Cloud Computing Services

Compute Power
Storage
Networking
Analytics

→ VM's Containers Serverless computing

Containers
- No guest operating system
- provide consistent, isolated execution env. for apps.
- application and all dependencies packaged into a container and run on standard runtime env. to execute
- Startup in few seconds as no OS to boot just the app.

Advantage: **Portability:**
Different components of application deployed independently into different containers.
Multiple containes run on same single machine
Containes moved between machines.
Portability between on premise and cloud with no change in app. itself.

Serverless Computing
- Run app. code without creating, configuring or maintaining a server.

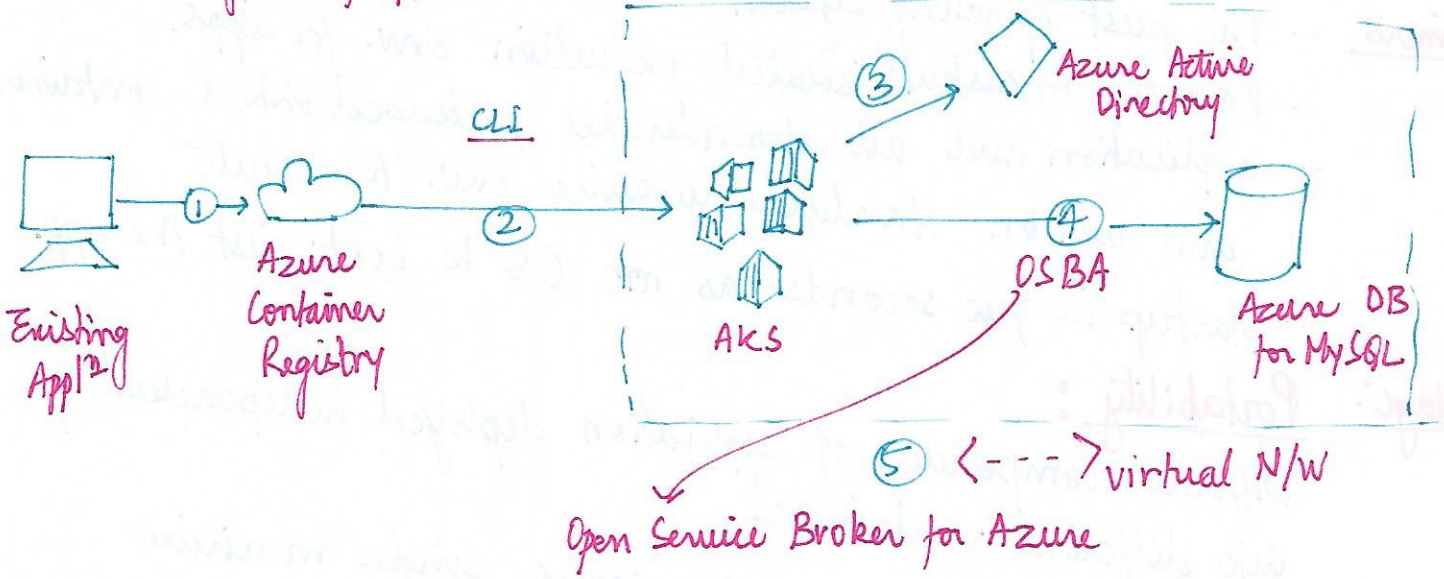
Core idea → Application is broken into seperate functions that run when triggered by some action.
- ideal for automated tasks

Eg: Send an email automatically when a purchase order completes.
- Pay only for processing time used by each function as it executes

CONTAINERS: OS level virtualization

1) Azure Container Instances (ACI): PaaS offering allowing to upload containers and execute directly with automatic elastic scale.
2) Azure Kubernetes Service (AKS): Orchestration service for containers with distributed architectures with multiple containers.

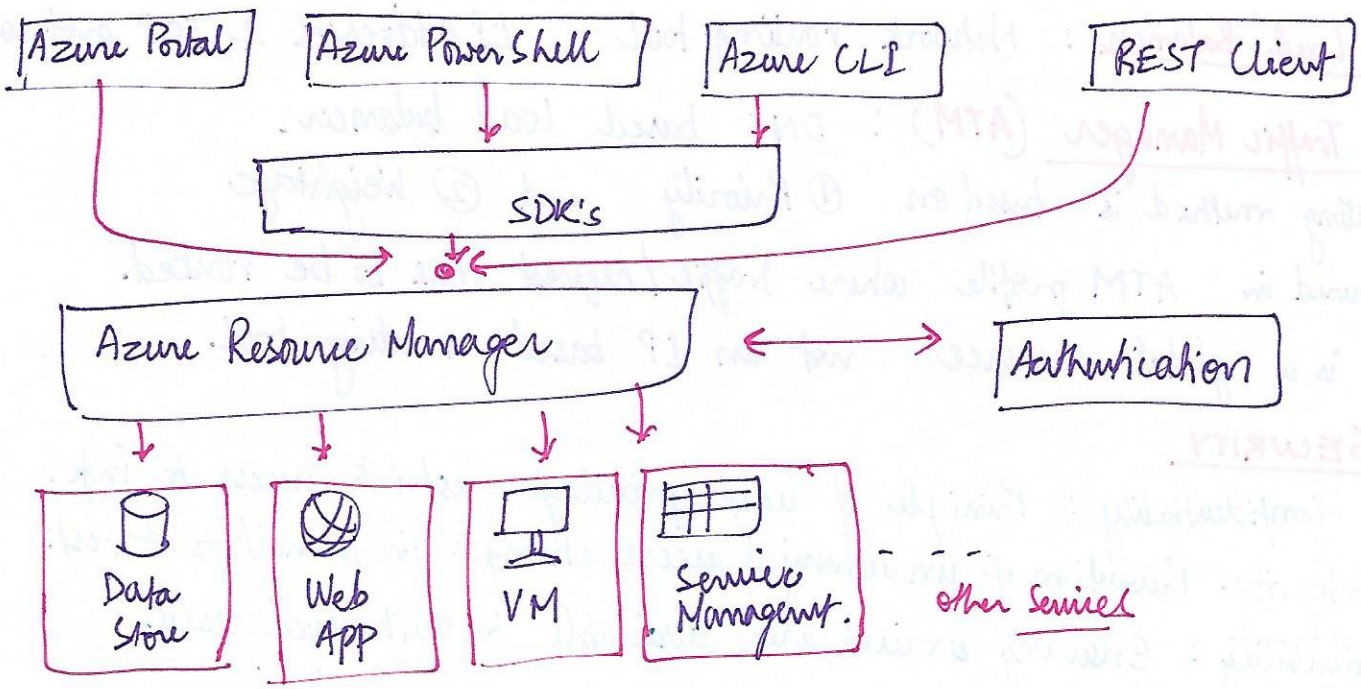
Migration of apps to containers via Microservices:



Azure App Service : PaaS offering to focus on website and API logic while Azure handles the infrastructure to run & scale the web applications.

Types:
1) Web Apps
2) API Apps
3) Web ~~Apps~~ Jobs
4) Mobile Apps

Azure Serverless Compute:
1) Azure Functions
2) Azure Logic Apps



Log analytics Workspace – storage account used to collect log & metric data from various Azure Resources.

Compliance Manager – track compliance with various international standards and gov. laws.

Azure Load Balancer : Network routing tool — IP Addresses & TCP protocol.

Azure Traffic Manager (ATM) : DNS based load balancer

Routing method is based on ① Priority and ② Weightage

→ configured in ATM profile where traffic / request has to be routed.

ATM is a global resource not an IP based routing tool.

SECURITY

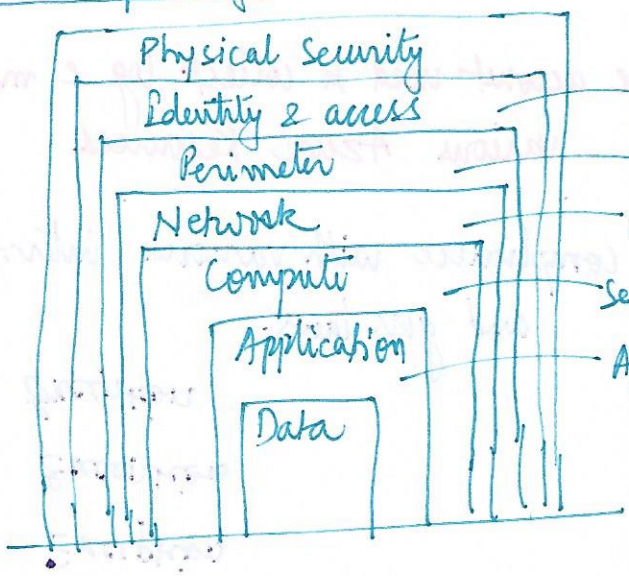
CIA

Confidentiality : Principle of least privilege restricts access to info.

Integrity : Prevention of unauthorized access changes in transit or at rest.

Availability : Ensuring services are available to authorized users.

Defense in depth layers



Infra & change control

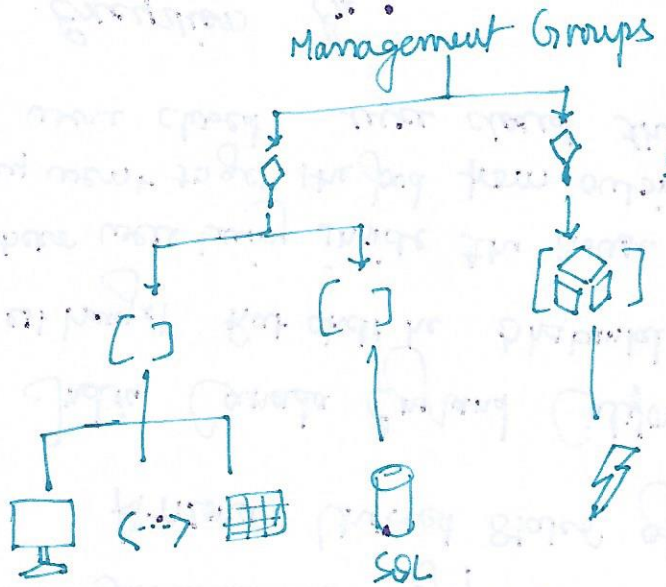
DDoS attack

Limit communication between resources.

Secure access to VM

App free of vulnerability & secure.

Organization of Resources in Azure.



Azure Resource Manager (ARM) model — common API

Azure Data Security Solutions

- Firewall that require whitelist IP
- Virtual Network Service Endpoints
- Non-public networks
- Encryption at rest
- Encryption over the wire
- Access keys and certificates stored securely (Key Vault)
- Managed Service Identity (via AAD)

TDE Transparent Data Encryption

- Data is stored to disk encrypted
- Azure manages the keys and does encryption/decryption without us knowing about this so transparent.

* Master Database is not encrypted (the one which stores keys for above encryption)

- On by default but can be switched off.

TDE supported operations:

- Geo restore
- Geo Replication
- Restoring a deleted database
- Copy a database
- self service point-in-time restore

* If you export a database, data is not encrypted.
  Stored in unencrypted BACPAC files.

Azure Storage Encryption:

↳ service for storage accounts

- Enabled by default for new storage
- cannot be disabled
- Supports BYOK Bring your own key. (not for managed disks)
- Azure DBs are sometimes stored in storage accounts.

Azure Key Vault

* One stop shop to store all secret and confidential info in encrypted format.
- Adheres to FIPS 140-2 Level 2 standard.
  Federal Information Processing standard.
- Centralized secrets, more secure, simplify management.

Storing ARM template (JSON) file — How?

Template:
```
"adminPassword" {
    "type": "secureString" }
```

(also uses similarly) Parameter file JSON
```
"adminPassword" {
    "reference": {
        "keyVault": {
            "id": "/subs/<RG>.../<KV>/(value)
        }
        "SecretName": "Examplesecret"}
}
```

* Supports two types of objects : ① Keys or ② Secrets
  - Keys → cryptographic keys
  - Secrets → Passwords, API keys (can hold text upto 25 kb)

- can consider encrypting secret as this is stored in plain in KV.
- Supports versioning so same secret may have different versions and key Valut URL can be generated for each of them.

* MS doesn't restricts you from reading secrets after it has expired or before it has been released. (info for application)
* App's can list the secrets they have access to using list command.

Certificates
- ↳ metadata
- ↳ key
- ↳ secret

Here KV can interact directly with some certificate authorities ↔ and generate/renew keys. Automatically or bring yr own CA. ⚠ manual work

COSMOS DB

- Read API vary like using Mongo etc.
- Storage and no. of RU Read Units charged.
- Global redundant storage with automatic backup copy made.
→ Primary write preferable to keep one location to avoid complexity.
- Add collections to DB later and use explorer to see / retrieve data.

Throughput → range is 400 - 1 million RU's

Interaction with Cosmos DB

in .Net → using Nuget package manager use Microsoft document DB and include Microsoft.Azure.Documents.Client.
↳ whole package.

→ Endpoint and key is in seperate lines.
→ Create app to get/retrieve data directly

Very similar to building query in SQL.

Default Consistency

- ① Session consistency is default and most widely used.
Each one will see same data but delay between each one of them. Best when bound to client.

② Strong & ③ Bounded Staleness (define time limit for consistency)

④ Consistent Prefix: Every one is guaranteed order of data but no guarantee who will get next update.

- ⑤ Eventual: Weakest one, no guarantee of data which order it would come. Data like retweets, likes etc.

⊗ RU's Request Units are affected by your query and stored data type.

Suppose cards < Plain / Ruled

Cross partition query.

Select * cards runs 2 queries one for plain partition and other one for Ruled partition and combined together to view.

But then it consumes more RU's thereby increasing / affecting costs for Cosmos DB.

(*) You can disable cross-partition query in Cosmos DB

Azure SQL Database

- SQL Database as a Service
- Best solution for a lot of situations
- advanced capabilities like collation Full Text indexing etc.

Azure SQL Managed Instance

MS takes care of everything like scaling, optimization, migration etc.

→ other options: MySQL, PostgreSQL, MariaDB

You can select your existing SQL DB backup in BACPAC file as source while setting up.

- Select/create a server for SQL a new Server.
- It has to be a fully qualified name. Public accessible DB. URL.
- white list people who are going to use.
- SQL DB service uses a firewall
  ↳ optional Advanced Threat Protection.

(*) SQL Elastic Pool - can be used when you have multiple databases and can share resources.

Pricing ① Basic ② Standard ③ Premium ← Pricing Model by Performance.

Purchasing by vCores. You get dedicated cores of CPU's

- After Geo replication you can initiate a Failover and make your backup secondary DB as Primary.
  ↳ usually done for reading performance.

(*) from on prem to cloud 90% of time you will need just connection string update. You can also use SQL Server Studio

Azure Service Fabric App:

AZURE

- Microservices take on Azure. Deploy all small microservices on all servers.
-
-
-

az sf cluster create --resource-group < > --location < > --certificate-output-folder . --certificate-password < > --certificate-subject-name < >

* Domain name has to match the domain name of cluster when creating.

- Download the pfx certificate
  - ↳ By default there is no password. Just click Ok.
- Just like kubernetes Service Fabric has a manager for services:

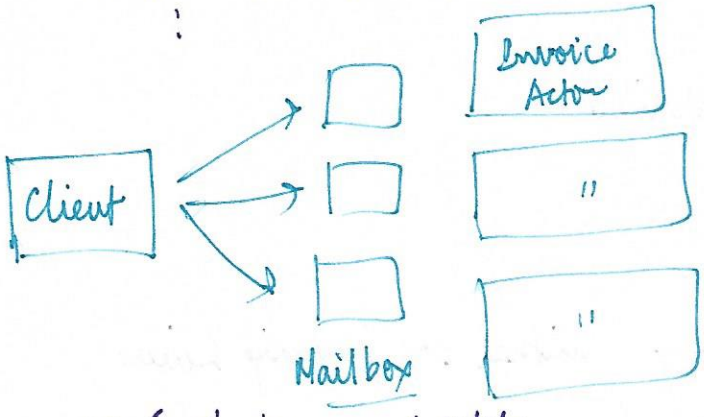
sfctl cluster select --endpoint < > --pem <>.pem --no-verify.

Secure cluster so we need a certificate to access it.

3 ways it can be developed:

1) Guest Executables → Existing code straight forward. (No tie to Service Fabric)
2) Reliable Services → Light framework to build Service Fabric Apps
   - call API's that allow SF runtime to manage
   - LOCK-IN the service.
   - SOA Service oriented Architecture.
   - State less like web server.
3) Reliable Actors
   - Communicate through messages
   - isolated from each other
   - Models ↔ Actor
   - MVC "controller"

Eg. User - name, email ...
Order - state, date, destination
:



- Good for scalability

Reliable Service API

- Query service fabric itself
- Report health
- Receive alerts & notification about system changes
- Use Reliable collections

Why Reliable Services:

Reliable
Available
Scalable
Consistent

Containers AKS Azure kubernetes Services

▷ Azure Arc - enable deployment of Azure services anywhere and extend Azure management to any infrastructure.

So you can import an existing kubernetes cluster into Azure.

Arc covers all → Multi cloud
→ On Premise
→ Edge

→ az aks get-credentials --resource-group <azvj> --name <cluster>
↳ Merged

→ kubectl get nodes → yaml file comes to current directory.

→ kubectl apply -f azure-vote.yaml

Kubectl get nodes → no effect. but get service will show

Kubectl get service → gives azure-vote-front → cluster
azure-vote-back → load balancer
kubernetes → cluster

Kubernetes dashboard
↳ Runs only on your local not on Azure Cloud shell.
Install CLI and authenticate to use it.

Run : ↳ az aks install-cli
az login
az aks get-credentials --resource-group <> --name < >
az aks browse --resource-group < > --name < >
↳ This fires up a web server and we see dashboard.
(local) port 8001.

Docker

After cloning the project / container

docker-compose up -d
docker container ls
docker ps
docker-machine ip default — virtual box having Linux

Message Base Integration Architecture

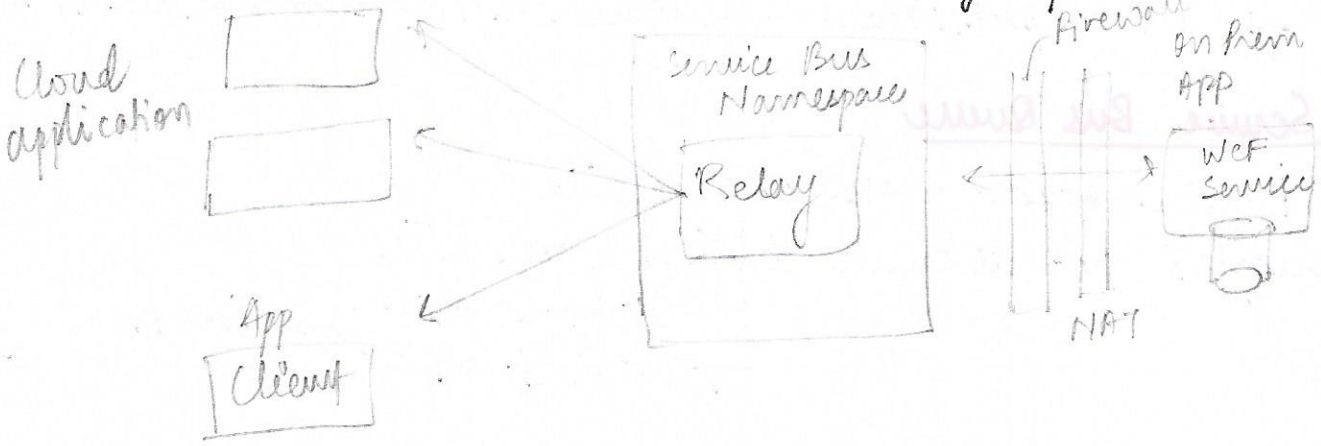
⊛ **Send Grid** : Email delivery service

↓ REST API which apps can use to send an receive email msgs.

- Scalable email service, 3rd party
- process incoming emails too
- needs send grid account.

Relay Service : allows msgs to be passed from Internet inside the company firewall to specific endpoints such as WCF

- No firewall changes required.
- Application inside network opens up outbound connection to Azure and the connection stays open.



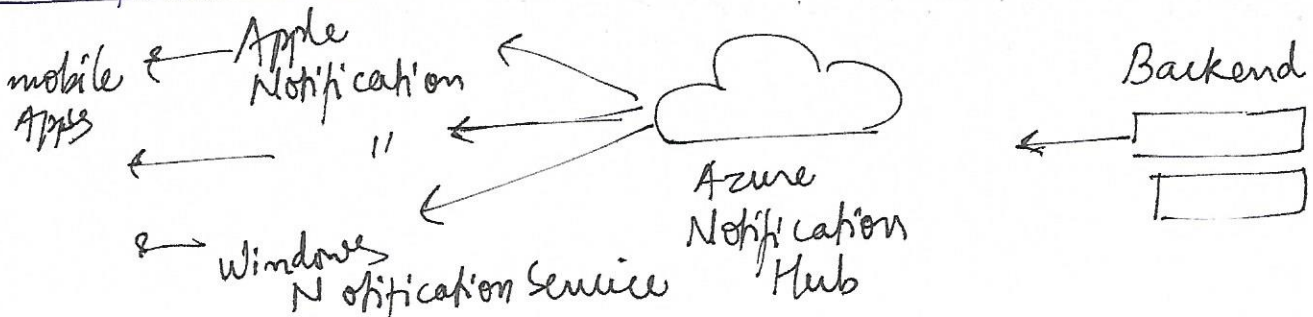
Hybrid connections – more sophisticated.

- Based on BizTalk Services
- uses standard Web Sockets and multi-platform support.

WCF Relay – It is for Windows Communication Framework (WCF) Services

- Legacy offering
- Only supports HTTP communication limited platform support

Notification Hub



Notification Hub is quite flexible.

- can choose to send push notification all at once or only to specific devices depending upon config.

Event Hub (IoT devices can use this)

Accepts all events like a service bus load on left does not affect apps on right.

- can be configured to get event messages in order.
- huge capacity kafka based event ingestion service.
  - ↳ can receive msgs from millions of producers.
- supports HTTPS and AMQP 1.0 protocol
- msgs can be partitioned – delivered in the same order by partition

Service Bus Queue

- Enterprise-grade messaging service
- supports two different delivery guarantees
  - ↳ at most once
  - ↳ at least once
  - > delivery.

Microsoft Graph API

- designed to interact with Office 365
  - ↳ outlook for email
- can be used to create and send messages.

Summary

usage: Developing highly-available and responsive apps

Disconnect between two system to relieve off load.

Scaling is expensive so use if possible / required.

AZURE DATA STORAGE

Semi Structured Eg: Product Catalog

Operations: High throughput high latency
Transaction support required

use: Azure Cosmos DB
↳ NOSQL data support by design
* ↳ Every property is indexed by default
↳ Supports SQL query
* ↳ ACID compliant
↳ Replication

UnStructured Data Eg: Photos and Videos

Operations: Retrival by ID, low latency and high throughput.
No transaction support

use: Azure Blob storage
↳ works with Azure CDN (Content Delivery Network)
↳ caching most frequently used content on edge servers.

uploading to Azure App Service can be used but for smaller data set.

Structured Data eg: Business Data

Operations: Read only, Complex analytical queries across several DBs
Some latency expected for complex queries
Transaction support required.

use: Azure SQL Database
↳ SQL is known query lang.
↳ pair with Azure Analysis service and BI tools

Azure SQL Data Warehouse supports OLAP and SQL query but cross-database query cannot be performed.

Azure SQL Database — DaaS or PaaS

Relational Database as a Service based on MS SQL Server

- use Azure Database Migration Service
- uses Microsoft Data Migration Assistant

Azure Data Lake Storage

- Data Lake is large repository to store both structured and unstructured data.
- Data Lake allows to perform analytics on data usage and prepare reports.
- Object storage with reliability and performance of Big Data File System
  ↳ Batch queries, interactive queries, Real time analytics machine learning, Data Warehouse

Azure Files

Managed File share in cloud using standard Server Message Block (SMB) protocol.

- usage: share files anywhere in the world, diagnostic data, app. data sharing.

Azure Queue Storage

- Service for storing large number of messages that can be accessed anywhere in the world.
- Build flexible applications using seperate functions for better durability across workloads.
- Asynchronous message queuing for communication between apps.

Disk Storage

Storage tiers:
- hot storage tier
- cool storage tier
- Archive storage tier

Encryption:
- Azure Storage Service Encryption (SSE)
- Client side encryption.