Microsoft Azure

Building Cloud Solutions

Microsoft

Host your applications on Azure

Microsoft Azure

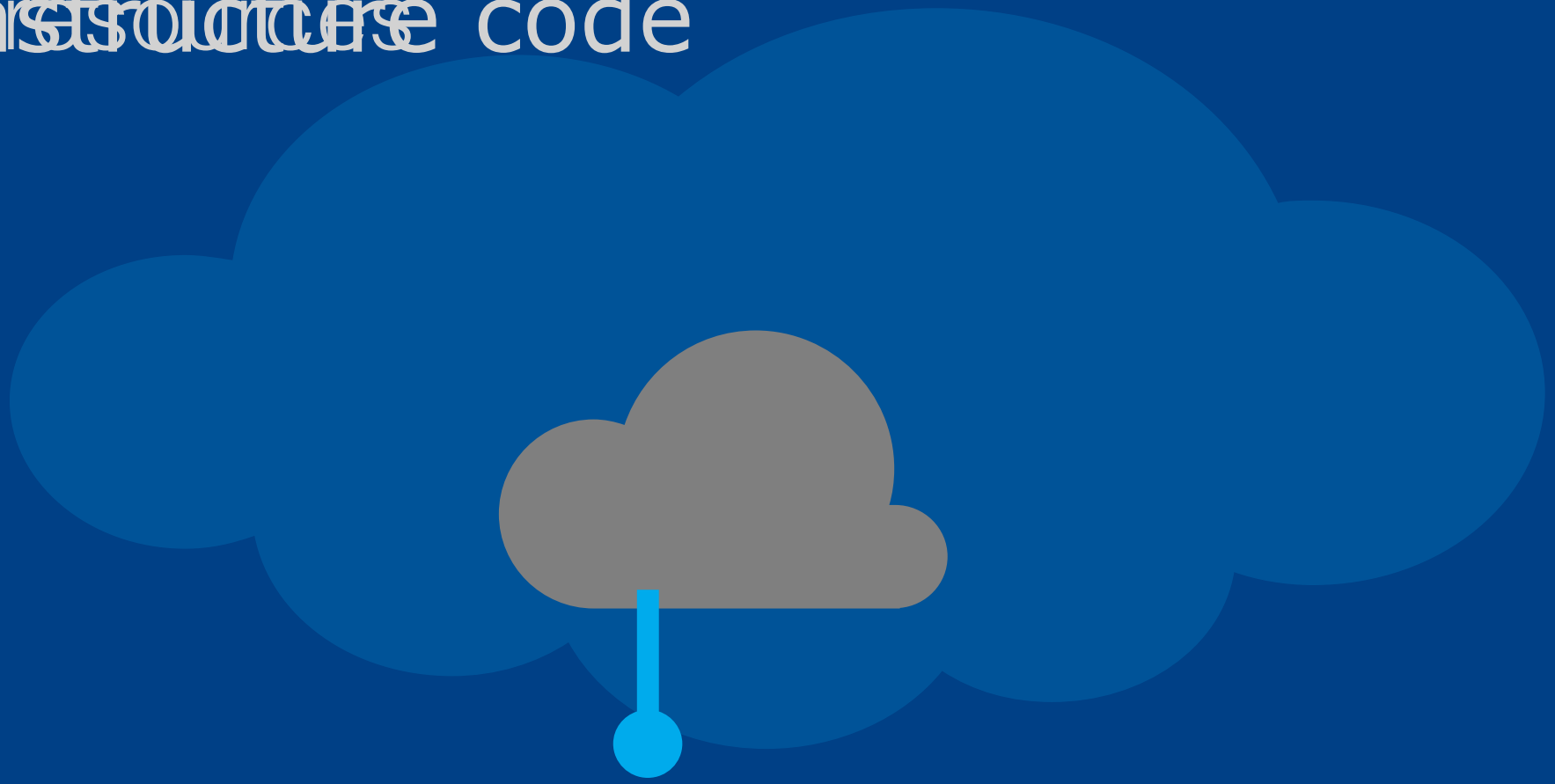
Infrastructure as a Service
IaaS

Platform as a Service
PaaS

Microsoft

Your service

- → Your application code
  - → Your infrastructure code



Automation

Innovation

Reduced cycle time

Elasticity

Versioning environments

Continuous Deployment

Speed to market

Agility

Continuous Improvements

Feed back

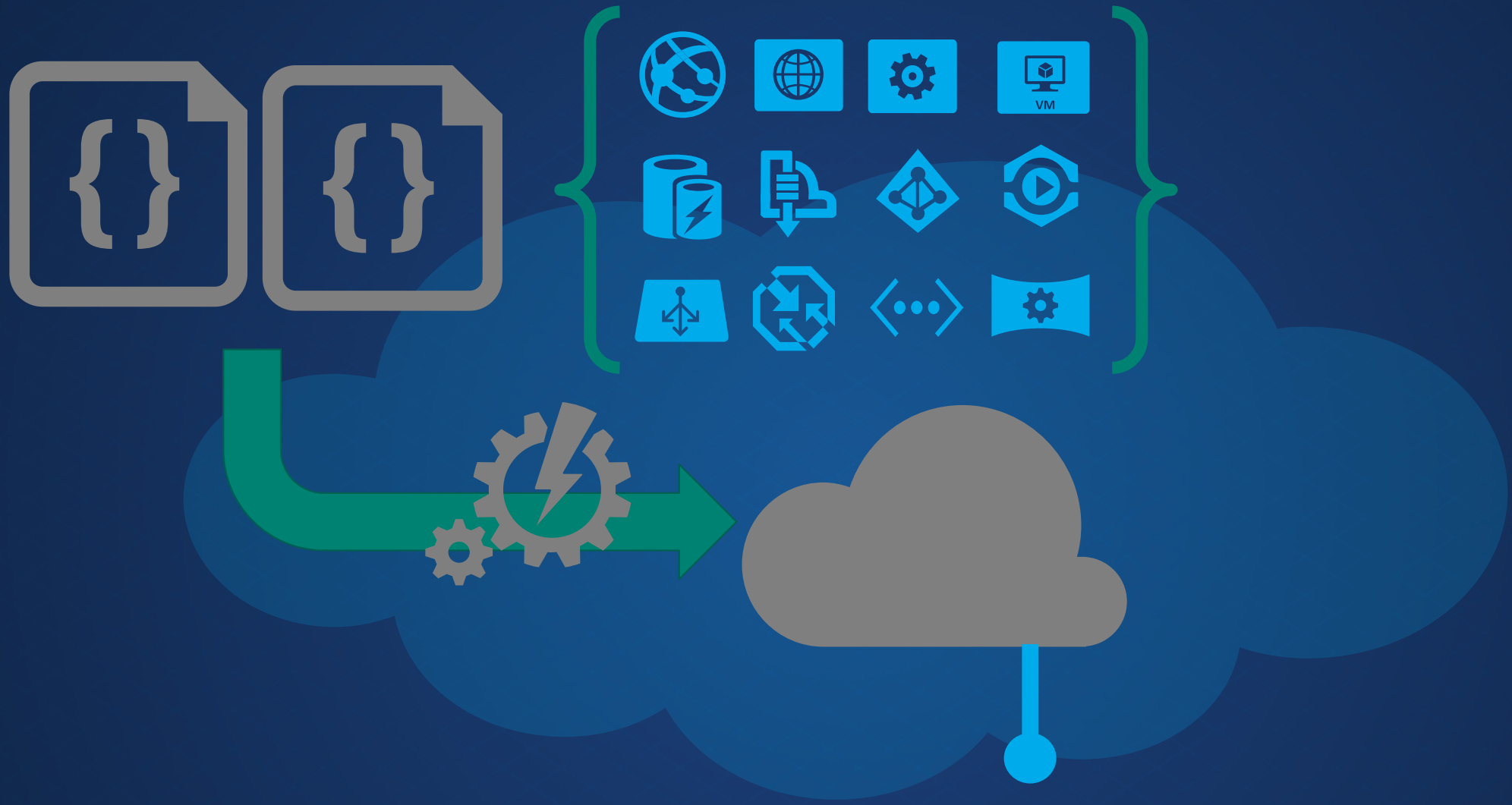
QoS

Availability

Growth

Insights

TCO



You: Code (application, infrastructure)
Azure: Resources (IaaS, PaaS, SaaS)

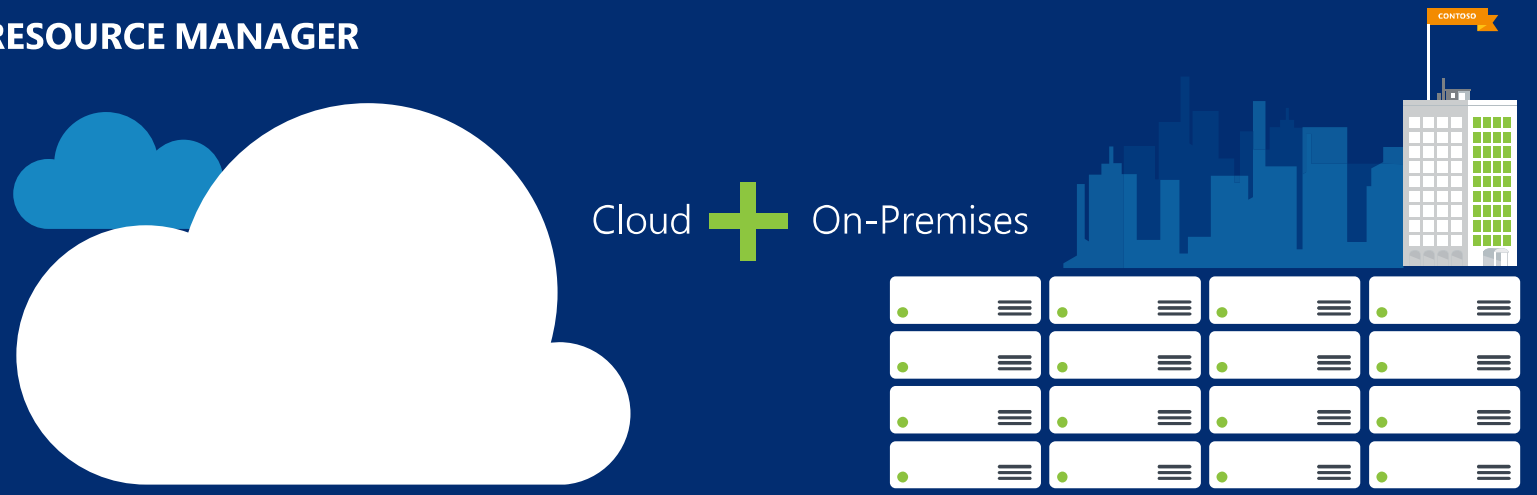
Consistent Management Layer

Tools

Microsoft Azure + Command Line + Visual Studio

SERVICE MANAGEMENT API

RESOURCE MANAGER

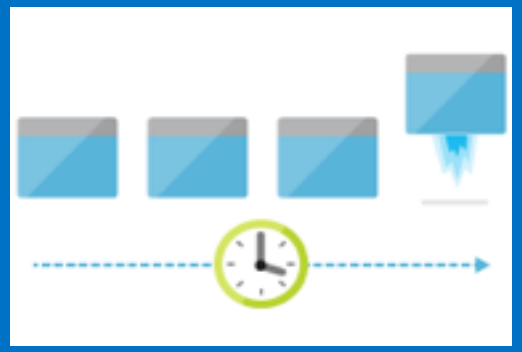


ADFS

AAD

RESOURCE PROVIDER CONTRACT

Provider Rest Points



Deploy

Organize

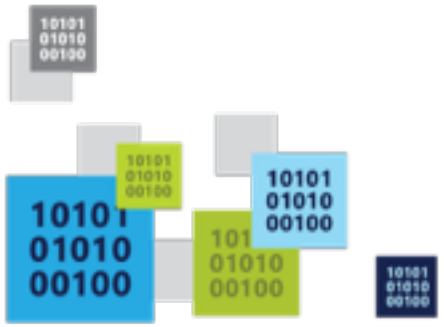
Control

- Deploy application lifecycle container with repeatable declarative model based template
- Organize resources by environment, role, department and user responsibility
- Control and monitor resources through RBAC, centralized audit and resource lock

Cloud Services

Cloud Services

- Focus on your application
- Scalability, availability and reliability
- Monitoring and diagnostics



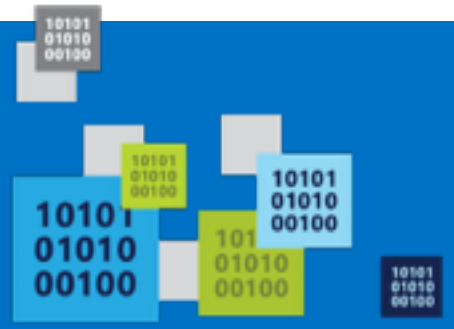
What is a Cloud Service?

A container of related service roles

Web Roles

Worker Roles

VMs

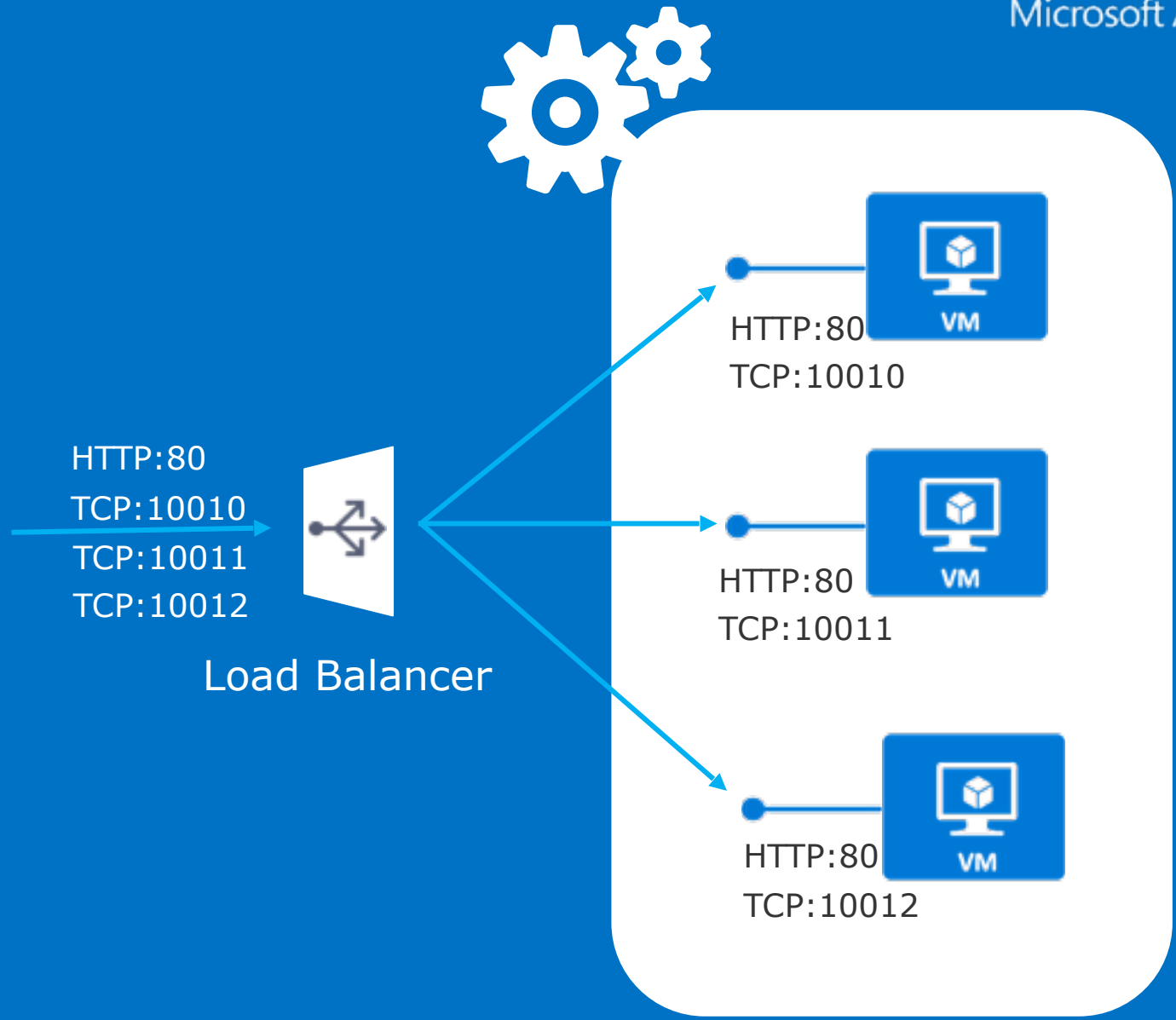


How do roles communicate?

- Public endpoints
  - Publicly accessible, load balanced
- Internal endpoints
  - Private to cloud service, not load balanced
- Instance Input endpoints
  - Address individual instance

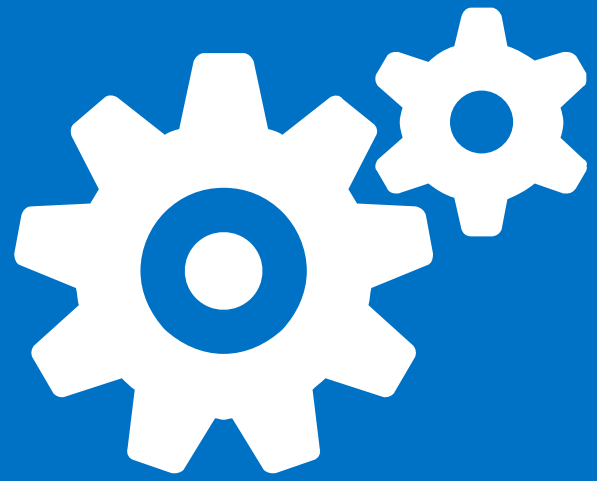
Endpoint

Name
Internal Port
Public Port
Protocol
IP



Worker Role

Microsoft Azure



Microsoft

Web Role

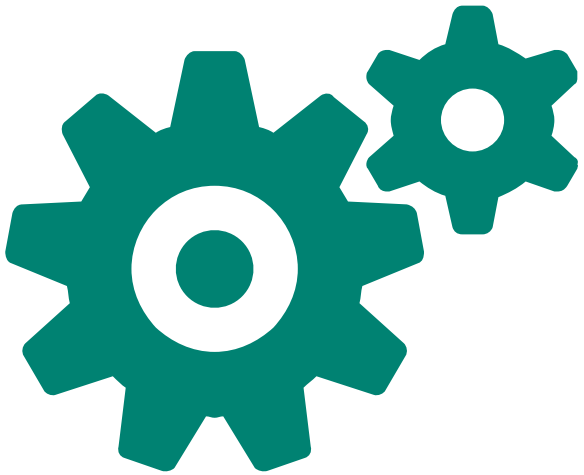
All features of a worker role + IIS 7, 7.5 or IIS 8.0*

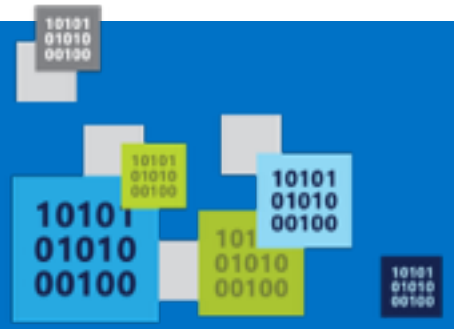
- All features of a worker role + IIS 7, 7.5 or IIS 8.0*
- ASP.NET 3.5 SP1, 4.0 or 4.5* – 64bit
- Hosts
  - Webforms or MVC
  - FastCGI applications (e.g. PHP)
  - Multiple Websites
- Http(s)
- Web/Worker Hybrid
  - Can optionally implement RoleEntryPoint

*with Windows Server 2012

Worker Role Patterns

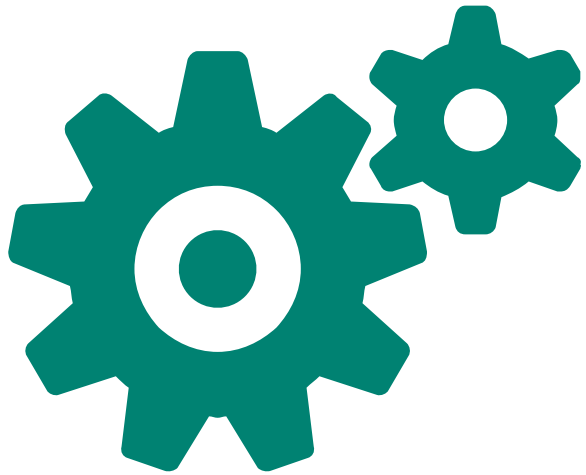
- Queue Polling Worker
  - Poll and Pop Messages within while(true) loop
  - E.g. Map/Reduce pattern, background image processing
- Listening Worker Role
  - Create TcpListener or WCF Service Host
  - E.g. Run a .NET SMTP server or WCF Service

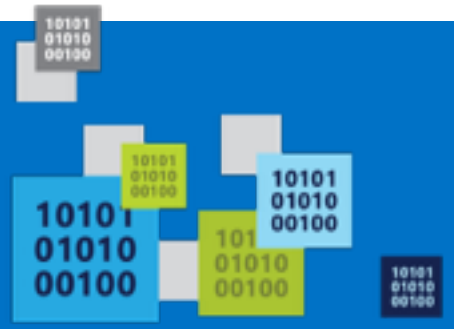




Worker Role Patterns

- External Process Worker Role
  - OnStart or Run method executes Process.Start()
  - Startup Task installs or executes background/ foreground process
- Custom Role Entry Point (executable or .Net assembly)
- E.g. Run a database server, web server, distributed cache





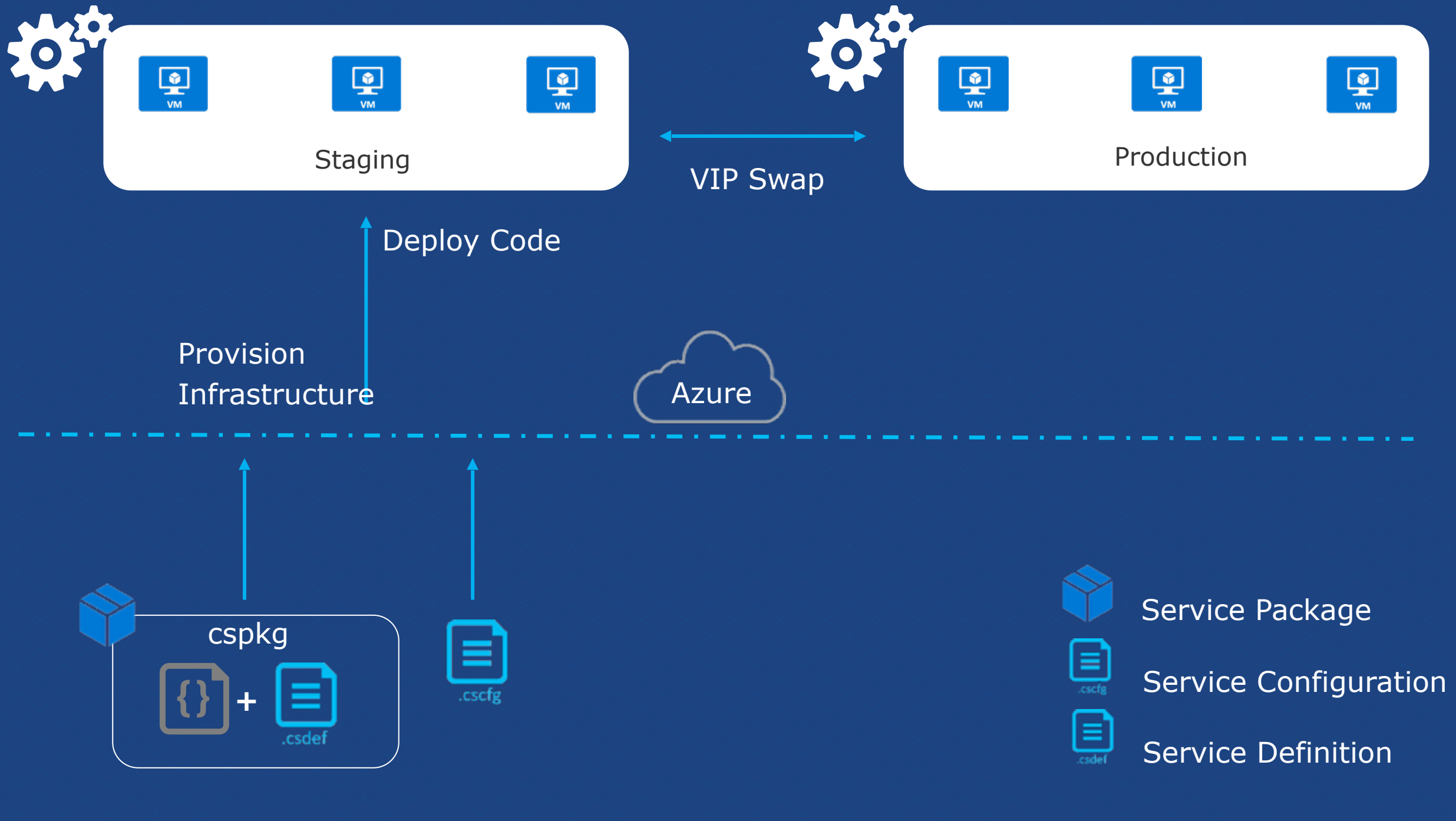
Roles and Instances

Roles are defined in a Hosted Service

A role definition specifies:
VM size
Communication Endpoints
Local storage resources
etc.

At runtime each Role will execute on one or more instances

A role instance is a set of code, configuration, and local data, deployed in a dedicated VM


VM
VM
VM
Staging
VIP Swap
VM
VM
VM
Production
Deploy Code
Provision
Infrastructure
Azure
cspkg
{}
+
.csdef
.cscfg
Service Package
.cscfg
Service Configuration
.csdef
Service Definition

Integrated development experience powered by Visual Studio + Azure SDK

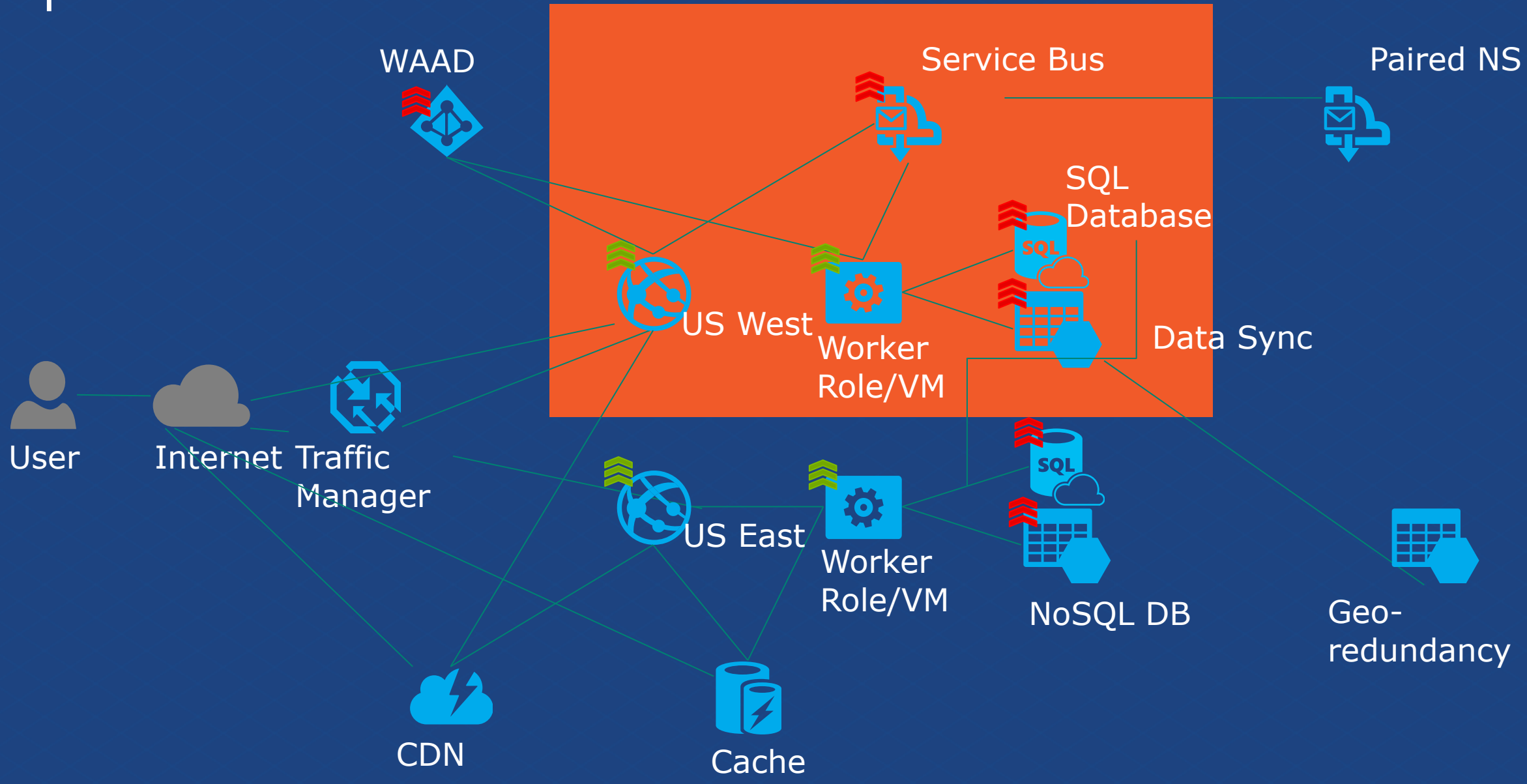
- Development experience using the Azure SDK, integrated seamlessly with Visual Studio.
- Deploy using any language you like, including .NET, Java, Node.js, PHP, Python, or Ruby.
- Test your application before deploying to the cloud using the Azure Emulator, which brings the platform's key functionality right to your dev machine.

Design for Cloud

A different mindset

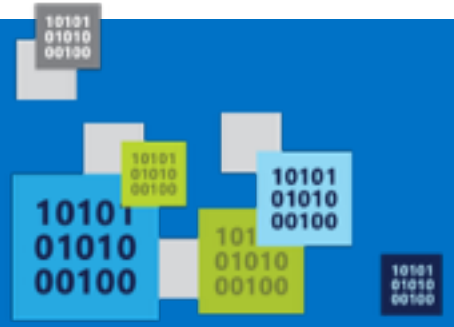
- → Embracing errors
- → Design for availability, reliability, scalability
- → Performance

Sample architecture



Redundancy in Microsoft Azure

- Storage Redundancy Options
- SQL Database Geo-Replication
- Caching with high availability option
- Load-balanced App Service Apps, Cloud Service and Virtual Machines
- Built-in redundancy in Azure Virtual Network gateways
- Failover with Azure Traffic Manager



Resiliency in Microsoft Azure

- Auto recovery – Service Healing
- Fault domain - prevent single point of failure
- Virtual machine Availability set – fault domain and rolling host updates
- Upgrade domain - service availability during upgrade
- Deployment Slots and VIP swap
- Emulator, Intellitrace and enhanced diagnostics
- Telemetry - native and 3rd party support
- Handle transient errors with Application Block

What does failsafe mean for my applications?

It depends… but some general practices apply.

- Take advantage of Microsoft Azure features
- Avoid single point of failure
- Failure mode analysis
- Transient errors
- Graceful degradation
- Eliminate human factors

Scaling in Microsoft Azure

- Scale up by choosing different VM sizes
- Scale out by adding more instances
- Auto-scale with Autoscaling Application Block
- Scale out by using multiple service entities
- CDN to distribute user traffic
- Caching to offset server workloads