Microsoft

Azure SQL Database Migration

Module 6

Learning Units covered in this Module

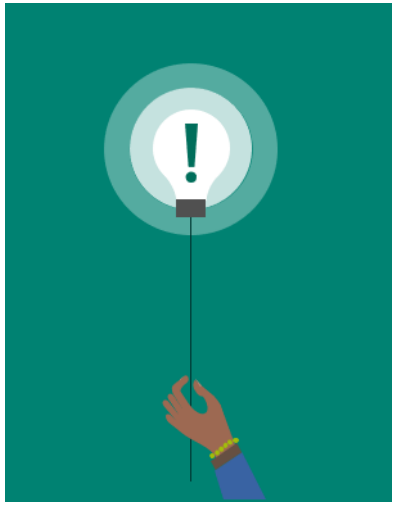
- Lesson 1: Steps to migrate your database to Azure SQL Database
- Lesson 2: Is your database ready to move to Azure SQL database?
- Lesson 3: Fix database migration compatibility issues
- Lesson 4: Identify the right Azure SQL Database SKU
- Lesson 5: Migrate a compatible SQL Server database to Azure SQL Database

Lesson 1: Steps to migrate your database to Azure SQL Database

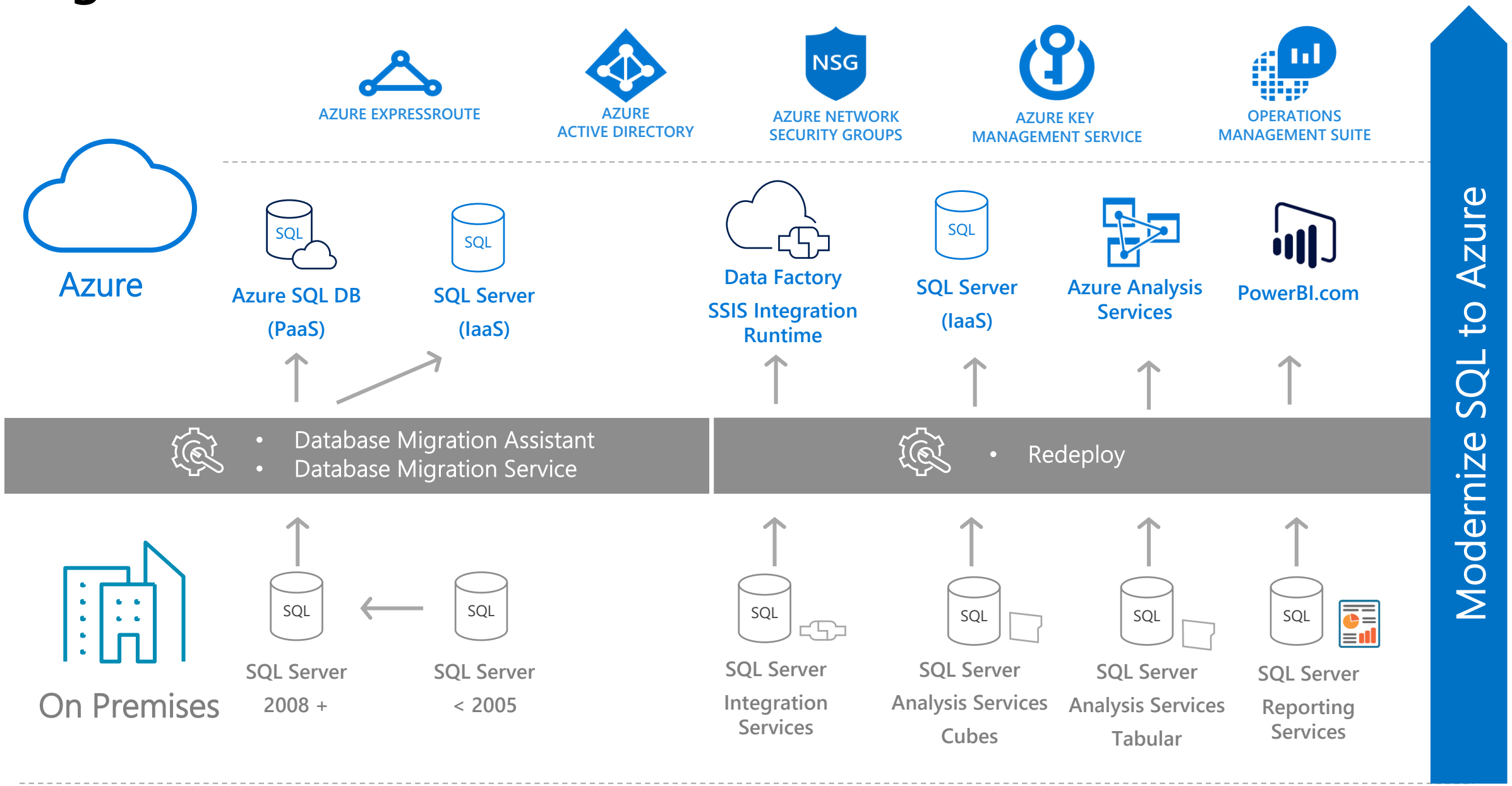
Objectives

After completing this learning, you will be able to:

- Get an overview of the migration steps. We will discuss these steps further in this module.

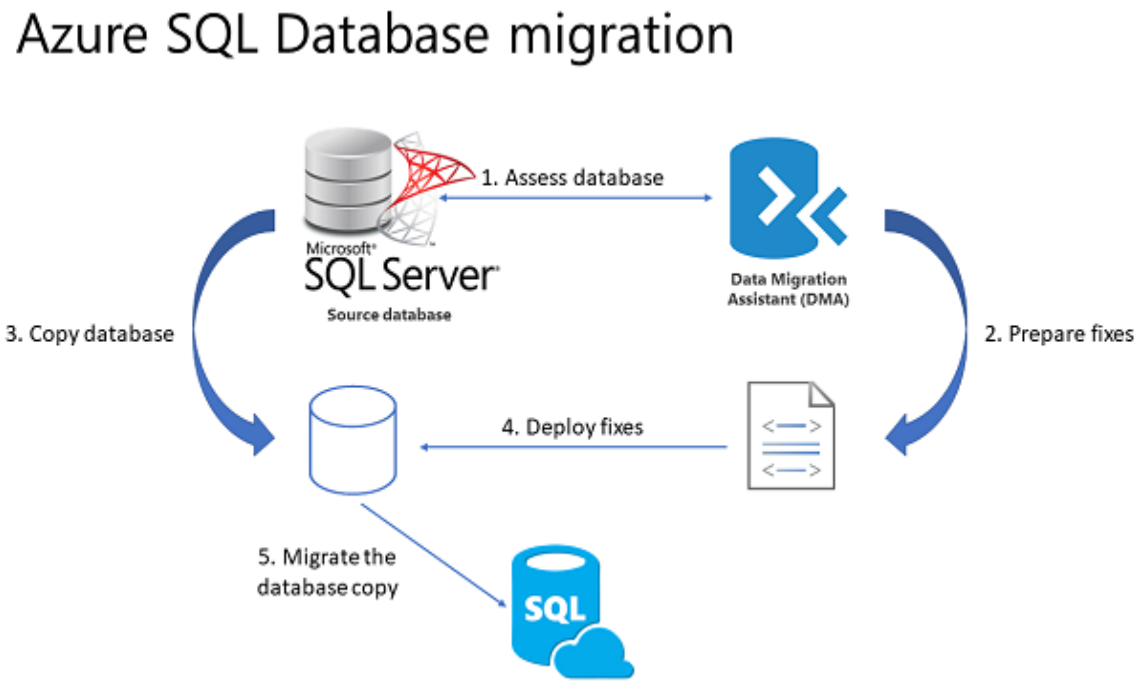


Migration of SQL Server to Azure - Overview



Migration Steps

- Assess the database
- Prepare any necessary fixes as Transact-SQL scripts.
- Make a transactionally consistent copy of the source database.
- Deploy the Transact-SQL scripts to apply the fixes to the database copy.
- Migrate the database copy to a new Azure SQL Database.



Determine Compatibility with DMA

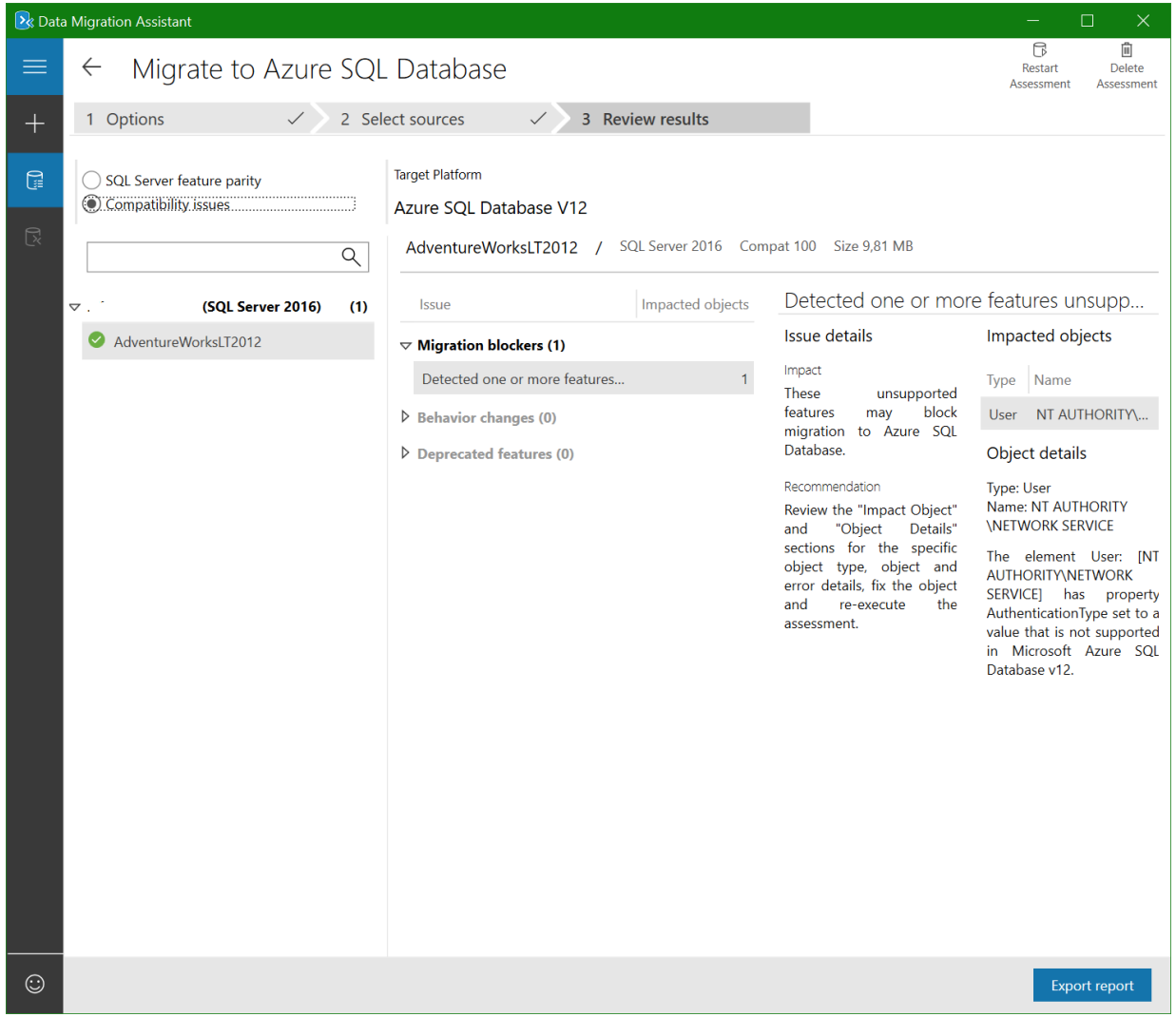
Open Data Migration Assistant (DMA).

Create a new Assessment Project with Azure SQL Database Target.

Connect to local instance.

Select the source database(s).

Verify Compatibility Report.



Fix Database Migration Compatibility Issues

Compatibility issues must be fixed before proceeding with the SQL Server Database migration.

You can use DMA + Extended Events to evaluate any ad hoc or dynamic SQL queries or any DML statements initiated through the application data layer.

A wide variety of compatibility issues.

Use the following resources:

- SQL Server database features not supported in Azure SQL Database
- Discontinued Database Engine Functionality in SQL Server 2019
- Discontinued Database Engine Functionality in SQL Server 2017
- Discontinued Database Engine Functionality in SQL Server 2016
- Discontinued Database Engine Functionality in SQL Server 2014
- Discontinued Database Engine Functionality in SQL Server 2012
- Discontinued Database Engine Functionality in SQL Server 2008 R2

Azure SQL SKU Recommendations

Console Command – Pre-requisites

1

Download and install the latest version of DMA. If you already have an earlier version of the tool, open it, and you'll be prompted to upgrade DMA.

2

Install the minimum version .NET Core 3.1 on the tools machine where the SKU recommendations console application is running.

3

Ensure the account used to connect to your SQL Server on-premises source has sysadmin permission.

Azure SQL SKU Recommendations

Console Command - Setup

Navigate to the SQL Assessment Console Folder

- CD "C:\Program Files\Microsoft Data Migration Assistant\SQLAssessmentConsole“

Collect Performance Data (Replace <instancename> with your SQL Server name. This step will take 15-20 minutes.

- .\SqlAssessment.exe PerfDataCollection --sqlConnectionStrings "Data Source= <instancename>; Initial Catalog=master; Integrated Security=True;" --outputFolder C:\Output

Azure SQL SKU Recommendations

Console Command – Reports

To get assessment for Azure SQL Database

- .\SqlAssessment.exe GetSkuRecommendation --outputFolder C:\Output --targetPlatform AzureSqlDatabase

To get assessment for Azure SQL Managed Instance

- .\SqlAssessment.exe GetSkuRecommendation --outputFolder C:\Output --targetPlatform AzureSqlManagedInstance --elasticStrategy true

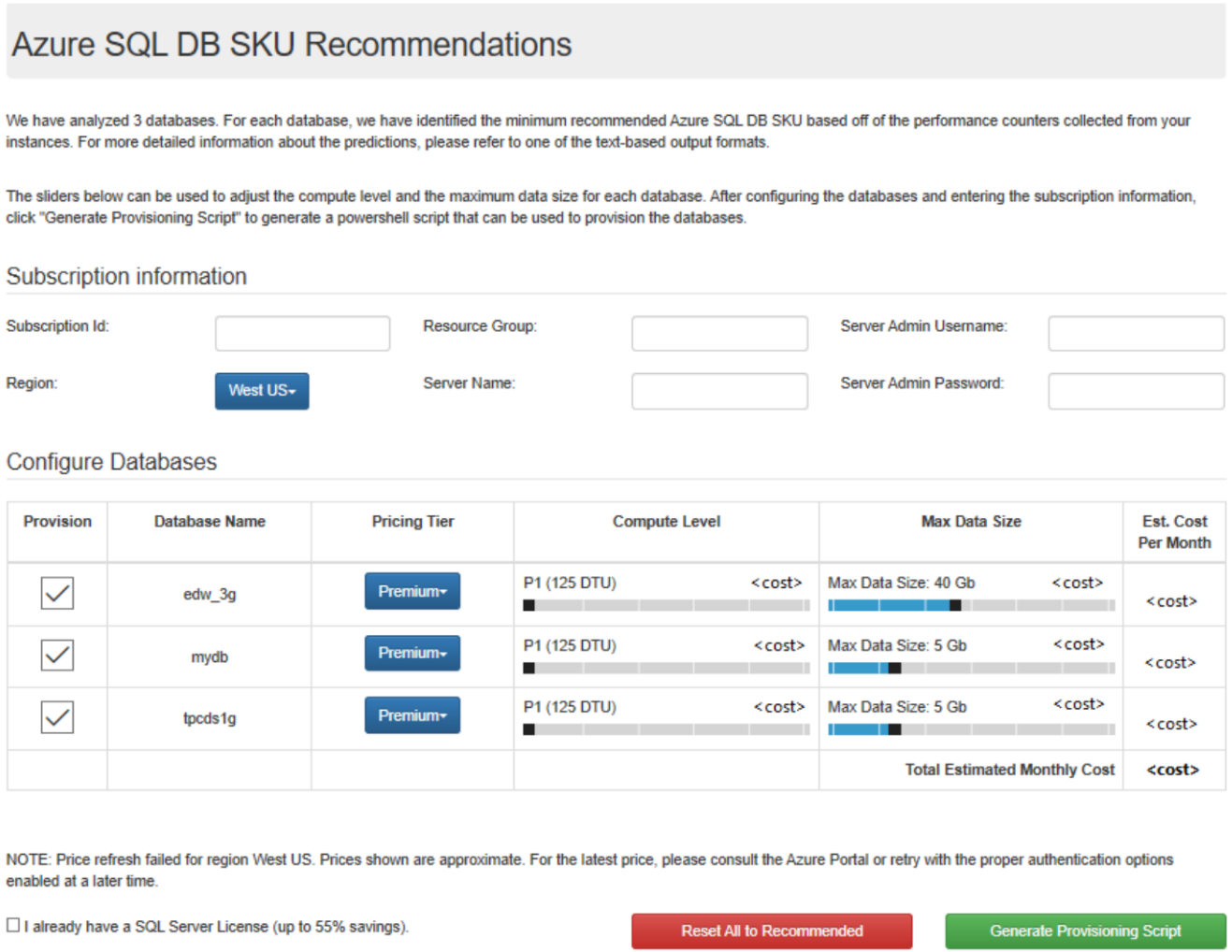
To get assessment for Azure SQL Virtual Machine

- .\SqlAssessment.exe GetSkuRecommendation --outputFolder C:\Output --targetPlatform AzureSqlVirtualMachine

Identify the right Azure SQL Database SKU for your on-premises database

Database Migration Assistant: (DMA)

- Provides SKU recommendations in a user-friendly output based on performance counters collected from the computer(s) hosting your databases.
- It has several deployment options, including:
  - Single database
  - Elastic pools
  - Managed instance



Azure SQL DB SKU Recommendations

We have analyzed 3 databases. For each database, we have identified the minimum recommended Azure SQL DB SKU based off of the performance counters collected from your instances. For more detailed information about the predictions, please refer to one of the text-based output formats.

The sliders below can be used to adjust the compute level and the maximum data size for each database. After configuring the databases and entering the subscription information, click "Generate Provisioning Script" to generate a powershell script that can be used to provision the databases.

Subscription information

Subscription Id: | Resource Group: | Server Admin Username:

Region: West US | Server Name: | Server Admin Password:

Configure Databases

Provision	Database Name	Pricing Tier	Compute Level	Max Data Size	Est. Cost Per Month
☑	edw_3g	Premium	P1 (125 DTU) <cost>	Max Data Size: 40 Gb <cost>	<cost>
☑	mydb	Premium	P1 (125 DTU) <cost>	Max Data Size: 5 Gb <cost>	<cost>
☑	tpcds1g	Premium	P1 (125 DTU) <cost>	Max Data Size: 5 Gb <cost>	<cost>
				Total Estimated Monthly Cost	<cost>

NOTE: Price refresh failed for region West US. Prices shown are approximate. For the latest price, please consult the Azure Portal or retry with the proper authentication options enabled at a later time.

☐ I already have a SQL Server License (up to 55% savings).

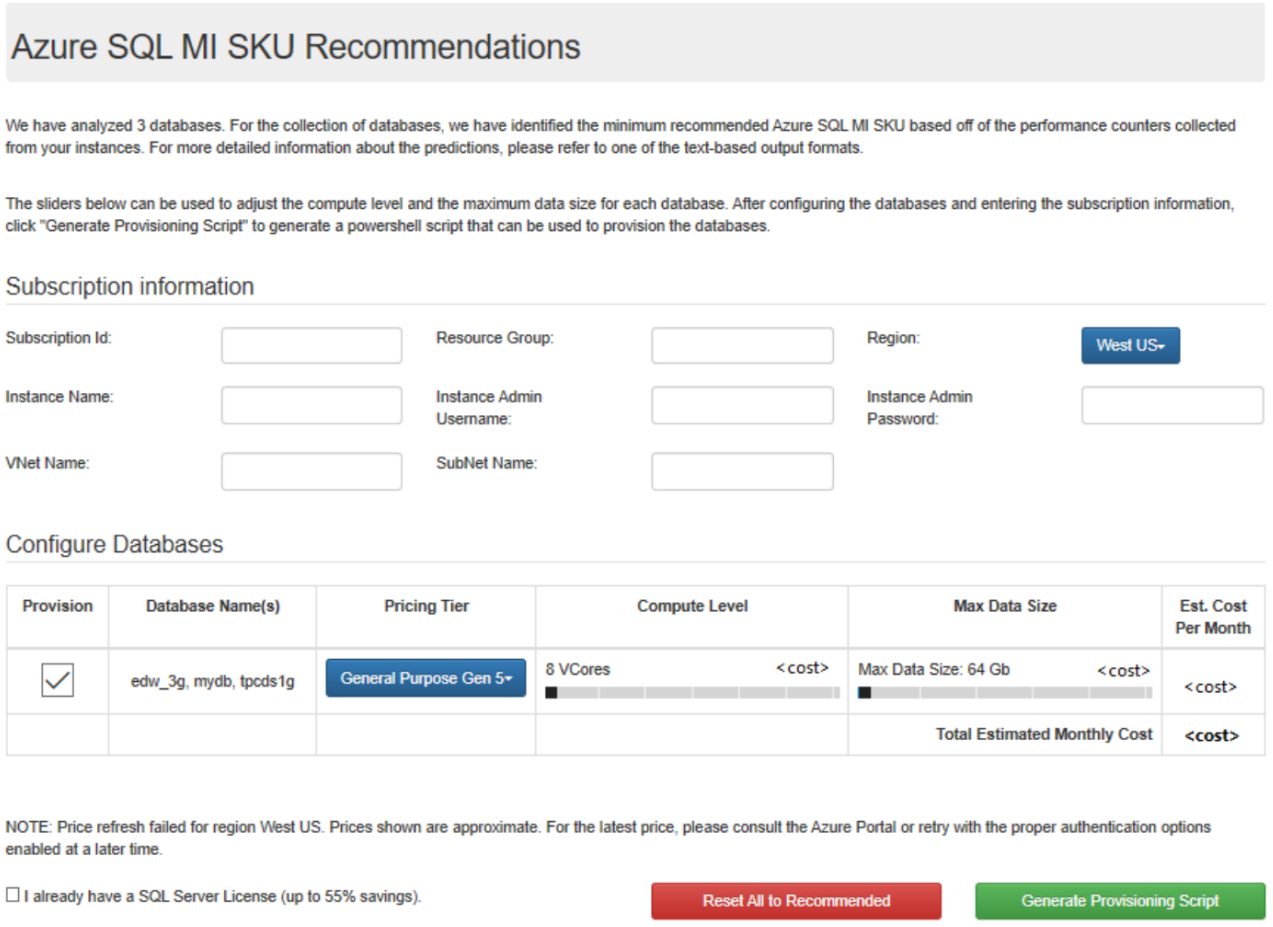
Reset All to Recommended | Generate Provisioning Script

Identify the right Azure SQL Database SKU for your on-premises database (continued)

This feature provides recommendations related to:

- pricing tier
- compute level
- max data size
- estimated cost per month.

Furthermore, it offers the ability to bulk provision single databases and managed instances in Azure for all recommended databases.



Migration Methods – DMA

Create a new migration project.

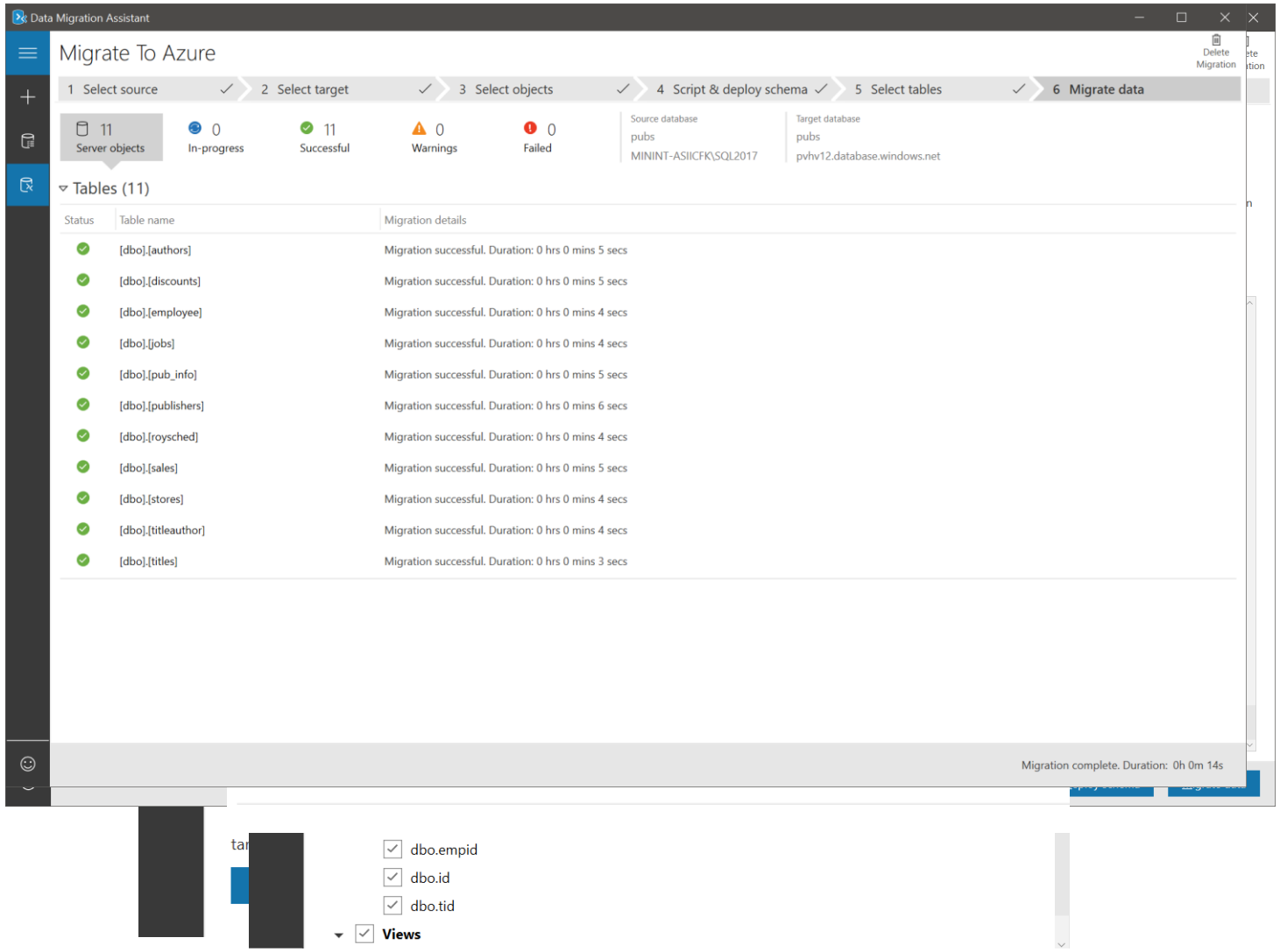
Connect to the Source.

Select the Target.

Select the objects & users to migrate.

Script & Deploy Schema.

Select tables & migrate data.

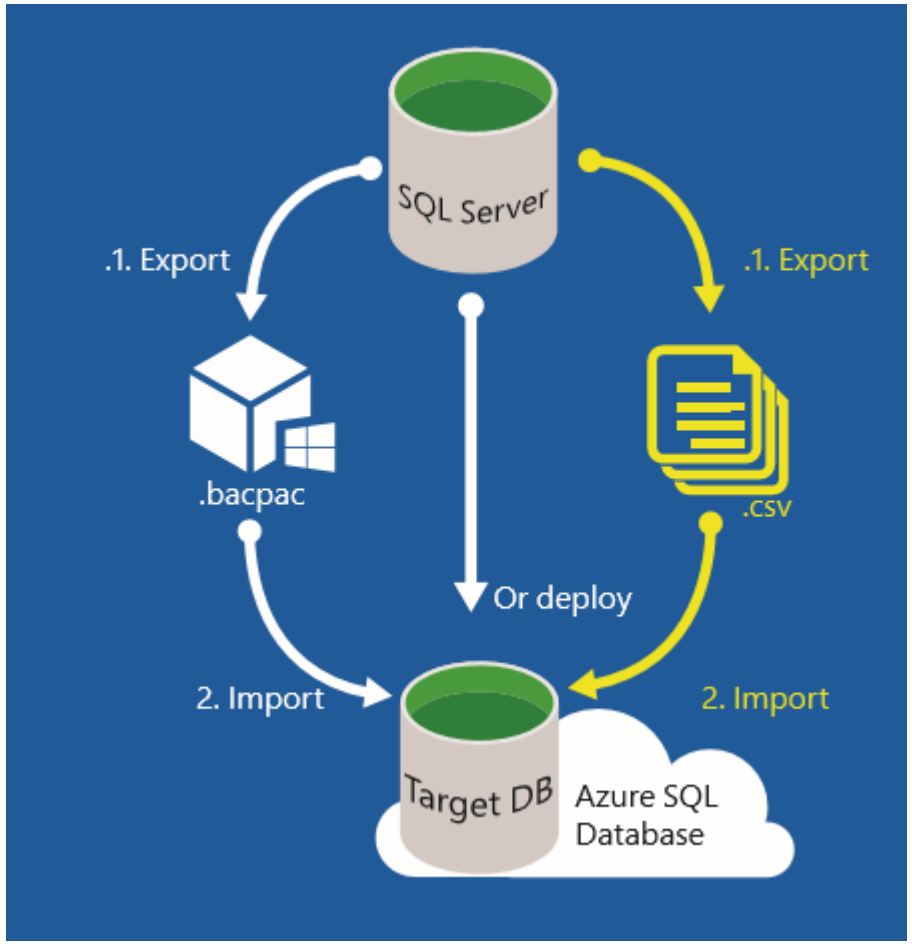


Migration Methods – Export/Import with DACPAC File and BCP

Used for much larger databases to achieve greater parallelization for increased performance.

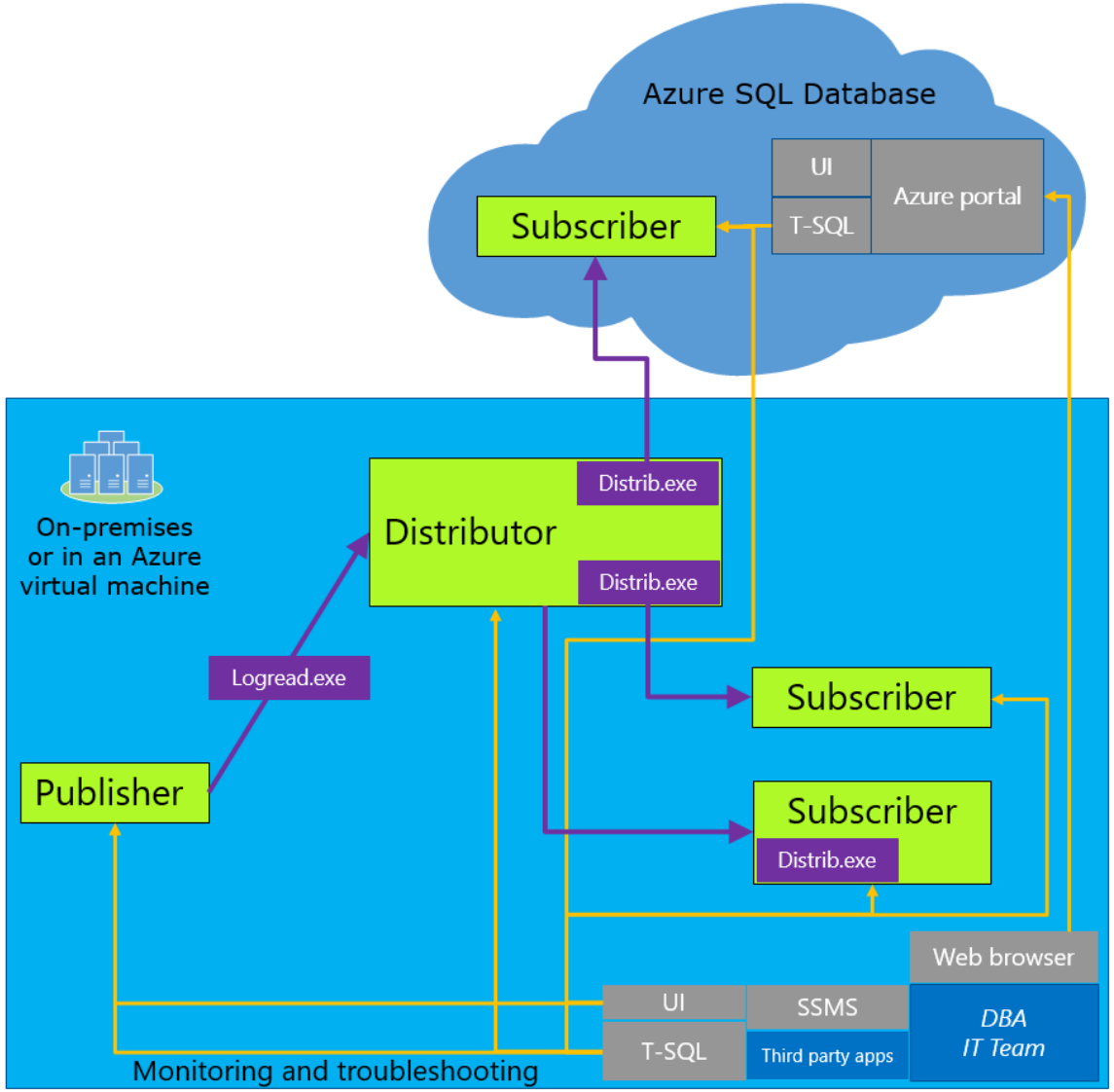
Migrate the schema and the data separately:

- Export the schema only to a DACPAC file.
- Import the schema only from the DACPAC File into SQL Database.
- Use BCP to extract the data into flat files and then parallel load these files into the Azure SQL Database.
- Investigate using SQLPackage.exe

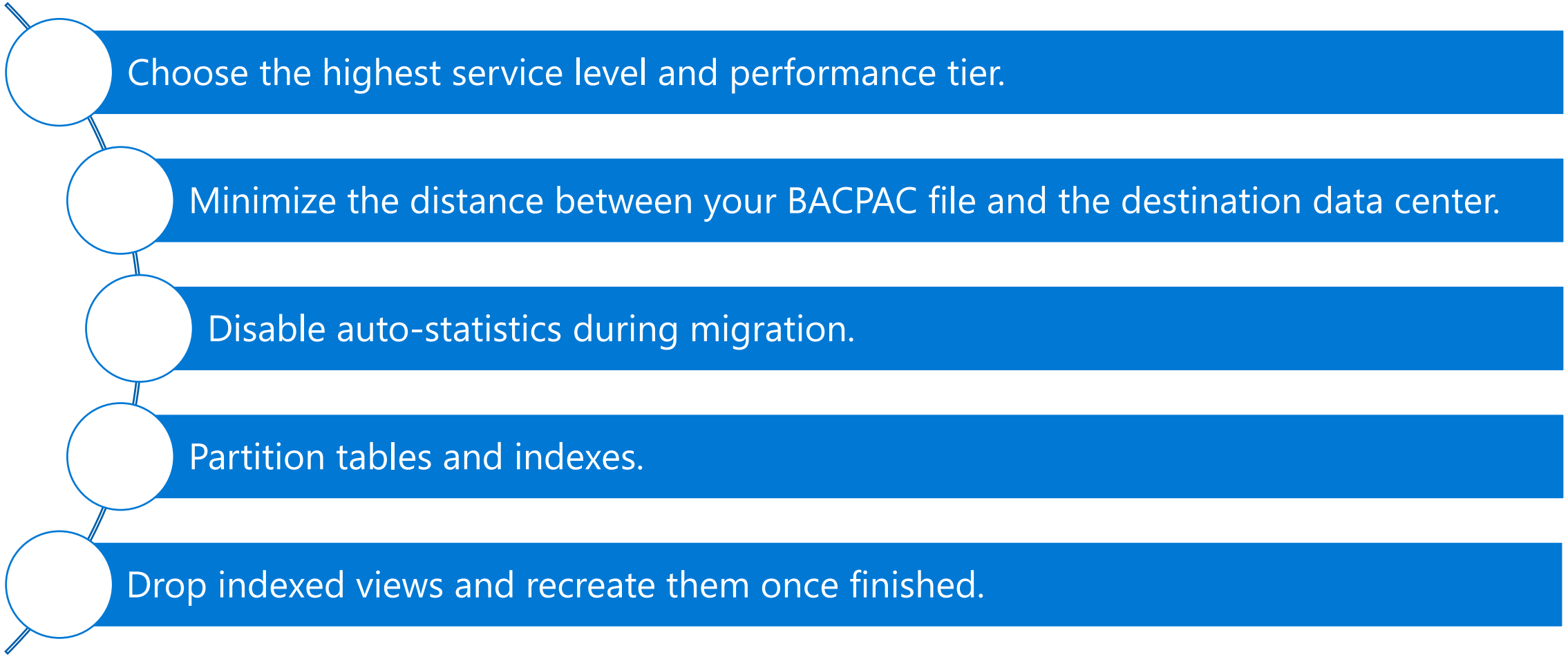


Transactional Replication

- Used will have minimal downtime during migration.
- Configure your Azure SQL Database as a subscriber.
- All changes to your data or schema show up in your Azure SQL Database.
- Synchronization is complete – change the connection string of your application.
- Remove Replication.



Optimizing data transfer performance during migration



Demonstration

Migration Methods – DMA

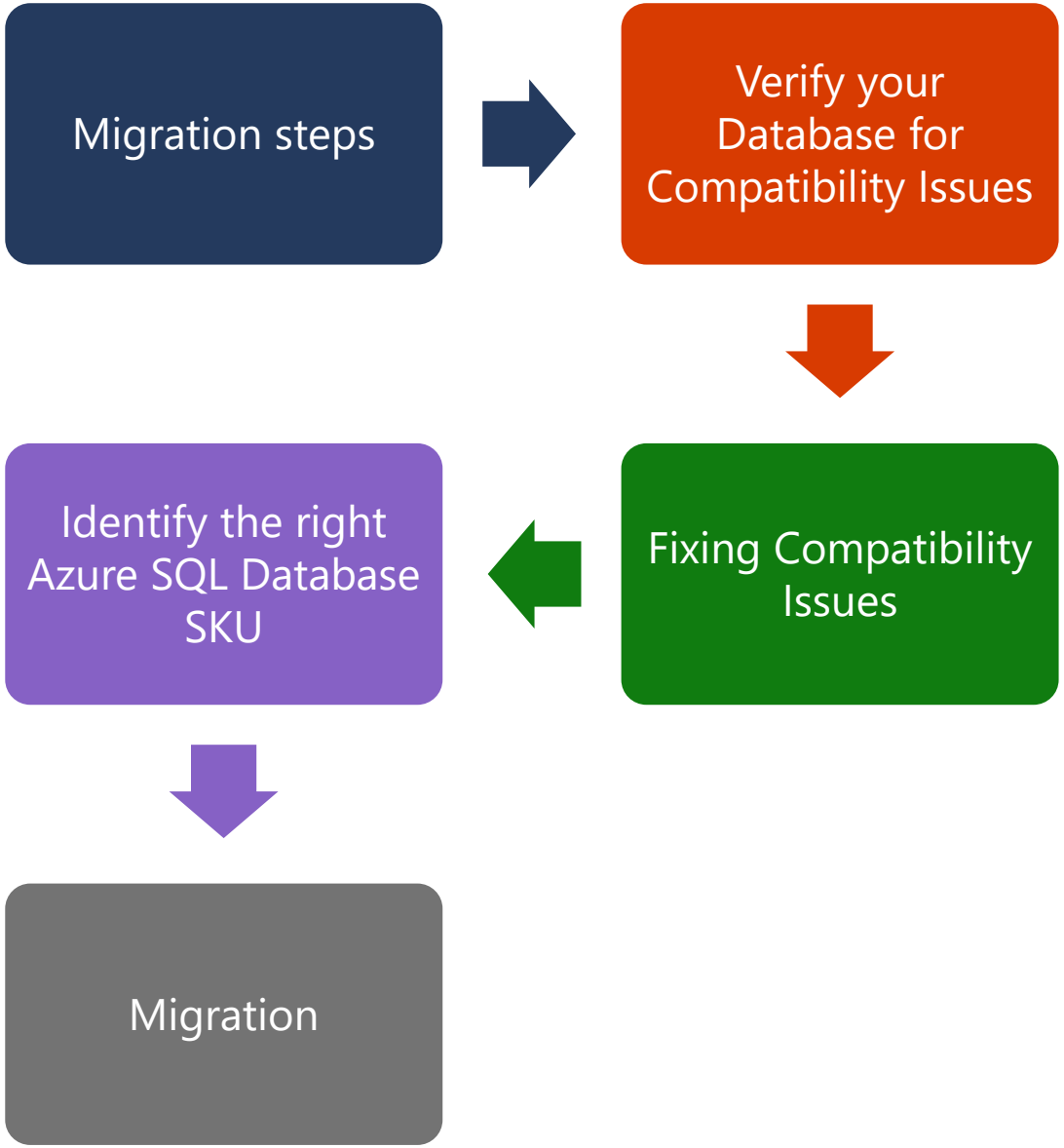
- Migrate your on-premises database with DMA.
- Migrate your on-premises database with DMS Hybrid mode.

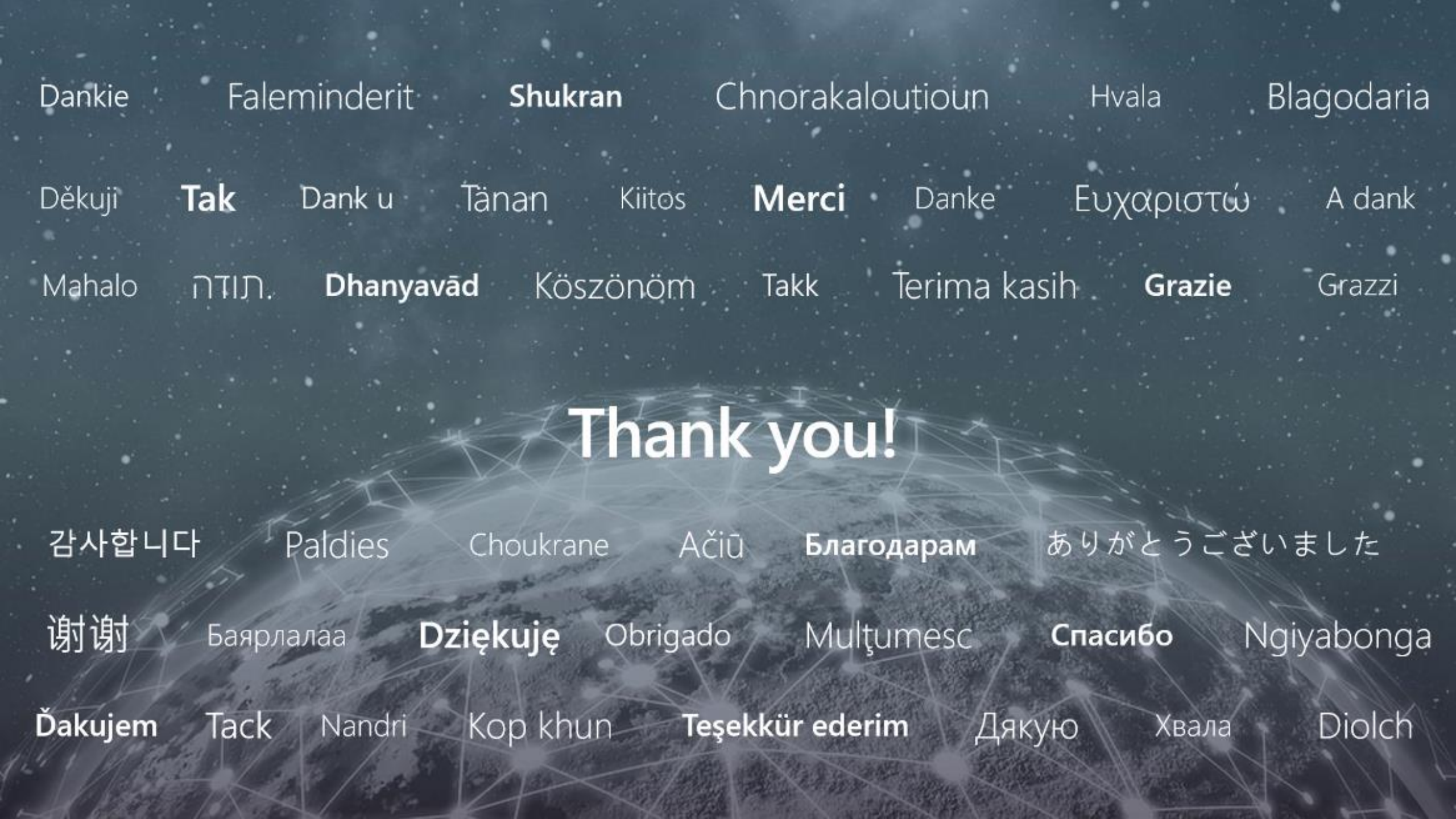
Migrate a Compatible SQL Server Database to Azure SQL Database

- **Exercise 1:** Analyze your SQL Server Database for compatibility issues
- **Exercise 2:** Fix database migration compatibility issues
- **Exercise 3:** Migrate a database to Azure with Data Migration Assistant
- **Exercise 4:** Migrate a database to Azure with SSMS
- **Exercise 5:** Migrate a database to Azure with Transactional Replication

Questions?

Module Summary




Dankie
Faleminderit
Shukran
Chnorakaloutioun
Hvala
Blagodaria
Děkuji
Tak
Dank u
Tänan
Kiitos
Merci
Danke
Ευχαριστώ
A dank
Mahalo
תודה.
Dhanyavād
Köszönöm
Takk
Terima kasih
Grazie
Grazzi
Thank you!
감사합니다
Paldies
Choukrane
Ačiū
Благодарам
ありがとうございました
谢谢
Баярлалаа
Dziękuję
Obrigado
Mulţumesc
Спасибо
Ngiyabonga
Ďakujem
Tack
Nandri
Kop khun
Teşekkür ederim
Дякую
Хвала
Diolch