



# DP-201T05: Designing for Scale and Resiliency



# Lesson Objectives

- Scaling Systems
- Optimizing Network Performance
- Optimizing Storage Performance
- Identifying Performance Bottlenecks
- High Availability
- Disaster Recovery
- Backup and Restore

# Scaling Systems

## Scaling up or down

Scaling up or down is the process where we increase or decrease the capacity of a given instance. It adjusts the amount of resources a single instance has available. You can use autoscaling so that this can be managed automatically

## Scaling in or out

Scaling out is the process of adding more instances to support the load of your solution. Scaling in is the process of removing instances that are no longer needed.

## Serverless Computing.

### Containers

A container is a method running applications in a virtualized environment. The virtualization is done at the OS level, making it possible to run multiple identical application instances within the same OS.

### Azure Kubernetes Service (AKS)

Azure Kubernetes Service allows you to set up virtual machines to act as your nodes. Azure hosts the Kubernetes management plane and only bills for the running worker nodes that host your containers.

### Azure Container Instance (ACI)

Azure Container Instances is a serverless approach that lets you create and execute containers on demand. You're charged only for the execution time per second.

# Optimizing Network Performance

Network performance can have a dramatic impact on a user's experience. In complex architectures with many different services, minimizing the latency at each hop can have a huge impact on the overall performance.

## The importance of network latency

Latency is a measure of delay. Network latency is the time needed to get from a source to a destination across some network infrastructure. A distributed approach can have an impact on the round-trip time of your network communications.

## Latency between Azure resources

The goal here is to minimize the network latency between each layer of the application. How this is solved depends on your application and data architecture, but Azure provides mechanisms to solve this on several services.

## SSO with Azure Active Directory

Latency between users and Azure resources can be improved by:

- Use a DNS load balancer for endpoint path optimization
- Use Azure Analysis Services as a Caching Service for Azure Data Warehouse
- Use CDN to cache content close to users
- Use ExpressRoute for connectivity from on-premises to Azure

# Optimizing Storage Performance

Virtual Machine  
Storage  
Performance

Storage  
Performance  
For  
Applications

Caching

Polyglot  
Persistence

# Performance Bottlenecks

## Importance of requirements

In theory, we could keep improving scalability and performance further and further without end. At some point, however, more improvement is prohibitively expensive, difficult, and doesn't have enough business impact.

## DevOps & application performance

The idea behind DevOps is that we don't have development and infrastructure silos. Instead, they work together to effectively build, deploy, monitor, and maintain apps in streamlined process.

## Performance monitoring options in Azure.

### Azure Monitor

Azure Monitor provides a single management point for infrastructure-level logs and monitoring for most of your Azure services.

### Log Analytics

With Log Analytics you can query and aggregate data across logs. This cross-source correlation can help you identify issues or performance problems that may not be evident when looking at logs or metrics individually.

### Application performance management

Telemetry can include individual page request times, exceptions within your application, and even custom metrics to track business logic. This telemetry can provide a wealth of insight into apps.

# High Availability

A highly-available service absorbs fluctuations in availability, load, and temporary failures in dependent services and hardware. The application remains online and available.

Determine the service-level agreement of your application

A service-level agreement (SLA) is an agreement between a service provider and a service consumer in which the service provider commits to a standard of service based on measurable metrics and defined responsibilities.

Evaluate the HA capabilities of the application

To evaluate the HA capabilities of your application, perform a failure analysis. Focus on single points of failure and critical components that would have a large impact on the application if they were unreachable, misconfigured, or started behaving unexpectedly.

Evaluate the HA capabilities of dependent applications

You'll need to understand not only your application's SLA requirements to your consumer, but also the provided SLAs of any resource that your application may depend on.

# Disaster Recovery

Disaster recovery is about **recovering from high-impact events** that result in downtime and data loss.

## Create a disaster recovery plan

A disaster recovery plan is a single document that details the procedures that are required to recover from data loss and downtime caused by a disaster and includes:

- Risk assessment and process inventory
- Recovery objectives
- Detailed recovery steps

## Designing for Disaster recovery

Disaster recovery is not an automatic feature. It must be designed, built, and tested, and should include the following:

- Data recovery and replication
- Process recovery
  - Azure Site Recovery
  - Service-specific features

## Testing disaster recovery

Disaster recovery planning doesn't end once you have a completed plan in hand. Testing the plan is a crucial aspect of disaster recovery, to ensure that the directions and explanations are clear and up-to-date

# Backup and Restore

- Establish backup and restoration requirements
- Azure backup and restore capabilities
  - Azure Backup
  - Azure Blob storage
  - Azure SQL Database
  - Azure App Service
- Verify backups and test restore procedures



# Lab: Designing for Scale and Resiliency



# Lab overview

The students will explore a range of resiliency and scale issues that would have to be considered when defining a solution architecture for an organization. They will first look at how they will incorporate scale into a solution. They will follow this by looking into storage and database performance, and how solutions can be made highly available. Finally, the student will investigate the issue of disaster recovery.

## Lab objectives

After completing this lab, you will be able to:

1. Adjust Workload Capacity by Scaling
2. Design for Optimized Storage and Database Performance
3. Design a Highly Available Solution
4. Incorporate Disaster Recovery into Architectures

# Lab scenario

You have recently been hired as a senior data engineer at AdventureWorks and are working with a consultant and architects to design a resilience and scale approach for a cloud data platform solution that meets the organizations technical and business requirements.

Based on the case study about AdventureWorks, you have been asked by the board to produce a short report that address their concerns regarding the resilience and scaling of the services to meet with the websites ability to process sales and the customers demand for service. The report should confirm the concerns that has been raised, and your report should propose how the issues are to be dealt with.

You should also address the concerns of how the organization should maintain operations in the event of intermittent failures. It should also address a disaster recovery approach to ensure the data is protected.

At the end of this lab, you will have:

1. Adjust Workload Capacity by Scaling
2. Design for Optimized Storage and Database Performance
3. Design a Highly Available Solution
4. Incorporate Disaster Recovery into Architectures

# Lab review

- Exercise 1 – Have any availability or disaster recovery requirements been missed?
- Exercise 2 – Was there any debate in the availability or disaster recovery options?
- Exercise 3 – Are there technologies that could not provide any availability or disaster recovery?
- Exercise 4 – Which availability or disaster recovery options are of importance to your own organization?

# Module Summary >

## In this module, you have learned about:

- Scaling Systems
- Optimizing Network Performance
- Optimizing Storage Performance
- Identifying Performance Bottlenecks
- High Availability
- Disaster Recovery
- Backup and Restore

## Next steps >

After the course, you can use this holistic [resilience](#) and [scalability checklist](#) to review your architecture from a scalability standpoint

