

Azure Core Services & Infrastructure - 3 Session Plan

Session 1: Azure Architecture Fundamentals (2 Hours)

Learning Objectives

By the end of this session, learners will be able to:

- Understand Azure's global infrastructure organization
- Differentiate between Regions, Availability Zones, and Region Pairs
- Explain Azure's hierarchical management structure
- Create and organize resource groups effectively

2.1 Azure Architectural Components

Introduction to Azure Infrastructure (30 mins) Azure Global Infrastructure Overview - Worldwide Presence: 60+ regions globally - Design Principle: Redundancy, scalability, and low latency - Key Components: Regions, Geographies, Availability Zones

What are Azure Regions? - Definition: Geographical locations containing multiple data centers - Key Characteristics: - Isolated from other regions - Contains one or more data centers - Resources are deployed to specific regions - Consider latency, compliance, and service availability when choosing

Hands-on Example:

```
# Common region codes:  
# East US      -> eastus  
# West Europe  -> westeurope  
# Southeast Asia-> southeastasia
```

Regions and Availability Zones (45 mins) Region Pairs: - Concept: Most Azure regions are paired with another region within the same geography - Benefits: - Automatic Replication: Platform-level replication for disaster recovery - Isolated Updates: Regions updated sequentially to prevent simultaneous downtime - Data Residency: Paired regions stay within the same geography for compliance

Example Region Pairs: | Geography | Primary Region | Paired Region | |-----|
|-----|-----|| North America | East US | West US || Europe | North Europe | West Europe || Asia | East Asia | Southeast Asia |

Availability Zones: - Definition: Physically separate locations within an Azure region - Composition: Each zone has independent: - Power source - Cooling systems - Networking infrastructure - Physical security

Zone Structure: - Minimum: 3 zones per enabled region - Design: Maximum physical separation within regional boundaries - Purpose: Protect against datacenter-level failures

Service Availability Categories: | Category | Description | Examples | |-----|-----|-----|
Zonal Services	Pin resource to specific zone	Virtual Machines, Managed Disks
Zone-Redundant	Platform replicates across zones	
Zone-redundant storage, SQL DB	Non-regional	No dependency on specific region
Azure AD, Traffic Manager		

Subscriptions and Management Groups (45 mins) Azure Subscription: - Definition: Fundamental unit of management, billing, and scaling - Key Roles: - Billing Boundary: Separate invoicing and cost management - Access Control Boundary: Role assignments scope - Scale Limits: Service quotas and limits apply per subscription

Subscription Types: - Free Account: \$200 credit for 30 days, limited free services
- Pay-As-You-Go: Flexible payment based on usage - Enterprise Agreement: Volume licensing for large organizations - Azure for Students: Free access to popular services

Management Groups: - Definition: Containers for managing multiple subscriptions - Hierarchy Benefits: - Apply governance policies at scale - Manage access centrally - Organize by department, environment, or project

Management Group Hierarchy Example:

```
graph TD; Root[Root Management Group] --> Production[Production (Management Group)]; Root --> Dev[Development (Management Group)]; Root --> Sandbox[Sandbox (Management Group)]; Production --> IT[IT Department Subscription]; Production --> Finance[Finance Department Subscription]; Dev --> DevTeam[Dev Team Subscription]; Dev --> TestTeam[Test Team Subscription]; Sandbox --> Training[Training Subscription];
```

Resource Groups: - Definition: Logical containers that hold related Azure resources - Key Characteristics: - Resources exist in only one resource group - No nesting of resource groups - Resources can be in different regions than their resource group - Lifecycle management - delete group deletes all contained resources

Resource Group Best Practices: 1. Group by Lifecycle: Resources with same deployment/update cycles 2. Consistent Naming: Use standardized naming conventions 3. Access Control: Apply RBAC at resource group level 4. Organization: Align with application architecture or teams

Practical Example - Resource Group Organization:

- rg-network-prod (VNet, NSGs, Route Tables)
 - rg-compute-prod (VMs, Scale Sets, App Services)
 - rg-data-prod (Storage, Databases, Data Factories)

- rg-security-prod (Key Vaults, Security Center)
-

Session 2: Compute Services Deep Dive (2 Hours)

Learning Objectives

By the end of this session, learners will be able to:

- Differentiate between Azure compute services
- Select appropriate compute options based on requirements
- Understand virtual machine configurations and scaling
- Explain container and serverless computing concepts

2.2 Compute and Networking - Part 1

Compute Types Overview (30 mins) Virtual Machines (IaaS) - The Foundation - What: Software emulation of physical computers - Management Responsibility: You manage OS, runtime, and applications - Use Cases: - Lift-and-shift migrations - Applications requiring custom configurations - Development and test environments - Legacy systems that can't be easily modernized

Virtual Machine Components: - vCPU: Virtual processors - Memory: RAM allocation - Storage: OS disk and data disks - Networking: Virtual network interface - Operating System: Windows or Linux distributions

Advanced Compute Options (45 mins) Virtual Machine Scale Sets: - What: Group of identical, load-balanced VMs - Key Features: - Automatic scaling based on metrics - High availability distribution - Centralized management - Use Cases: - Large-scale applications - Batch processing - Compute-intensive workloads - Microservices architectures

App Service (PaaS) - Managed Web Applications: - What: Fully managed platform for web applications and APIs - Supported Stacks: - .NET, .NET Core - Java, Node.js - Python, PHP - Ruby, Docker containers - Benefits: - No infrastructure management - Automatic scaling and patching - Built-in continuous deployment - Multiple pricing tiers

Container and Serverless Computing (45 mins) Azure Container Instances (ACI): - What: Simplest way to run containers in Azure - Use Cases: - Simple applications - Task automation - Build jobs - Development/testing - Benefits: - No VM management required - Per-second billing - Fast startup times

Azure Kubernetes Service (AKS): - What: Managed Kubernetes container orchestration - Key Concepts: - Pods: Smallest deployable units - Nodes: Underlying VMs running containers - Services: Network endpoints for pods - Deployments: Declarative updates for pods - Benefits: - Automated updates and scaling - Self-healing capabilities - Enterprise-grade security

Serverless Computing with Azure Functions: - What: Event-driven serverless compute service - Execution Model: Code runs in response to events - Triggers Examples: - HTTP requests (webhooks, APIs) - Timer/schedule (cron jobs) - Storage events (new files in Blob storage) - Message queues (Service Bus, Event Grid) - Benefits: - Pay only when code executes - Automatic scaling - No infrastructure management

Session 3: Application Hosting & Networking (2 Hours)

Learning Objectives

By the end of this session, learners will be able to: - Choose appropriate application hosting solutions - Understand Azure networking fundamentals - Design basic network architectures - Implement load balancing and connectivity solutions

2.2 Compute and Networking - Part 2

Application Hosting Strategies (60 mins) Compute Service Selection Framework:

Decision Flowchart:

Application Requirements →

Need full OS control?
Single instance? → Virtual Machines
Multiple instances? → VM Scale Sets

Web application/API?
App Service (PaaS)

Container-based?
Simple container? → Container Instances
Orchestration needed? → AKS

Event-driven tasks?
Azure Functions (Serverless)

Scaling Considerations: | Service | Scaling Type | Scaling Trigger | |——|——
——|——| | Virtual Machines | Manual | Administrative action | | VM Scale Sets | Automatic | Metrics-based rules | | App Service | Automatic | Metrics or schedule | | AKS | Automatic | Metrics or HPA | | Azure Functions | Automatic | Event-driven, instantaneous |

Cost Optimization Guide: | Scenario | Recommended Service | Cost Benefit |

Web Application	App Service	No VM management costs	Batch Processing	Azure Functions	Pay only during execution	Spiky Workloads
Development/Test	Dev/Test Lab	Reduced pricing				
VM Scale Sets	Scale down during low usage					

Azure Networking Fundamentals (60 mins) Virtual Network (VNet) - The Backbone: - Definition: Isolated network infrastructure in Azure - Key Capabilities: - Connect Azure resources securely - Filter network traffic - Route traffic between resources - Connect to on-premises networks

Core Networking Components:

1. Subnets: - Purpose: Segment VNet into smaller networks - Best Practices: - Plan CIDR blocks carefully - Create separate subnets for different tiers - Reserve IP space for future growth

Example Subnet Design:

VNet: 10.1.0.0/16

Web Subnet:	10.1.1.0/24	(App Services, VMs)
App Subnet:	10.1.2.0/24	(Application logic)
Data Subnet:	10.1.3.0/24	(Databases, Storage)
Gateway Subnet:	10.1.254.0/24	(VPN/ExpressRoute)

2. Network Security Groups (NSG): - What: Firewall for Azure resources - Rules: Allow/deny inbound and outbound traffic - Application: Can be applied to subnets or individual network interfaces

3. Route Tables: - Purpose: Control how traffic is routed between subnets - Use Cases: - Force tunneling through firewalls - Custom routing paths - Network virtual appliances

Network Connectivity Options:

1. VNet Peering: - What: Connect VNets seamlessly - Types: - Regional Peering: Within same region - Global Peering: Across different regions - Benefits: Low latency, high bandwidth

2. VPN Gateway: - Site-to-Site: Connect entire networks (office to Azure) - Point-to-Site: Connect individual devices to Azure - Use Cases: Hybrid connectivity, remote access

3. ExpressRoute: - What: Private, dedicated connection to Azure - Benefits: - Higher reliability - Faster speeds - Lower latency - Enhanced security

Load Balancing and Traffic Management (30 mins) Azure Load Balancing Services:

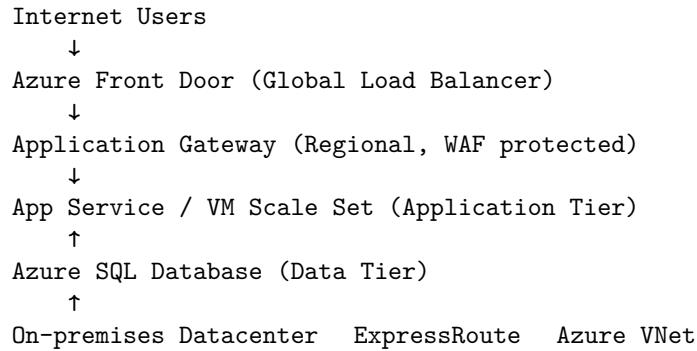
1. Azure Load Balancer: - Layer: 4 (TCP/UDP) - Features: - Port forwarding - Outbound rules - Health probes - Use Cases: Non-HTTP traffic, basic load

distribution

2. Application Gateway: - Layer: 7 (HTTP/HTTPS) - Features: - SSL termination
- Web Application Firewall (WAF) - URL-based routing - Cookie-based session affinity - Use Cases: Web applications, API management

3. Azure Front Door: - Scope: Global HTTP load balancer - Features: - Anycast protocol - TLS termination - DDoS protection - URL redirection - Use Cases: Global applications, CDN integration

Complete Network Architecture Example:



Session Wrap-up & Hands-on Practice

Key Takeaways: 1. Start with PaaS when possible for reduced management overhead
2. Use availability zones for production workloads requiring high availability
3. Plan networking before deploying resources
4. Implement security at multiple layers (NSG, WAF, etc.)

Recommended Next Steps: - Create a free Azure account and explore the portal
- Deploy a simple web app using App Service - Experiment with creating virtual networks and subnets - Review AZ-104 certification path for administrator roles

Practice Exercise: Design a three-tier application architecture including:
- Web tier with auto-scaling - Application tier with load balancing
- Database tier with high availability - Secure network design with proper segmentation

This 3-session plan provides comprehensive coverage of Azure core services, preparing learners for real-world implementations and further Azure certification paths.