1/27/2021  
**Overview – Microsoft Azure: The Big Picture**

Broad course with little depth. Briefly touched key features provided by Azure.

IaaS: Infrastructure as a service – provided infrastructure needed to create virtual environments. Rather than using on prem servers to store data, might be using virtual machine in MS datacenter to store data.

PaaS: Abstract away infrastructure needed to run/build applications. Managing platform level components (just managing a SQL server instance) rather than hardware.

SaaS: Only using a service, no concern for platform, development, and certainly not infrastructure.

Hybrid solutions connect on prem resources to cloud resources.

**Compute**

VMs: Can be created with variety of parameters. Linux or Windows, drive type, CPU speed, etc.. typical VM optionality. VM images.

Example: Migrating on prem VMs to Azure VMS. Azure Migration tool will analyze VMs and tell you what needs to be done to migrate them to Azure VMs

Scale sets allow you to scale number of instances of a VM as needs increase.

Managed disks: Allows you to manage VM storage in concise manner.

App Services: Allow you to run web apps in a hosted environment (PaaS).

Containers: Run applications in containers. Supports Kubernetes (AKS). Azure Container Instances lets you run containers without setting up infrastructure. Azure Container Registry lets you catalog Containers for storage.

Serverless compute: run code without regard for IaaS. Azure functions allow you write code and run it on demand. I don’t believe you have to worry about where the code is being executed (infrastructure). Logic Apps, Event Grid.

Batch – Execute code at scale. Multiple instances, running things in parallel.

**Data storage and processing**

Cloud dbs need to be scalable, available, and global.

Can choose to self manage dbs – run the db on a cloud VM and manage it essentially the same way we do now. Management of computation and storage.

Can choose service based: Provision instances, choose scale/load, managed by Azure. Patching is done automatically.

Types of databases: Azure SQL, My SQL, Maria DB, PostgreSQL (all relational). Table storage, blob storage (for large objects), Queues, Redis Cache. MongoDB, Cassandra, Neo4j are all wrapped into Azure CosmosDB -- multi-model storage database, allowing you to use multiple database types in a single database. Azure files – store files.

Azure Data Lakes – Large scale data storage for analytics. File/Blob storage.

Example: A web app might require multiple types of data storage. All types of storage can be hosted in Azure and access through the web app.

Data processing: Data factory can pull in data from various resources and push it into Azure based databases.

Analyzing data: SQL Data Warehouse, Analysis Services. Stream analytics allows you to process data as it is fed into applications (real time analysis).

**Integration**

Connecting systems and applications.

Service Bus: Brokered and relayed messages. Event grid focuses on notifying user of events in Azure. API Management securing and managing applications.

Integration accounts allow you to export Logic Apps output to enterprise type formats (Csv, xml…)

Example: Service bus Relay – Two servers in different networks. Can communicate through Service Bus Relay hosted in Azure.

**Networking**

Virtual networks, Public IP addresses, Network security groups, Service endpoint policies.

CDN – Content delivery network. Data cached near customers. Traffic manager -- manages traffic from clients. Load balancers to handle traffic as well. DNS zones to handle name resolution within a network.

For security, DDOS protection, Application Gateway, Front Door service to create entry points into microservices. A lot of this seems to overlap.

Example: Hybrid network – Web app in Azure and on Prem database. These can be connected through Express Route to setup VPN connection between the resources. Generally, these resources are used to safely connect Azure resources.

**Management**

Portal, CLI/PowerShell, Cloud Shell, Mobile App (??).

Backup utilities, Automation and scheduling, Policy. RBAC

Azure Resource Manager allows you to define resources, resource groups, services, etc.. in Json and deploy it.

Azure monitor lets you monitor statuses of various Azure services.

Example: Deploying and monitoring an application. Deploy Json template which creates resource group need for application to work (maybe an REST api, a network, a database…). Then use Azure Monitor to *monitor* the resource group. Maybe use Scheduler to set when this Resource group should be kicked off.

**Developing for Azure**Azure provides numerous SDKs to develop applications in different languages. Many are cross platform. Resources for running containers, Kubernetes… Already mentioned above. CI/CD options.

Azure DevOps: Azure Boards, Azure Repos, Azure Pipelines… This is already in use.

**Identify**

Azure Active Directory: Provides core directory services – manage users and multi-factor authentication.

Azure AD Domain Services: Manages domain related services. Such as setting up VMs under a domain. Adding and removing resources from domain.

Azure AD B2C: Business to consumer – identify store for clients.

Data protection: Azure Security Center provides information on security details of services.

Example: web app calls API, header needs a specific key to access API. Web app can retrieve key from Key Vault, and then pass that to the API.

Example: Authenticating users in production application – Store and retrieve user credentials using Azure AD B2C.