Introduction to Windows Azure

Cloud Computing
A Game Changing Technology

- Location independent computing
- Shared servers → resources, software, and data
- Elasticity
- Total Cost of Operation
- Natural evolution of:
  - **Virtualization**
  - **Service-Oriented Architecture**
  - **Utility computing**
- Details are abstracted from consumers

Cloud Fundamentals

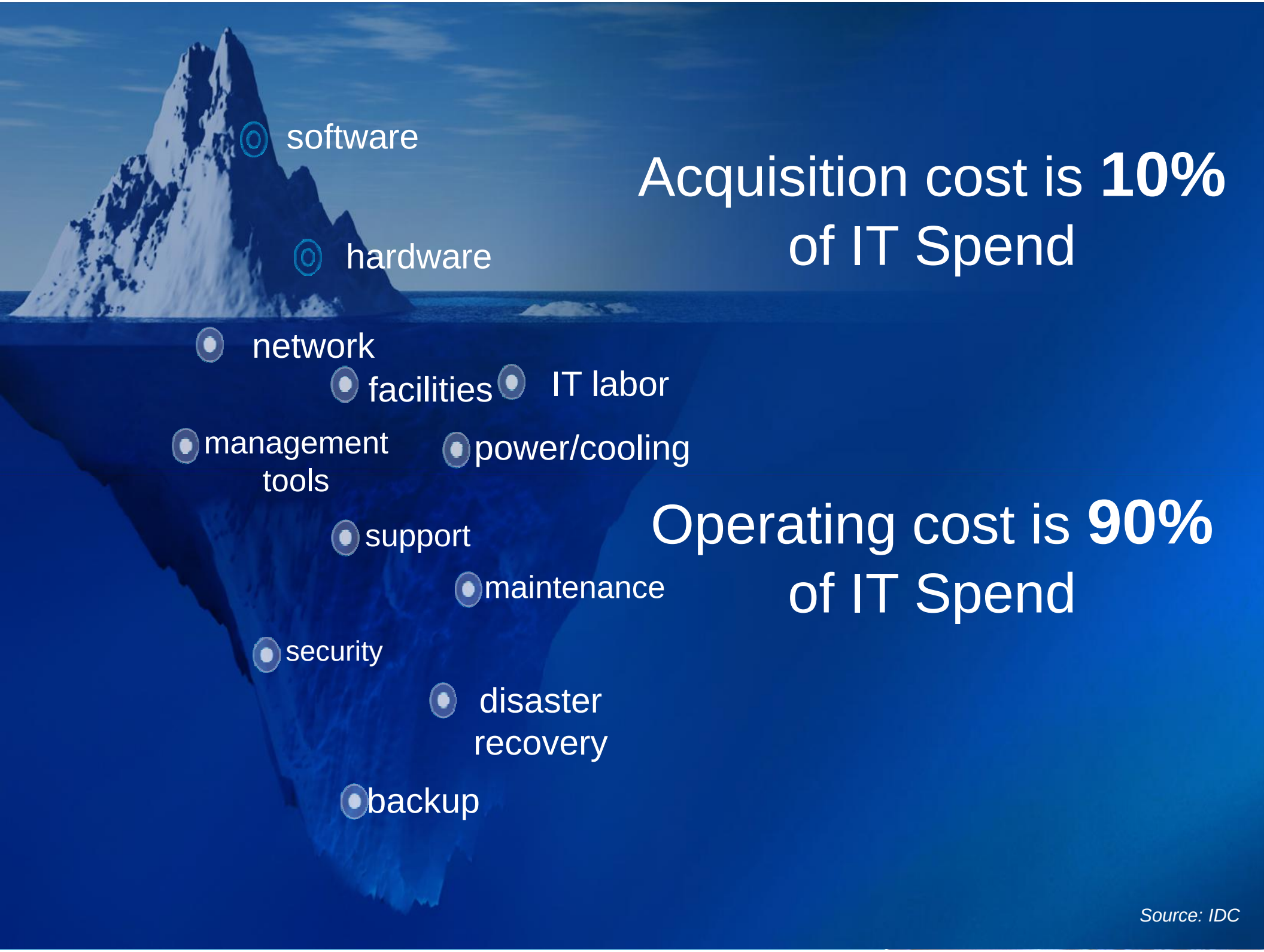
- Infrastructure as a Service (IaaS): basic compute and storage resources
  - **On-demand servers**
  - **Amazon EC2, VMWare vCloud**
- Platform as a Service (PaaS): cloud application infrastructure
  - **On-demand application-hosting environment**
  - **E.g. Google AppEngine, Salesforce.com, Windows Azure, Amazon**
- Software as a Service (SaaS): cloud applications
  - **On-demand applications**
  - **E.g. GMail, Microsoft Office Web Companions**

The Benefits of the Cloud

- The Cloud is about cheap, on-demand capacity

Windows Azure

= Managed for You	Standalone Servers	IaaS	PaaS	SaaS
Applications	✗	✗	✗	✓
Runtimes	✗	✗	✓	✓
Database	✗	✗	✓	✓
Operating System	✗	✗	✓	✓
Virtualization	✗	✓	✓	✓
Server	✗	✓	✓	✓
Storage	✗	✓	✓	✓
Networking	✗	✓	✓	✓


software
hardware
network
facilities
IT labor
management tools
power/cooling
support
maintenance
security
disaster recovery
backup
Acquisition cost is 10% of IT Spend
Operating cost is 90% of IT Spend
Source: IDC

Windows Azure

- Platform as a Service
  - **Application Platform in the Cloud**
- Provides:
  - ***Compute***
    - Web, Worker & VM Role
  - ***Storage***
    - *Blob, Table, Queue & Azure SQL Server*
  - **Application *Fabric***
    - *Service Bus, Access Control, (Future: Cache, Integration & Composite)*

INSIDE WINDOWS AZURE: THE CLOUD OPERATING SYSTEM

Windows Azure

- Windows Azure is an OS for the data center
  - **Model: Treat the data center as a machine**
  - **Handles resource management, provisioning, and monitoring**
  - **Manages application lifecycle**
  - **Allows developers to concentrate on business logic**
- Provides shared pool of compute, disk and network
  - **Virtualized storage, compute and network**
  - **Illusion of boundless resources**
- Provides common building blocks for distributed applications
  - **Reliable queuing, simple structured storage, SQL storage**
  - **Application services like access control and connectivity**

Windows Azure Components

	Windows Azure PaaS
Applications	Windows Azure Service Model
Runtimes	.NET 3.5/4, ASP .NET, PHP
Operating System	Windows Server 2008/R2-Compatible OS
Virtualization	Windows Azure Hypervisor
Server	Microsoft Blades
Database	SQL Azure
Storage	Windows Azure Storage (Blob, Queue, Table)
Networking	Windows Azure-Configured Networking

Modeling Cloud Applications

- A cloud application is typically made up of different components
  - **Front end: e.g. load-balanced stateless web servers**
  - **Middle worker tier: e.g. order processing, encoding**
  - **Backend storage: e.g. SQL tables or files**
  - **Multiple instances of each for scalability and availability**

The Windows Azure Service Model

- A Windows Azure application is called a “service”
  - **Definition information**
  - **Configuration information**
  - **At least one “role”**
- Roles are like DLLs in the service “process”
  - **Collection of code with an entry point that runs in its own virtual machine**
- There are currently three role types:
  - **Web Role: IIS7 and ASP.NET in Windows Azure-supplied OS**
  - **Worker Role: arbitrary code in Windows Azure-supplied OS**
  - **VM Role: uploaded VHD with customer-supplied OS**

Role Contents

- Definition:
  - **Role name**
  - **Role type**
  - **VM size (e.g. small, medium, etc.)**
  - **Network endpoints**
- Code:
  - **Web/Worker Role: Hosted DLL and other executables**
  - **VM Role: VHD**
- Configuration:
  - **Number of instances**
  - **Number of update and fault domains**

Service Model Files

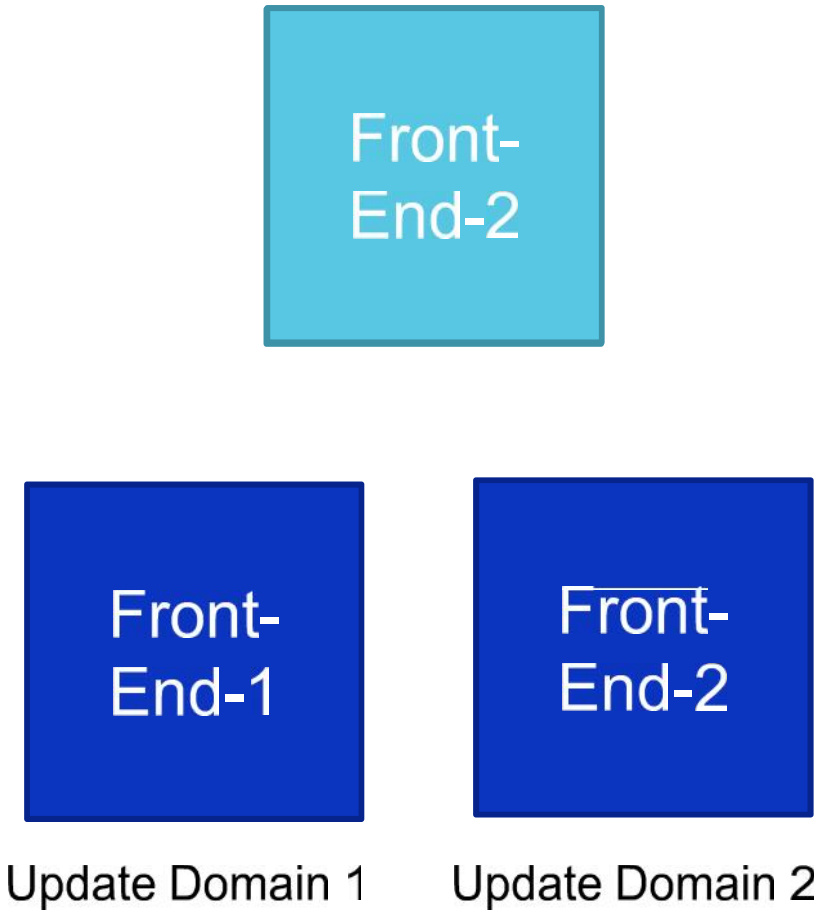
- Service definition is in ServiceDefinition.csdef
- Service configuration is in ServiceConfiguration.cscfg
- CSPack program Zips service binaries and definition into service package file (service.cscfg)

```
<?xml version="1.0" encoding="utf-8"?>
<ServiceDefinition name="Thumbnails" xmlns="http://schemas.micros
  <WorkerRole name="Thumbnails_WorkerRole">
    <ConfigurationSettings>
      <Setting name="DataConnectionString" />
      <Setting name="DiagnosticsConnectionString" />
    </ConfigurationSettings>
  </WorkerRole>
  <WebRole name="Thumbnails_WebRole">
    <InputEndpoints>
      <!-- Must use port 80 for http and port 443 for https when
      <InputEndpoint name="HttpIn" protocol="http" port="80" />
    </InputEndpoints>
    <ConfigurationSettings>
      <Setting name="DataConnectionString" />
      <Setting name="DiagnosticsConnectionString" />
    </ConfigurationSettings>
  </WebRole>
</ServiceDefinition>
```

```
<?xml version="1.0"?>
<ServiceConfiguration serviceName="Thumbnails" xmlns="ht
  <Role name="Thumbnails_WorkerRole">
    <Instances count="2" />
    <ConfigurationSettings>
      <!-- Add your storage account information and unco
        <Setting name="DataConnectionString" value="Defa
        <Setting name="DiagnosticsConnectionString" valu
      -->
      <Setting name="DataConnectionString" value="Defaul
      <Setting name="DiagnosticsConnectionString" value=
    </ConfigurationSettings>
  </Role>
  <Role name="Thumbnails_WebRole">
    <Instances count="1" />
    <ConfigurationSettings>
      <!-- Add your storage account information and unco
        <Setting name="DataConnectionString" value="Defa
        <Setting name="DiagnosticsConnectionString" valu
      -->
      <Setting name="DataConnectionString" value="Defaul
      <Setting name="DiagnosticsConnectionString" value=
    </ConfigurationSettings>
  </Role>
</ServiceConfiguration>
```

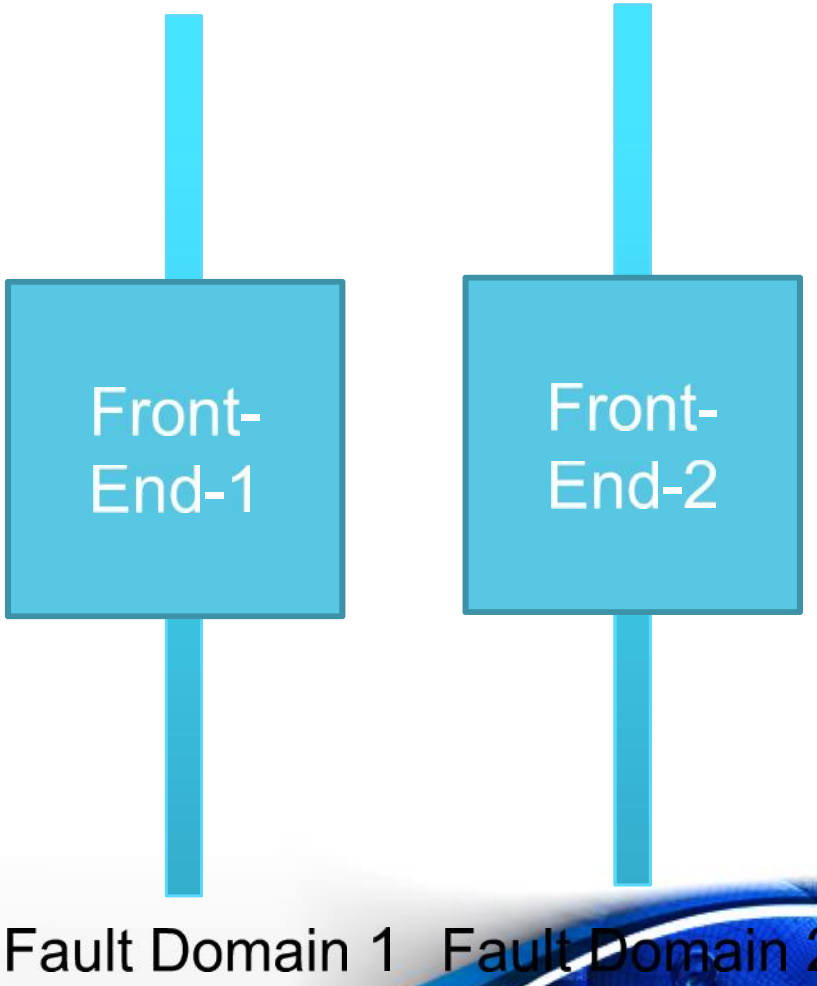
Availability: Update Domains

- Purpose: Ensure service stays up while updating and Windows Azure OS updates
- System considers update domains when upgrading a service
  - **Update domains/Instance count = percent of service that will be offline**
  - **Default and max is 5, but you can override with upgradeDomainCount service definition element**
- The Windows Azure SLA is based on at least two update domains and two role instances in each role



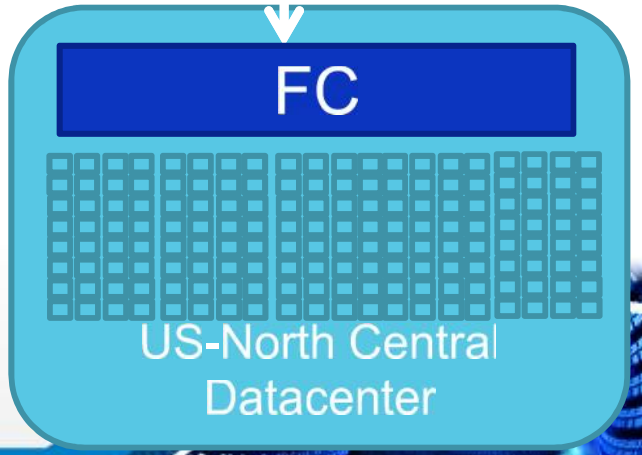
Availability: Fault Domains

- Purpose: Avoid single points of failures
  - **Similar concept to update domains**
  - **But you don't control the updates**
- Unit of failure based on data center topology
  - **E.g. top-of-rack switch on a rack of machines**
- Windows Azure considers fault domains when allocating service roles
  - **E.g. don't put all roles in same rack**



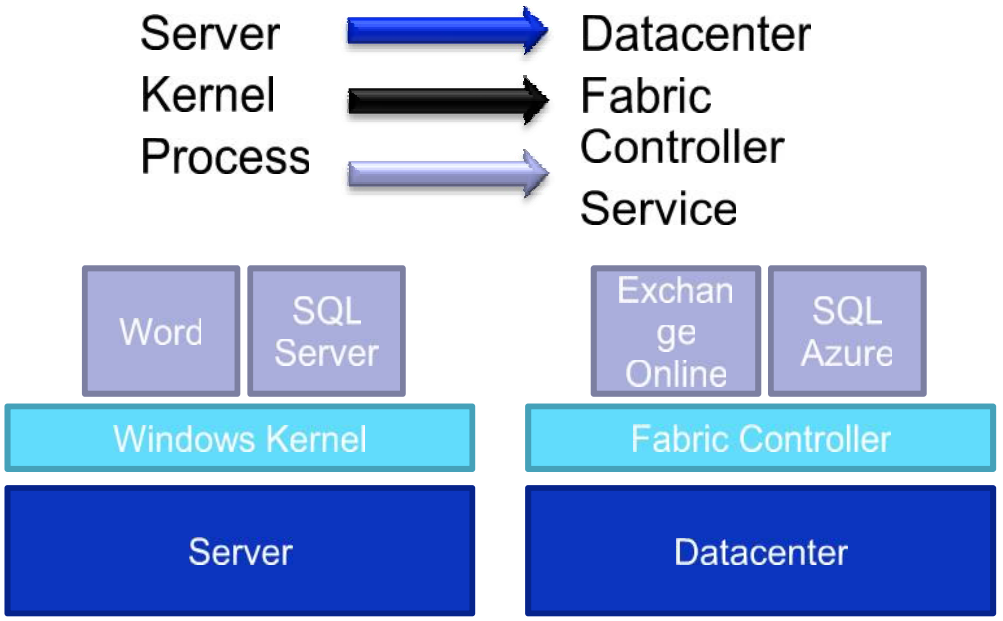
Deploying a Service
The 10,000 foot view

- Service package uploaded to portal
  - **Windows Azure Portal Service passes service package to “Red Dog Front End” (RDFE) Azure service**
  - **RDFE converts service package to native “RD” version**
- RDFE sends service to Fabric Controller (FC) based on target region
- FC stores image in repository and deploys and activates service

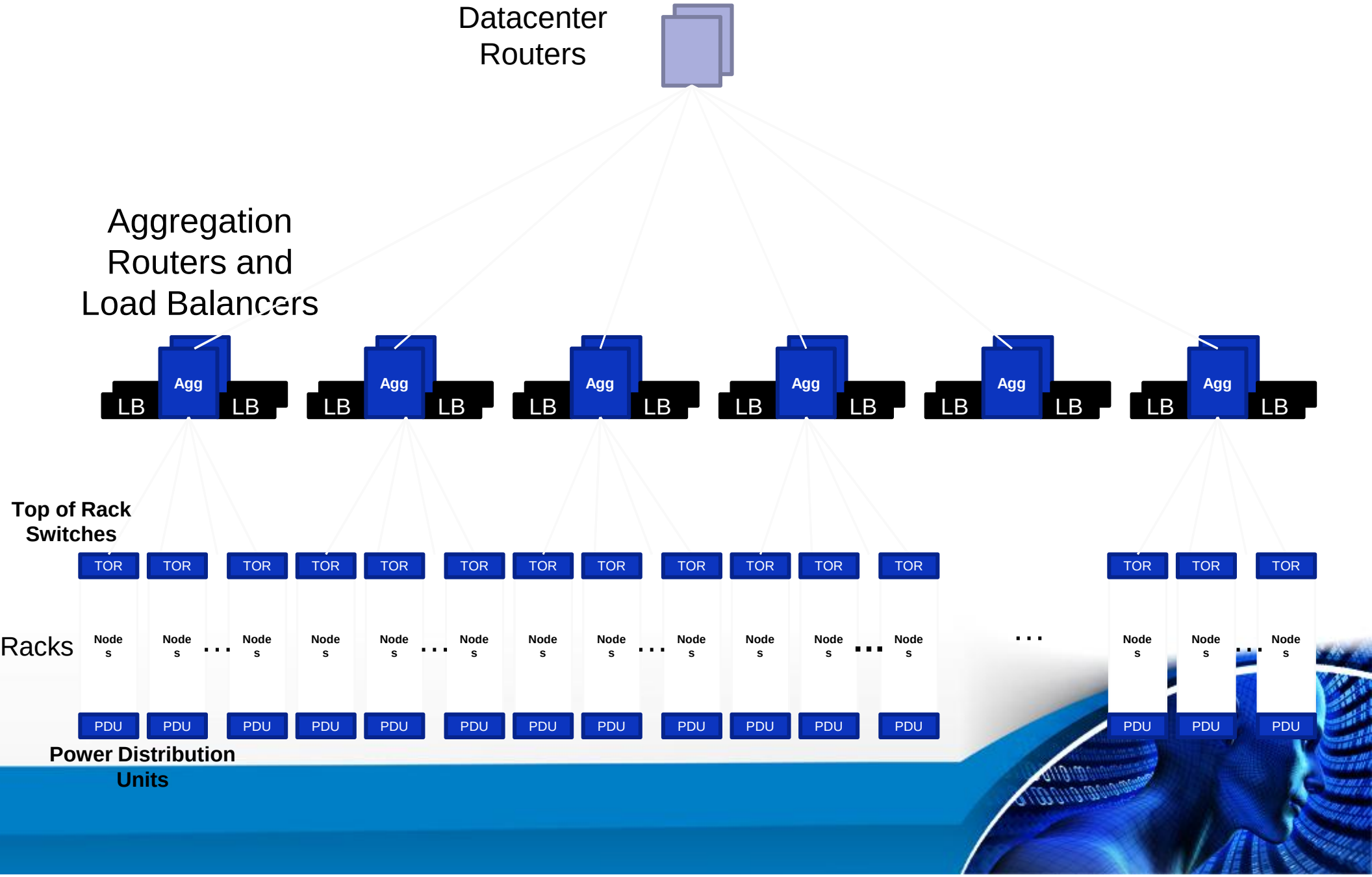


The Fabric Controller (FC)

- The “kernel” of the cloud operating system
  - **Manages datacenter hardware**
  - **Manages Windows Azure services**
- Four main responsibilities:
  - **Datacenter resource allocation**
  - **Datacenter resource provisioning**
  - **Service lifecycle management**
  - **Service health management**
- Inputs:
  - **Description of the hardware and network resources it will control**
  - **Service model and binaries for cloud applications**

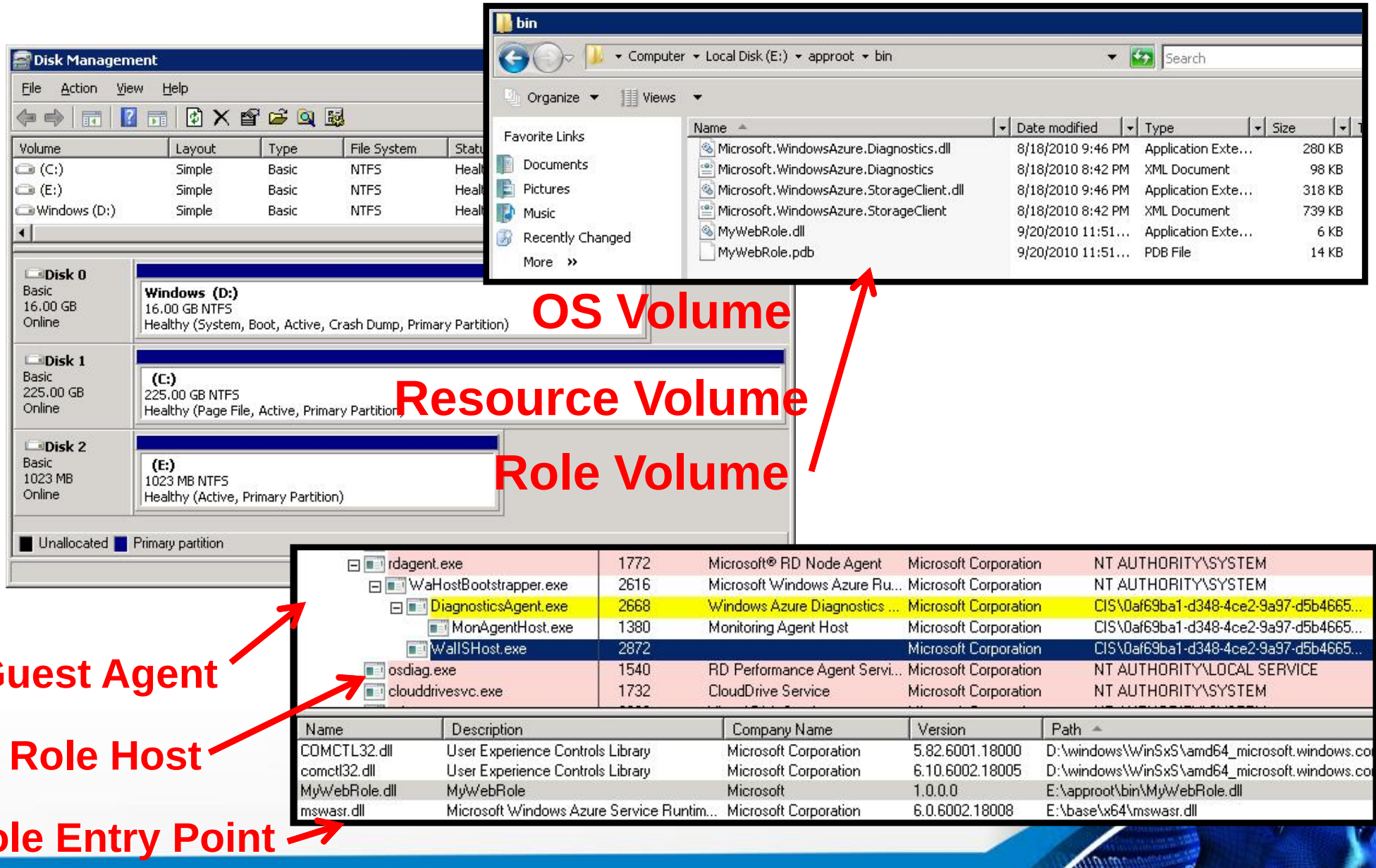


Datacenter Architecture



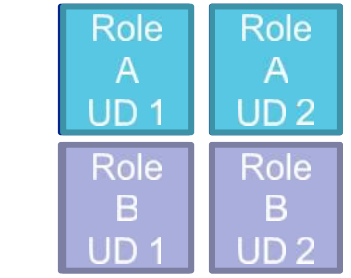
Windows Azure Datacenters

Inside a Role VM

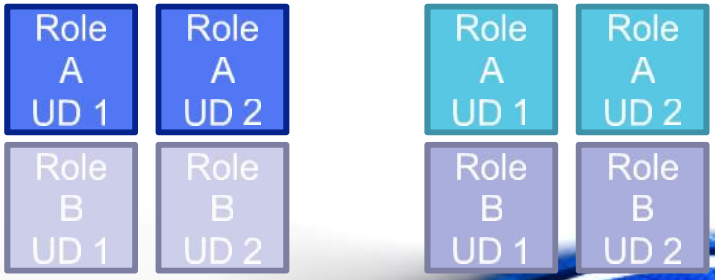


Update Types

- There are two update types:
  - **In-place update:**
    - Supports changes to configuration or binaries, not service definition
    - Role instances upgraded one update domain at a time
    - Two modes: automatic and manual
  - **VIP swap update:**
    - Service definition can change, but external endpoints must remain the same
    - New version of service deployed, external VIP/DIP mapping swapped with old
- Changes to external endpoint count require a new deployment



In-Place Update

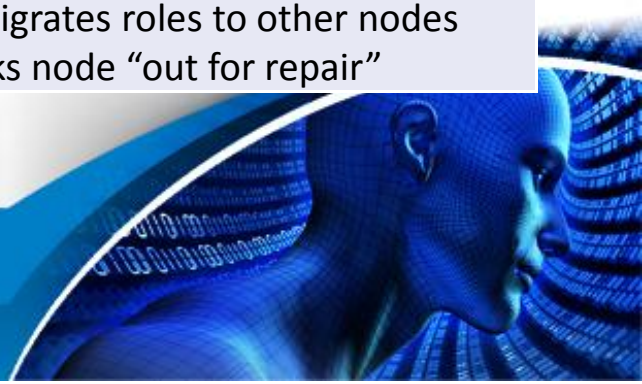


VIP Swap Update

Node and Role Health Maintenance

- FC maintains service availability by monitoring the software and hardware health
  - **Based primarily on heartbeats**
  - **Automatically "heals" affected roles**

Problem	How Detected	Fabric Response
Role instance crashes	FC guest agent monitors role termination	FC restarts role
Guest VM or agent crashes	FC host agent notices missing guest agent heartbeats	FC restarts VM and hosted role
Host OS or agent crashes	FC notices missing host agent heartbeat	Tries to recover node FC reallocates roles to other nodes
Detected node hardware issue	Host agent informs FC	FC migrates roles to other nodes Marks node "out for repair"



Azure Architecture Summary

- Platform as a Service is all about reducing management and operations overhead
- The Windows Azure Fabric Controller is the foundation for Windows Azure's PaaS
  - **Provisions machines**
  - **Deploys services**
  - **Configures hardware for services**
  - **Monitors service and hardware health**
  - **Performs service healing**

End