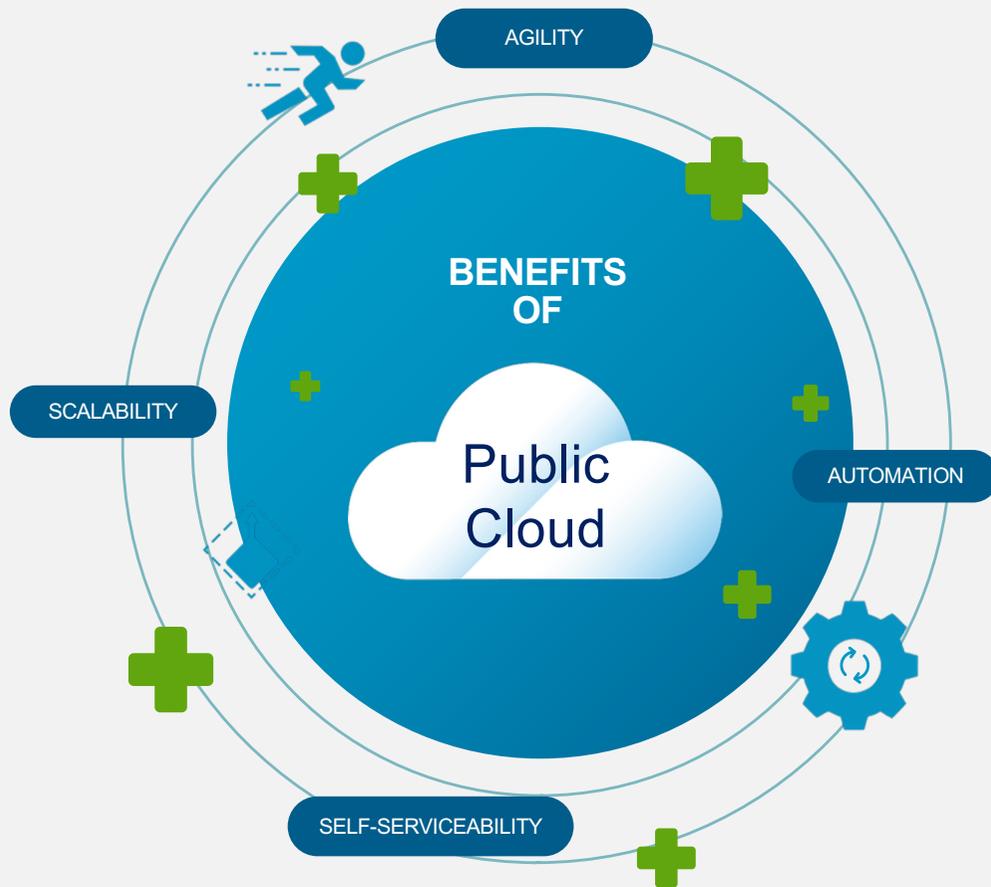


Enabling The Private Cloud Operating Model

with **VMware Cloud Foundation 9**

Yves.Hertoghs@broadcom.com
Principal Technologist

December 2025



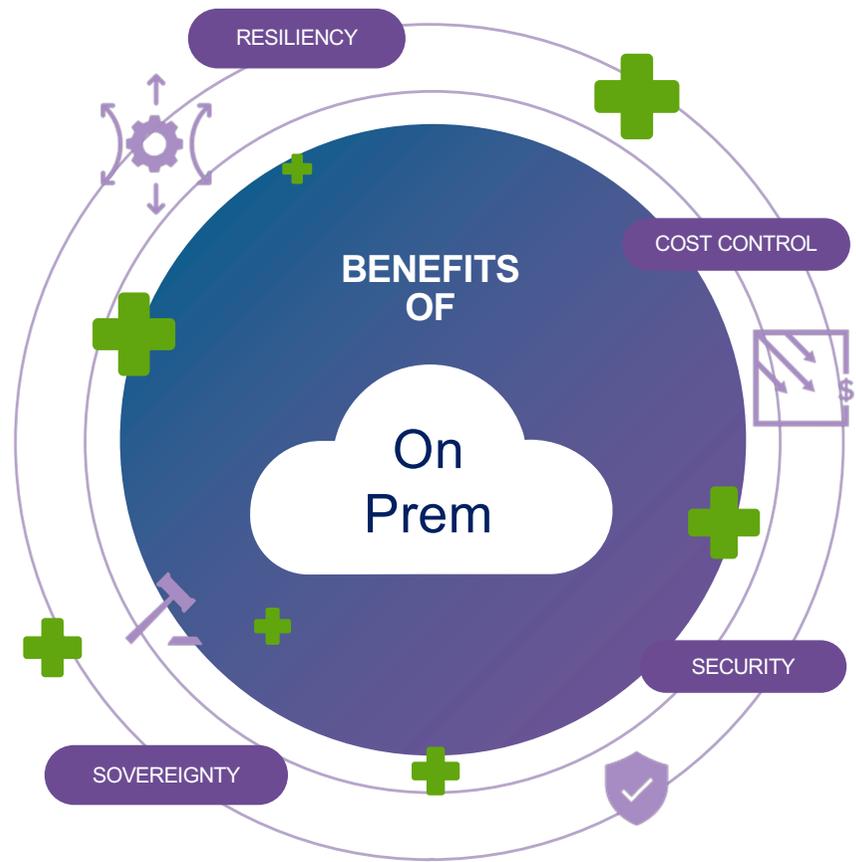
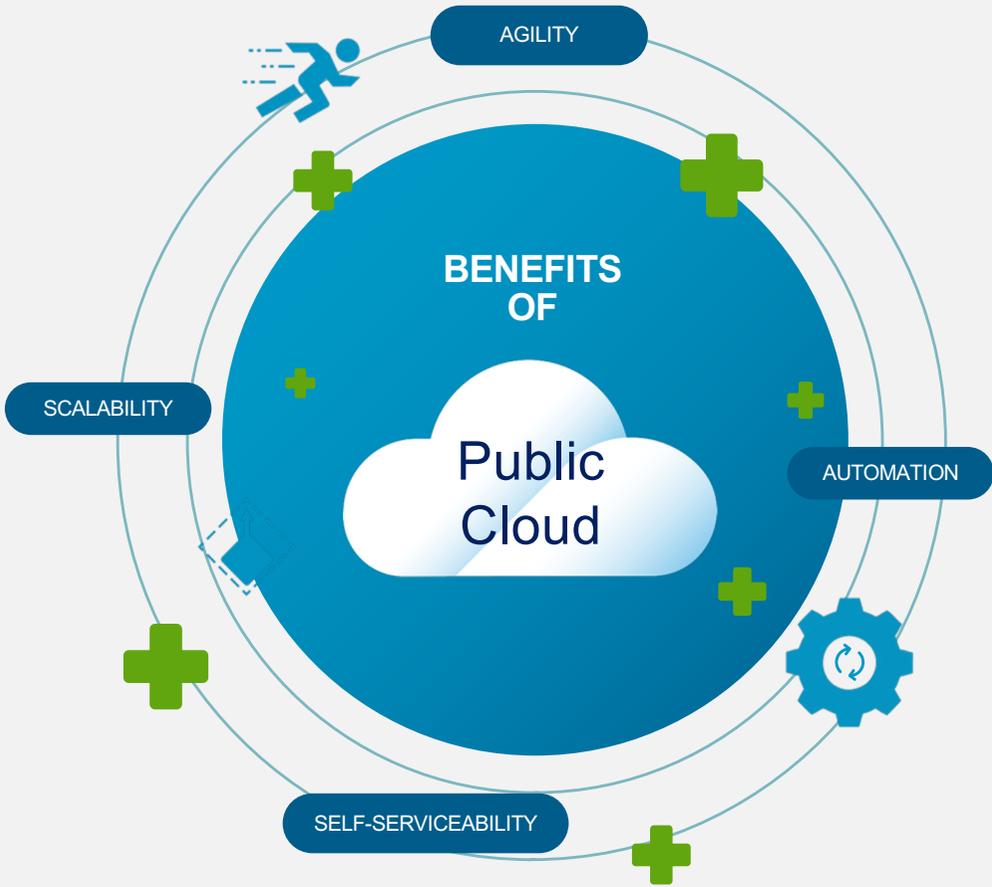
Public Cloud 'customers' are '**consumers**'.

They don't need to know what's under the hood (Rightfully so)

Hyperscalers are '**providers**' of **Services**, NOT **infrastructure**

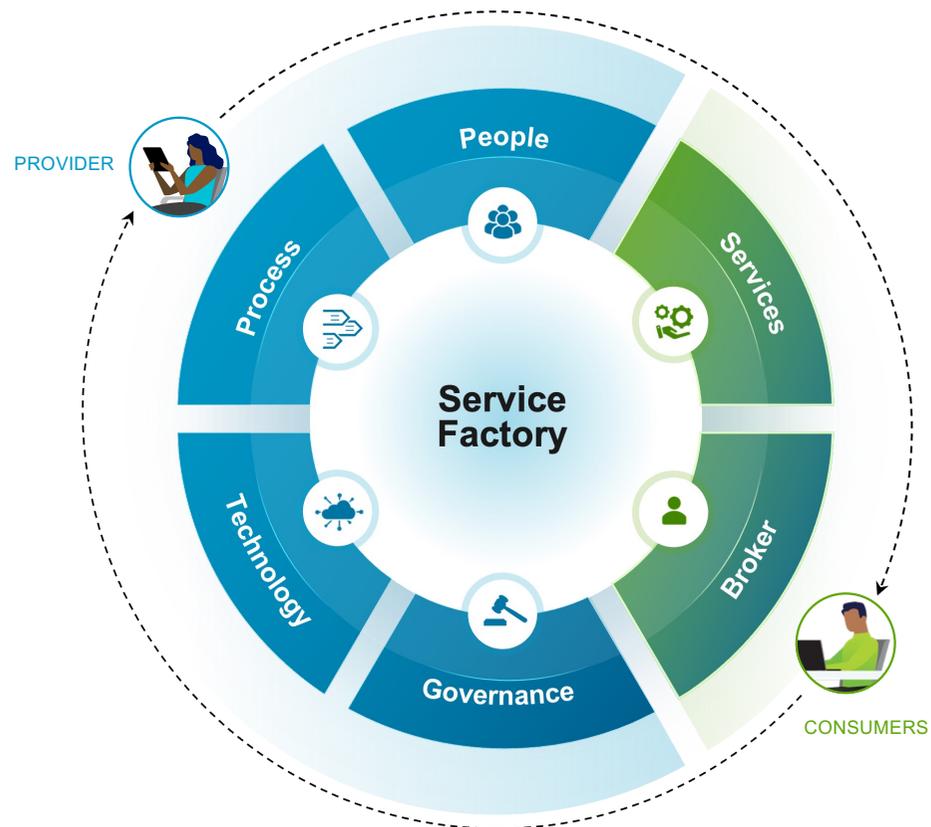
Infrastructure is hidden/abstracted

*Why would this '**Consumption Focus**' only be possible on the public cloud?*

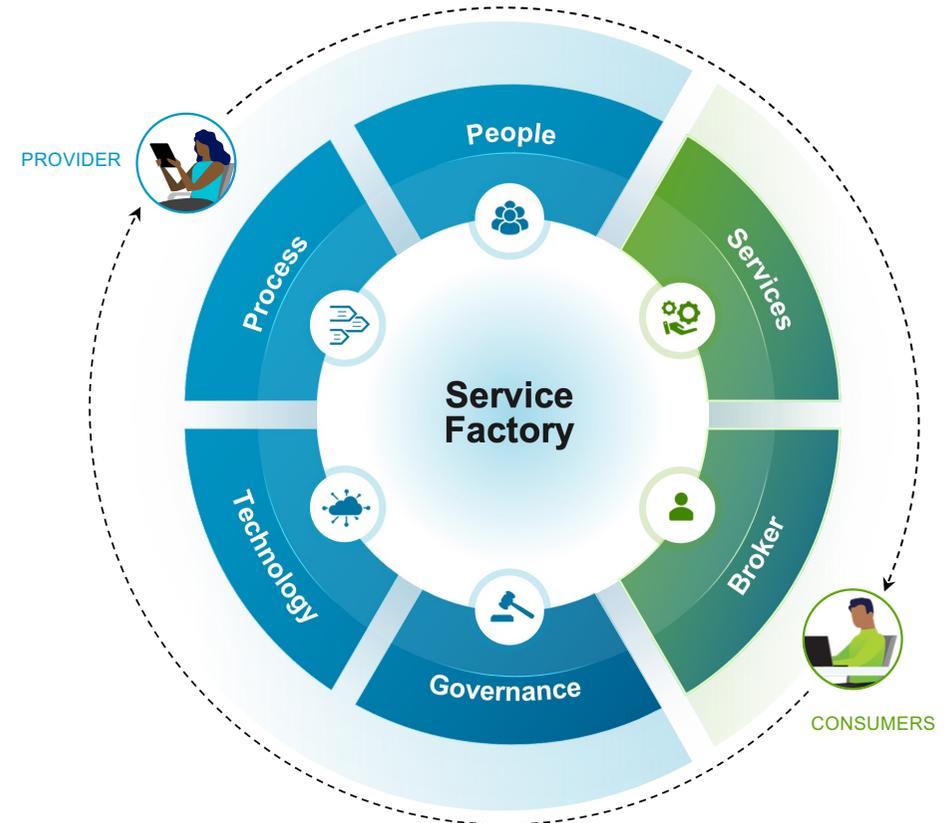
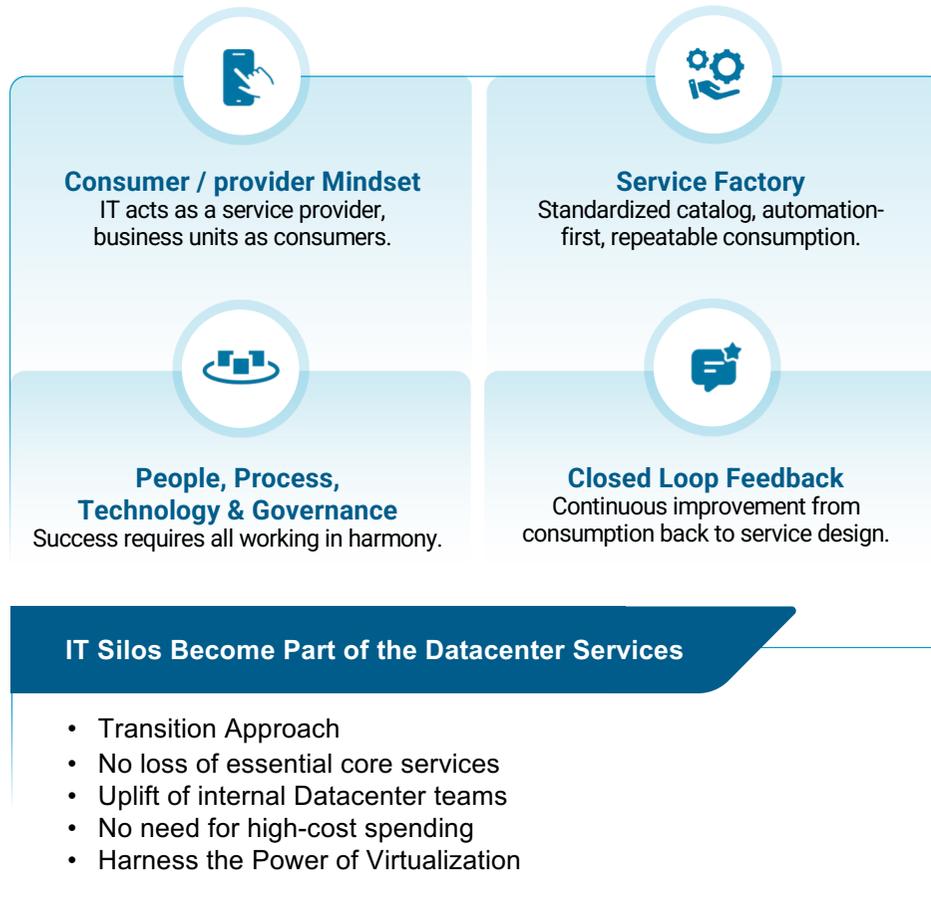


FUTURE CLOUD OPERATING MODEL

From IT Silos to Unified Service Delivery

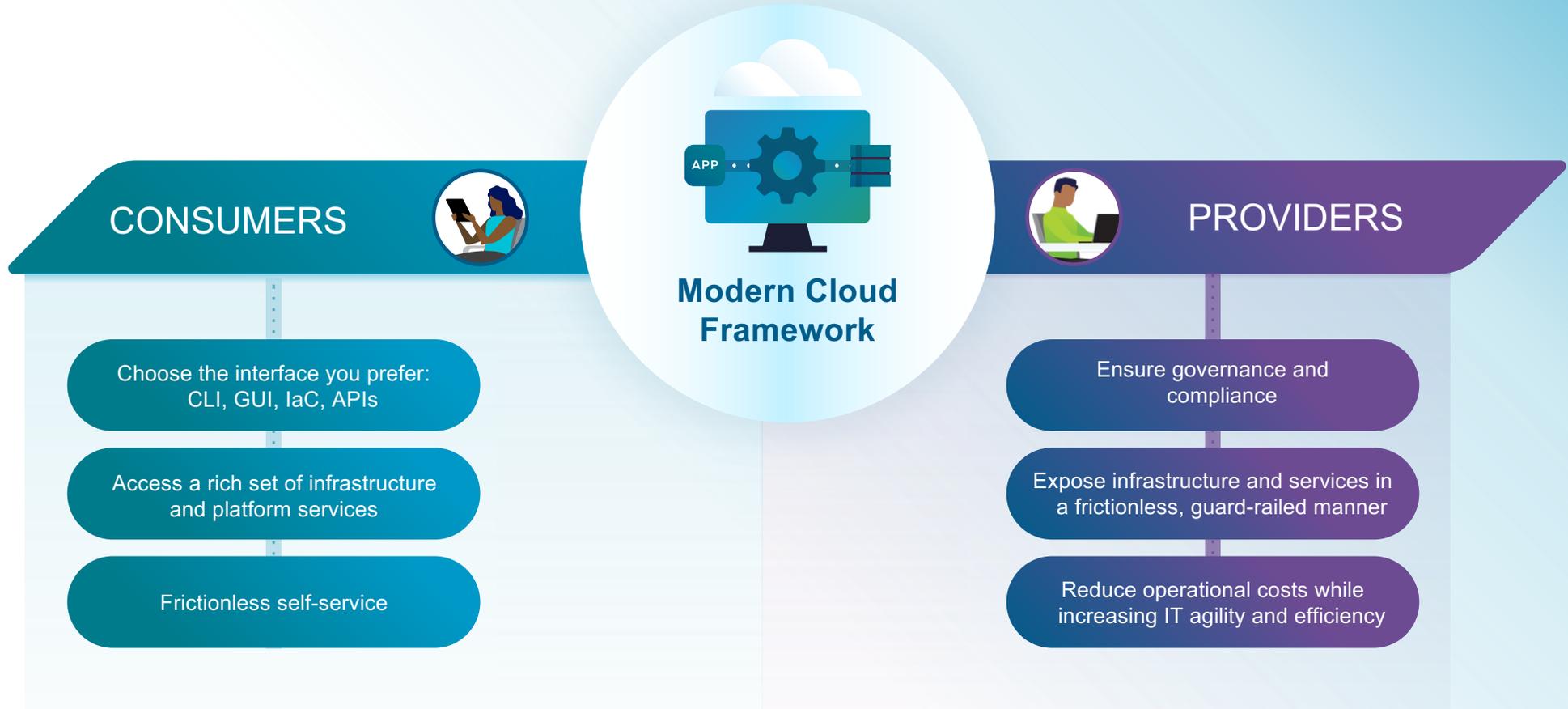


The Core Pillars of a Private Cloud Operating Model



Consumers & Providers

Requirements to support a cloud-like experience



Process

From Intent to Service Delivery on the Service Factory



Business Intent

Service Definition

Service Delivery

Standardization ensures repeatability and quality.

Automation accelerates speed and reduces human error.

Governance and guardrails are embedded at every stage.

Continuous measurement drives improvement.



Business Intent

CIO/IT Leadership,
Business Stakeholders.



Use Cases

Mapped to organizational needs.



Capabilities (Service Blueprint)

Standardized service designs and reusable definitions.



Operational Excellence

Reliability, security, compliance, and cost efficiency.



Service Delivery

Business outcomes, agility, faster time-to-market.



Governance

Measure & Improve cycle

People



Cross-functional roles:

Architects, Platform Engineers, SRE, SecOps, FinOps, Product Owners

From projects to products:

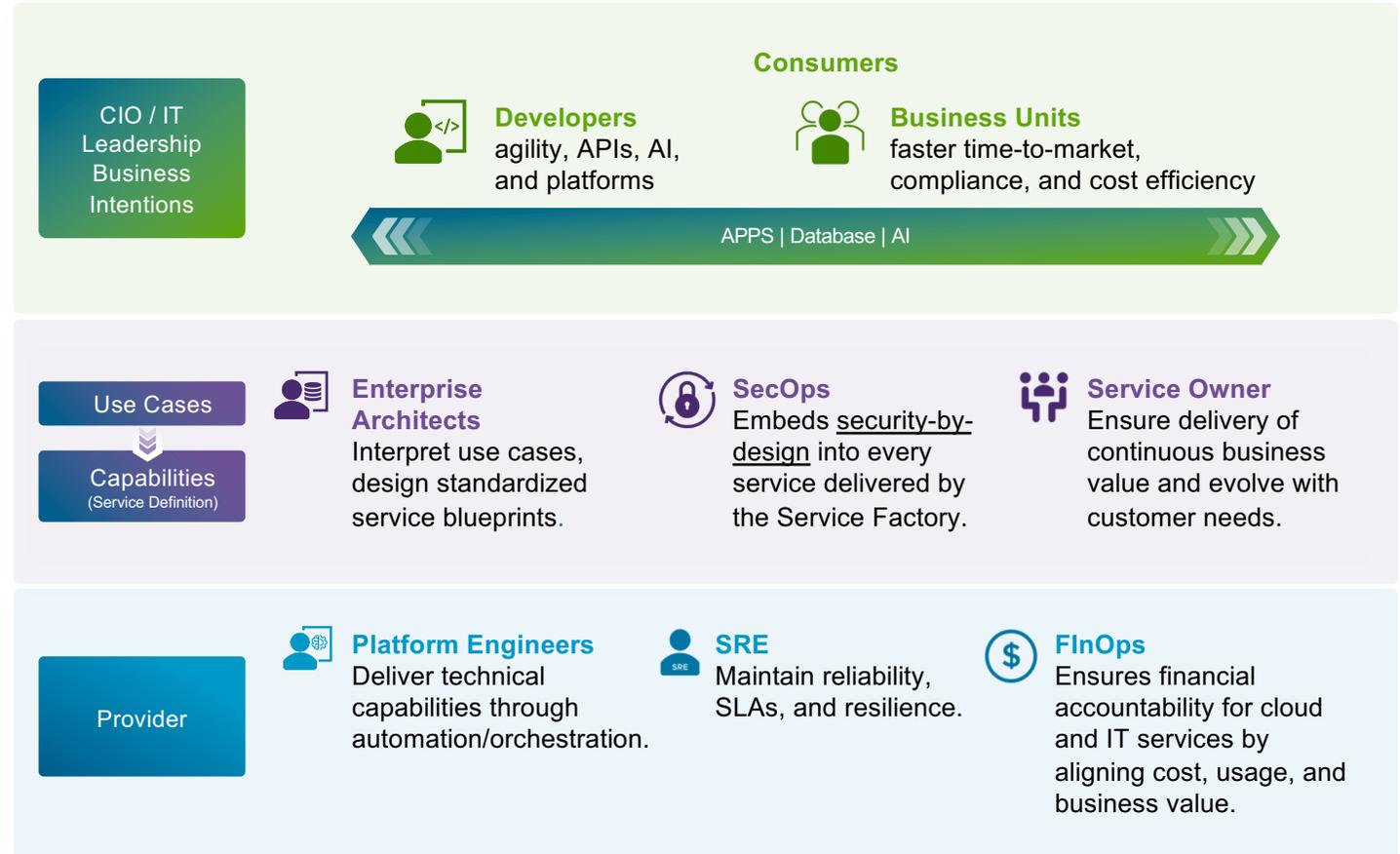
Services owned, versioned, and continuously improved

Shared accountability:

Collaboration across teams, not silos

Evolving skills:

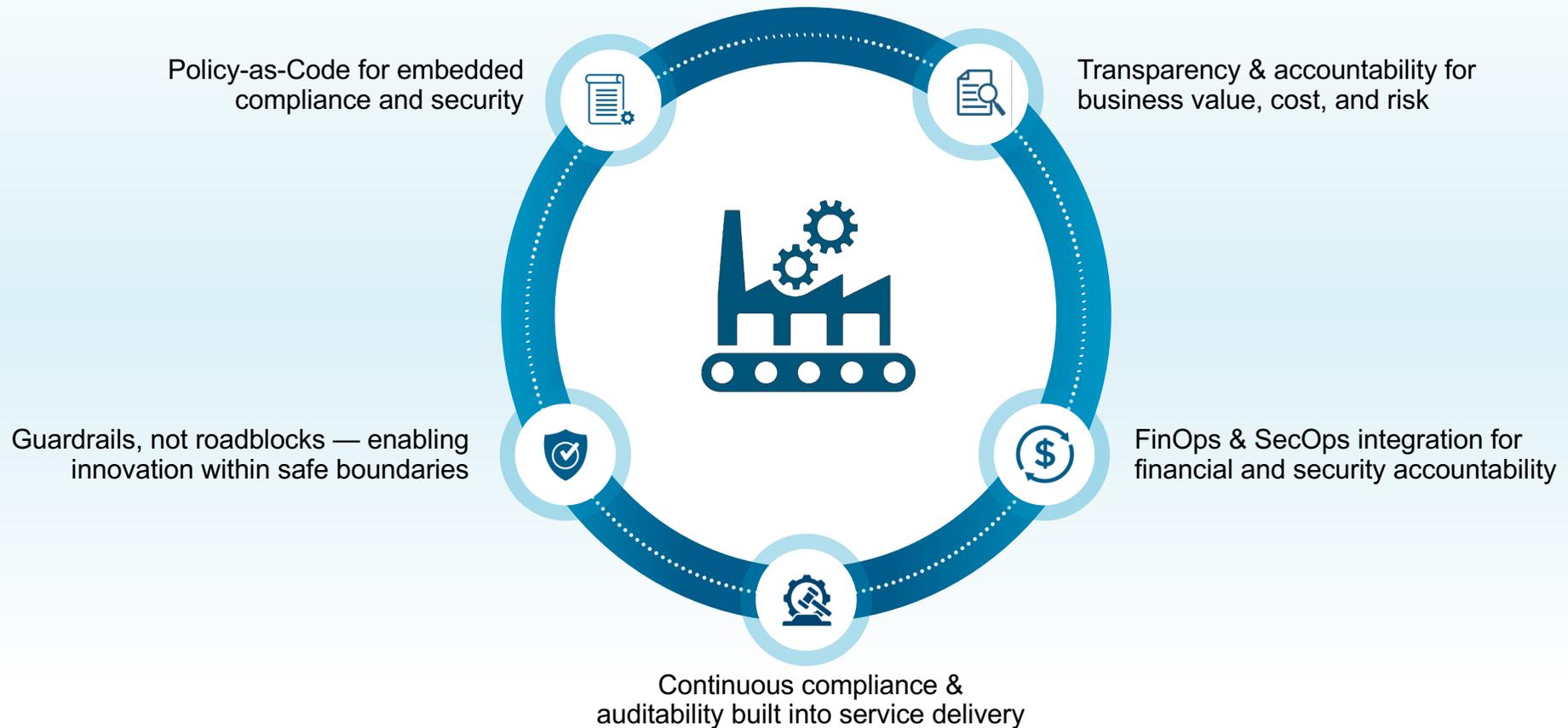
DevOps, GitOps, FinOps, Policy-as-Code





Governance

Ensuring Policy-By-Design, Compliance & Trust



Technology Building Blocks for a Modern Cloud



Choice of interfaces for consumers to consume services

Management and Governance for admins to set quotas and policies (guard rails)

Service Factory: providing extensibility

Abstraction Layer: hiding and slicing the underlying virtual infrastructure

Virtual Infrastructure—Used and trusted for decades: Virtual Compute, Storage, Network



Different Personas want Different things from the Cloud



Cloud Consumption API/Portal



DEVELOPER/
CONSUMER

"I don't want to think about infrastructure. Give me easy access to services that just work with my tools."



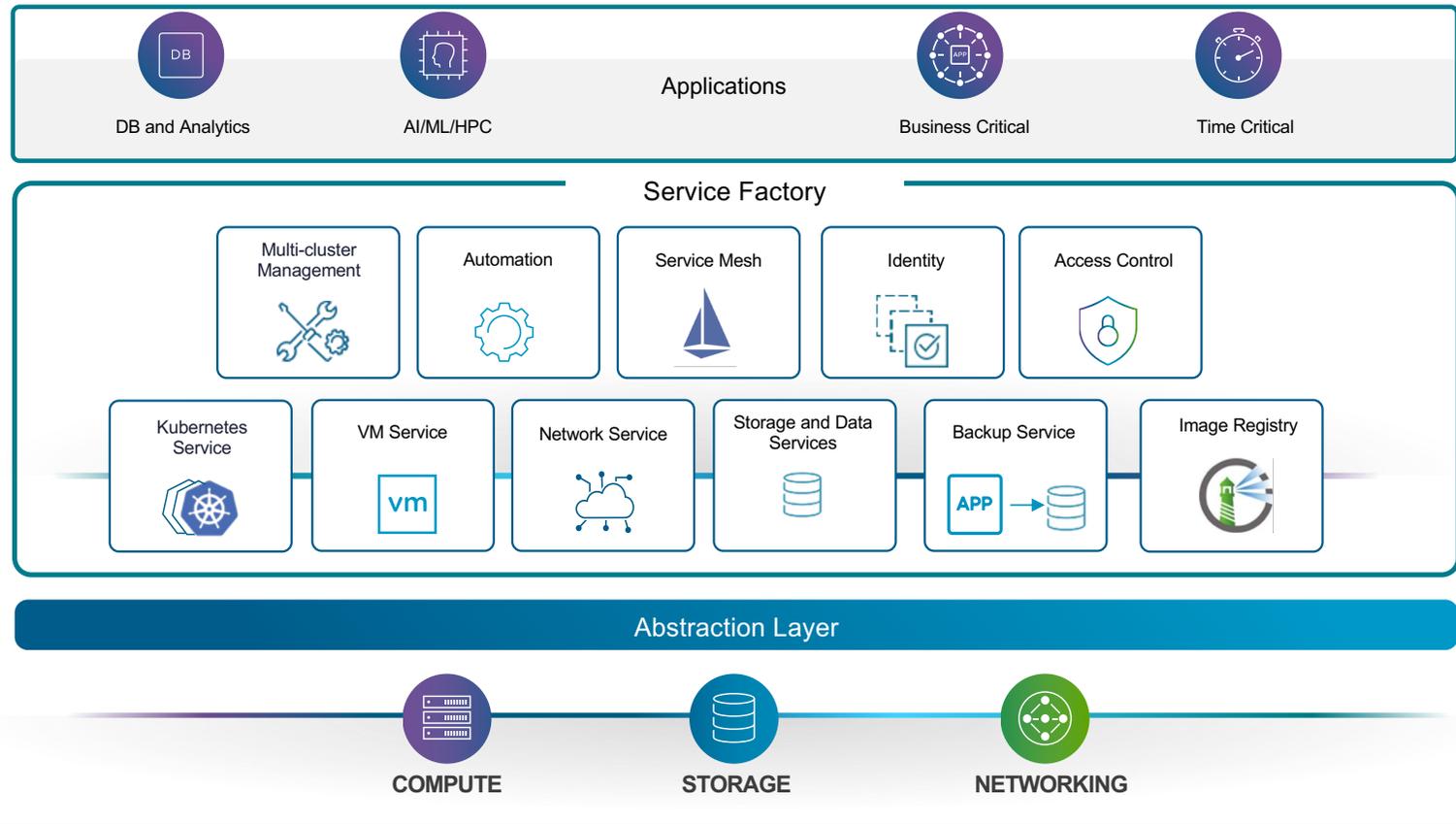
PLATFORM
ENGINEER /
SecDevOps

"I need to provision secure, reliable infra and cloud services, at scale, for developer teams."



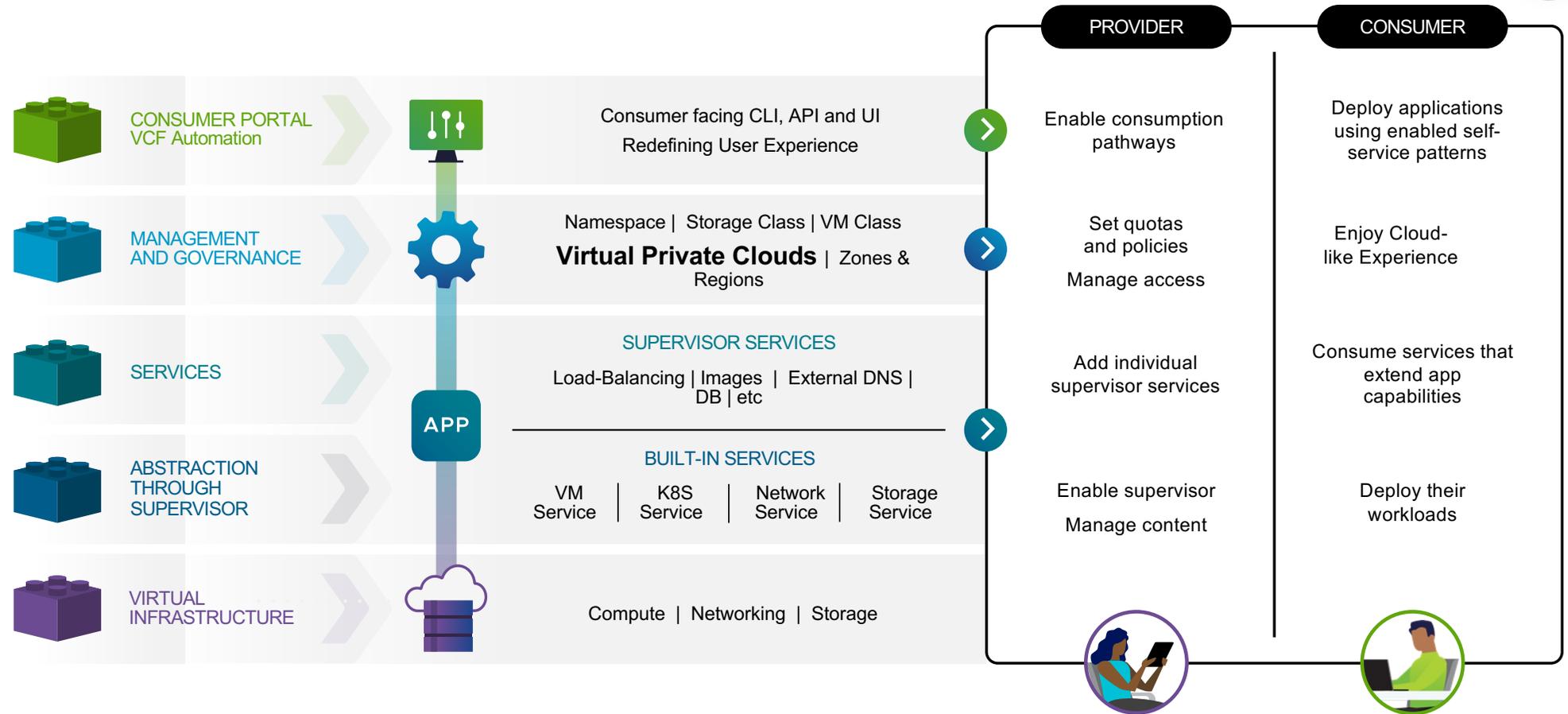
CLOUD ADMIN

"I need to ensure availability, security, and operational efficiency."



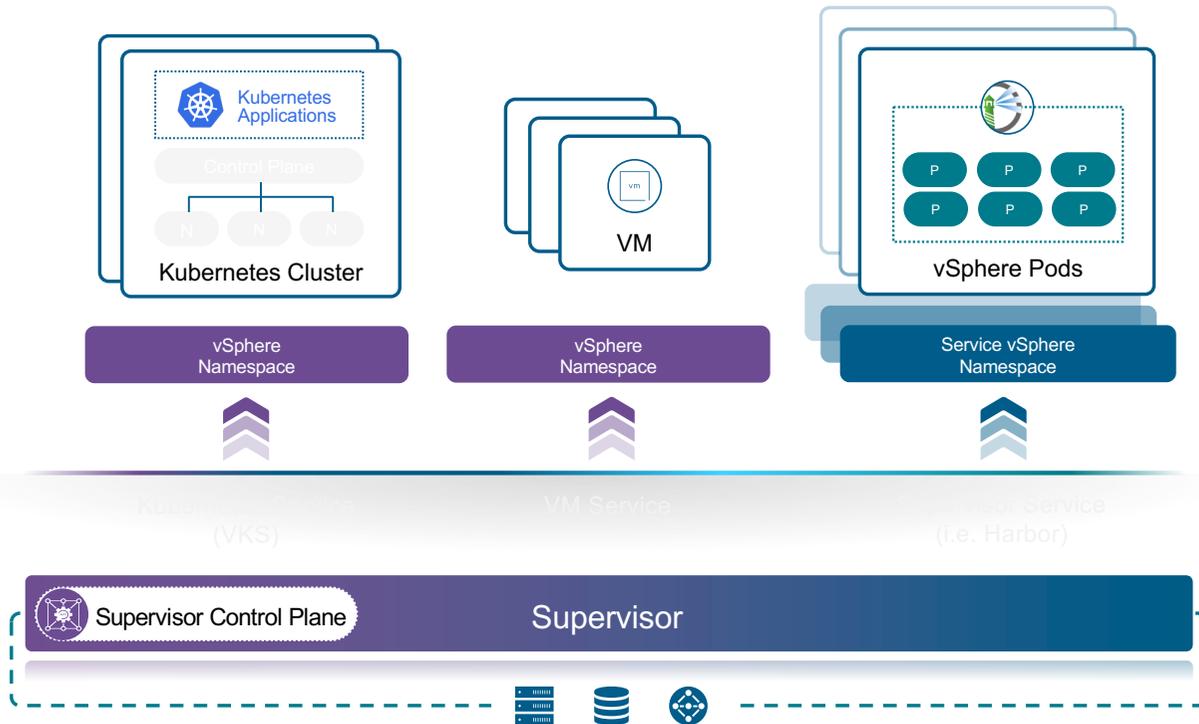
Building the Modern Cloud Interface with VCF9

Evolving the consumption experience



vSphere Supervisor

Basis for enabling the Private Cloud Operating Model



vSphere Supervisor is an integrated **Kubernetes-based Declarative API**

Define **what** you want, let the system figure out **how to** make it happen

Deploy **Kubernetes Clusters** with **vSphere Kubernetes Service**

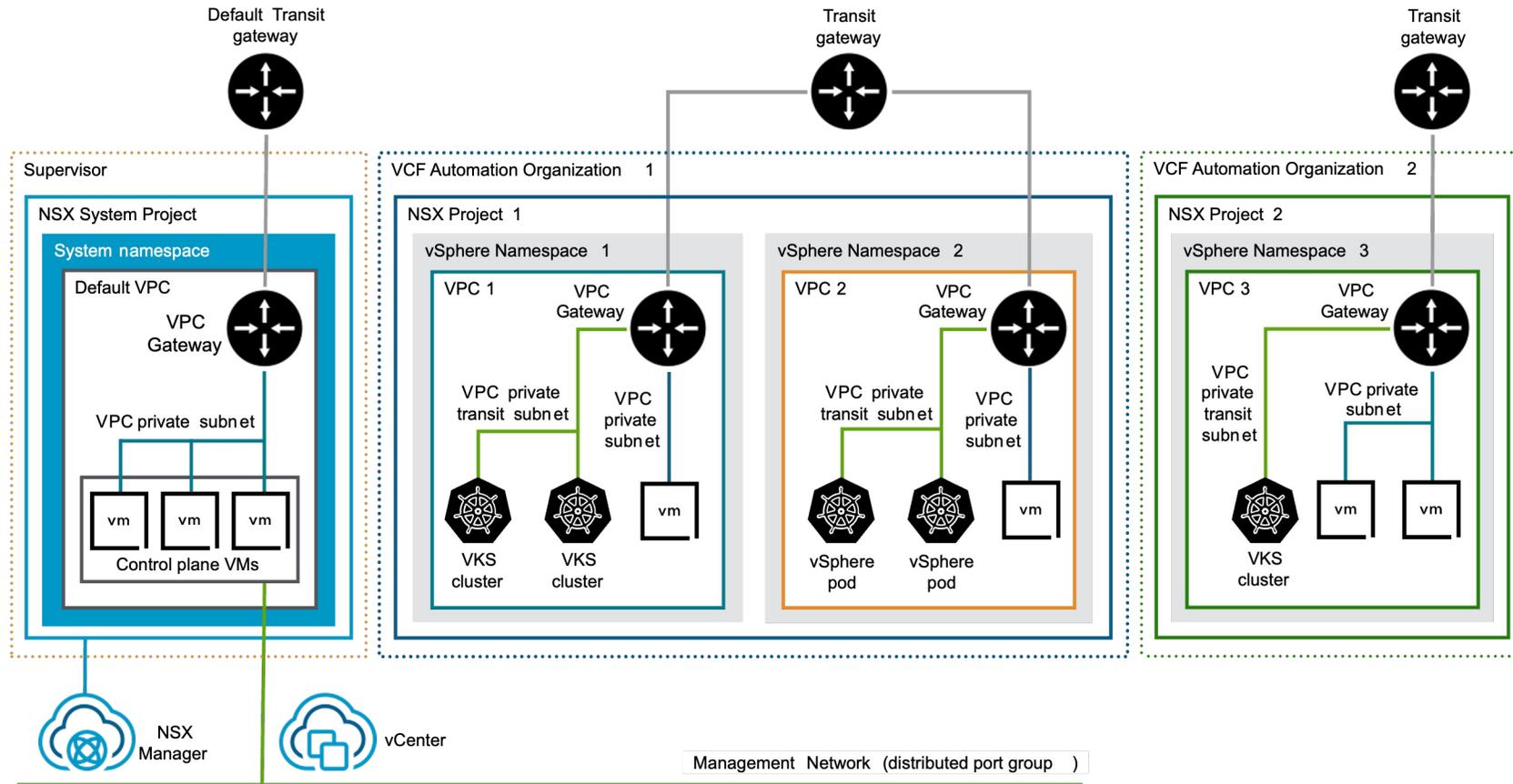
Deploy **Virtual Machines** with **VM Service**

Add additional services to **extend the capabilities** of the platform and support your apps



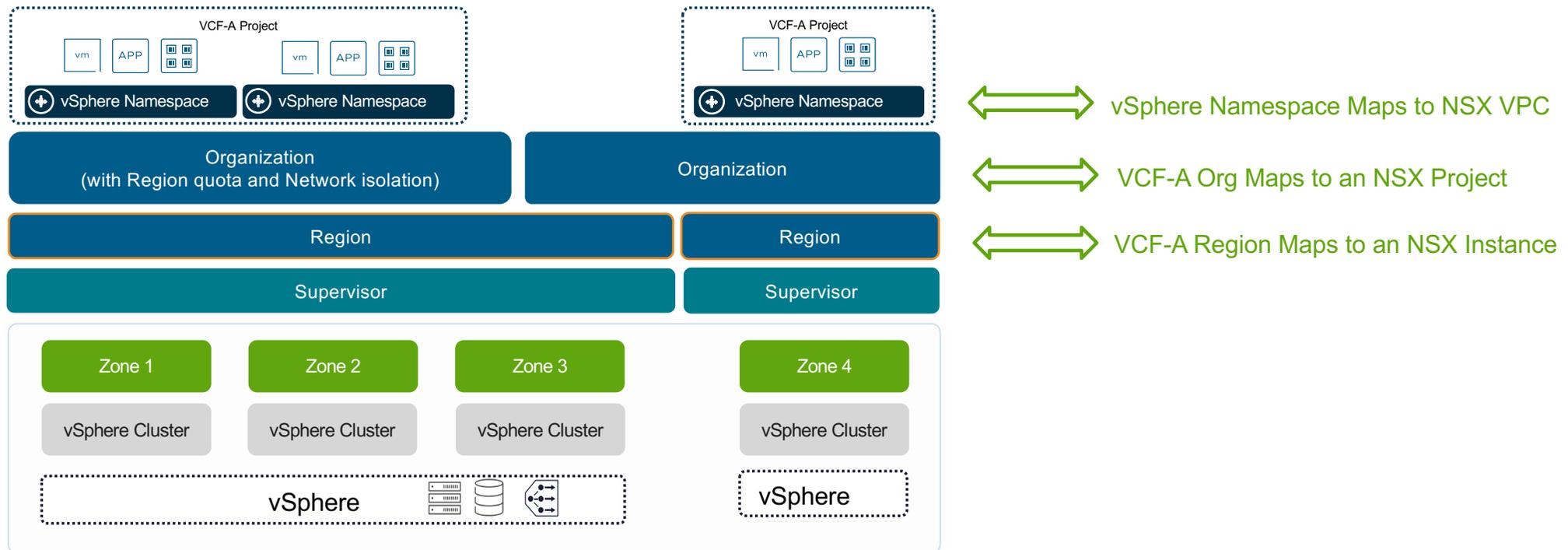
VCF9 connectivity across tenants and apps

A Union between vSphere Supervisor and NSX Virtual Private Clouds (VPCs)



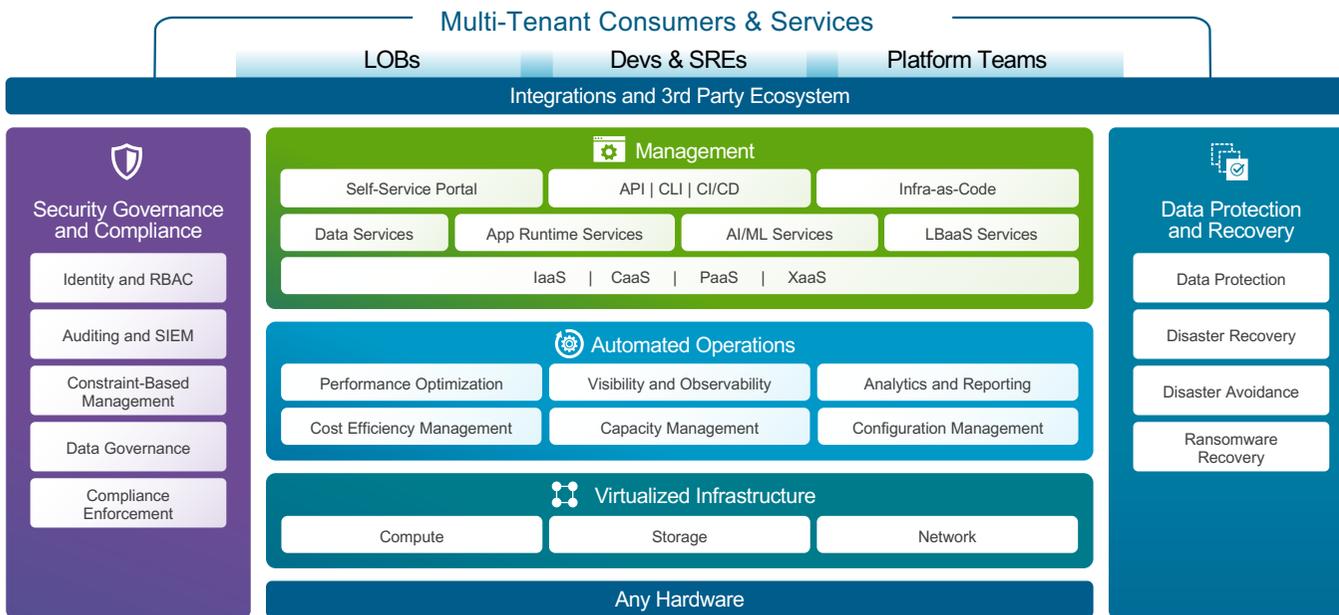
VCF9 Consumption across Locations & Tenants

A Union between VCF Automation, NSX VPCs, and vSphere Supervisor



Private Cloud Technology

Powered by VMware Cloud Foundation 9



- Infrastructure as Code & Automation** for consistency
- Orchestration & Catalogs** for scale and self-service
- Policy-as-Code** for embedded governance
- Observability & Feedback** for continuous improvement

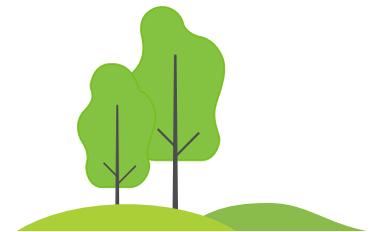
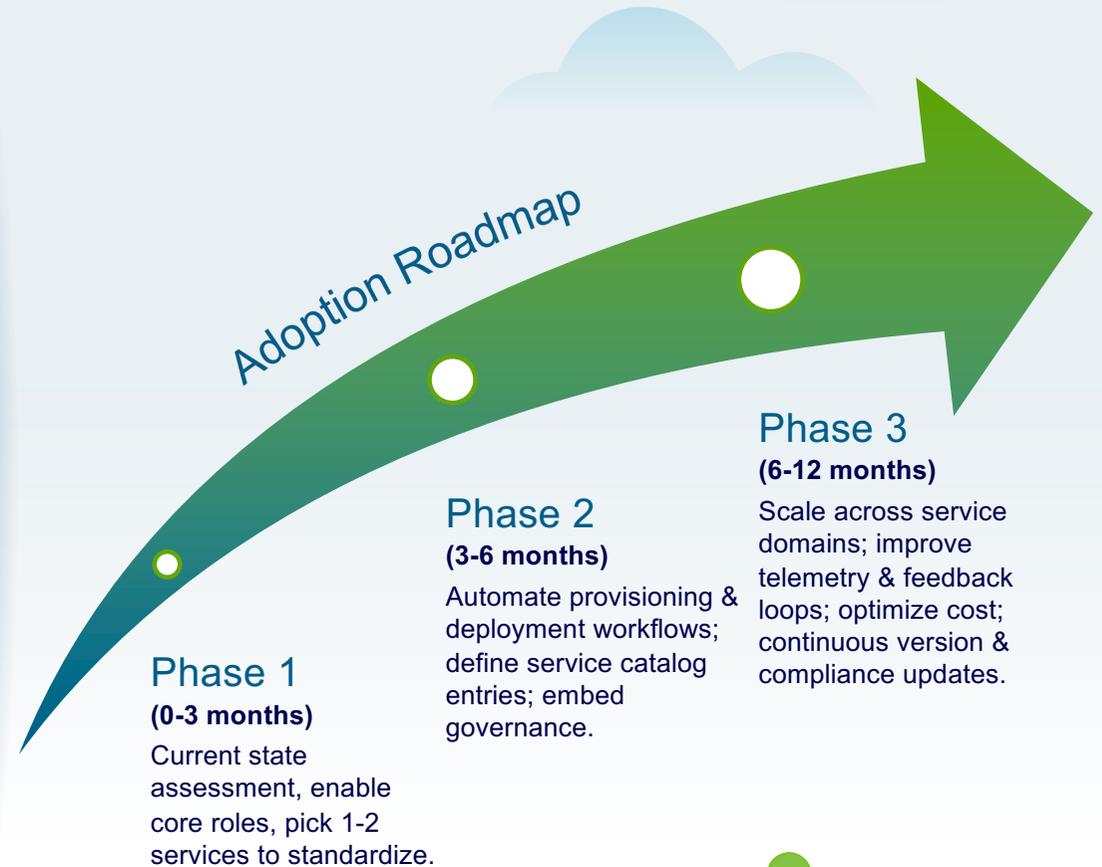
Private Cloud Operating Model Roadmap & Ownership for Success

What Must Happen

- Clear role ownership: Service Owner, SRE, SecOps, FinOps, Platform Engineers.
- Full lifecycle discipline: Define → Design → Catalog → Deliver → Improve.
- Standardize automation, templates, versioning, guardrails.
- Embed measurement & feedback: SLIs/SLOs, usage, cost, defect rate, consumer satisfaction.

Expected Outcomes

- Faster service adoption and higher platform stability.
- Less unplanned work, fewer incidents.
- Greater transparency in cost & resource usage.
- Improved satisfaction from both consumers and provider teams





Thank You