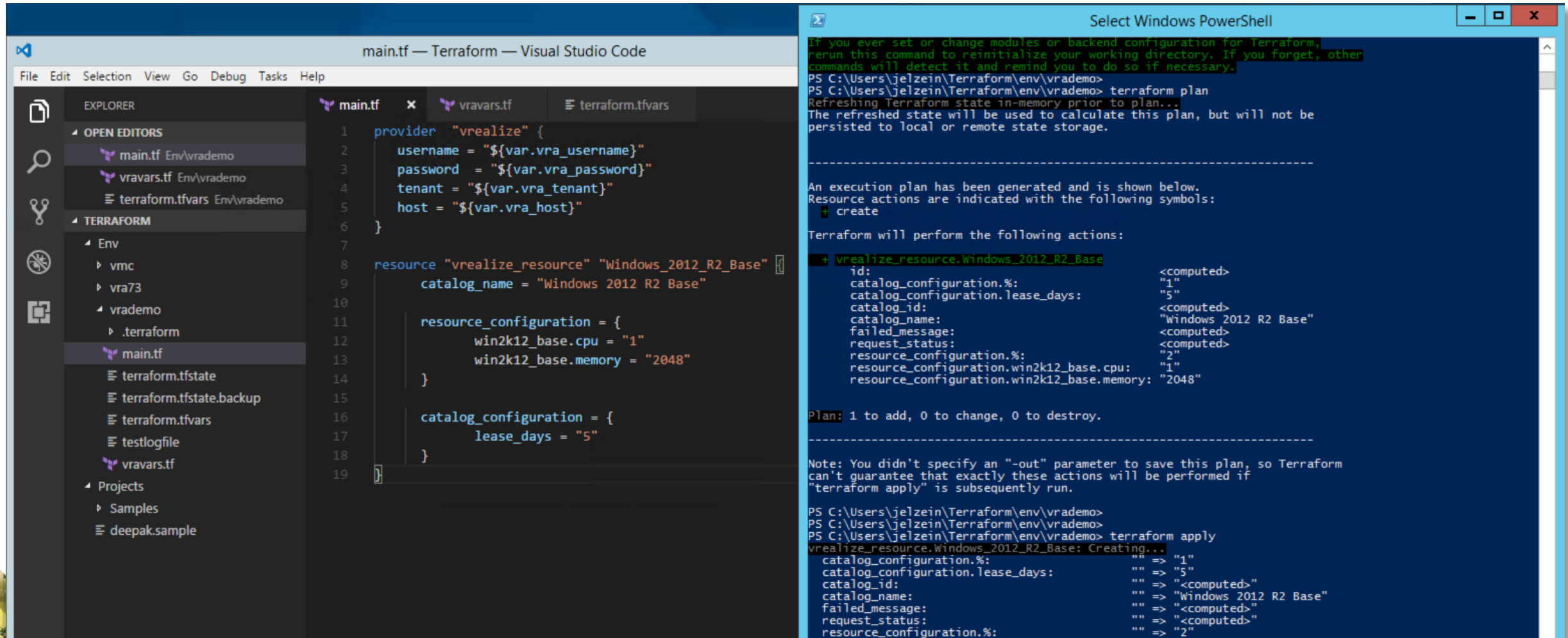
Interoperable

```
id: Wordpress.4.3.1
name: Wordpress 4.3.1
components:
  LB:
    type: Infrastructure.Machine.vSphere
    data:
      machine:
        cpu: {max: 4, default: 2}
        memory: 1024
  wpApache:
    type: Software.wpApache_1
    data:
      host: '${_resource~LB~machine}'
  ...
  wpMySQLDB:
    type: Software.wpMySQLDB_1
    data:
      db_port: 3306
  WebApp:
    type: data:wpMySQL_Config_1

    db_port: ${MySQL~db_port}
    db_username: ${DB_Setup~db_username}
  ...
```



# vRA Terraform Provider



The image shows a Visual Studio Code editor window titled "main.tf — Terraform — Visual Studio Code". The Explorer sidebar on the left shows a project structure with folders "Env", "vra73", "vrademo", and "Projects". The "vrademo" folder is expanded, showing files like ".terraform", "main.tf", "terraform.tfstate", "terraform.tfstate.backup", "terraform.tfvars", "testlogfile", "vravars.tf", and "deepak.sample". The "main.tf" file is open in the editor, showing Terraform configuration for the "vrealize" provider and a resource "vrealize\_resource" of type "Windows\_2012\_R2\_Base".

```
1 provider "vrealize" {
2     username = "${var.vra_username}"
3     password = "${var.vra_password}"
4     tenant   = "${var.vra_tenant}"
5     host     = "${var.vra_host}"
6 }
7
8 resource "vrealize_resource" "Windows_2012_R2_Base" {
9     catalog_name = "Windows 2012 R2 Base"
10
11     resource_configuration = {
12         win2k12_base.cpu = "1"
13         win2k12_base.memory = "2048"
14     }
15
16     catalog_configuration = {
17         lease_days = "5"
18     }
19 }
```

On the right, a PowerShell terminal window titled "Select Windows PowerShell" shows the output of the Terraform plan and apply commands. The output indicates that the plan was generated and the resource "vrealize\_resource.Windows\_2012\_R2\_Base" is being created.

```
PS C:\Users\jelzein\Terraform\env\vrademo> terraform plan
Refreshing Terraform state in-memory prior to plan...
The refreshed state will be used to calculate this plan, but will not be
persisted to local or remote state storage.

-----

An execution plan has been generated and is shown below.
Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

+ vrealize_resource.Windows_2012_R2_Base
  id: <computed>
  catalog_configuration.%: "1"
  catalog_configuration.lease_days: "5"
  catalog_id: <computed>
  catalog_name: "Windows 2012 R2 Base"
  failed_message: <computed>
  request_status: <computed>
  resource_configuration.%: "2"
  resource_configuration.win2k12_base.cpu: "1"
  resource_configuration.win2k12_base.memory: "2048"

Plan: 1 to add, 0 to change, 0 to destroy.

-----

Note: You didn't specify an "-out" parameter to save this plan, so Terraform
can't guarantee that exactly these actions will be performed if
"terraform apply" is subsequently run.

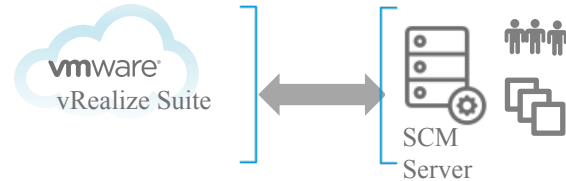
PS C:\Users\jelzein\Terraform\env\vrademo> terraform apply
vrealize_resource.Windows_2012_R2_Base: Creating...
  catalog_configuration.%: "" => "1"
  catalog_configuration.lease_days: "" => "5"
  catalog_id: "" => <computed>
  catalog_name: "" => "Windows 2012 R2 Base"
  failed_message: "" => <computed>
  request_status: "" => <computed>
  resource_configuration.%: "" => "2"
```

# Software Deployment Using Configuration Automation

Connect



Jason : Cloud Admin  
Setup and automate cloud infra



Register CM as an  
Endpoint

Discover Roles /  
Templates



Configure



Jason : Cloud Admin  
Setup and automate cloud infra

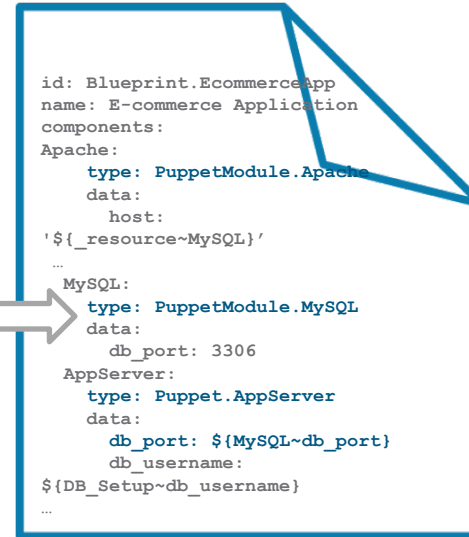
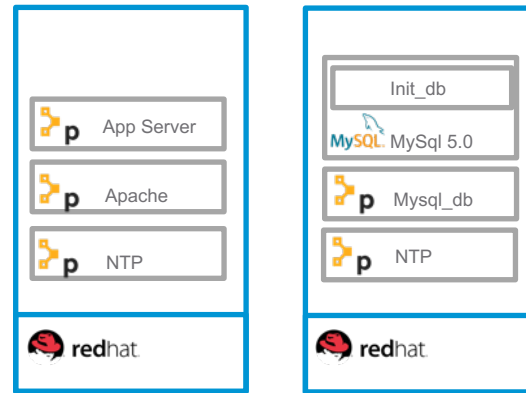


Shauna : DevOps Admin  
Provide & support dev env



Scott : Developer  
Write & deploy code

Blueprint



Manage



Shauna : DevOps Admin  
Provide & support dev env



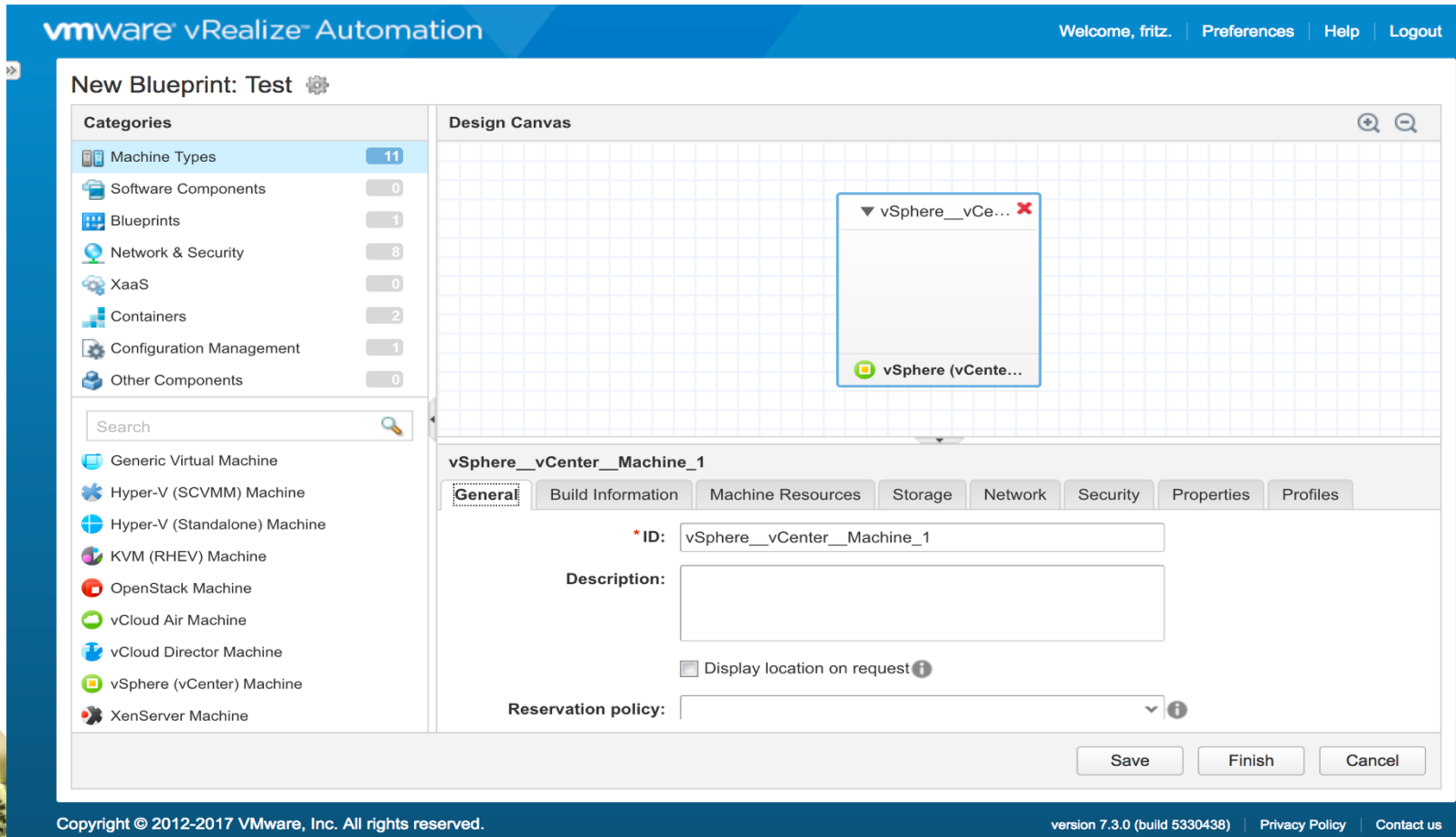
Scott : Developer  
Write & deploy code

Deregister Machine  
from CM Server

Change role of the  
machine

Drift Remediation

# Configuration Management as a “First Class Citizen”

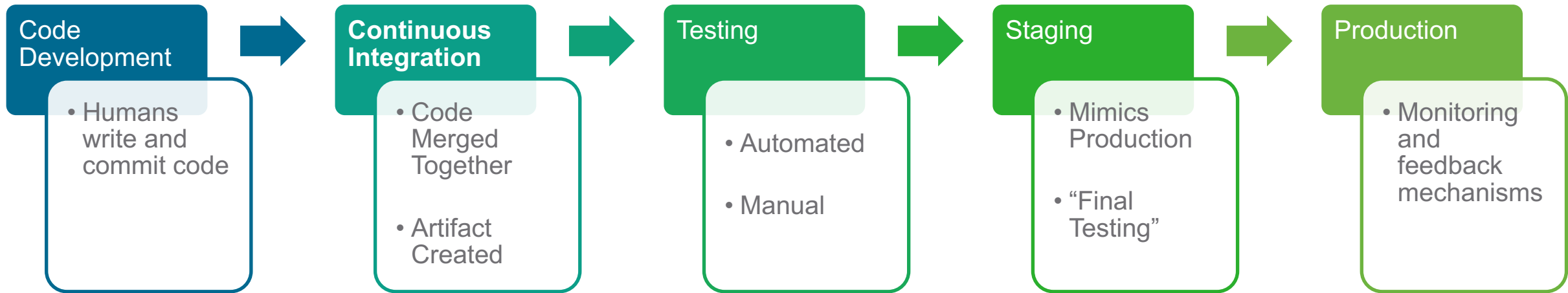


- Make the vRO plug-in actions **invisible**
- Configuration Management as one of the **GUI options** in Blueprint canvas
- Align with existing **workflows and governance**
- First Integration with **Puppet Enterprise**

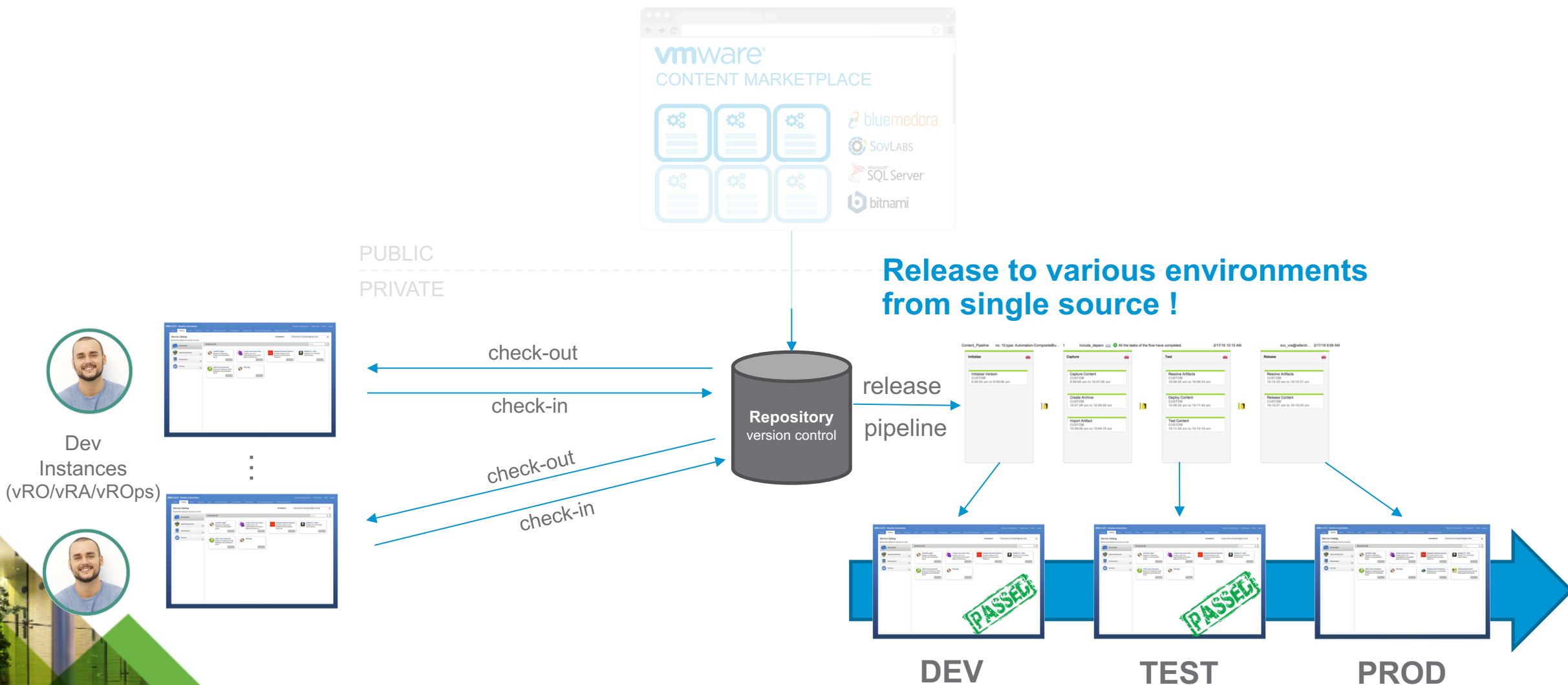


# CI/CD Pipeline

Recap a generic software development process



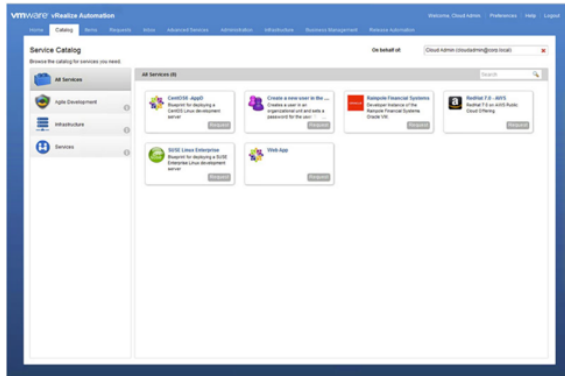
# Details: Content Versioning



# Enable Content Authoring with vRCS Management Pack

Houdini

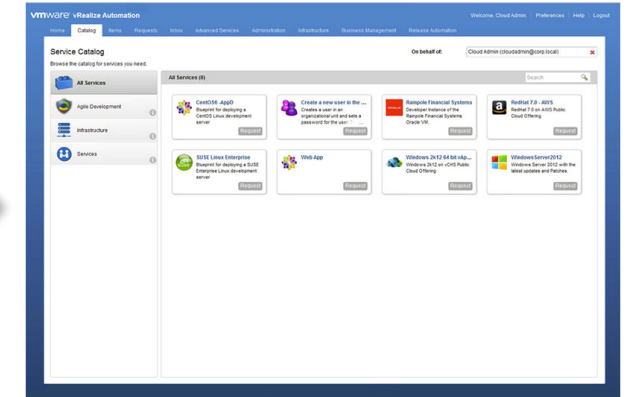
DEV/TEST



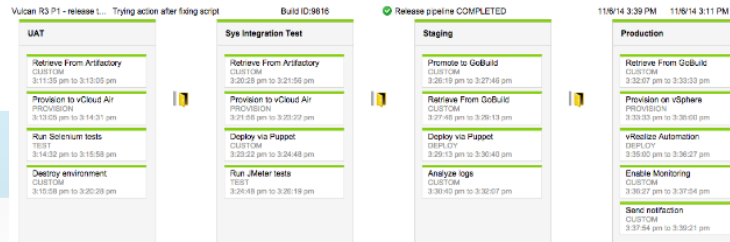
- Blueprints & Services
- Templates & workflows
- Configurations & scripts
- Recipes, manifests, etc.



PRODUCTION



Upstream



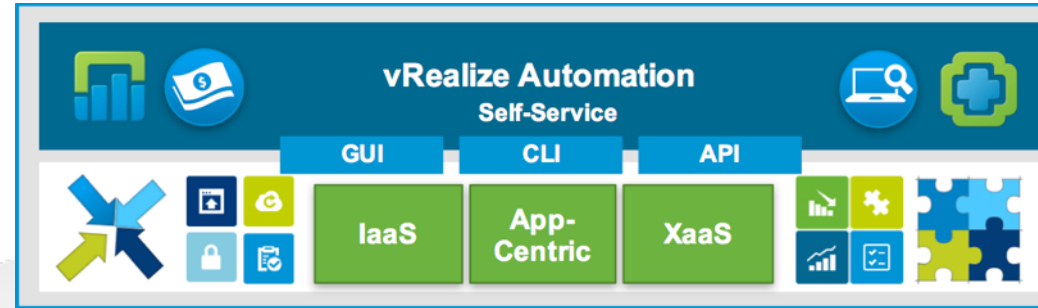
vRealize Code Stream

Downstream

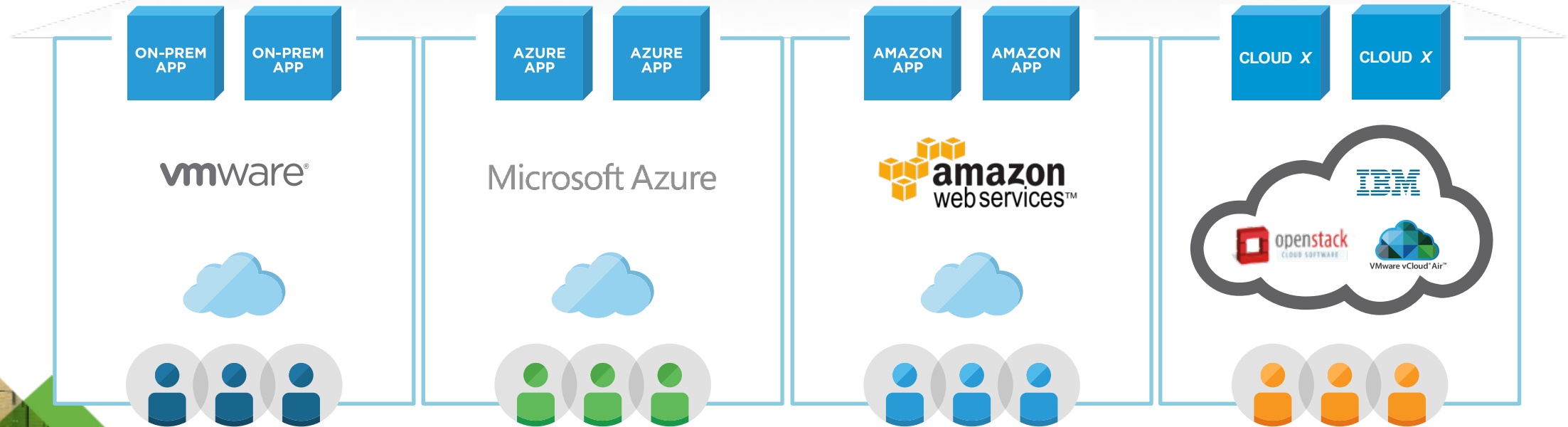


# Extending Public Cloud with vRA

Deliver Public Cloud Services to the Enterprise



- Federated Authentication
- Governance & Approvals
- Service Entitlements
- Catalog Based Self-Service
- LifeCycle Management



# vRA + AWS EC2

## Native AWS EC2 Integration



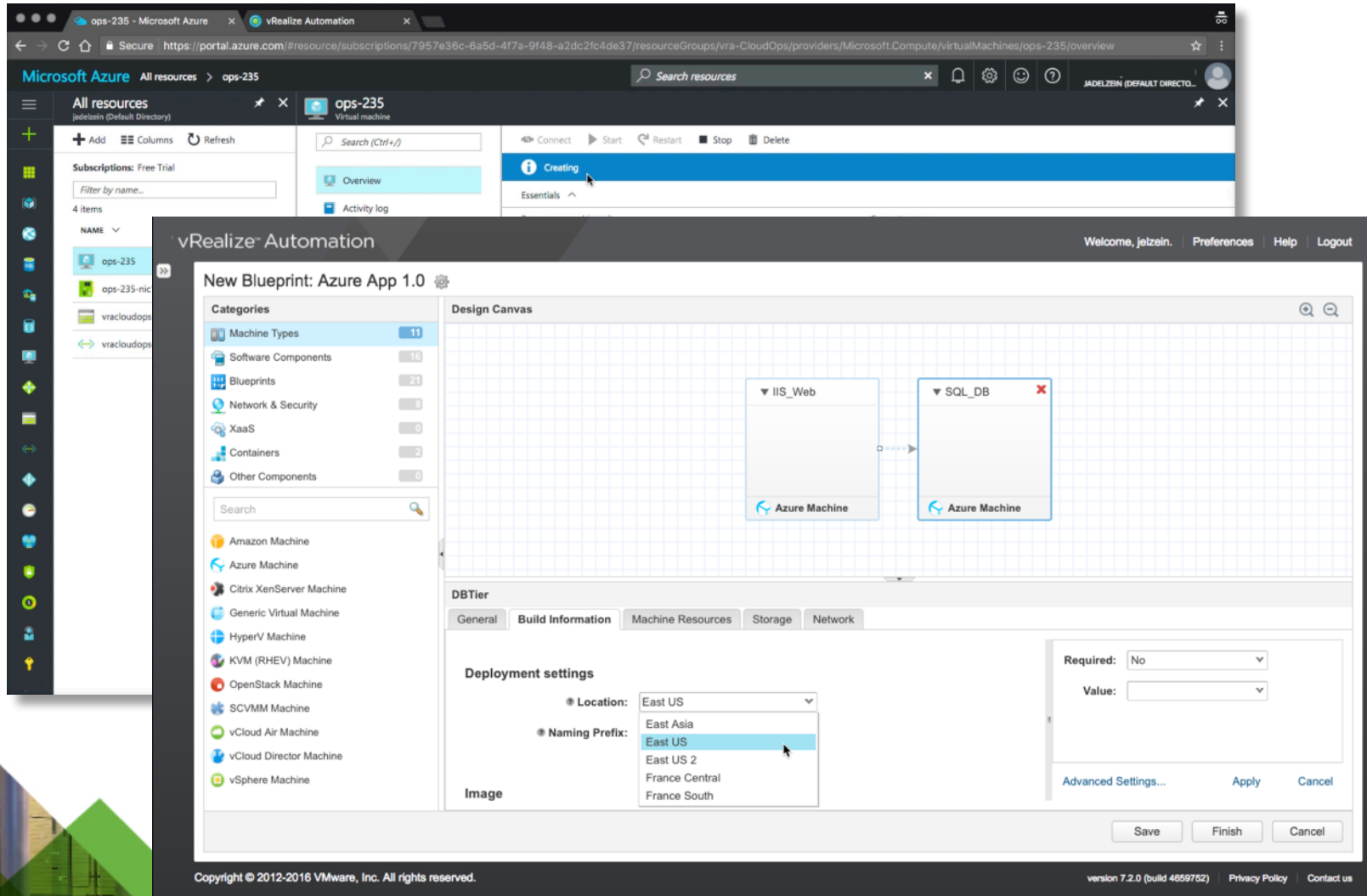
The screenshot displays the vRealize Automation console interface. At the top, there's a navigation bar with 'AWS' and 'Services' tabs. Below it, a progress bar shows steps from '1. Choose AMI' to '7. Review'. The main content area is titled 'Step 1: Choose an Amazon Machine Image (AMI)' and includes a 'Quick Start' section. On the left, a sidebar lists various machine types and components. The central 'Design Canvas' shows a blueprint named 'Amazon\_Machine\_1' with fields for 'Blueprint type' (Server), 'Provisioning workflow' (CloudProvisioningWorkflow), and 'Amazon machine image' (ami-00624765). A table at the bottom lists instance types with their respective CPU, memory, and storage specifications.

Name	CPU	Memory (GB)	Storage (GB)
t2 - Micro Instance	1	1024	0
t2 - Small Instance	1	2048	0
t2 - Medium Instance	2	4096	0
m3 - Medium Instance	1	3840	4

- Build, provision, and management EC2-based services
- Supports all EC2 Instance types
- Blueprint creation with EC2 VMs, storage disks, and nics
- Software Authoring Support
- EC2 Networking Options
  - VPC's
  - Security Policies

# vRA + Azure Public Cloud

## Native Azure Integration

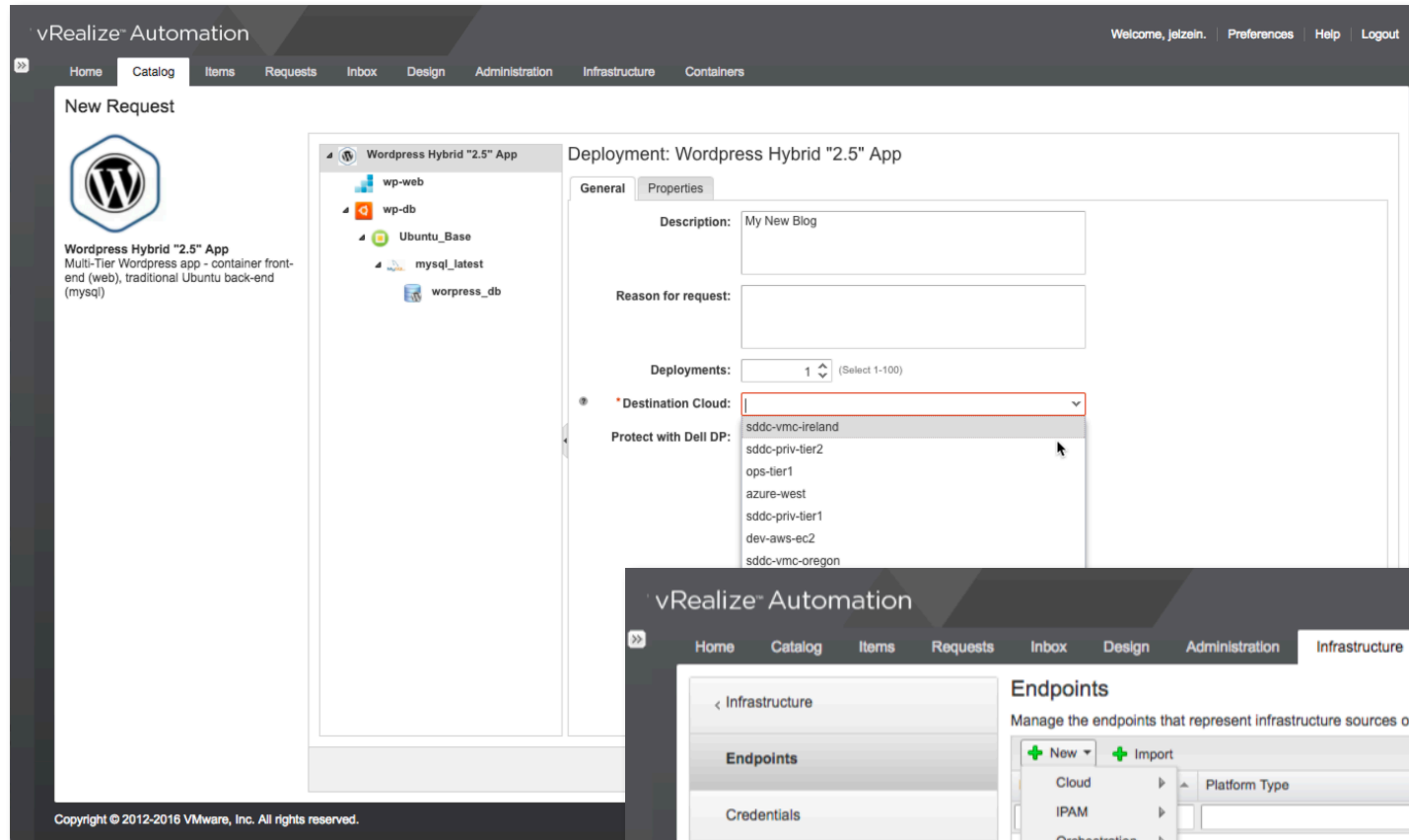


- Azure Endpoint with subscription and Active Directory users information
- Reservations and integration with governance model
- Blueprint creation with Azure VMs, storage disks, and nics
- Software Authoring Support
- Azure Networking Support
  - Subnets
  - Load balancers



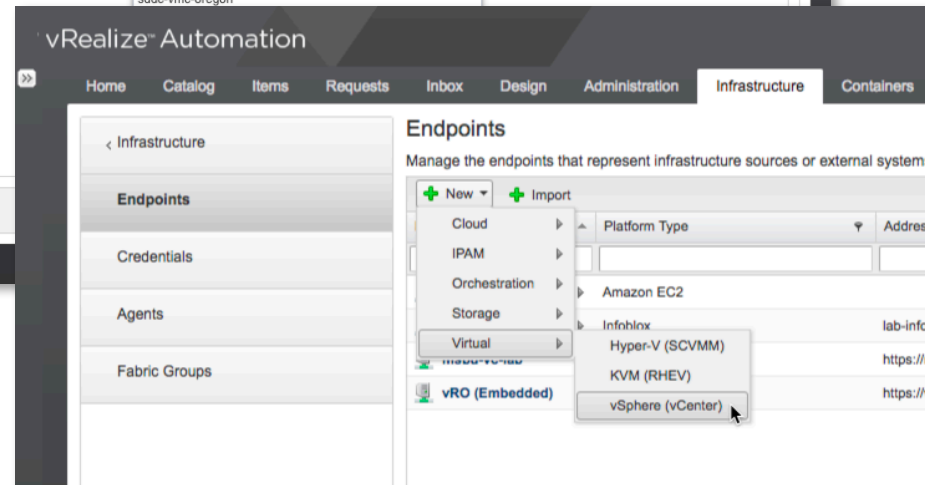
# Ready for VMware Cloud on AWS

## VMC Managed Endpoint



## Manage vCenter in VMware Cloud on AWS

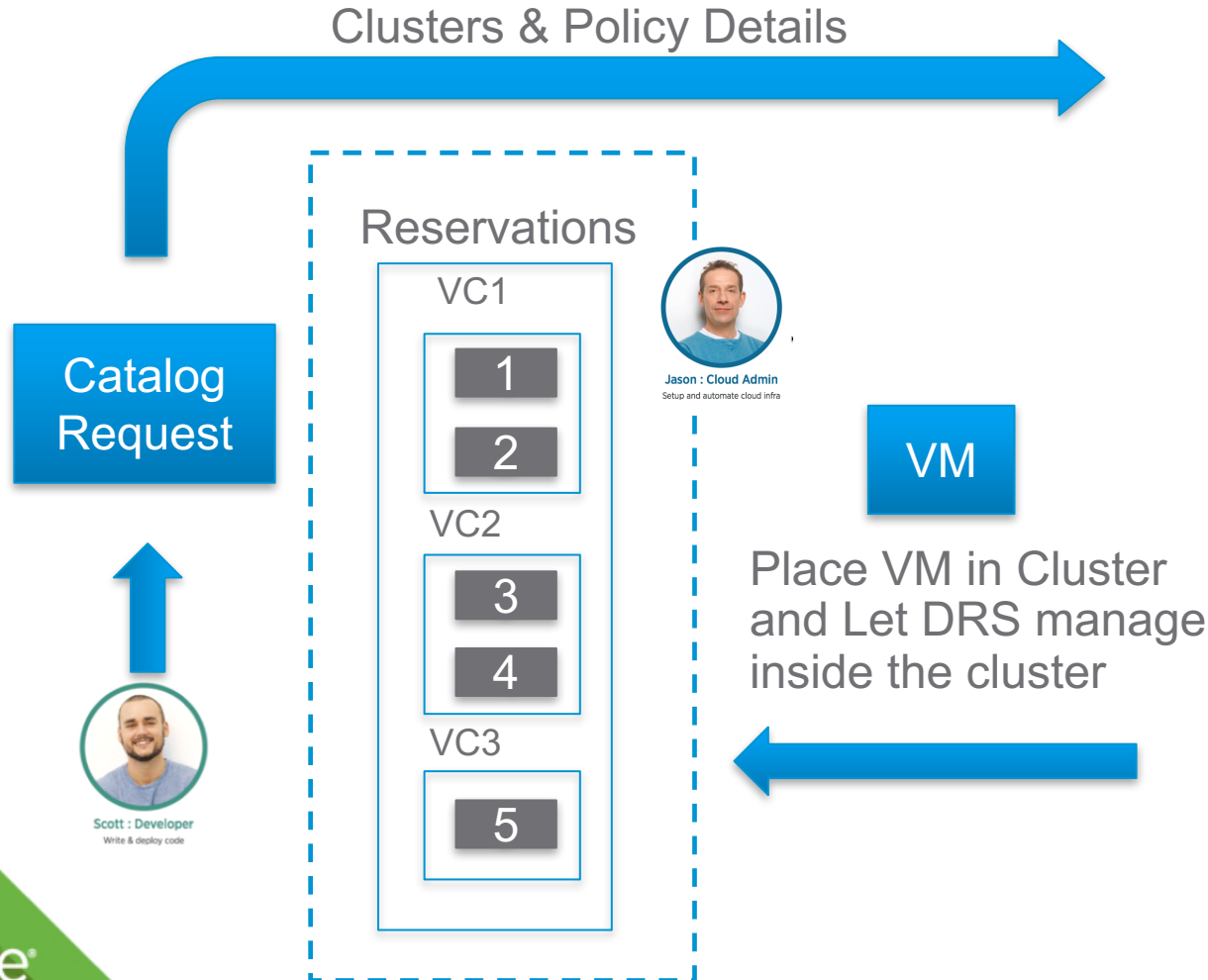
- Treated as a traditional vSphere / vCenter Endpoint
- Build an IaaS Fabric using VMware Cloud SDDC Resources
- Leverage Reservation Policies for machine placement



# Optimized Placement Using vR Ops Analytics

Intelligent Workload Placement (WLP)

## vRA Requests



## vR Ops Policies

Balance (Performance)



Consolidate



Performance  
& Capacity based  
Recommendation

In flight capacity  
reservation

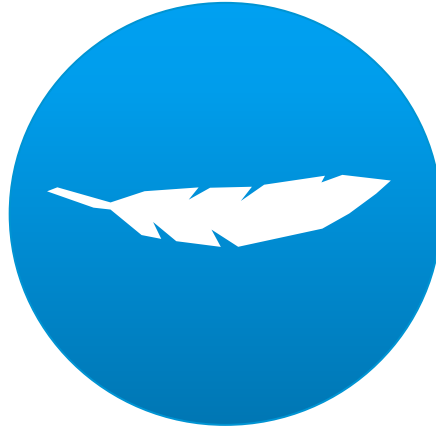
# Benefits of Containers



Simple



Boot environments  
rapidly



Lightweight



Minimal resources  
needed



Portable



Ability to move  
containers freely

# Container Use Cases for Enterprise

The Move to Containerization Will Come in Waves



New Apps

Repackaging

Refactoring



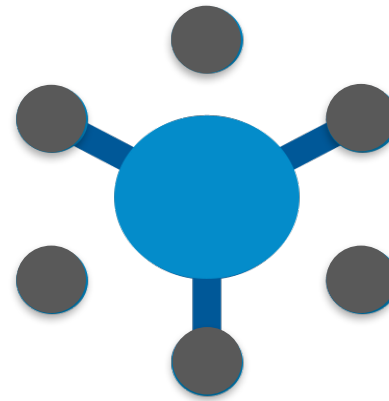
# There Are Many Challenges



Security



Monitoring



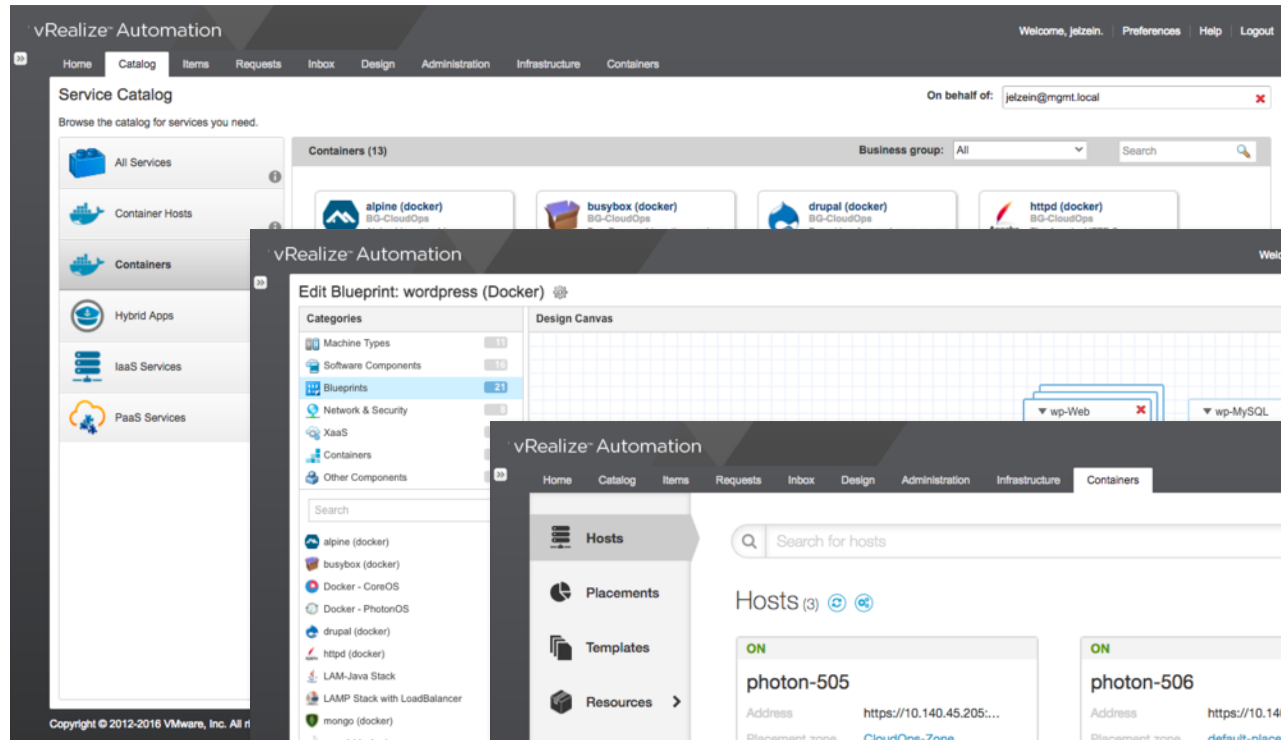
Networking



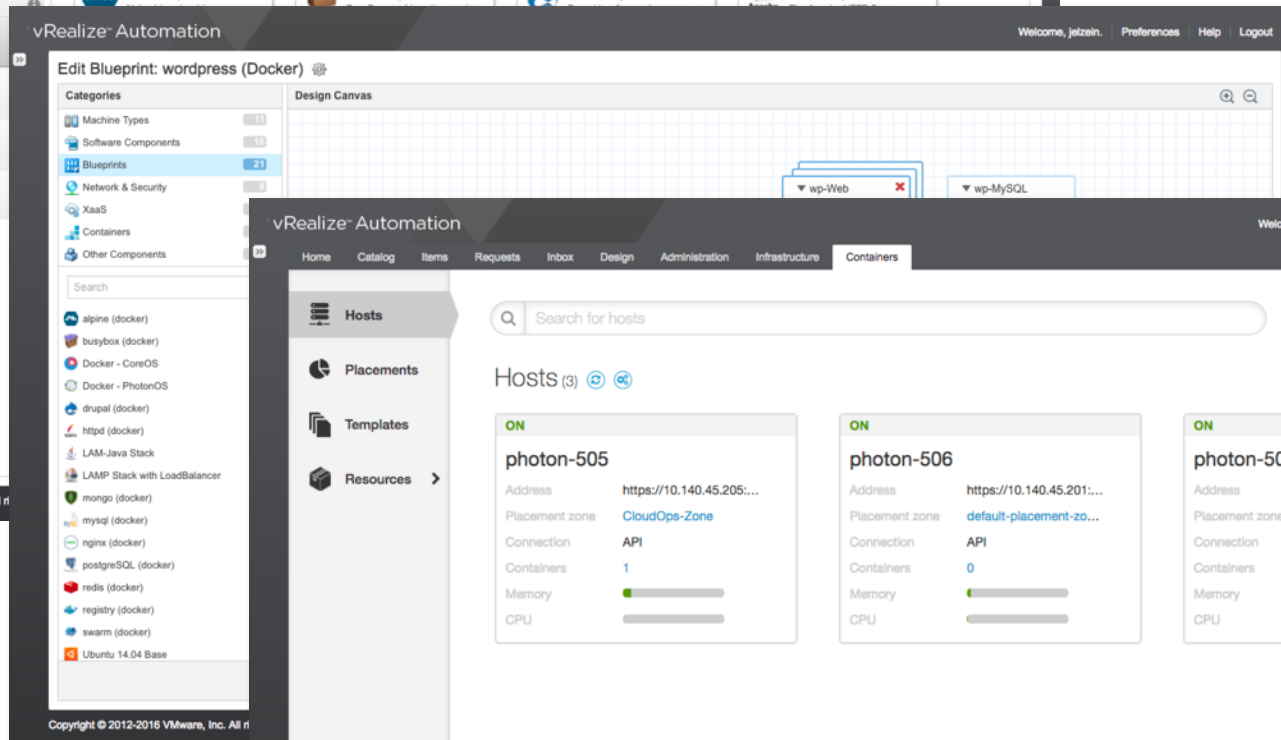
Compliance

Production Environment

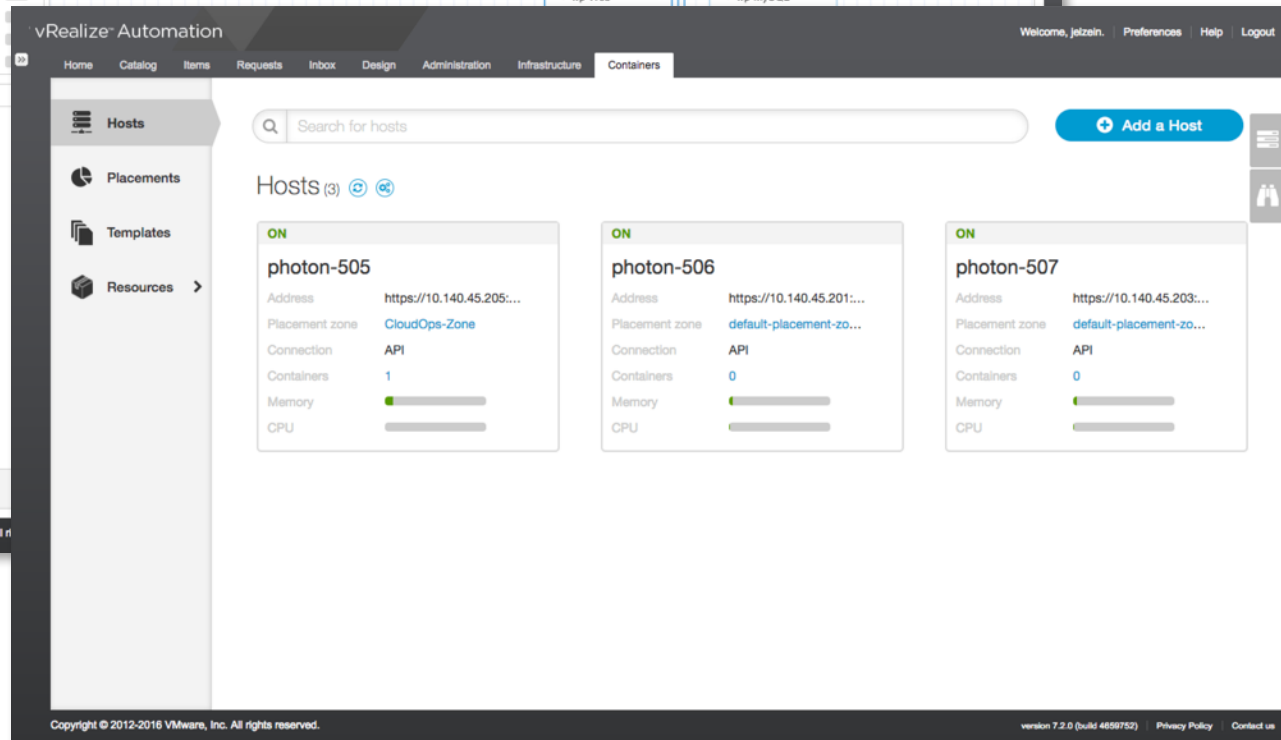
# vRA Container Management in Action



Self-Service Provisioning for Container Applications and Container Hosts

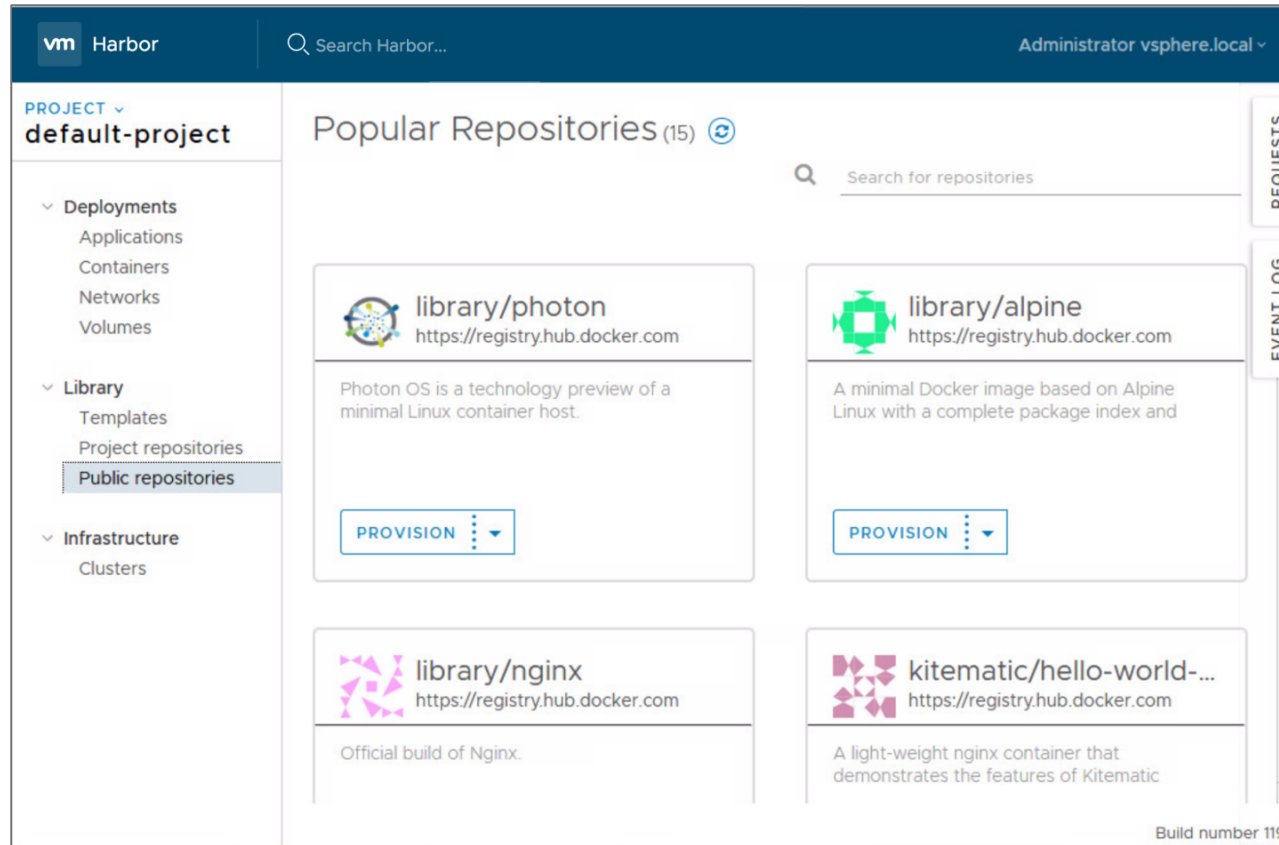


Design Traditional, Container or Hybrid (VM + Container) Applications



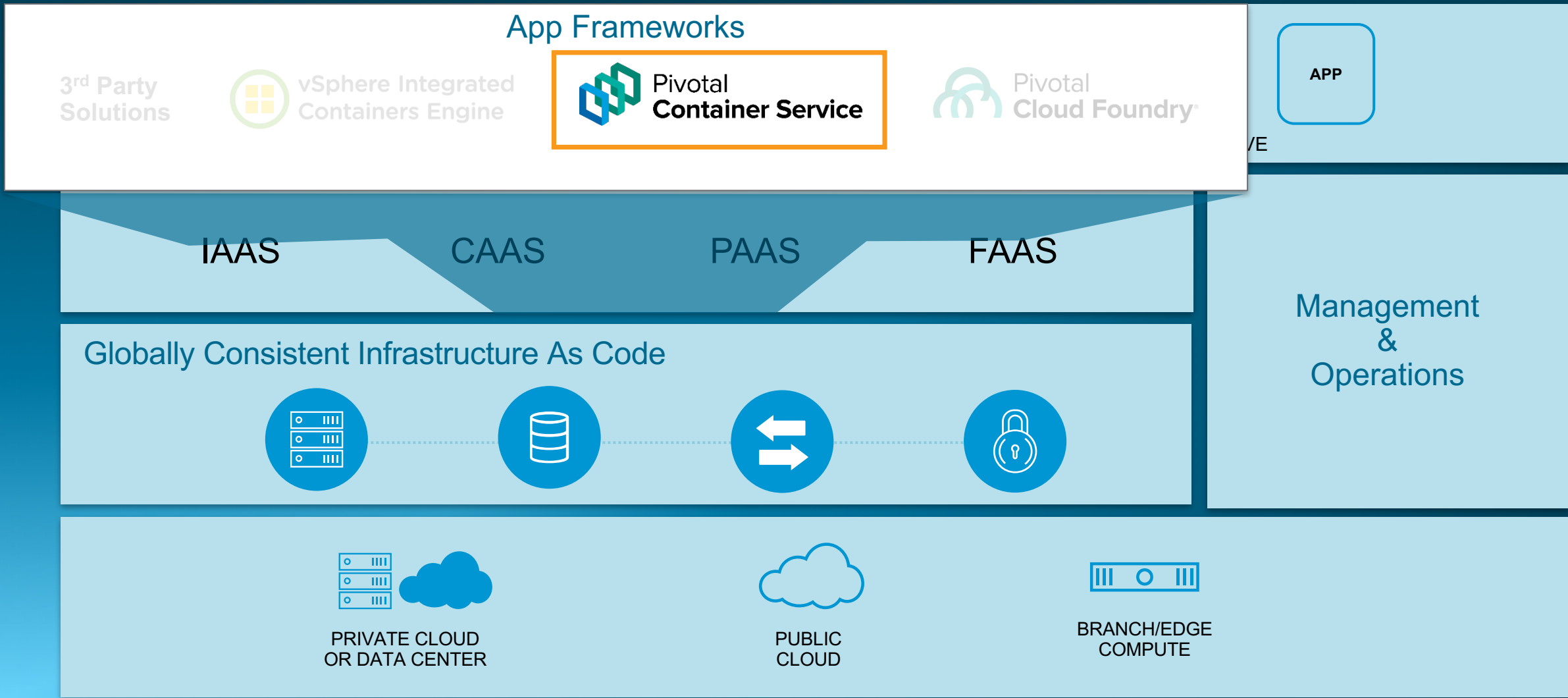
Discovery and Management of Container Hosts and Containers

# Registry – Enterprise-grade Private Registry



user management & access control  
role-based **access control**  
**AD/LDAP** integration  
security  
**vulnerability scanning**  
content trust - **image signing**  
policy based image **replication**  
**audit and logs**  
restful **API**  
**lightweight & easy** deployment  
**open-source** under Apache 2 license

# Production-Grade Kubernetes



# VMware Pivotal Container Service (PKS)

**Purpose-built container service to operationalize Kubernetes**

Fully Supported  
Mainline Kubernetes Distribution

Constant Compatibility  
with GKE

Runs on vSphere, VCF

Deep Integration  
with NSX-T

Service  
Hardened,  
Production-grade

HA, Security,  
Multi-tenancy, Tools



# PKS Leverages Project Kubo

## Day 1 “Build”

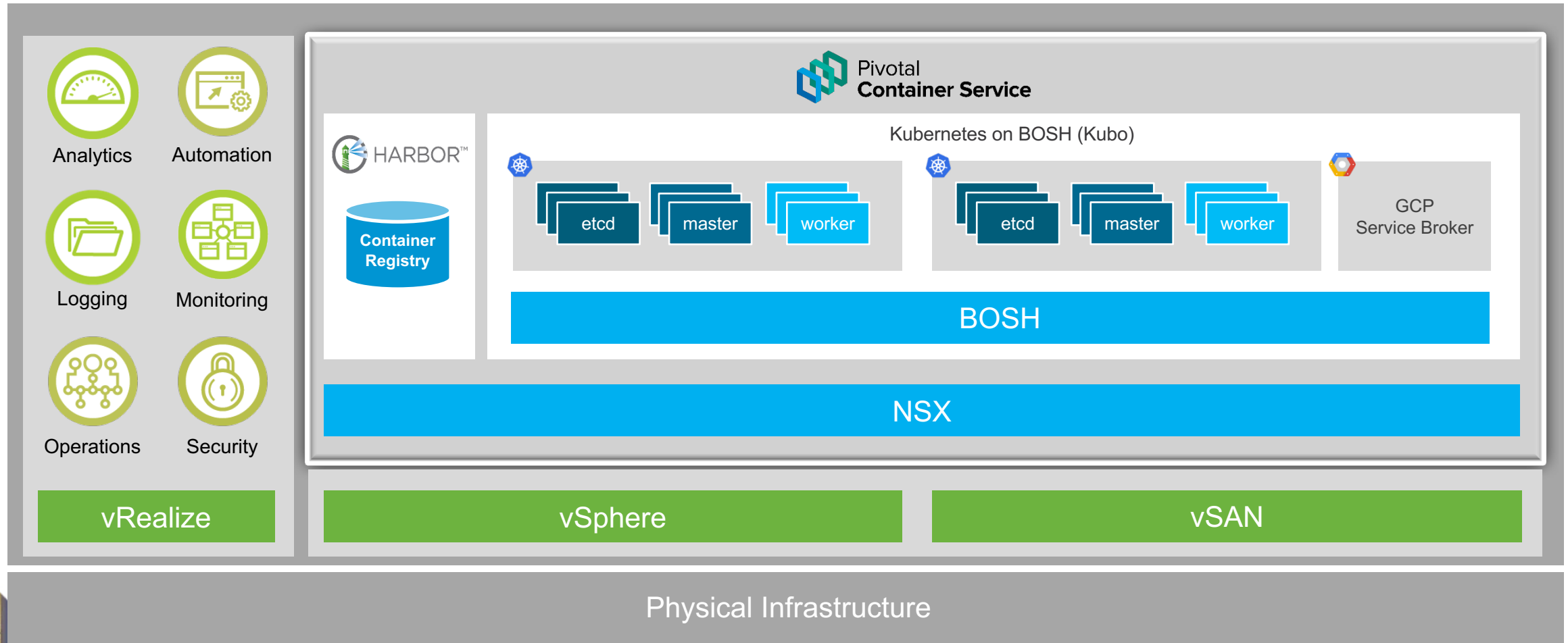
Deploy Kubernetes cluster via BOSH  
Deploy Kubernetes clusters on demand

## Day 2 “Operate”

Self-healing VMs and monitoring  
Elastic scaling for clusters  
Rolling upgrades to latest Kubernetes release  
High availability and multiple clusters & resource pools

# Container Infrastructure for Cloud-Native Apps

Rapidly deliver and operationalize next generation apps

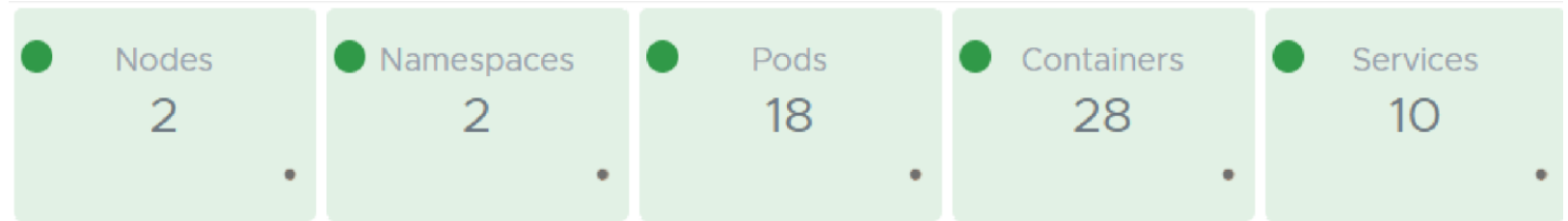


# vRops & PKS (Operations & Monitoring)

## vRealize Operations & K8s

- Operator KPIs
- Single Pane for SDDC & K8s clusters monitoring
- vRLI Integrated
- Alert on K8s KPIs
- Entity Relationship
- Capacity Planning
- **Integrated with PKS**

2. Summary of the Selected Cluster with Historical Trends



7. Pods running on this Node?



Platform  
Operator



# VMware Cloud

Run, Manage, Connect, Secure Any App on Any Cloud to Any Device



## Existing Apps

Reduce Costs • Security • Reliability • Control

## Cloud Native Apps

Time to market • Innovation • Scale • Differentiation

## VMware Cloud Services



VISIBILITY



OPERATIONS



AUTOMATION



SECURITY



GOVERNANCE

## Cloud Management Platform



## VMware Cross-Cloud Architecture



## Public Cloud IaaS



vmware  
Cloud Foundation

amazon  
webservices™  
VMware Cloud on AWS

IBM Cloud  
for VMware

vmware  
Cloud Providers

OVH.com

amazon  
webservices™

Microsoft  
Azure

IBM Cloud



Google Cloud Platform

virtustream

## Consistent Infrastructure

VM Infrastructure • Container Infrastructure

## Consistent Operations

Management and Operations • Across Clouds

vmware

# VMware Cloud Services

Manage, Govern and Secure Public and Private Cloud Apps

vm VMware Cloud Services

?

Nikhil Girdhar  
ACME Fitness Comp...  
▼

⋮

Welcome, Nikhil

ACME Fitness Company

Cost Insight

Analyze and optimize cloud costs

OPEN

Discovery

Automatically detect cloud inventory

OPEN

Network Insight

Simplify cloud network and security operations

OPEN

VMware Cloud on AWS

VMware Cloud on AWS

REQUEST ACCESS

VMware NSX Cloud™ Beta

Consistent networking and security for applications running natively in public clouds.

REQUEST ACCESS



## Discovery

Visibility into apps and resources they consume. Analyze usage and utilization across clouds.



## Cost Insight

Accounting and cost optimization for multiple clouds. Track and analyze your costs and trends.



## NSX Cloud

Secure networks with micro-segmentation  
Create private networks within or across clouds.



## Network Insight

Operational visibility, control, and compliance across clouds. Optimize performance, health, and availability.



## Wavefront

Metrics-driven monitoring and real-time analytics.



## AppDefense

Governance for running workloads.

vmware®  
ON PREMISES DATA CENTER

Microsoft  
Azure

amazon  
web services™

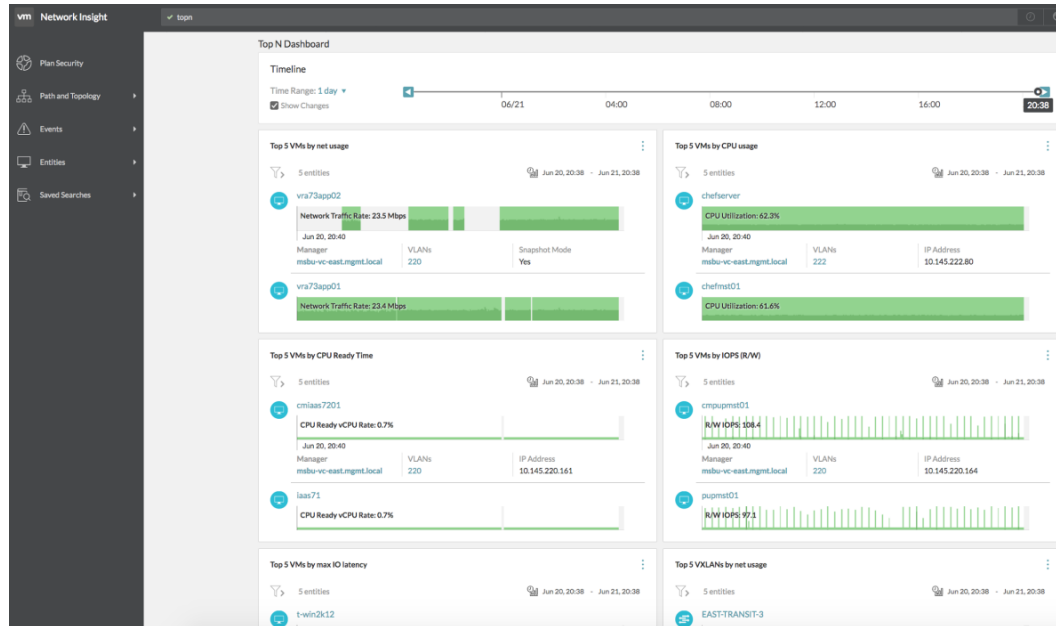
Google Cloud Platform





# Network Insight: Simplify Cloud Network and Security Operations

## Purpose-built for Network Virtualization and Public Clouds



## Plan and manage application security

- Understand application dependencies by analyzing traffic flow patterns between VMs
- Accelerate micro-segmentation planning and use firewall rule recommendations to improve cloud security
- Continuously monitor, troubleshoot and audit cloud security posture over time

## Troubleshoot network with visibility across clouds



- Discover AWS, VMW and physical network infrastructure resources including AWS VPCs, security groups, and cloud tags
- Troubleshoot network connectivity issues within and between public and private clouds
- Rapidly identify issues through pro-active events and alerts

## Ensure health and availability of VMware NSX deployments

- Scale across large NSX deployments with powerful visualizations for topology and health
- Avoid configuration issues with NSX deployments based on health checklists
- Quickly pinpoint issues for resolution with the help of intuitive UI and search



# Discovery Service :Within minutes get a holistic view of all cloud resources

 <b>Amazon Web Services</b>	<b>Microsoft Azure</b>		
<b>Compute</b> AWS EC2 Instances 369	<b>Compute</b> Virtual Machines 1	<b>Compute</b> Virtual Machines 377 Servers 43 Clusters 8	<b>Resource Groups</b> Total Groups 4 <a href="#">CREATE GROUP</a>
<b>Storage</b> EBS Volumes 389 S3 Buckets 109	<b>Network</b> Virtual Networks 2 Network Interfaces 7 Subnets 4	<b>Storage</b> Virtual Hard Disk 0	<b>Cloud Accounts</b> Amazon Web Services 2 Microsoft Azure 1 <a href="#">ADD CLOUD ACCOUNT</a>
<a href="#">VIEW AWS RESOURCES</a>	<a href="#">VIEW AZURE RESOURCES</a>	<a href="#">VIEW VCENTER RESOURCES</a>	

Public and Private

Compute, Storage and Network



# Multi-Cloud Cost and Visibility Challenges for IT



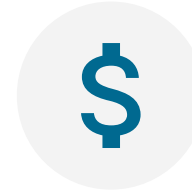
## Lack of Visibility

Resources, usage and spend



## Cloud Waste

Powered off, underutilized Resources



## Cost Overruns

Challenges with budget control



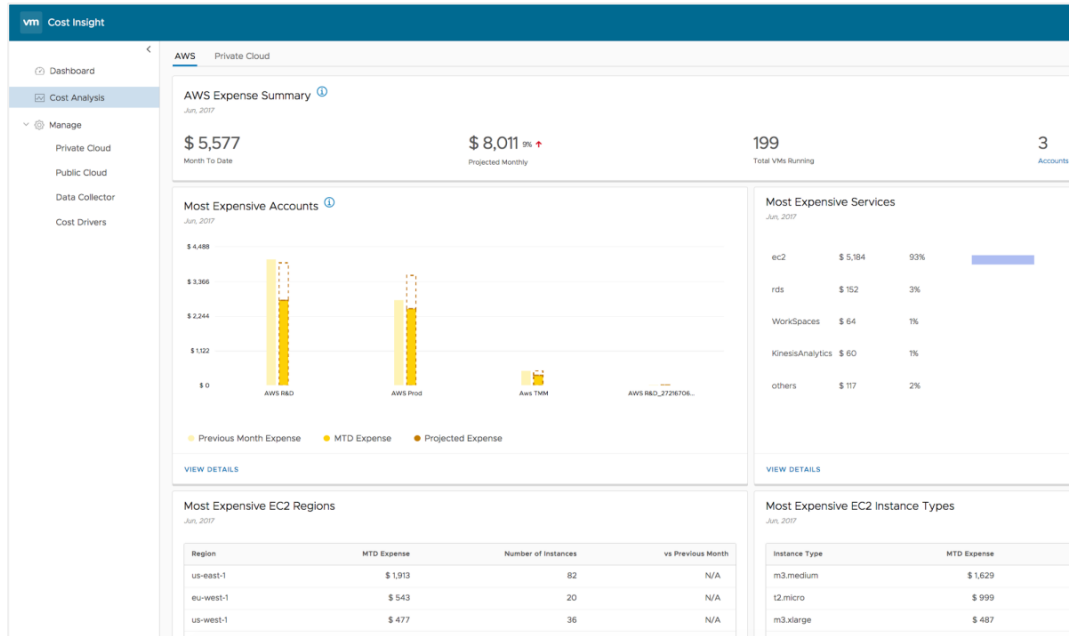
## No Accountability

Lack of communication



# Cost Insight: Make Intelligent Business Decisions

## Cost visibility and control for AWS, Azure and VMware



## Cost Visibility

- Estimate total cloud spend across public and private clouds
- Compare spend by cloud providers, regions, accounts, or other custom groups
- Analyze costs and drill deeper to identify key cost drivers

## Cost Control

- Track cloud costs over time and project future costs based on historical data
- Compare actual spend with assigned budgets
- Share cloud costs and budget comparisons with application teams

## Cost Efficiency

- Identify powered off virtual machines
- Identify unused cloud storage resources
- Customize threshold limits for identifying unused resources



# Start By Modernizing Your Data Center...

## VMWARE CLOUD ARCHITECTURE™

**Private Cloud**

**Hybrid Cloud**

**Public Cloud**

VMware Cloud Services

VMware vRealize®



VMware Cloud Foundation™

Software-Defined Data Center

Microsoft Azure



Google Cloud Platform

IBM

VMware Cloud on  
AWS

OVH

Vmware Cloud  
Providers



vmware®



# **vINNOVATION**

THIS IS HOW TOMORROW IS BUILT

**Thank You !**

