

# Migrate SQL Server to Azure MI Without Getting a Migraine



**EIGHTKB**

# HELLO!

## I am Josephine Bush

10+ years DBA experience  
Microsoft Data Platform MVP  
Manning liveProject Author  
MBA IT Management  
MS Data Analytics  
[@hellosqlkitty](https://twitter.com/hellosqlkitty)  
[sqlkitty.com](http://sqlkitty.com)



EIGHTKB

# Database Migration Process

Prepare for Migration

1

Monitor  
and Audit

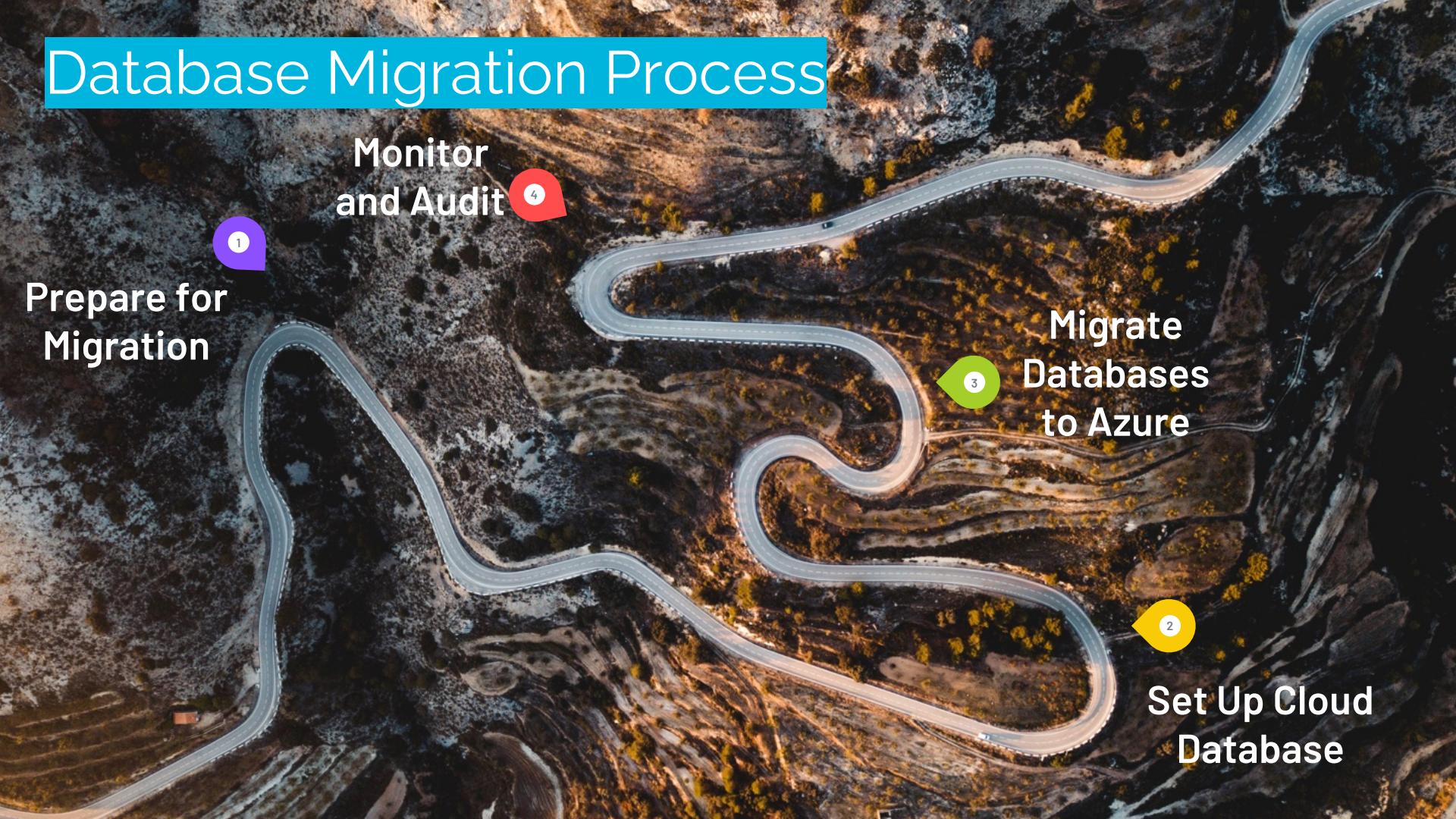
4

Migrate  
Databases  
to Azure

3

2

Set Up Cloud  
Database

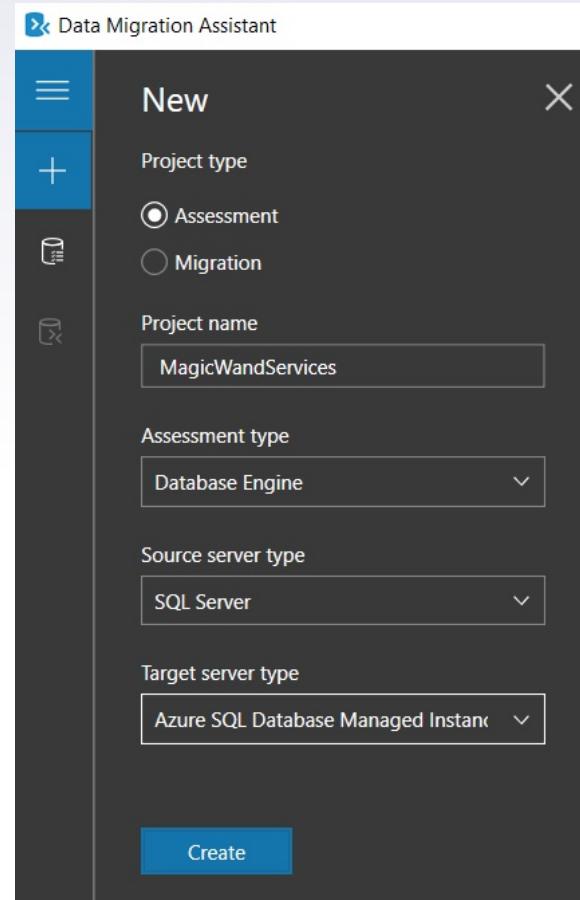


# Prepare for Migration

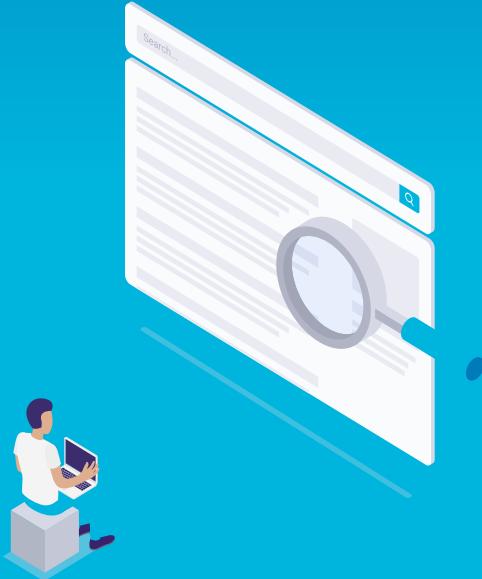
- **Analyze/Audit**  
Using Microsoft  
Data Migration  
Assistant and SQL  
Server Audit
- **Clean Up**  
Remove  
unsupported and  
unused objects
- **Create  
Checklist**  
Catalogue items  
migrating to the  
cloud

# 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



“ 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.

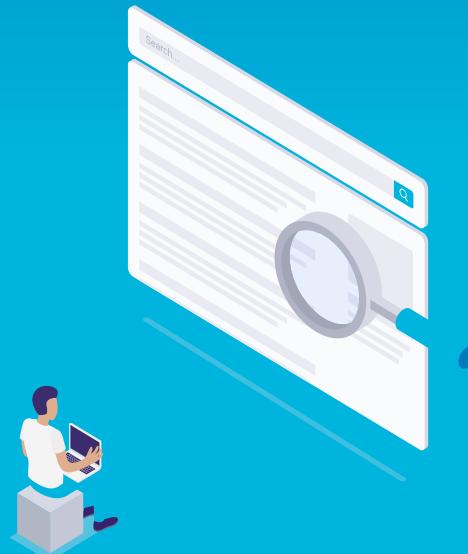


# DMA Configuration

The screenshot shows the DMA configuration interface. At the top, there's a navigation bar with a back arrow, the title "MagicWandServices", and two tabs: "1 Options" (which is selected) and "2 Select sources". On the left, a sidebar has icons for "Data Migration Assistant", "New migration", "Recent migrations", and "Select report type". The main area is titled "Select report type" and contains two options, each with a checkbox and a database icon:

- Check database compatibility**  
Discover migration blocking issues and deprecated features by analyzing databases you can access.
- Check feature parity**  
Discover unsupported or partially-supported features and functions that your application uses.

“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

Assess

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 sup...	1

**Unsupported features (1)**

PowerShell job step is not sup...	1
-----------------------------------	---

**Compatibility 150 (1)**  Compatibility 140 (1)

**Issue**  **Impacted objects**

Breaking changes (0)

Behavior changes (1)  
Full-Text Search has changed si... 3

Deprecated features (0)

Information issues (0)

**Full-Text Search has changed since SQL Server 2008**

**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	HumanResources.Jo...
FullTextIndex	Production.Document

**Object details**

Type: FullTextIndex  
Name:  
HumanResources.JobCandidate

EIGHTKB

# 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



# Set Up Cloud Database

- **Create a Baseline**

Check configuration settings with a SQL script

- **Create Azure SQL Managed Instance**

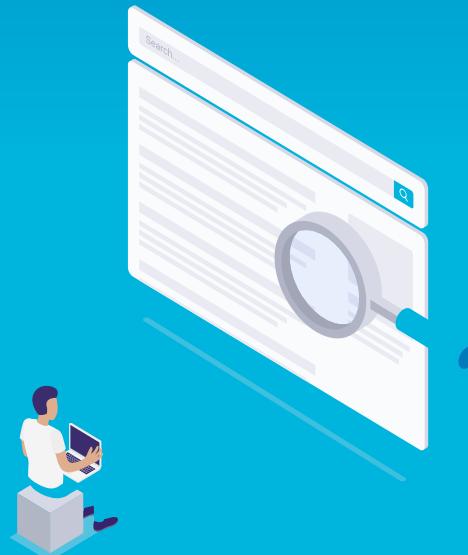
Create MI via the Azure portal

- **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

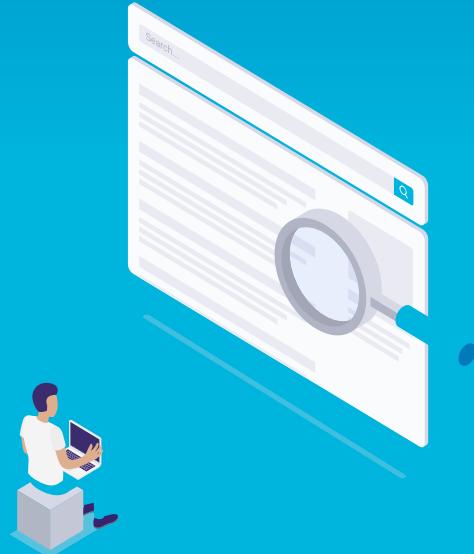
```
/*setup for querying SQL Server configuration */
DECLARE @tf TABLE (TraceFlag nvarchar(35), status bit,global bit, session bit)
INSERT INTO @tf execute('DBCC TRACESTATUS(-1)');

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 \* [Visual Studio Professional Subscription](#)  
Resource group \* [\(New\) AzureSQLMI\\_RG](#) [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 \* [josephinemi](#)  
Region \* [\(US\) East US 2](#)  
[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.

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

Managed Instance admin login \* [miadmin](#)

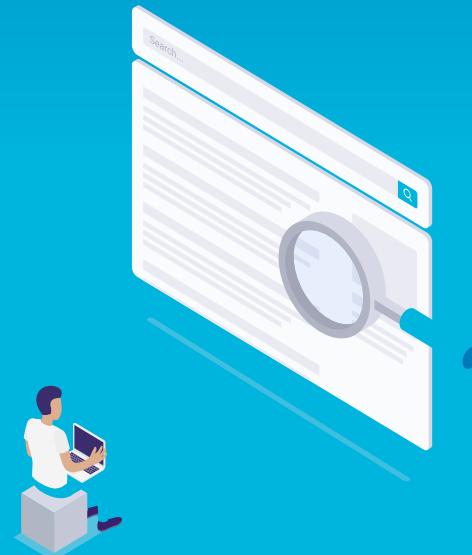
Password \* [\\*\\*\\*\\*\\*](#)

Confirm password \* [\\*\\*\\*\\*\\*](#)

EIGHTKB

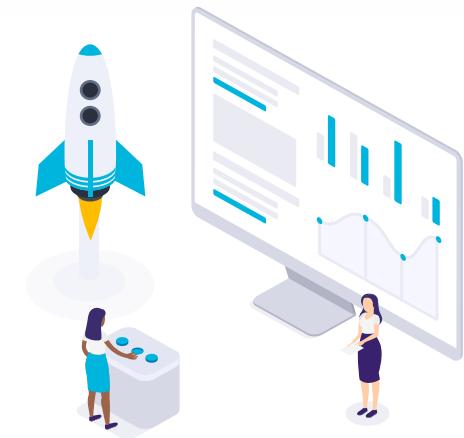
“ Azure SQL Managed Instance requires use of a dedicated vnet.

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.
- ▶ 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.publicxxxxxxxxxxxx.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



# 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**



# Setting up DMS in ADS

The image shows two screenshots of the Azure Data Studio application.

**Left Screenshot:** The Extensions page. The search bar at the top contains "azure sql migration". A card for the "Azure SQL Migration" extension by Microsoft is displayed, showing a blue icon with a white "SQL" and an upward arrow, the extension name, a brief description, and an "Install" button. The sidebar on the left has icons for Home, Search, Database, Notebook, Query, and Object Explorer.

**Right Screenshot:** The Object Explorer page. The sidebar shows "CONNECTIONS" and "SERVERS". Under SERVERS, "SQLServerOnPrem" is expanded, showing "Databases", "System Databases", "MagicWandServices" (which is selected and highlighted in blue), "Security", and "Server Objects". A context menu is open over the "MagicWandServices" node, with "Manage" highlighted in blue and a cursor pointing at it. Other options in the menu include "New Query", "New Notebook", "Refresh", a separator line, "Backup", and "Restore".

EIGHTKB

# Set Up Migration in ADS

The screenshot shows the Azure Data Studio interface. On the left, the 'CONNECTIONS' sidebar lists 'SQLServerOnPrem' as the active connection. Under 'SERVERS', 'SQLServerOnPrem' is expanded, showing 'Databases', 'System Databases', 'MagicWandServices' (which is selected), 'Security', and 'Server Objects'. The main pane displays the 'Dashboard' for 'MagicWandServices'. At the top of the dashboard, there are links for 'New migration', 'New support request', and 'Feedback'. Below this, the title 'Azure SQL Migration' is displayed, followed by the sub-instruction: 'Determine the migration readiness of your SQL Server instances, identify a Server instance to Azure SQL Managed Instance, SQL Server on Azure Virt'. A call-to-action button labeled 'Migrate to Azure SQL' is present, along with a list of requirements: 'Things you need': An Azure account (n, A source SQL Server machine running in I, An Azure Managed database(s) to, Your database backu for Azure SQL Datab'. A 'Learn more' link is also provided.

EIGHTKB

# 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 – Online or offline
- ▶ Step 5: Data source configuration – along with location of db backups on the network
- ▶ Step 6: Azure Database Migration Service – includes installing self hosted integration runtime
- ▶ Step 7: Summary



# DMS Setup: Step 1



## Step 1: Databases for assessment

Select the databases that you want to assess for migration to Azure SQL.

Search

1/1 databases selected

<input checked="" type="checkbox"/> Database	Status	Size (MB)
<input checked="" type="checkbox"/>  MagicWandServices	ONLINE	336

EIGHTKB

# DMS Setup: Step 2

1

## Step 2: Assessment results and recommendations

- ✓ We have completed the assessment of your SQL Server instance

Based on the assessment results, all 1 of your databases in an online state can be migrated to Azure SQL.

⟳ Refresh assessment

2

3

## Choose your Azure SQL target

4

5

6

7



Azure SQL Managed  
Instance

ASSESSMENT RESULTS

1/1 databases can be  
migrated without issues

RECOMMENDED  
CONFIGURATION

Azure recommendation is  
not available. Click "Get  
Azure recommendation"  
button below



SQL Server on Azure  
Virtual Machine

ASSESSMENT RESULTS

1/1 databases can be  
migrated without issues

RECOMMENDED  
CONFIGURATION

Azure recommendation is  
not available. Click "Get  
Azure recommendation"  
button below



Azure SQL Database  
(PREVIEW)

ASSESSMENT RESULTS

1/1 databases can be  
migrated without issues

RECOMMENDED  
CONFIGURATION

Azure recommendation is  
not available. Click "Get  
Azure recommendation"  
button below

EIGHTKB

# DMS Setup: Step 3

Step 3: Azure SQL target

Select an Azure account and your target Azure SQL Managed Instance.

Azure account \*  
Bush, Josephine

[Link account](#)

Azure AD tenant  
Dev

Subscription \* ⓘ  
Dev

Location \* ⓘ  
East US 2

Resource group \* ⓘ  
dbops

Azure SQL Managed Instance \* ⓘ  
dbopsml



EIGHTKB

# DMS Setup: Steps 4 & 5

- ▶ Step 4: Migration Mode – Offline or Online
- ▶ Step 5: Data source configuration



# DMS Setup: Step 6

The screenshot shows the 'Step 6: Azure Database Migration Service' configuration screen. On the left, a vertical blue line with numbered circles 1 through 6 indicates the progress. The main area contains the following fields:

- Subscription:** Dev
- Location:** East US 2
- Resource group \***: A dropdown menu showing 'dbops'.
- Azure Database Migration Service \***: A dropdown menu showing 'DMSMI'.
- Create new**: A link below the second dropdown.
- Connection status**: Shows a green checkmark and the message: 'Azure Database Migration Service 'DMSMI' is connected to self-hosted integration runtime running on the node'.
- Authentication keys**: A section at the bottom.

EIGHTKB

# Migration Status in ADS

The screenshot shows the Azure Database Migration Service (ADS) portal interface. The top navigation bar includes 'Welcome' and 'SQLServerOnPrem' tabs. The main navigation menu on the left lists 'Home', 'Databases', 'Administration', 'SQL Agent', and 'General'. The 'General' section is currently selected, with 'Azure SQL Migration' highlighted. The main content area displays the 'Database migration status' for the 'DMSMI (change)' service, which is associated with the 'Azure Database Migration Service'. It shows one database migration in progress and zero completed migrations.

Migration Status	Count
Database migrations in progress	1
Database migrations completed	0

EIGHTKB

# 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 shows the 'General' tab of the Azure SQL Migration tool. It displays the following information:

Source version	Target version
SQL Server 2019 15.0.4261.1	dbops

Below this, under 'Active backup files (1 item)', there is a table:

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

The screenshot shows a user interface for managing database migrations. At the top, there's a navigation bar with 'Dashboard' and 'Migrations' tabs, and a sub-menu 'Migrations > MagicWandServices'. Below this, there are two buttons: 'Complete cutover' (with a rocket icon) and 'Cancel migration' (with a cancel icon). A modal window titled 'Complete cutover' is open. It displays the following steps: 'Perform the following steps before you complete cutover.' followed by a numbered list: 1. Stop all incoming transactions to the source database. 2. Create a final transaction log backup and store it on the network share. 3. Verify that all log backups have been restored on the target database. The 'Log backups pending restore' value is shown as 0, with a 'Refresh' button next to it. A note says 'Last scan completed: 01:16 PM'. A checked checkbox says 'I confirm there are no additional log backups to provide and want to complete cutover.' A warning message in an orange box states: '⚠ Completing cutover without restoring all the backups may result in a data loss.' At the bottom of the modal are 'Complete cutover' and 'Cancel' buttons.

EIGHTKB

# 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



# 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 > dbopsml | Alerts >

Alert rules ... X

+ Create ≡ Columns ↻ Refresh ⬇️ Export to CSV 🔍 Open query | trash Delete ▶ Enable ☐ Disable

🔍 Search Target resource type : all Target scope : dbopsml Subscription : all Signal type : all Severity : all Status : Enabled

No grouping

Name ↑↓	Condition	Severity ↑↓	Target scope	Target resource type	Signal type ↑↓	Status ↑↓	...
<input type="checkbox"/> MICPUPERCENT90	avg_cpu_percent > 90	2 - Warning	dbopsml	SQL managed instance	Metrics	<span style="color: green;">● Enabled</span>	<span style="color: red;">...</span>

EIGHTKB

# Security Check Tips

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

```
1 $Date = Get-Date -Format "yyyy-MM-dd"
2 Invoke-DbcCheck -Show Fails -SqlInstance "dbopsMI.public.rand0m1234.database.windows.net,3342" -SqlCredential (Get-Credential yourmiadmin) ` 
3 -ExcludeCheck Backup, HADR, Domain, LogShipping, AgentServiceAccount, IdentityUsage, FutureFileGrowth, FKCKTrusted, GuestUserConnect, ` 
4 ValidDatabaseOwner, InvalidDatabaseOwner, InstanceConnection, SqlEngineServiceAccount, TempDbConfiguration, BackupPathAccess, DefaultFilePath, ` 
5 DAC, MaxMemory, OrphanedFile, ServerNameMatch, MemoryDump, SupportedBuild, DefaultBackupCompression, ErrorLog, CrossDBOwnershipChaining, DefaultTrace, ` 
6 OLEAutomationProceduresDisabled, RemoteAccessDisabled, SystemFull, UserFull, UserDiff, Userrog ` 
7 -Passthru | Convert-DbcResult | Set-DbcFile -FilePath C:\windows\temp\ -FileName DbcCheck_$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!

EIGHTKB

# 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/>

