

DP-050-Migrating-SQL-Workloads-to-Azure

During this course, the students will first explore the preparation steps that are required to help with a successful data modernization project, and the tools that can help support them in the various stages of such a project. They will then explore the different methods of migrating SQL workloads to a range of different SQL services that run in Azure.

The following is a summary of the lab objectives for each module:

Lab 1 - Azure Architecture Considerations

The students will use the information gained in this module to demonstrate an understanding of data platform modernization, and why and how organizations should undertake a modernization project at a high level. They will learn how to determine the cost of moving to Azure. They will also gather information about the environment that they are working on. Students will finally determine a data platform modernization strategy to use for a given scenario.

Lab 2 - Choosing the right tools for data migrations

The students will use the Data Migration Assistant in the prescribed data platform modernization stages to investigate their environment in an automated way. They will also identify any premigration compatibility issues and define a plan on how to address those issues before performing a migration of an on-premises server. Finally, they will assess how a workload will perform on a target version of Azure SQL Database.

Lab 3 – Migrate SQL Workloads to SQL Server in Azure Virtual Machine

The students will initially evaluate the migration process that they will use to migrate from an on-premises SQL Server 2008 R2 instance to an SQL Server 2017 running in a virtual machine. They will then perform a migration using Data Migration Assistant to move databases. Finally, they will assess a successful migration.

Lab 4 – Migrate SQL Workloads to Azure SQL Database

The students will perform a migration to Azure SQL Database. First, you will perform a schema migration and create the destination database as a pre-requisite, then you migrate from a database running in an on-premises SQL instance to Azure SQL Database. You will perform an online migration using Database Migration Service (DMS), to keep the data in synchronization between the source and the target databases until cut over to the new database.

DP 050 - Migrating SQL Workloads to Azure

Lab 1 - Azure Architecture Considerations

Estimated Time: 50 minutes

Pre-requisites: There are no pre-requisite steps to perform this lab.

Lab files: The files for this lab are in the *Allfiles\Labfiles\Starter\DP-150.1* folder.

Lab overview

The students will use the information gained in this module to demonstrate an understanding of data platform modernization, and why and how organizations should undertake a modernization project at a high level. They will learn how to determine the cost of moving to Azure. They will also gather information about the environment that they are working on. Students will finally determine a data platform modernization strategy to use for a given scenario.

Lab objectives

After completing this lab, you will be able to:

- Describe data platform modernization.
- List the stages of migration.
- Describe data migration technologies.

Scenario

You have recently been hired as a senior data engineer at AdventureWorks and are working with a consultant and architects to initiate a data platform modernization project that meets the organization's technical and business requirements. You will calculate the Total Cost of Ownership (TCO) of migrating to Azure. You will ask a junior data engineer to collect information about the SQL Server that will be migrated to Azure. Then you will choose the best Azure data platform technology to migrate to for a given scenario.

Exercise 1: Understand data platform modernization

In this lab, you will calculate the costs for a migrating databases to Azure.

Estimated Time: 15 Minutes

The main tasks for this exercise are the following:

1. Open the Microsoft Total Cost of Ownership (TCO) Calculator
2. Define a server-based workload
3. Add storage
4. Add networking
5. Adjust the currency
6. Adjust the electricity costs
7. Adjust other assumptions
8. Compare the on-premises and Azure costs

Open the Total Cost of Ownership (TCO) Calculator

1. Open your browser and navigate to the Azure TCO Calculator, [+++https://azure.microsoft.com/pricing/tco/calculator/+++](https://azure.microsoft.com/pricing/tco/calculator/).
2. Under **Define your workloads**, delete any existing workloads in the **Servers** section.

Enter the Database workload

1. Under **Databases**, delete any existing workloads.
2. Select **+ Add database**.
3. In the **Name** textbox, type **Accounting**.
4. In the **Source** section, choose these values:

Property	Value
Database	Microsoft SQL Server
License	Enterprise
Environment	Physical Servers
Operating system	Windows

Property	Value
Operating System License	Datacenter
Servers	1
Procs per server	1
Core(s) per proc	4
RAM (GB)	64
Optimize by	CPU
SQL Server 2008/2008R2	Slide to the right

5. In the **Destination** section, choose these values:

Property	Value
Service	SQL Server VM
Disk type	SSD
IOPS	5000
SQL Server storage	32 GB
SQL Server Backup	32 GB

[!NOTE] SSDs are the recommended disk type for production servers in Azure

Enter the Storage and Networking workloads

1. Under **Storage**, delete any existing workloads, and then select **+ Add storage**.
2. Enter these values:

Property	Value
Name	Accounting Local Disks
Storage Type	Local Disk/SAN
Disk Type	HDD
Capacity	3 TB
Backup	1 TB
Archive	0 TB

3. Under **Networking**, in the **Outbound bandwidth** controls, select **1 GB**.

4. At the bottom of the page, select **Next**.

Adjust assumptions

1. In the **Adjust assumptions** section, in the **Currency** list, select your preferred currency.
2. Under **Software Assurance coverage (provides Azure Hybrid Benefit)**, slide the **Windows Server Software Assurance coverage** slider to the right.
3. Slide the **SQL Server Software Assurance coverage** slider to the right.

[!NOTE] You can use the links in the **Software Assurance** section to learn more about the assurance that is available. There are similar links in subsequent sections.

4. Under **Geo-redundant storage (GRS)**, ensure that the toggle beside **GRS replicates your data to a secondary region that is hundreds of miles away from the primary region** is turned off.
5. Under **Virtual Machine costs**, ensure that the toggle beside **Enable this for the Calculator to not recommend B-series virtual machines** is turned off.
6. Under **Electricity costs**, in the **Price per KW hour** textbox, enter a realistic value for your location.

[!NOTE] You can find approximate electricity prices at [Global electricity prices](#). These prices are in USD (\$). Convert them to an approximate value in your preferred currency.

7. Under **Storage costs**, keep app the default values.
8. Under **IT labor costs**, keep all the default values.
9. Under **Other assumption**, expand each section and examine the associated costs.
10. At the bottom of the page, select **Next**.

Investigate the 5 year report

1. On the **View report** page, note that the **Timeframe** defaults to **5 years**.
2. Scroll down to display the report, and note the estimated costs for on-premises systems compared to Azure. Make a note of this information:
 - Where is the majority of the spend for on-premises workloads?
 - Where the largest cost saving if you decide to migrate to Azure?
3. Expand each section in turn and investigate the breakdown of costs.
4. Change the region based on your location, network latency, or data sovereignty requirements, and examine how the costs compare.

Investigate the 3 year report

1. Scroll to the top of the page and then, in the **Timeframe** textbox, select **3 years**.
2. Scroll down the report and investigate the estimated breakdown of costs for on-premises systems and Azure. Make a note of this information:
 - Where is the majority of the spend for on-Premises?
 - Where the largest cost saving if you decide to migrate to Azure?
3. Expand each section in turn and investigate the breakdown of costs

After completing this exercise, you have used the Azure TCO calculator to identify cost differences between on-premises and Azure deployments for AdventureWorks Corporation's accounting server and its associated databases.

Exercise 2: Understand the stages of migration

In this exercise, you will examine the specifications of your existing server.

Estimated Time: 15 Minutes

The main tasks for this exercise are:

1. Identify the amount of memory used by the existing server
2. Discover the type of CPU of the existing server
3. Determine the disk configuration of the existing server

Determine the amount of memory that is being used on the server

1. Sign in to the **LON-DEV-01** virtual machine running in the classroom environment. The username is **administrator**, and the password is **Pa55w.rd**.
2. Use a method of your choice to determine the amount of memory being used by the server.

Tip: Use System Information, an application included in all modern versions of Windows, to obtain specifications about your server or desktop device. To open System Information, select Start, and then enter System Information into the search field.

3. Make a note of the value in Notepad.

Determine the amount of CPU that is being used on the server

1. Use a method of your choice to determine how CPU resources are being used on the server.
2. Make a note of your observations in Notepad.

Determine the disk configuration that is being used on the server

1. Use a method of your choice to determine the disk configuration that is configured on the server.
2. Make a note of the configuration in Notepad.
3. Save the Notepad into the *Allfiles\Labfiles\Starter\DP-150.1* folder with the name of ServerSpecs.txt

After completing this exercise, you will have collected information on the hardware specifications of the server.

Exercise 3: Data Migration Technologies

In this exercise, you will select a data platform technology to help a customer migrate from an on-premises SQL Server to Azure.

Estimated Time: 15 Minutes

The tasks for this exercise are as follows:

1. Identify the database technology required to facilitate a migration in scenario A.
2. Identify the database technology required to facilitate a migration in scenario B.

Identify the database technology required to facilitate a migration in scenario A

1. Read the following scenario

The customer has an application that uses many databases currently residing in an on-premises version of Microsoft SQL Server 2008. The total database footprint is large at 12 TB, and rapidly growing by several terabytes per year. The application also has tight integration with SQL Server Reporting Services for its reporting functionality. The existing SAN storage, which contains the databases, is almost at capacity, expensive to expand, and nearing the end of its life. The application is critical to the company, with a moderate transaction rate, and any downtime would have significant business impact. Small maintenance windows are available in which to make changes to maximize the availability of the application. The high growth rate has seen more and more time being spent by DBAs and sysadmins to keep everything running.

2. From the scenario, which database platform technology would be appropriate?

Identify the database technology required to facilitate a migration in scenario B

1. Read the following scenario

The customer keeps departmental data in databases on a quad-core SQL Server with 16 GB of memory. The server is used as a backend to simple data access for spreadsheets and an Access form. There are six databases in total that take up 350 MB of storage. The maximum number of concurrent connections to this server is twelve.

2. From the scenario, which database platform technology would be appropriate?

After completing this exercise, you will have identified the appropriate Azure database technologies for your migration project.

Lab Review

After approximately 45 minutes, the instructor will bring a close to this lab. The class will discuss the findings of each group.

DP 050 - Migrating SQL workloads to Azure

Lab 2 - Choose the right tools for data migration

Estimated Time: 25 minutes

Pre-requisites: There are no pre-requisite steps to perform this lab.

Lab files: There are no lab files for this lab

Lab overview

The students will use the Data Migration Assistant in the prescribed data platform modernization stages to investigate their environment in an automated way. They will also identify any premigration compatibility issues and define a plan on how to address those issues before performing a migration of an on-premises server. Finally, they will assess how a workload will perform on a target version of Azure SQL Database.

Lab objectives

After completing this lab, you will be able to:

1. Identify migration candidates using Data Migration Assistant

Scenario

You are the senior data engineer of AdventureWorks and are preparing for a data modernization project. You will begin by performing an inventory of the SQL Servers that exist in your environment using a tool. You will then perform an assessment of the server to identify any issues that need to be resolved prior to any modernization work that will be undertaken. One of your on-premises SQL Servers is used to process sales for the business, and you want to ensure that your new Azure services can support your workloads after they are migrated.

Exercise 1: Identify migration candidates using Data Migration Assistant

In this exercise, you will use the Data Migration Assistant to look for SQL Server feature parity and compatibility issues between SQL Server and Azure SQL Database.

Estimated Time: 20 Minutes

The main tasks for this exercise are:

1. Install the Data Migration Assistant (DMA) on Windows.
2. Use the Data Migration Assistant.

Install the Data Migration Assistant (DMA) on Windows

[!NOTE] Follow the steps in this task even if DMA is already installed, to ensure that you are using the latest version of the tool.

1. Sign in to the **LON-DEV-01** virtual machine running in the classroom environment. The username is **administrator**, and the password is **Pa55w.rd**.
2. Open up your web browser, and go to the download page for Microsoft Data Migration Assistant,
+++<https://www.microsoft.com/download/details.aspx?id=53595>+++.
3. Confirm that your environment supports the software by checking the requirements list.
4. To download **DataMigrationAssistant.msi**, select **Download**, and then select **Run**.
5. On the **Welcome** screen, select **Next**.
6. Select **I accept the terms in the License Agreement**, and then select **Next**.
7. Read the **Privacy Statement**, and then select **Install**.
8. If prompted, allow UAC control for this application.
9. To complete install, select **Finish**.

Use the Data Migration Assistant to prepare for migration from SQL Server to Azure SQL Database

1. Select the **Start** button, type **Data Migration**, and then select **Microsoft Data Migration Assistant**.
2. To create a new project, on the left, select **+**.
3. On the **New** page, enter these values, and then select **Create**:

Property	Value
Project type	Assessment
Project name	+++Migration Assessment SQL DB+++
Assessment type	Database Engine
Source server type	SQL Server
Target server type	Azure SQL Database

4. On the **Options** page, leave the default values, and then select **Next**.
5. On the **Connect to a server** page, enter these values, and then select **Connect**:

Property	Value
Server name	localhost
Authentication type	Windows Authentication
Encrypt connection	Deselected
Trust server certificate	Selected

6. On the **Add sources** page, select the **AdventureWorks** database, and then select **Add**.
7. On the **Select sources** page, select **Start Assessment**.
8. Review the SQL Server feature parity and compatibility issues found. Note how many feature parity issue there are.

Use the Data Migration Assistant to prepare for migration from SQL Server to Azure SQL Database Managed Instance

1. To create a new project, on the left, select **+**.
2. On the **New** page, enter these values, and then select **Create**:

Property	Value
Project type	Assessment

Property	Value
Project name	+++Migration Assessment SQL DB MI+++
Assessment type	Database Engine
Source server type	SQL Server
Target server type	Azure SQL Database Managed Instance

3. On the **Options** page, leave the default values, and then select **Next**.
4. On the **Connect to a server** page, enter these values, and then select **Connect**:

Property	Value
Server name	localhost
Authentication type	Windows Authentication
Encrypt connection	Deselected
Trust server certificate	Selected

5. On the **Add sources** page, select the **AdventureWorks** database, and then select **Add**.
6. On the **Select sources** page, select **Start Assessment**.
7. Review the SQL Server feature parity and compatibility issues found. Note how many feature parity issue there are.

Use the Data Migration Assistant to prepare for migration from SQL Server to SQL Server on an Azure Virtual Machine

1. To create a new project, on the left, select **+**.
2. On the **New** page, enter these values, and then select **Create**:

Property	Value
Project type	Assessment
Project name	+++Migration Assessment Azure VM+++
Assessment type	Database Engine
Source server type	SQL Server
Target server type	SQL Server on Azure Virtual Machines

3. On the **Options** page, leave the default values, and then select **Next**.
4. On the **Connect to a server** page, enter these values, and then select **Connect**:

Property	Value
Server name	localhost
Authentication type	Windows Authentication
Encrypt connection	Deselected
Trust server certificate	Selected

5. On the **Add sources** page, select the **AdventureWorks** database, and then select **Add**.
6. On the **Select sources** page, select **Start Assessment**.
7. Review the SQL Server feature parity and compatibility issues found. Note how many feature parity issue there are.

After completing this exercise, you have collected information about the SQL Server feature parity and compatibility issues found between the instance of SQL Server on premises and various SQL Server hosting options in Azure.

Lab Review

After approximately 20 minutes, the instructor will bring a close to this lab. The class will discuss the findings of each group.

DP 050 – Migrating SQL Workloads to Azure

Lab 3 – Migrate SQL Workloads to SQL Server in Azure Virtual Machine

Estimated Time: 60 minutes

Pre-requisites: There are no pre-requisite steps to perform in this lab.

Lab files: There are no lab files for this lab.

Lab Overview

The students will initially evaluate the migration process that they will use to migrate from an on-premises SQL Server 2008 R2 instance to an SQL Server 2017 running in a virtual machine. They will then perform a migration using Data Migration Assistant to move databases. Finally, they will assess a successful migration.

Lab Objectives

After completing this lab, you will be able to:

- Create a new virtual machine on Azure that runs SQL Server 2017.
- Create an Azure Storage Account and a file share.
- Perform a migration of SQL Server 2008 R2 databases to SQL Server on an Azure VM.

Scenario

You are the senior database management lead of AdventureWorks and are preparing to run a data modernization project. You will prepare the necessary environment to migrate a set of databases to SQL Server on an Azure virtual machine and perform test migrations using Data Migration Assistant.

Exercise 1: Create a new virtual machine on Azure that runs SQL Server 2017

In this exercise, you will create a new virtual machine on Azure, using the Azure Portal.

Estimated Time: 20 Minutes

The task for this exercise is:

1. Create a new virtual machine in the Azure Portal

Provision a SQL Server 2017 virtual machine

[!NOTE] If you are running this lab in a hosted lab environment, perform these steps inside that environment.

1. In the [Azure portal](#), select **Create a resource**.
2. In the Marketplace search box, type **SQL Server 2017 on Windows Server 2019** and then press Enter. Under **Showing All Results** select **SQL Server 2017 on Windows Server 2019**.
3. In the **Select a plan** dropdown list, select **Free SQL Server License: SQL Server 2017 Developer on Windows Server 2019**, and then select **Create**.
4. In the **Create a virtual machine** wizard, on the **Basics** page, enter these values, and then select **Next: Disks >**:

Property	Value
Subscription	Choose your subscription
Resource group	Create a new resource group called DP-050-Training
Virtual machine name	sql2017vm
Region	Select a region close to you
Availability options	No infrastructure redundancy required
Image	Free SQL Server License: SQL Server 2017 Developer on Windows Server 2019 - Gen1

Property	Value
Azure Spot instance	No
Size	Standard_D2_v2
Username	sqladmin
Password	Pa55w.rdPa55w.rd
Confirm Password	Pa55w.rdPa55w.rd
Public inbound ports	Allow selected ports
Select inbound ports	RDP (3389)
Would you like to use an existing Windows license?	No

5. On the **Disks** page, accept the default settings, and then select **Next: Networking >**.
6. On the **Networking** page, accept the default settings, and then select **Next: Management >**.
7. On the **Management** page, in the **Boot diagnostics** list, select **Disable**, and then select **Next: Advanced >**.
8. On the **Advanced** page, accept the default settings, and then select **Next: SQL Server settings >**.
9. On the **SQL Server Settings** page, enter these values, and then select **Review + create**:

Property	Value
SQL connectivity	Public (Internet)
Port	1433
SQL Authentication	Enable
Login name	sqladmin
Password	Pa55w.rdPa55w.rd
Azure Key Vault integration	Disable

10. On the **Review + create** page, select **Create**.

[!NOTE] This step could take about 10 minutes to complete.

11. When the deployment is complete, select **Go to resource**.

12. Locate and record the **Public IP address** for your VM. You will need this address later.
13. Next to **DNS name**, select **Configure**.
14. In the **DNS name label (optional)** textbox, type a unique DNS name and record it.

For example: sql2017vmxxxx.centralus.cloudapp.azure.com

15. Select **Save**.

Results: After completing this exercise, you have a SQL Server 2017 instance running in an Azure virtual machine.

Exercise 2: Create an Azure storage account and file share

In this exercise, you will create a new storage account in Azure, by using the Azure Portal.

Estimated Time: 15 Minutes

The main tasks for this exercise are:

1. Create an Azure storage account.
2. Create a file share in the Azure storage account.

Create an Azure storage account

1. In the [Azure portal](#), select **Create a resource**.
2. In the **Search the Marketplace** textbox, type **Storage account**, and then press **Enter**.
3. Under **Showing All Results** select **Storage account**, and then select **Create**.
4. In the **Create storage account** wizard, on the **Basics** page, enter these values:

Property	Value
Subscription	Select your subscription
Resource group	DP-050-Training
Storage account name	dp050storagexxxx where xxxx is a random sequence of characters
Location	Select the same location as you used for the virtual machine
Performance	Standard
Account kind	StorageV2 (general purpose v2)
Replication	Locally-redundant storage (LRS)

[!NOTE] Make a careful note of the storage account name you use. You will need this name later in the lab.

5. Select **Review + create**.
6. On the **Review + create** page, select **Create**.

[!NOTE] This deployment could take a few minutes
7. When the deployment is complete, select **Go to resource**
8. Under **Settings**, select **Access keys**.
9. On the **Access keys** page, select **Show keys**, and then under **key1**, record the contents of the **Key** textbox.

Create a file share

1. In the storage account page, under **File service** in the left menu, select **File shares**.
2. On the **File shares** page, select **+ File share**
3. In the **New file share** page, enter these values:

Property	Value
Name	backupshare
Quota	200 GiB

4. Select **Create**.

Results: You have now successfully created an Azure file share which will be used as a shared access location for SQL Server database backup files. In the next exercise you will configure the SQL instances to access the shared location.

Exercise 3: Create a connection for the SQL Server instances to connect to the Azure file share

In this exercise, you will configure the SQL Server environment to access the Azure file share on both the on-premises server and the new Azure VM.

Estimated Time: 10 Minutes

The main tasks for this exercise are:

1. Register the file share through SQL Management Studio by mapping a network drive.
2. Connect to the file share.

Register the server instances in SQL Management Studio

1. Sign into the **LON-DEV-01** virtual machine running in the classroom environment. The username is **administrator**, and the password is **Pa55w.rd**.
2. Start **SQL Management Studio**, and then connect to the local instance (LONDON).
3. In SQL Management Studio Object Explorer, select **Connect** and then select **Database Engine**.
4. In the **Connect to Server** dialog, enter these values and then select **Connect**:

Property	Value
Server name	Enter the fully qualified domain name or IP address of your SQL 2017 VM in Azure. For example: sql2017vmxxx.centralus.cloudapp.azure.com
Authentication	SQL Server
Login	sqladmin
Password	Pa55w.rdPa55w.rd

Connect the on-premises SQL instances to the file shares

[!NOTE] In order for SQL Server to be able to connect to a drive letter residing on a file share, you have to map the network drive by running `xp_cmdshell` in SQL Server Management Studio, so that the SQL service account can access the share. Data Migration Assistant uses the SQL service account to backup the database. For security reasons, command line access should be limited to SQL Server service accounts. By default SQL command line is disabled.

1. To configure the connection on the on-premises SQL Server, in SQL Management Studio, in the **Object Explorer**, right-click the **LONDON** server and then select **New Query**.
2. Enter this Transact-SQL code:

```
sql EXECUTE sp_configure 'show advanced options', 1;
RECONFIGURE; EXECUTE sp_configure 'xp_cmdshell', 1;
RECONFIGURE; GO EXECUTE xp_cmdshell 'net use U: \\
<storageaccountname>.file.core.windows.net\backupshare
/persistent:Yes /u:Azure\<storageaccountname>
<storageaccountkey>'; EXECUTE xp_cmdshell 'dir U:';
```

3. In the query text, replace `<storageaccountname>` with the name of the storage account you created earlier. The name must be entered in two places.
4. Replace `<storageaccountkey>` with the primary access key you recorded for the storage account.
5. Execute the query and check that there are no error messages in the results.

[!NOTE] The SQL code maps the network drive U: to the storage account in Azure. However, because this is done in the context of the SQL Service account, you will not see the U: drive in File Explorer or when you use the `net use` command.

6. Save the query in the Labfiles folder as **MapNetworkDrive.sql**
7. Start a new query window and disable `xp_cmdshell` on the LONDON SQL Server by running this query:

```
sql EXECUTE sp_configure 'xp_cmdshell', 0; RECONFIGURE;
```

8. Close all the queries windows and don't save any files.

Connect the Azure VM SQL instances to the file shares

1. To configure the connection on the Azure VM SQL Server, in the **Object Explorer**, right-click the SQL Server in Azure, and then select **Connect**.
2. In the **Connect to Server** dialog, in the **Password** textbox, type **Pa55w.rdPa55w.rd** and then select **Connect**.
3. On the **File** menu, select **Open/File** and then open the **MapNetworkDrive.sql** file that you saved above.
4. In the status bar at the bottom of the query window, check that you are connected to the Azure VM.
5. To map the U: drive on the Azure VM, execute the query and check that there are no error messages in the results.
6. Start a new query window and disable `xp_cmdshell` by running this query:

```
sql EXECUTE sp_configure 'xp_cmdshell', 0; RECONFIGURE;
```

7. Close SQL Server Management Studio.

Exercise 4: Perform a Database Migration using SQL Server Data Migration Assistant

In this exercise, you will migrate the data from an on-premises SQL Server to a VM in Azure.

Estimated Time: 10 Minutes

The main tasks for this exercise are:

1. Migrate databases by using Data Migration Assistant.
2. Validate a successful migration of the database.

Migrate SQL Databases using Data Migration Assistant

1. In the **LON-DEV-01** virtual machine, open **Microsoft Data Migration Assistant** and then select **+**.
2. In the **New** page, enter these values:

Property	Value
Project type	Migration
Project name	Migration to Azure VM
Source server type	SQL Server
Target server type	SQL Server on Azure Virtual Machines

3. Select **Create**.
4. On the **Specify source & target** page, under **Source server details**, enter these values:

Property	Value
Server name	localhost
Authentication type	Windows Authentication
Encrypt connection	No
Trust server certificate	Yes

5. Under **Target server details**, enter these values:

Property	Value
Server name	Enter the IP address or DNS name of the virtual machine in Azure
Authentication type	SQL Server Authentication
Username	sqladmin
Password	Pa55w.rdPa55w.rd
Encrypt connection	Yes
Trust server certificate	Yes

6. Select **Next**.

7. On the **Add databases** page, deselect all the databases except for **AdventureWorks** and **AdventureWorksLT2008TR2**.

8. In the **Shared location** textbox, type **U:** and then select **Next**.

9. Review the **Select logins** window. There are no logins to migrate. Select **StartMigration**

[!NOTE] All databases will be backed up to the shared network drive in the Azure Storage file share.

10. Monitor the migration process.

11. When the migration is complete, close Data Migration Assistant.

Validate a successful migration

1. In the **LON-DEV-01** virtual machine, open **SQL Management Studio**.
2. In the **Connect to Server** dialog, in the **Server name** list, choose the IP address or DNS name of the Azure VM.
3. In the **Password** textbox, type **Pa55w.rdPa55w.rd** and then select **Connect**.
4. In Object Explorer, expand the expand the **Databases** list.
5. Verify that the **AdventureWorks** database has been successfully migrated.
6. On the **File** menu, select **New/Query with Current Connection**.

7. Type and execute this query to validate the database compatibility level of each of the databases.

```
sql SELECT name, compatibility_level FROM sys.databases
```

8. Alter the database compatibility level for the **Adventureworks** database using this query:

```
sql ALTER DATABASE AdventureWorks SET COMPATIBILITY_LEVEL  
= 110; GO
```

9. Backup the Adventureworks database using this query:

```
sql BACKUP DATABASE Adventureworks TO DISK =  
'U:\Adventureworks' WITH FORMAT, MEDIANAME =  
'Adventureworks', NAME = 'Full Backup of Adventureworks';
```

10. Upon successful completion of the backup close **SQL Management Studio**.

[!IMPORTANT] Don't delete the SQL Server VM running in Azure at this end of this lab. You will use it as the source for migration to Azure Database in Lab 4.

Results: You have now completed the successful migration of SQL Server databases to SQL Server 2017 running in an Azure VM.

DP 050 – Migrating SQL Workloads to Azure

Lab 4 – Migrate SQL Workloads to Azure SQL Database

Estimated Time: 60 minutes

Pre-requisites: There are no pre-requisite steps to perform in this lab.

Lab files: There are no lab files for this lab

Lab Overview

In this lab you will perform a migration to Azure SQL Database. First, you will perform a schema migration and create the destination database as a pre-requisite, then you migrate from a database running in an on-premises SQL instance to Azure SQL Database. You will perform an online migration using Database Migration Service (DMS), to keep the data in synchronization between the source and the target databases until cut over to the new database.

Lab Objectives

At the end of this lab, you will be able to:

- Create a database in Azure SQL Database using Azure Cloud Shell.
- Configure Azure Data Migration Service.
- Migrate a database schema to Azure SQL Database.
- Perform an online migration using Data Migration Service.

Scenario

You are the senior database management lead at AdventureWorks and are preparing to run a data modernization project. You will prepare the necessary environment to migrate a set of databases to SQL Server in an Azure Virtual Machine (VM) and perform test migrations using Data Migration Assistant.

Exercise 1: Create a database in Azure SQL Database using Azure Cloud Shell

In this exercise you'll be using Azure Cloud Shell to:

- Create a new resource group.
- Create a new Azure SQL Database server instance.
- Configure the Azure SQL Database server firewall.
- Create a new general-purpose database.

Estimated Time: 20 minutes

There are many options in Azure to automate the installation, management, and deployment of services. One of the options could be installing the Azure CLI command line tool, a lightweight cross-platform command-line tool. Another option would be to automate and script using Azure Cloud Shell.

Azure Cloud Shell is an interactive shell environment hosted in Azure and managed through your browser. Cloud Shell lets you use either Bash or PowerShell to work with Azure services. You can use the Cloud Shell pre-installed commands to run the code in this article without having to install anything on your local environment.

Task 1: Log on and configure Azure Cloud Shell

[!NOTE] This task can be completed entirely in the Azure Cloud Shell.

1. Go to the [Azure Shell](#)
2. Log on to your Azure Subscription with the credentials used for this training.
3. Select **Bash** as the scripting environment.

Task 2: Create a new storage account and share for Azure Cloud Shell

1. When prompted that you have no storage account created for Azure Cloud Shell, select **Show Advanced Settings**, enter these

values, and then select **Create storage**:

Property	Value
Subscription	Select the Azure subscription used for this lab
Cloud Shell Region	Select an available Cloud Shell region close to your location
Resource group	Create a new resource group called dp050lab4rg
Storage account	Create a new storage account called dp050sa<youridentifier> where <youridentifier> is a unique string
File share	Create a new file share called dp050share<youridentifier> where <youridentifier> is a unique name

2. When the storage account has been created, the Azure Cloud Shell opens.

Task 3: Create the Azure SQL Database server instance and database

1. Select the {} icon to open the code editor.
2. Paste this script into the code editor:

```
```bash
```



**bash script to create a new sql db server instance and sql db**

**Disclaimer: this script is a sample script on how to create an Azure database but uses least restrictive firewall settings for lab purposes. Do not use this script**

**defining a name for the resource group**

`resourcegroup=dp-050-labresourcegroup`

**edit the variable below to  
provide a unique server name**

servername=dp-050-servername

**edit the variable below to  
provide the location – azure  
locations can be listed by typing  
az account list-locations -o table  
in the shell command interface**

location=eastus

adminuser=sqladmin password=Pa55w.rdPa55w.rd  
firewallrule=dp-050-access

# **edit the script to provide a unique database name**

labdatabase=dp-050-Adventureworksxxx

# **creates a resource group**

```
az group create --name $resourcegroup --location $location
```

# **creates a sql db servername**

```
az sql server create --name $servername --resource-group
$resourcegroup --admin-user $adminuser --admin-password
$password --location $location
```

# shows current firewall list

```
az sql server firewall-rule list --resource-group $resourcegroup --
server $servername
```

**creates a server-based firewall  
– note – you should restrict the  
start/end ip range based on  
your environment**

```
az sql server firewall-rule create --resource-group $resourcegroup --
server $servername --name $firewallrule --start-ip-address 0.0.0.0 --
end-ip-address 255.255.255.255
```



# creates a general-purpose SQL database

```
az sql db create --name $labdatabase --resource-group
$resourcegroup --server $servername -e GeneralPurpose ``
```

3. In the script, change the value `dp-050-servername` to a unique servername, such as `dp-050-serverxxx`. Don't use upper case in this name.
4. In the script, change the value `eastus` to an Azure location near you.
5. In the script, change the value `dp-050-Adventureworksxxx` to a unique database name.
6. To save the script, press `CTRL + S`, type **CreateSQLDBandServer.sh**, and then select **Save**.
7. To exit the code editor, press `CTRL + Q`.
8. To execute the script, type **sh CreateSQLDBandServer.sh** and then press `Enter`.

When the script has completed successfully, you can validate that the resource group, server, and database have been created in your Azure account subscription.

## Exercise 2: Migrate a database to Azure SQL Database

In this exercise, you'll migrate the database schema of a database in a SQL Server instance running on-premises and load the schema into the SQL Database created in the previous exercise. The source SQL Server instance for the database schema runs on the VM LON-DEV-01.

At the end of this lab, you will be able to:

- Create a new migration project using Data Migration Assistant.
- Migrate a database a schema.

**Estimated Time:** 20 minutes

[!NOTE] This exercise is completed using Data Migration Assistant running on the LON-DEV-01 VM.

### Task 1: Create a Migration Project using Data Migration Assistant

1. Sign into the **LON-DEV-01** virtual machine running in the classroom environment. The username is **administrator**, and the password is **Pa55w.rd**.
2. To create a new data migration project, open **Microsoft Data Migration Assistant**, and then **+**.
3. In the **New** page, enter these values, and then select **Create**:

Property	Value
Project type	Migration
Project name	SQLSchemaMigration
Source server type	SQL Server
Target server type	Azure SQL Database
Migration scope	Schema only

### Task 2: Perform the schema migration

1. On the **Select source** page, under **Connect to source server**, enter these values, and then select **Connect**:

Property	Value
Server name	localhost
Authentication type	Windows Authentication
Encrypt connection	No
Trust server certificate	Yes

2. In the list of databases, select **AdventureworksLT2008R2** and deselect the **Assess database before Migration** checkbox.

3. Select **Next**.

4. On the **Select target** page, under **Connect to target server**, enter these values, and then select **Connect**:

Property	Value
Server name	Type the <b>servername</b> of the server you created in the previous exercise. List the server by its <b>full qualified name</b> , for example: dp-050-servername.database.windows.net
Authentication type	SQL Server Authentication
Username	sqladmin
Password	Pas55w.rdPa55w.rd
Encrypt connection	No
Trust server certificate	Yes

5. In the list of databases, select the target database you created in the previous exercise, and then select **Next**
6. Review all the schema objects, and then select **Generate SQL script**

The Data Migration Assistant window will look similar to:

! Data Migration Assistant Window  
/home/ll/Azure\_clone/Azure\_new/DP-050-Migrating-SQL-  
Workloads-to-Azure/Instruction//images/dbmigrate.png

7. Select **Deploy Schema**.
8. When the deployment is complete, close Data Migration Assistant.

## Exercise 3: Migrate on-premises databases to Azure SQL Database

In this exercise you'll perform an online migration of a database using the Azure Data Migration Service.

At the end of this exercise, you will be able to:

- Configure the source instance for replication and make sure the pre-requisites for migration are met.
- Perform a live migration using Azure Data Migration Service.

**Estimated Time:** 20 minutes

### Task 1: Configure SQL Server as a replication distributor

You will complete this task using SQL Management Studio on the SQL 2008 R2 Virtual Machine but connected to the **SQL Server 2017 instance**.

[!NOTE] Perform these tasks on LON-DEV-01 while connected to the SQL Server 2017 instance in Azure, to avoid having to configure VPN access between the hosted SQL 2008 R2 VM and Azure. In a non-lab environment, you would typically configure VPN or ExpressRoute.

1. In SQL Management Studio Object Explorer, connect to the SQL Server 2017 instance you created in Lab 3.

If the SQL Server 2017 instance was not registered, select **Connect/Database Engine**, and then use these values:

Property	Value
Server name	The IP address or fully qualified domain name of your SQL 2017 VM, for example sql2017vmxxxxx.centralus.cloudapp.azure.com
Authentication	SQL Server Authentication
Login	sqladmin
Password	Pa55w.rdPa55w.rd

2. In the **Object Explorer**, right-click **Replication**, and then select **Configure Distribution**.
3. In the **Configure Distribution Wizard**, accept the default settings on each of the Configure Distribution pages. If the wizard prompts you to **start the SQL Server Agent Automatically**, select **Yes**.
4. On the **Complete the wizard** page, select **Finish**
5. If the **SQL Server Agent Fails to start**, start the agent manually by right-clicking on **SQL Server Agent in SQL Management Studio | Object Explorer | SQL Server Agent** and Start the service
6. To set the database recovery model for the **AdventureworksLT2008R2** database to FULL, execute this query in a new query window:

```
sql ALTER DATABASE AdventureworksLT2008R2 SET RECOVERY
FULL WITH NO_WAIT
```

7. Perform a FULL Backup of the database by executing the following query

```
sql BACKUP DATABASE AdventureworksLT2008R2 TO DISK =
'd:\awlt2008r2backup.bak'
```

8. As the backup completes, close **SQL Management Studio**.

## **Task 2: Perform an online migration using Azure Database Migration Service**

In this task you will configure the Azure Database Migration Service to enable a live migration between the database running on a VM and Azure SQL Database.

[!NOTE] Complete all these tasks in the Azure Portal.

1. In the Azure Portal, select **Create a resource**.
2. In the **Search the Marketplace** textbox, type **Azure Database Migration Service** and then press **Enter**.
3. Select **Create**.
4. In the **Create Migration Service** wizard, on the **Basics** page, enter these values, and then select **Next: Networking >>**:

<b>Property</b>	<b>Value</b>
Subscription	Select your subscription.
Resource group	dp050lab4rg
Migration service name	<b>dp050dmsxxx</b> where <b>xxx</b> is a unique value.
Location	Select a location near you.
Service mode	Azure
Pricing tier	Premium

5. On the **Networking** page, in the **Virtual network name** textbox, type **OnPremGateway**, and then select **Review + create**.

6. On **Review + create** page, select **Create**.

[!NOTE] This deployment can take up to 10 minutes.

7. When the deployment is complete, select **Go to resource**.

8. Select **+ New Migration Project**.

9. In the **New migration project** page, enter these values, and then select **Create and run activity**:

<b>Property</b>	<b>Value</b>
Project name	DP050OnlineMigration
Source server type	SQL Server
Target server type	Azure SQL Database
Choose type of activity	Online data migration

10. In the **SQL Server to Azure SQL Database Online Migration Wizard**, on the **Select source** page, enter these values, and then select **Next: Select target >>**:

<b>Property</b>	<b>Value</b>
Source SQL Server instance name	Fully qualified domain name of your SQL 7 VM in Azure
Authentication type	SQL Authentication
User Name	sqladmin
Password	Pa55w.rdPa55w.rd

Property	Value
Encrypt connection	Deselected
Trust server certificate	Selected

11. On the **Select target** page, enter these values, and then select **Map to target databases >>**:

Property	Value
Target server name	Fully qualified domain of the Azure SQL Database server
Authentication type	SQL Authentication
User name	sqladmin
Password	Pa55w.rdPa55w.rd
Encrypt connection	Deselected
Trust server certificate	Selected

12. On the **Map to target databases** page, select the database for which you did the schema migration, select the target database, and then select **Configure migration settings >>**.
13. On the **Configure migration settings** page, examine the notes for each table, and then select **Summary >>**.
14. Name the **Activity name** textbox, type **dp050lab4activity**, and then select **Run Migration**.

[!NOTE] In a production environment, you would increase the database tier of the SQL database to perform a faster load of the data.

15. During the migration process, select **Refresh** occasionally and monitor the **STATUS** column, until it displays **Ready to cutover**.

[!NOTE] During the migration, any changes made to data in the source database will be synchronized to the target database.

If you have time, you can insert some records in the **ProductCategory** table in the source database using this SQL



command. Ensure that you execute this command on the source VM:

```
``sql USE [AdventureWorksLT2008R2] GO
```

```
INSERT INTO [Sales LT].[ProductCategory]
([ParentProductCategoryID] ,[Name] ,[ModifiedDate]) VALUES
(1, ' Myproduct', getdate()) ``
```

16. When the data migration is complete, select the database and then select **Start Cutover**.

17. Check **Confirm** and then select **Apply**.

Results: You have now successfully completed an online migration to Azure SQL Database. There are no solution files for this lab There are no solution files for this lab There is no lab for this module There are no starter files for this lab There are no starter files for this lab There are no starter files for this lab There are no starter files for this lab There is no lab for this module