

Benefits of using Cloud Services:

Availability, scalability, reliability, predictability, security and governance, and manageability.

### **High Availability (HA)**

Ability of a system to remain operations to users during planned or unplanned outages. It is important to remember that it is near impossible to have 100% availability.

Planned outages include operating system security patches, application updates, hardware replacement, migrating to a new hosting provider.

Unplanned outages include hardware failure, network disruptions, power outages, natural disasters, cyber attacks, software bugs, poor scaling/architecture design.

Methods to mitigate planned outages:

1. Gradual deployment strategy (do not deploy to all the servers at once). Start with handful servers and gradually grow deployment.
2. Testing and monitoring deployment (errors (files missing, corrupted) are increasing over the time).
3. Easy rollback plan (have a plan to go back if there are major issues). MS Azure has tools to make rolling back easy.
4. Small deployments (less features and less changes in every deployment)
5. Frequent deployments (become expert at deployments)
6. Automation (instead of manual deployments, follow standards)

Methods to mitigate unplanned outages:

1. Every single core component has redundancy (we will not have a single server, single web app, single network, single region dependency for application). This may increase the cost, but it will help to mitigate the unplanned outages.
2. Use Azure's built-in features for availability: Availability Sets, Availability Zones, Cross-Region Load Balancing/Front Door.

3. Constant health monitoring/probes (real time monitoring systems).
4. Automation (automate fall back strategies).
5. Strong security practices (for cyber attacks, ...).
6. Be geographically distributed.
7. Have a disaster recovery plan (similar to fire evacuation plan).
8. Test that disaster recovery plan (similar to fire drill).
9. Load testing (this for scaling). Identify where are the capacity limits. This helps to identify bottleneck and fix them.

## **Scalability**

The ability of a system to accommodate increasing demand by adding or removing resources as needed.

Note: Some system cannot be easily scaled.

Scalability allows a system to adapt to changing usage patterns and handle increased traffic without requiring changes to the application code or system design (system is automatically designed to handle increasing/decreasing resources requirements)

Not every system has traffic that fluctuates based on day or day of the year. Examples: E-commerce websites have Black Friday, School registrations are busy during the enrolment times, tax systems are busy in April

Questions:

Can you expand the capacity of a system very easily by adding more servers? Or will it be a massive undertaking to do that?

Even after adding 10/100 servers and still cannot handle users, then that system is not scalable.

Types of scaling:

Vertical scaling (scaling up/scaling down): This is we have a single server, and we are adding more resources (more memory, number of CPUs) to the server. The issue is there is an upper limit to this (Azure – 96 vCPUs, 384 memory). Also, single server does not improve availability.

Horizontal scaling (scaling out/scaling in): Adding more servers to a system. Here no limits to scaling. However, this adds complexities for load balancing. Note that this will improve availability.

When discuss scalability, it has impact on system cost. Adding more resources to a system adds to cost (in Azure cost will double when number of CPU doubled).

Reducing resources can reduce cost.

Thus, having a scalable system allows for a system to be perfectly sized. In the cloud we can easily add/remove resources and this optimizes the cost by reducing wasted computing resources.