

# Migrate SQL Server to Azure MI Without Getting a Migraine



# HELLO!

## I am Josephine Bush

10+ years DBA experience  
Microsoft Data Platform MVP

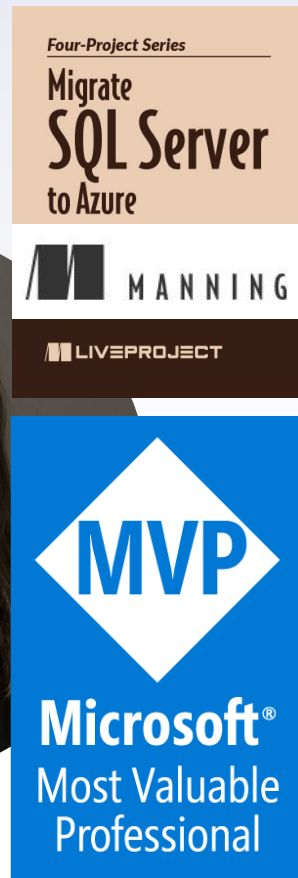
Manning liveProject Author

MBA IT Management

MS Data Analytics

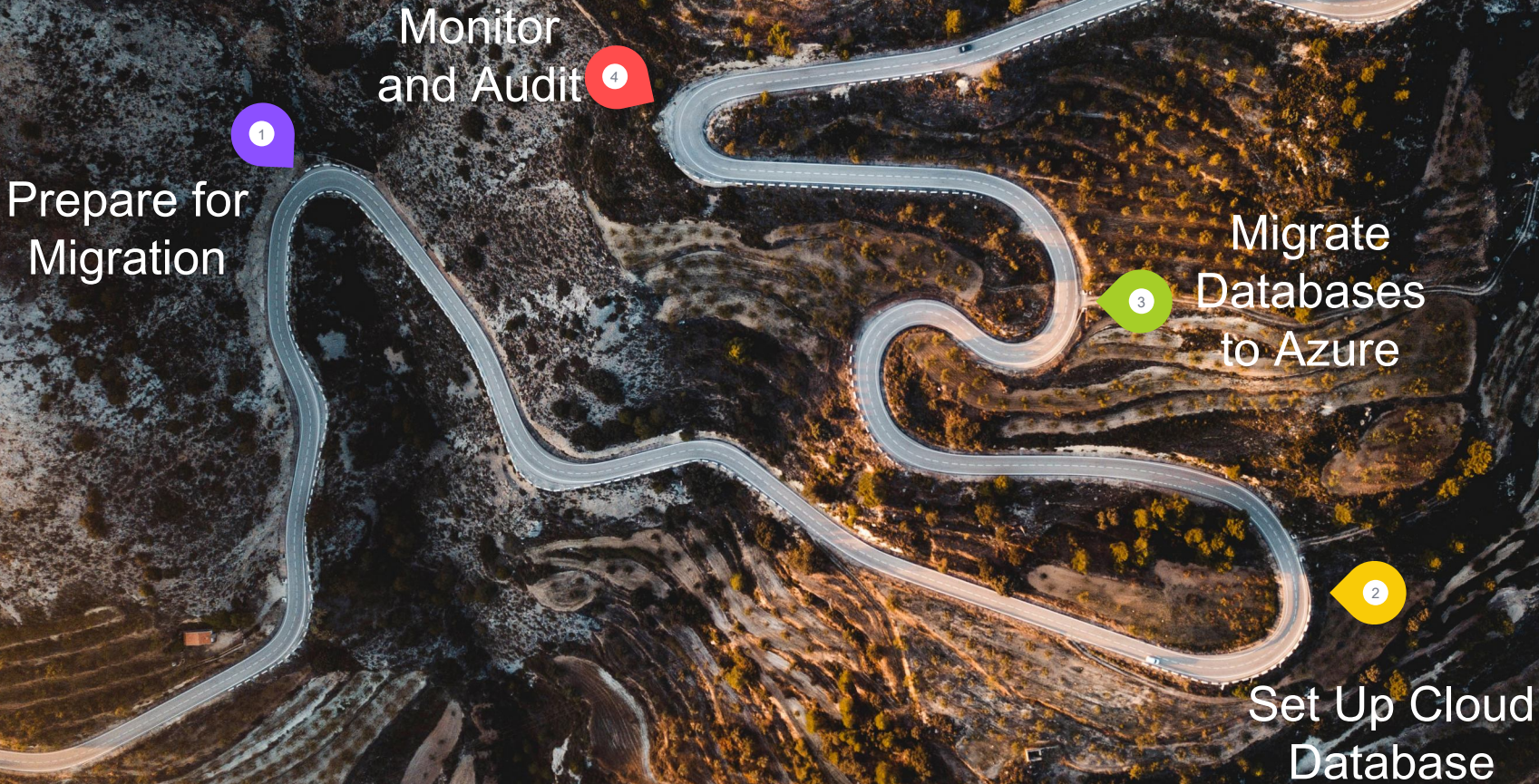


@hellosqlkitty  
sqlkitty.com





# Database Migration Process



# 1

# Prepare for Migration

## ➤ **Analyze/Audit**

Using Microsoft  
Data Migration  
Assistant and SQL  
Server Audit

## ➤ **Clean Up**

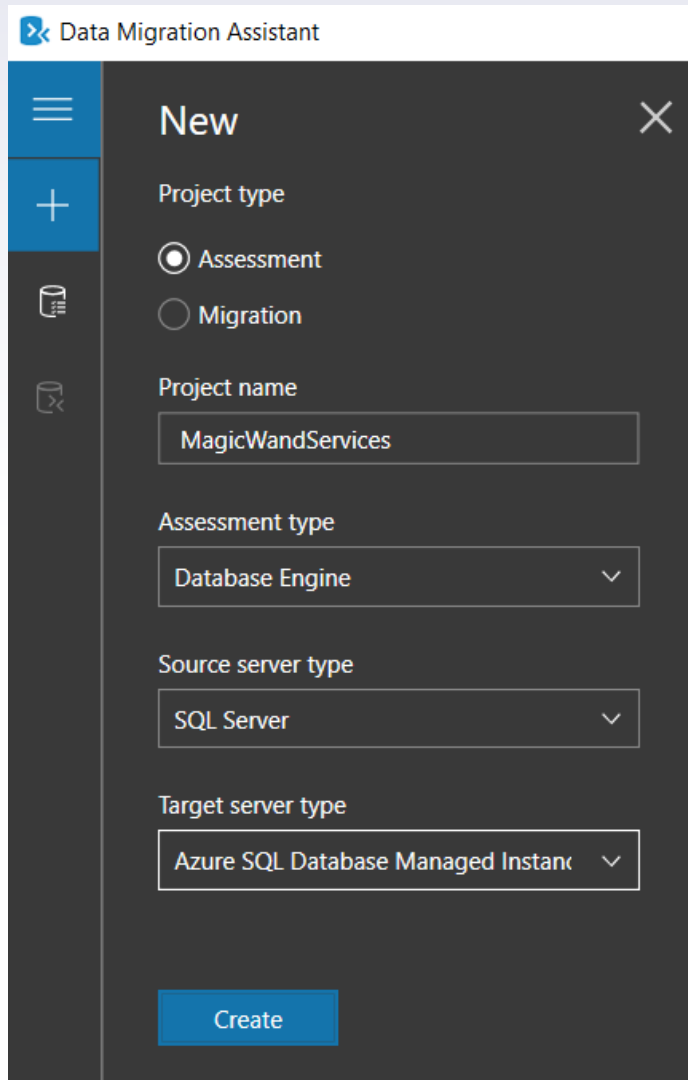
Remove  
unsupported and  
unused objects

## ➤ **Create Checklist**

Catalogue items  
migrating to the  
cloud

# Microsoft Data Migration Assistant (DMA)

Enables you to upgrade to a modern data platform by detecting compatibility issues that can impact database functionality on your new version of SQL Server



The screenshot shows the 'Data Migration Assistant' application window. On the left is a dark sidebar with icons for a menu, a plus sign, a document, and a code editor. The main area is titled 'New' and contains the following fields:

- Project type:** Two radio buttons are present. 'Assessment' is selected, and 'Migration' is unselected.
- Project name:** A text input field containing the value 'MagicWandServices'.
- Assessment type:** A dropdown menu with 'Database Engine' selected.
- Source server type:** A dropdown menu with 'SQL Server' selected.
- Target server type:** A dropdown menu with 'Azure SQL Database Managed Instance' selected.

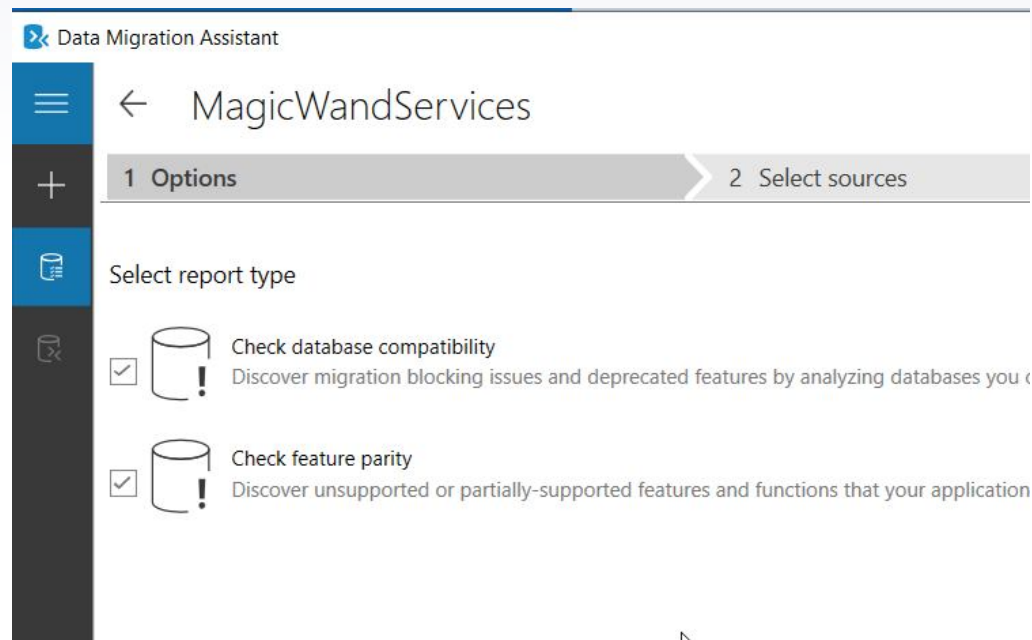
At the bottom of the main area is a blue button labeled 'Create'.



Note that the Microsoft Data Migration Assistant won't warn you about all issues you might encounter. **This is why it's always important to test your migrations from start to finish in a lower environment.**



# DMA Configuration





Azure uses Azure Active Directory, as opposed to Active Directory. This means that when your users are logging in with a domain account, it will change the way they have to log in.





# DMA Results

Assessment

1 Options ✓ 2 Select sources ✓ 3 Review results

☒ SQL Server feature parity  
☐ Compatibility issues

Search server instance name

DESKTOP-15

Target Platform  
Azure SQL Database Managed Instance

DESKTOP-15 / SQL Server 2019

Feature parity (1)

Recommendation | Impacted objects

PowerShell job step is not supported in Azure SQL M

Details | Impacted

Impact

It is a job step that runs a PowerShell script. PowerShell job step is not supported in Azure SQL Managed Instance.

Type | Job step

Compatibility 150 (1) | Compatibility 140 (1)

Issue | Impacted objects

Breaking changes (0)

Behavior changes (1)

Full-Text Search has changed since SQL Server 2008 3

Deprecated features (0)

Information issues (0)

Issue details

Impact

Full-Text Search has changed since SQL Server 2008.

Recommendation

Many full-text search options and settings have changed. Therefore, when you upgrade to SQL Server 2014 or SQL Server 2016 Full-Text Search, some of your settings might need modification. We recommend you to test your applications leveraging the

Impacted objects

Type	Name
FullTextIndex	HumanResourcesJo...
FullTextIndex	Production.Document

Object details

Type: FullTextIndex  
Name: HumanResources.JobCandidate  
Full Text indexes: Full text index on

# ▶ SQL Server Audit Tips

- ▶ I use this on all prod servers to audit DDL and security changes
- ▶ For migrations, I may audit more to see if something is in use or not
- ▶ Be careful to not over audit

# Prepare for Migration

- ▶ Analyze/audit
- ▶ Clean up
- ▶ Create checklist



# 2

## Set Up Cloud Database

### ➤ **Create a Baseline**


Check configuration settings with a SQL script

### ➤ **Create Azure SQL Managed Instance**

Create MI via the Azure portal

### ➤ **Migrate Server-Level Objects**

Using a PowerShell module named dbatools



SQL Managed Instance currently supports deployment only on the following types of subscriptions:

- ▶ Enterprise Agreement (EA)
- ▶ Pay-as-you-go
- ▶ Cloud Service Provider (CSP)
- ▶ Enterprise Dev/Test
- ▶ Pay-as-you-go Dev/Test
- ▶ Subscriptions with monthly Azure credit for Visual Studio subscribers



# Create Configuration Baseline

```
/*setup for querying SQL Server configuration */
DECLARE @tf TABLE (TraceFlag nvarchar(35), status bit, global bit, session bit)
INSERT INTO @tf execute('DBCC TRACESTATUS(-1)');

DECLARE @config TABLE (
    name nvarchar(35),
    default_value sql_variant
)

/*not all of these settings are in all versions of sql server*/
INSERT INTO @config (name, default_value) VALUES
('access check cache bucket count',0),
('access check cache quota',0),
('ADR cleaner retry timeout (min)', 0),
('ADR Preallocation Factor', 0),
('Ad Hoc Distributed Queries',0),
('affinity I/O mask',0),
('affinity64 I/O mask',0),
('affinity mask',0),
('affinity64 mask',0),
('Agent XPs',0), --Changes to 1 when SQL Server Agent is started. Default value is
('allow filesystem enumeration', 0),
('allow polybase export', 0),
('allow updates',0),
('awe enabled',0),
('backup checksum default', 0),
```





**Very important** information  
about costs

A single managed instance  
Standard-series Gen 5 will  
cost approximately \$1/hour  
depending on your  
subscription



# Create Managed Instance

[Home](#) > [SQL managed instances](#) >

## Create Azure SQL Managed Instance

Microsoft

[Basics](#) [Networking](#) [Security](#) [Additional settings](#) [Tags](#) [Review + create](#)

SQL Managed Instance is a fully managed PaaS database service with extensive on-premises SQL Server compatibility and native virtual network security. [Learn more](#)

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ

Resource group \* ⓘ

[Create new](#)

### Managed Instance details

Enter required settings for this instance, including picking a location and configuring the compute and storage resources.

Managed Instance name \*

Region \*

[Not seeing a region?](#)

Compute + storage \* ⓘ

#### General Purpose

Standard-series (Gen 5), 4 vCores, 32 GB storage, Locally-redundant backup storage

[Configure Managed Instance](#)

### Authentication

Select your preferred authentication methods for accessing this Managed Instance. Create a Managed Instance admin login and password to access your Managed Instance with SQL authentication, select only Azure AD authentication [Learn more](#) using an existing Azure AD user, group, or application as Azure AD admin [Learn more](#), or select both SQL and Azure AD authentication.

#### Authentication method

- ☐ Use only Azure Active Directory (Azure AD) authentication
- ☐ Use both SQL and Azure AD authentication
- ☒ Use SQL authentication

#### Managed Instance admin login \*

#### Password \*

#### Confirm password \*

# Sample Costs



## Compute + storage ...

SQL managed instance



Feedback

Select from the latest vCore service tiers available for Azure SQL Managed Instance including General Purpose and Business Critical. [Learn more](#)

Service tier ⓘ

- ☒ General Purpose (4-80 vCores, 32 GB-16 TB storage capacity, Fast storage) - for most production workloads
- ☐ Business Critical (4-80 vCores, 32 GB-4 TB storage capacity, Super fast storage) - for IO-intensive and compute-intensive workloads

### Compute Hardware

Configure compute hardware that will run your Azure SQL Managed Instance. [Learn more](#)

Hardware generation ⓘ

- ☒ Standard-series (Gen 5) - Intel Broadwell, 5,1 GB RAM/vCore
- ☐ Premium-series - Intel Ice Lake, 7 GB RAM/vCore, up to 560 GB
- ☐ Premium-series - memory optimized - Intel Ice Lake, 13,6 GB RAM/vCore, up to 870,4 GB

vCores ⓘ



Storage in GB ⓘ



ⓘ Azure Hybrid Benefit is provided free of charge for dev/test subscriptions.

SQL Server License ⓘ

Pay-as-you-go Azure Hybrid Benefit Hybrid failover rights

☒ I confirm that I have a SQL Server License with Software Assurance to apply this Azure Hybrid Benefit for SQL Server.

### Cost summary

#### Standard-series (Gen 5) General Purpose

Cost per vCore (in USD)	111.12
vCores selected	x 4
Azure Hybrid Benefit discount (in USD)	- 0.00

Cost per GB (in USD)	0.12
Max storage selected (in GB)	x 32
32 GB storage included (in USD)	- 3.84

ESTIMATED COST / MONTH **444.48 USD**

#### Additional charge per usage

See [pricing details](#) for more detail.



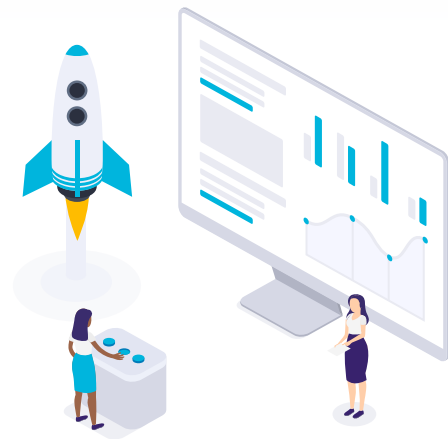
**Azure SQL Managed Instance requires use of a dedicated vnet.**

**Make sure to check with your networking team to make sure you don't configure this incorrectly.**



# Tips for Managed Instance

- ▶ It can take up to 6 hours to create. 30 min create is in preview, but not available in all subscriptions.  
Deployment of the first instance in the subnet takes longer than deploying into a subnet with existing managed instances.
- ▶ Start/stop is in preview, but not available in all subscriptions
- ▶ For testing, you can enable public endpoint, but don't do this for production
- ▶ To connect via SSMS with public endpoint:  
yourminame.**public**.xxxxxxxxxxxxx.database.windows.net, 3342



# Migrate Server-Level Objects

- ▶ Using dbatools
  - ▶ Install module in PowerShell

```
1  $scred = Get-Credential sa
2  $dcred = Get-Credential miadmin
3  $params = @{
4      Source = "your_sql_server_name"
5      Destination = "copy_your_mi_name_from_azure,3342"
6      SourceSqlCredential = $scred
7      DestinationSqlCredential = $dcred
8  }
9
10 Start-DbaMigration @params -Force -Exclude Databases -Verbose
```



# Set Up Cloud Database

- ▶ Create baseline
- ▶ Create managed instance
- ▶ Migrate server-level objects



# 3

# Migrate Databases To Azure

➤ **Set Up  
Database  
Migration  
Service (DMS)**

Using Azure portal  
and Azure Data  
Studio (ADS)

➤ **Perform Online  
Migration**

Using Azure Data  
Studio

➤ **Verify  
Migration**

Check database  
configuration  
settings with a  
SQL script

# Before You Set Up DMS

- ▶ Configure a VPN or ExpressRoute
- ▶ Register Microsoft.DataMigration
- ▶ Create a storage account
- ▶ Take a full backup of your on-premises database WITH CHECKSUM



# VPN Setup

^ Connect to managed instance using Point-to-Site VPN connection

## To setup environment using Azure Portal

Read documentation Quickstart: [Configure a point-to-site connection to an Azure SQL Database managed instance from on-premises](#)

## To setup environment using PowerShell

1. Create VPN Gateway, configure it for Point-to-Site access and attach it to the virtual network. You could run PowerShell code shown below from your computer to automate this step.

```
$scriptUrlBase = 'https://raw.githubusercontent.com/Microsoft/sql-server-samples/master/samples/manage/azure-sql-db-managed-instance/attach-vpn-gateway'

$parameters = @{
    subscriptionId = '244eb28e-a9b8-42d4-9260-c0c553ae92e1'
    environmentName = 'AzureCloud'
    resourceGroupName = 'azuremi'
    virtualNetworkName = 'vnet-josephinemi'
    certificateNamePrefix = 'cert4josephinemi'
}

Invoke-Command -ScriptBlock ([Scriptblock]::Create((iwr ($scriptUrlBase+'/attachVPNGateway.ps1?t=' + [DateTime]::Now.Ticks)).Content)) -ArgumentList $parameters, $scriptUrlBase
```



2. [Install the VPN client and connect to virtual network](#)
3. Run SSMS from the virtual machine and connect to Managed Instance. Below are the parameters needed to establish a connection

Login

miadmin

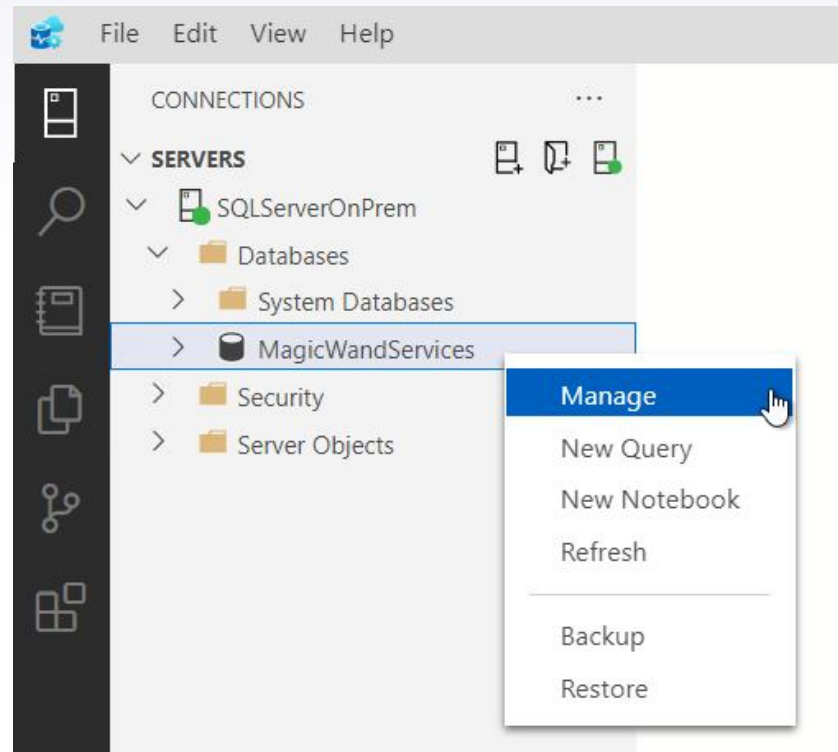
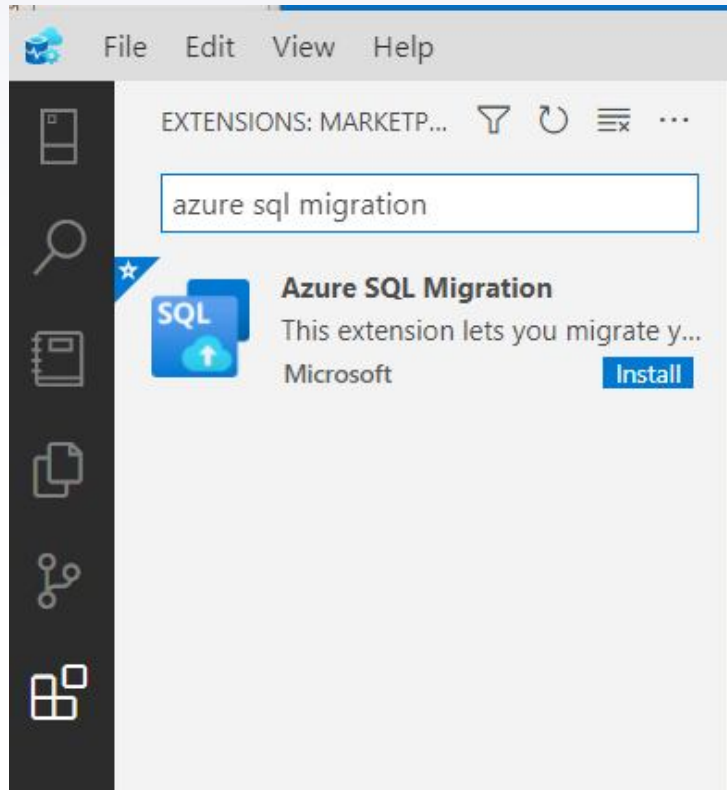


Server name

josephinemi.a57eaffdb3ca.database.windows.net



# Setting up DMS in ADS



# Set Up Migration in ADS

The screenshot displays the Azure Data Studio (ADS) interface. On the left, the 'CONNECTIONS' pane shows a tree view under 'SERVERS' with 'SQLServerOnPrem' expanded, containing 'Databases', 'System Databases', 'MagicWandServices' (selected), 'Security', and 'Server Objects'. The main pane shows the breadcrumb 'Home > SQLServerOnPrem > MagicWandServices'. Below this is a 'General' tab with an 'Azure SQL Migration' button. The right pane shows the 'Migrations' tab with a 'New migration' button and a 'Feedback' button. The main content area is titled 'Azure SQL Migration' and includes a description: 'Determine the migration readiness of your SQL Server instances, identify a Server instance to Azure SQL Managed Instance, SQL Server on Azure Virtual Machine, or Azure SQL Database.' Below this is a 'Migrate to Azure SQL' button with a hand cursor, and a list of 'Things you need to know' including: 'An Azure account (n', 'A source SQL Server machine running in t', 'An Azure SQL Mana database(s) to.', and 'Your database backu for Azure SQL Datab'. A 'Learn more' link is at the bottom right.

CONNECTIONS

SERVERS

- SQLServerOnPrem
  - Databases
  - System Databases
  - MagicWandServices**
  - Security
  - Server Objects

Home > SQLServerOnPrem > MagicWandServices

Dashboard Migrations

+ New migration New support request Feedback

## Azure SQL Migration

Determine the migration readiness of your SQL Server instances, identify a Server instance to Azure SQL Managed Instance, SQL Server on Azure Virtual Machine, or Azure SQL Database.

**Migrate to Azure SQL**

Migrate a SQL Server instance to Azure SQL.

Things you need to know

- An Azure account (n
- A source SQL Server machine running in t
- An Azure SQL Mana database(s) to.
- Your database backu for Azure SQL Datab

[Learn more](#)



# Steps to Set Up Migration

- ▶ Step 1: Databases for assessment
- ▶ Step 2: Assessment results and recommendations
- ▶ Step 3: Azure SQL target setup
- ▶ Step 4: Migration Mode
- ▶ Step 5: Data source configuration
- ▶ Step 6: Azure Database Migration Service – includes installing self hosted integration runtime
- ▶ Step 7: Summary



# Migration Status in ADS

The screenshot displays the Azure Data Studio (ADS) interface. At the top, there's a breadcrumb navigation bar with 'Welcome' and 'SQLServerOnPrem'. Below it, a sidebar on the left contains navigation links: 'Home', 'Databases', 'Administration', 'SQL Agent', 'General', and 'Azure SQL Migration' (which is highlighted). The main content area shows the 'Database migration status' section. It includes a 'Refresh' button and a 'DMSMI (change)' link for the 'Azure Database Migration Service'. Below this, there are two summary cards: 'Database migrations in progress' with a count of 1, and 'Database migrations completed' with a count of 0. A 'Learn more' link is also visible in the top right of the main content area.

Home > SQLServerOnPrem

Database migration status [Refresh](#)

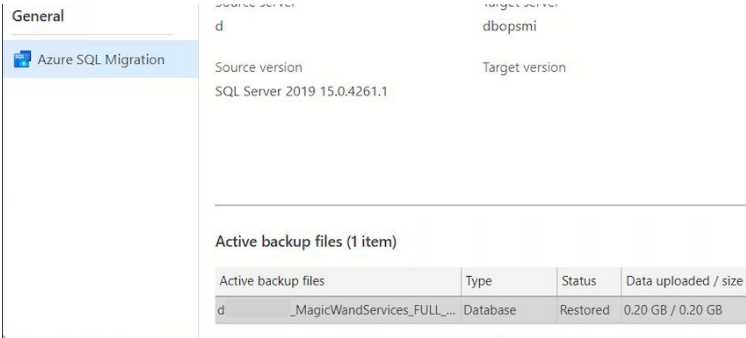
**DMSMI (change)**  
Azure Database Migration Service

	Database migrations in progress	1
	Database migrations completed	0

[Learn more](#)

# Begin Online Migration

- ▶ Stop applications and users from using the on-premises database
- ▶ Take a final log backup WITH CHECKSUM
- ▶ Verify final backup is restored
- ▶ Don't take your on-prem db offline

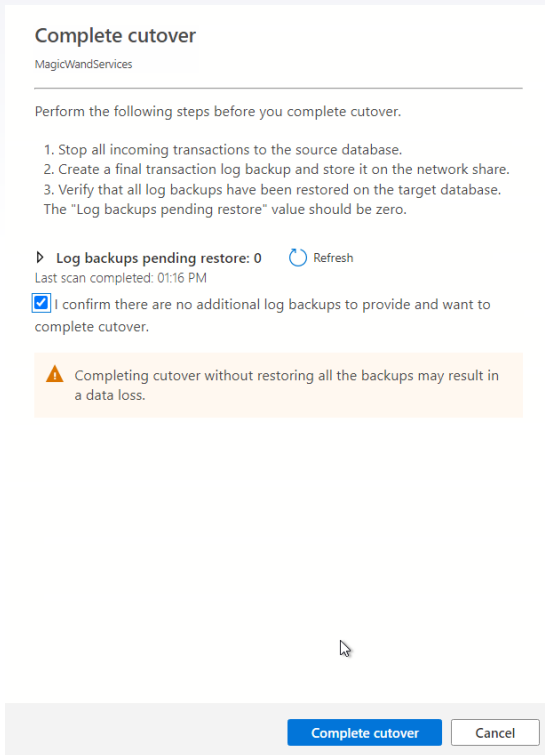
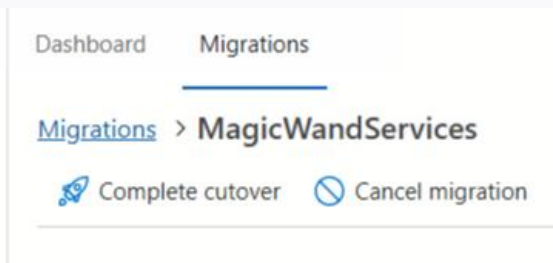


The screenshot displays the 'General' tab of the Azure SQL Migration tool. It shows the source server as 'd' and the target server as 'dbopsmi'. The source version is 'SQL Server 2019 15.0.4261.1'. Below this, a table titled 'Active backup files (1 item)' lists the backup files. The table has four columns: 'Active backup files', 'Type', 'Status', and 'Data uploaded / size'. The first row shows a file named '\_MagicWandServices\_FULL\_...' of type 'Database' with a status of 'Restored' and a size of '0.20 GB / 0.20 GB'.

Active backup files	Type	Status	Data uploaded / size
d _MagicWandServices_FULL_...	Database	Restored	0.20 GB / 0.20 GB

# Complete Online Migration

## Click Complete Cutover



# Verify Migration & Migrate DB Settings

```
1 DECLARE @dbname varchar(20);
2 SET @dbname = 'MagicWandServices';
3
4 DECLARE @config TABLE (
5     name nvarchar(35),
6     value sql_variant
7 )
8
9 INSERT INTO @config (name, value)
10 SELECT name, CASE
11     WHEN value = 1 then 'ON'
12     WHEN value = 0 then 'OFF'
13     ELSE value
14     END AS value
15 FROM sys.database_scoped_configurations
16 WHERE name <> 'MAXDOP'
17
18 INSERT INTO @config (name, value)
19 SELECT name, value
20 FROM sys.database_scoped_configurations
21 WHERE name = 'MAXDOP'
22
23
24 SELECT name = CONCAT('DBCONFIG:',dsc.name), dsc.value, is_value_default,
25     'USE ' + @dbname + '; ALTER DATABASE SCOPED CONFIGURATION SET ' + dsc.name + '=' + convert(nvarchar(35), c.value) + ';' as SQLscript
26 FROM sys.database_scoped_configurations dsc
27 INNER JOIN @config c
28 ON c.name = dsc.name
29 WHERE is_value_default <> dsc.value
30
```

# Migrate Databases to Azure

- ▶ Set up DMS
- ▶ Perform online migration
- ▶ Verify migration





# 4

## Monitor and Audit

### ➤ **Configure Auditing**

Using SQL Server Audit

### ➤ **Configure Alerting**

Monitor and alert on things such as CPU usage

### ➤ **Security Check**

Make sure your MI complies with CIS benchmarks

# ▶ SQL Server Audit with MI tips

- ▶ Create a storage account to hold the audit files
- ▶ Use URL for audit destination
- ▶ Query that URL to get audit data

# Monitoring/Alerting Tips

- ▶ Make sure to monitor and alert on things like
  - ▶ CPU usage
  - ▶ Memory usage
  - ▶ Storage space used
- ▶ Create an alert rule and alert action group

[Home](#) > [dbopsmi](#) | [Alerts](#) >

## Alert rules ...

×

[+ Create](#) [Columns](#) [Refresh](#) [Export to CSV](#) [Open query](#) | [Delete](#) [Enable](#) [Disable](#)

Target resource type : all

Target scope : dbopsmi

Subscription : all

Signal type : all

Severity : all

Status : Enabled

Showing 1 to 1 of 1 Alert rules.

No grouping ▼

Name <span>↑↓</span>	Condition	Severity <span>↑↓</span>	Target scope	Target resource type	Signal type <span>↑↓</span>	Status <span>↑↓</span>	
<input type="checkbox"/> MICPUPERCENT90	avg_cpu_percent > 90	2 - Warning	dbopsmi	SQL managed instance	Metrics	✓ Enabled	...

# Security Check Tips

- ▶ Use the PowerShell module dbchecks
  - ▶ CIS benchmarks help you ensure your managed instance is as secure as possible.
  - ▶ Must install Pester module

```
1 $Date = Get-Date -Format "yyyy-MM-dd"
2 Invoke-DbcCheck -Show Fails -SqlInstance "dbopsmi.public.rand@m1234.database.windows.net,3342" -SqlCredential (Get-Credential yourmiadmin) `
3 -ExcludeCheck Backup, HADR, Domain, LogShipping, AgentServiceAccount, IdentityUsage, FutureFileGrowth, FKCKTrusted, GuestUserConnect, `
4 ValidDatabaseOwner, InvalidDatabaseOwner, InstanceConnection, SqlEngineServiceAccount, TempDbConfiguration, BackupPathAccess, DefaultFilePath, `
5 DAC, MaxMemory, OrphanedFile, ServerNameMatch, MemoryDump, SupportedBuild, DefaultBackupCompression, ErrorLog, CrossDBOwnershipChaining, DefaultTrace, `
6 OLEAutomationProceduresDisabled, RemoteAccessDisabled, SystemFull, UserFull, UserDiff, Userrog `
7 -Passthru | Convert-DbcResult | Set-DbcFile -FilePath C:\windows\temp\ -FileName DbcCheck_$Date -FileType csv
```

# Monitor and Alert

- ▶ Configure auditing and alerting
- ▶ Security check



# ▶ Migrating to MI

## **Prepare to Migrate**

Analyze and audit your on-premises database server, clean up unused and unsupported objects, and prepare a checklist



## **Set Up Cloud Database**

Create a baseline of your on-premises db server, create MI, and migrate server-level objects



## **Migrate Databases to Azure**

Create DMS, do online migration, and verify migration



## **Monitor and Alert**

Configure auditing, alerting, and do a security check

# 100%

Total success!



Want the whole picture?

Get the Manning liveProject Series.

Four-Project Series

Migrate  
**SQL Server**  
to Azure

 LIVEPROJECT

 MANNING

# ► Resources

- ▶ [Manning liveProject Series](#)
- ▶ [Migrate Without a Migraine GitHub Repository](#)
- ▶ [Managed Instance Features](#)
- ▶ [SQL Server Audit Presentations and Code](#)
- ▶ [dbatools PowerShell module](#)
- ▶ [dbachecks PowerShell module](#)



# ► Credits

Special thanks to all the people who made and released these awesome resources for free:

- Presentation template by [SlidesCarnival](#)
- Illustrations by [Sergei Tikhonov](#)
- Photographs by [Unsplash](#)

# THANKS!

## Any questions?

You can find me at:

- ▶ [@hellosqlkitty](https://twitter.com/hellosqlkitty)
- ▶ [hellosqlkitty@gmail.com](mailto:hellosqlkitty@gmail.com)
- ▶ <https://sqlkitty.com/>

