

Migrate SQL Server to Azure MI Without Getting a Migraine



HELLO!

I am Josephine Bush

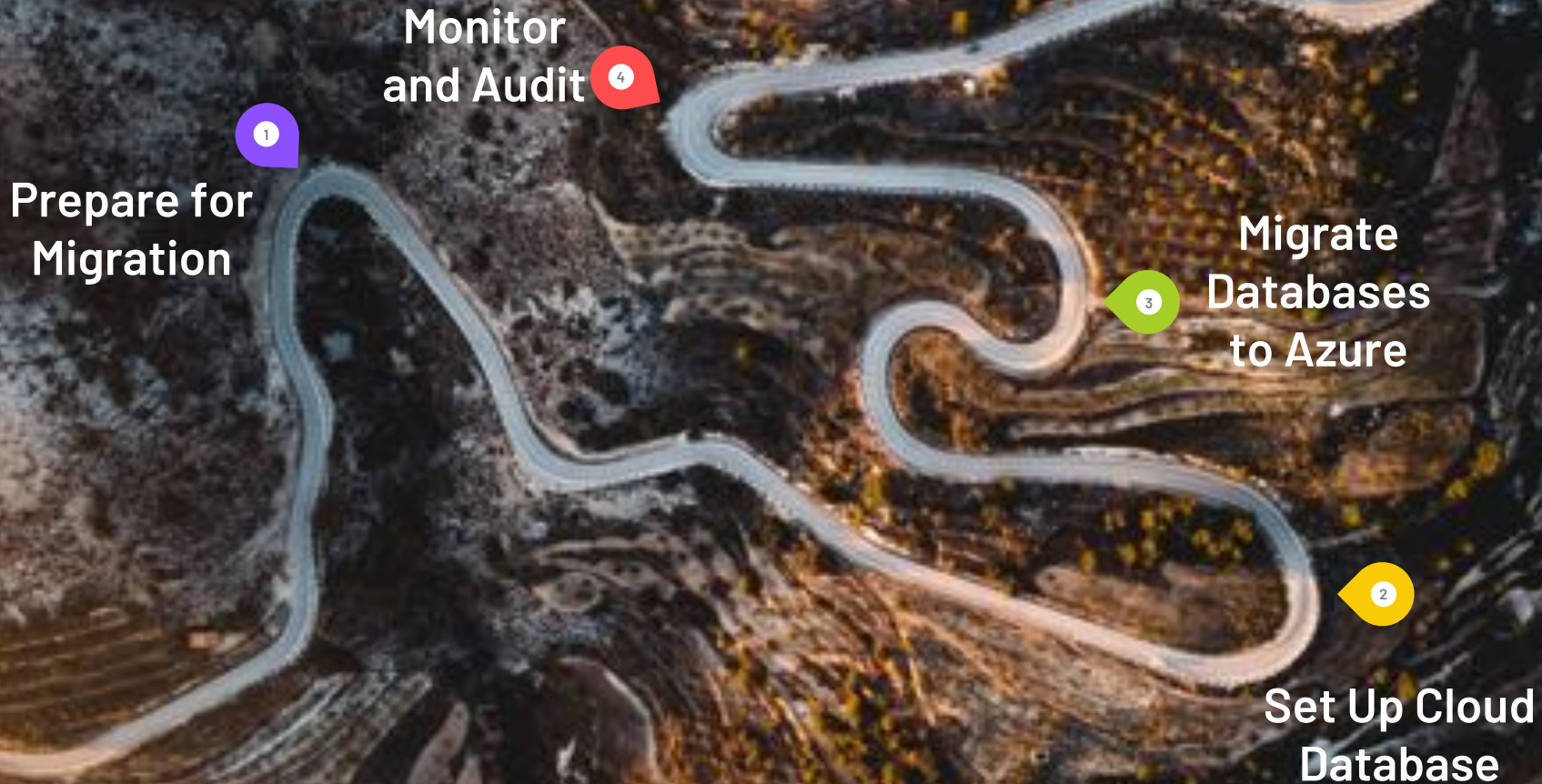
10+ years DBA experience
Microsoft Data Platform MVP
Manning liveProject Author
MBA IT Management
MS Data Analytics



@hellosqlkitty
sqlkitty.com



Database Migration Process



1

Prepare for Migration

➤ **Analyze/Audit**

Using Microsoft
Data Migration
Assistant and SQL
Server Audit

➤ **Clean Up**

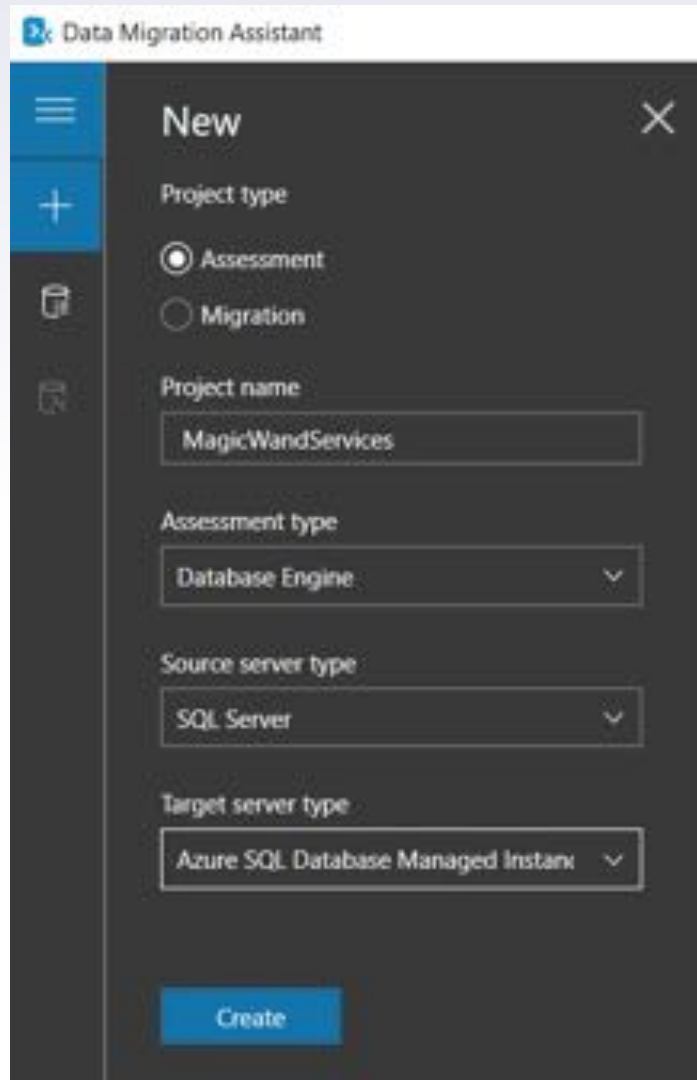
Remove
unsupported and
unused objects

➤ **Create Checklist**

Catalogue items
migrating to the
cloud

Microsoft Data Migration Assistant (DMA)

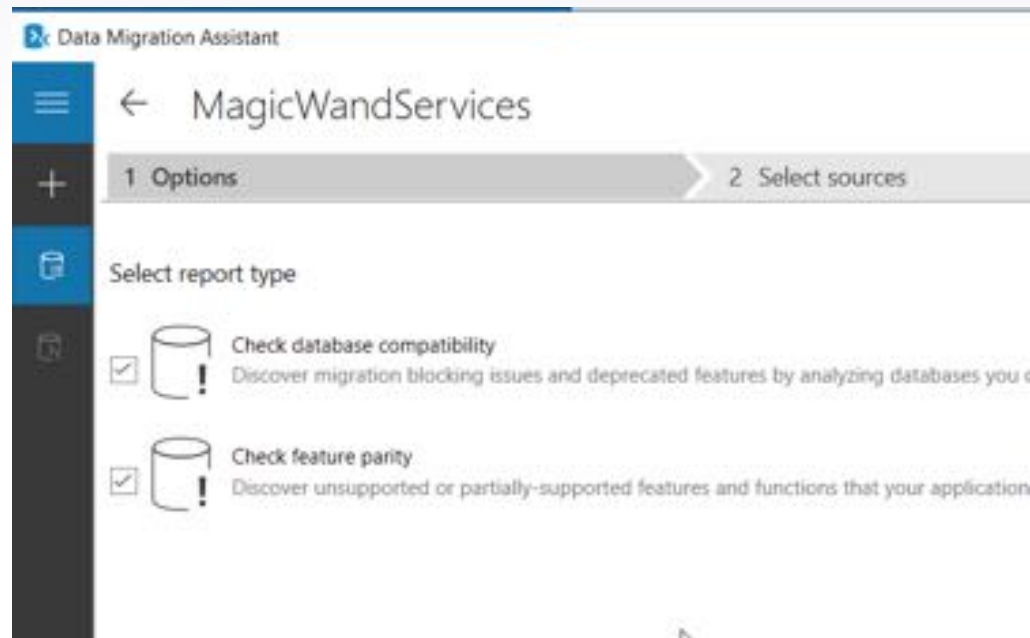
Enables you to upgrade to a modern data platform by detecting compatibility issues that can impact database functionality on your new version of SQL Server



“
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

The screenshot displays the DMA tool interface, which is divided into three main steps: 1. Options, 2. Select sources, and 3. Review results. The 'Options' step is currently active, showing the 'Target Platform' as 'Azure SQL Database Managed Instance' and the 'Source' as 'DESKTOP-15 / SQL Server 2019'. The 'Feature parity' section indicates that the 'PowerShell job step is not supported in Azure SQL Database Managed Instance'. The 'Compatibility' section shows a 'Compatibility 150 (1)' score, with a list of issues including 'Full-Text Search has changed since SQL Server 2008'. The 'Impacted objects' section lists the affected objects, including 'FullTextIndex' and 'HumanResources.JobCandidate'.

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

Unsupported features (1)

PowerShell job step is not sup... 1

PowerShell job step is not supported in Azure SQL Database Managed Instance

Details

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

Impacted

Type | Job step

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.JobCandidate

FullTextIndex | Production.Document

Object details

Type: FullTextIndex
Name: HumanResources.JobCandidate

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

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](#) or 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 *

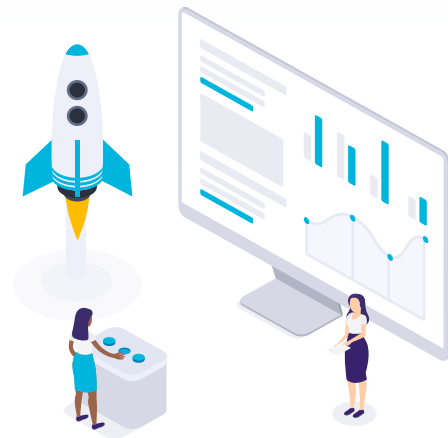
“ Azure SQL Managed Instance requires use of a dedicated vnet.

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.public.xxxxxxxxxxxxxx.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

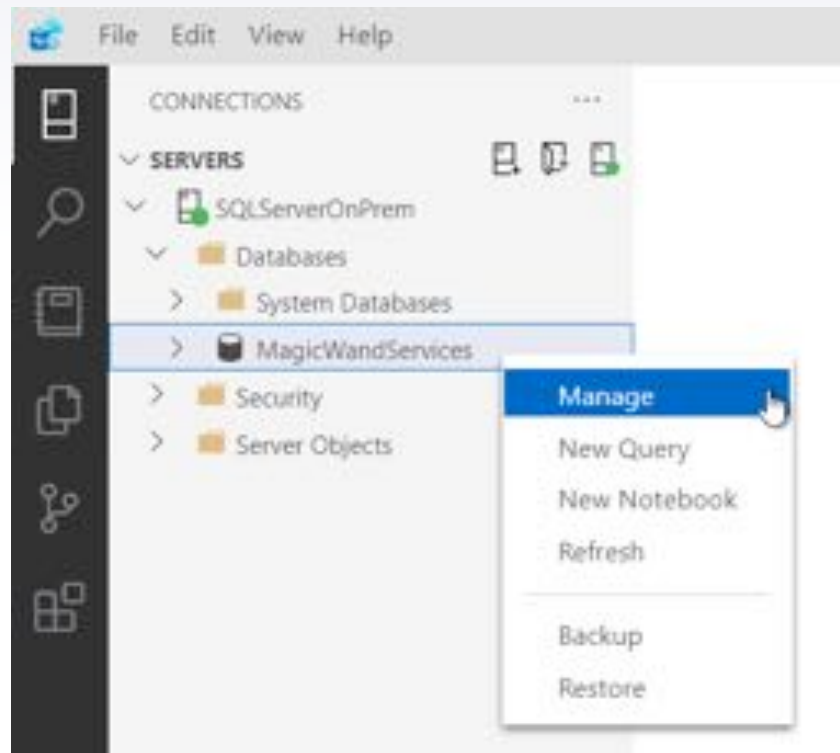
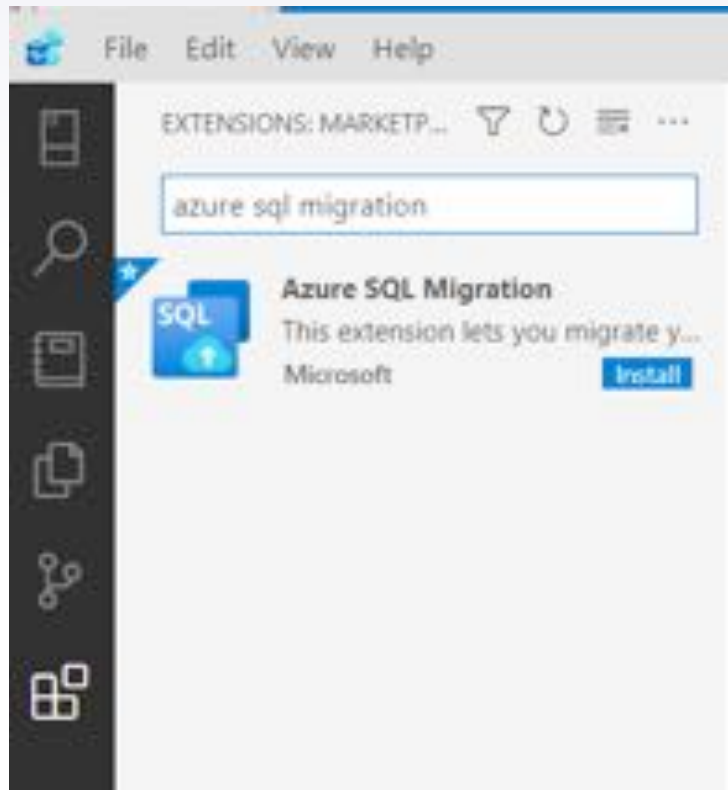
➤ **Verify Migration**
Check database configuration settings with a SQL script

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**



Setting up DMS in ADS



Set Up Migration in ADS

The screenshot displays the Azure SQL Migration portal. On the left, a 'CONNECTIONS' sidebar shows a tree view under 'SERVERS' with 'SQLServerOnPrem' expanded, containing 'Databases', 'System Databases', 'MagicWandServices' (selected), 'Security', and 'Server Objects'. The main content area has a breadcrumb path 'Home > SQLServerOnPrem > MagicWandServices'. Below this is a 'General' section with a highlighted 'Azure SQL Migration' link. The right pane shows the 'Azure SQL Migration' dashboard with tabs for 'Dashboard' and 'Migrations'. It includes links for '+ New migration', 'New support request', and 'Feedback'. The main heading is 'Azure SQL Migration', followed by a description: 'Determine the migration readiness of your SQL Server instances, identify a Server instance to Azure SQL Managed Instance, SQL Server on Azure Virt'. A prominent box titled 'Migrate to Azure SQL' states 'Migrate a SQL Server instance to Azure SQL.' To the right, a section 'Things you need to' lists requirements: 'An Azure account (n', 'A source SQL Server machine running in I', 'An Azure SQL Mana database(s) to', and 'Your database back 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 Virt

Migrate to Azure SQL

Migrate a SQL Server instance to Azure SQL.

Things you need to

- An Azure account (n
- A source SQL Server machine running in I
- An Azure SQL Mana database(s) to
- Your database back 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



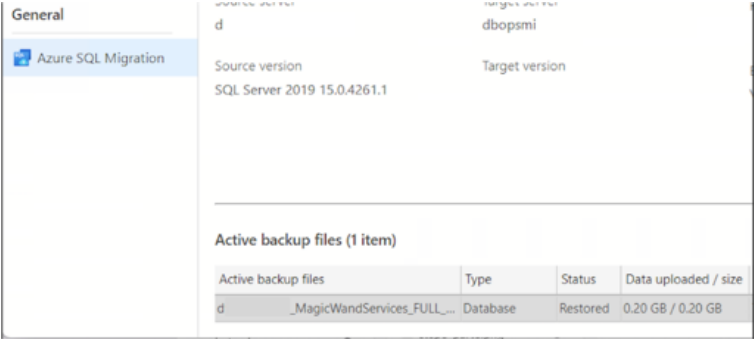
Migration Status in ADS

The screenshot displays the Azure Data Studio (ADS) interface. At the top, there are tabs for 'Welcome' and 'SQLServerOnPrem'. Below the tabs, a breadcrumb trail shows 'Home > SQLServerOnPrem'. On the left side, a sidebar contains navigation links: 'Home', 'Databases', 'Administration', 'SQL Agent', 'General', and 'Azure SQL Migration' (which is currently selected and highlighted in blue). The main content area on the right is titled 'Database migration status' and includes a 'Refresh' button. Below the title, there is a section for 'DMSMI (change)' with the subtitle 'Azure Database Migration Service'. This section contains two rows of migration status data:

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

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

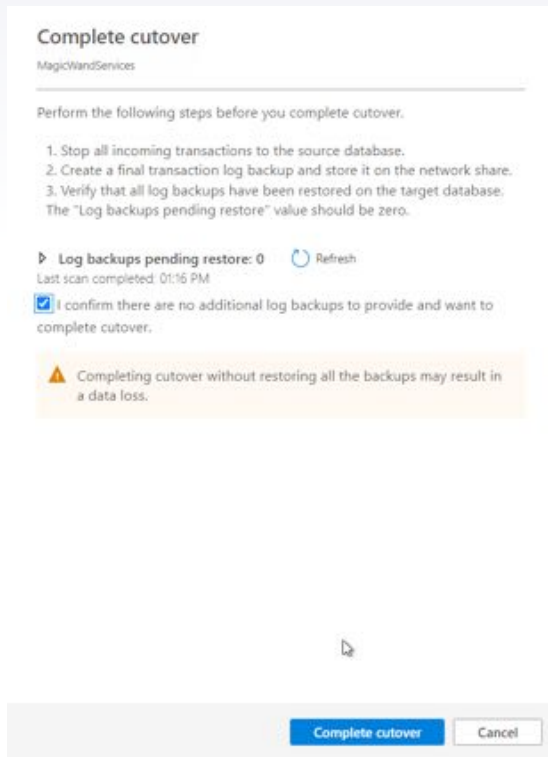
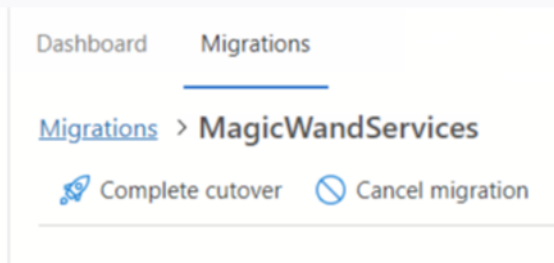


The screenshot displays the 'General' tab of the Azure SQL Migration tool. It shows the source version as 'SQL Server 2019 15.0.4261.1' and the target version as 'dbopsmi'. Below this, a table titled 'Active backup files (1 item)' lists the backup files. The table has four columns: 'Active backup files', 'Type', 'Status', and 'Data uploaded / size'. One item is listed with the name 'd..._MagicWandServices_FULL...', type 'Database', status 'Restored', and size '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

Home > dboponi | Alerts >

Alert rules

+ Create Columns Refresh Export to CSV Open query Unlink Enable Disable

Search Subscription: all Signal type: all Severity: all Status: Enabled

Showing 1 to 1 of 1 Alert rules.

Name	Condition	Severity	Target scope	Target resource type	Signal type	Status
<input type="checkbox"/> MACPUPERCENT90	avg_cpu_percent > 90	Warning	dboponi	SQL managed instance	Metrics	Enabled

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

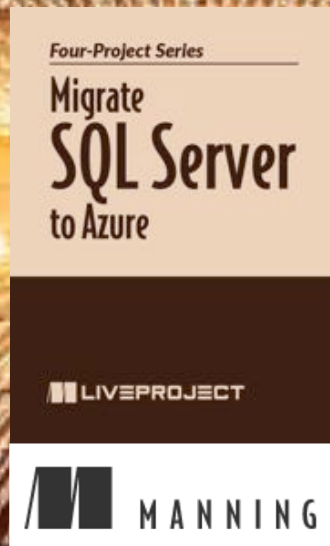
Configure auditing, alerting, and do a security check

100%

Total success!

Want the whole picture?

Get the Manning liveProject Series.



► 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
- ▶ <https://sqlkitty.com/>

