MIGRATION PLAYBOOK · DATABASES

How to migrate SQL Server to Azure SQL

Choose between Azure SQL Database, Azure SQL Managed Instance and SQL Server on Azure VMs, assess compatibility, pick the current migration tool and cut over.

Updated 4 Nov 2026 6 phases 21 detailed steps 11 official sources

START HERE

How do you migrate SQL Server to Azure SQL?

SHORT ANSWER

This moves SQL Server databases from your own servers into Azure. The recommended route copies each database to Azure SQL Managed Instance (a version of SQL Server that Microsoft runs for you, keeping most server features) with the Managed Instance link, which keeps the Azure copy up to date until you switch the application over.

- **Recommended route:** Azure SQL Managed Instance with the Managed Instance link (from Azure Arc)
- **Order:** Assess first, build the target and the server-level settings next, prove the copy on one database, then move the rest in waves.
- **6 phases:** Assess and choose, Prepare Azure, Copy to Managed Instance, Or copy to SQL Database, Switch over and Check and sign off.
- **You're done when:** Every database shows ONLINE on the target and its migration job completed with no errors.

You start it from Azure Arc (Microsoft's service for managing your own servers from the Azure portal). The link copies databases only: you recreate logins, SQL Agent jobs and other server-level settings yourself. Azure Data Studio and its migration extension were retired on 28 February 2026, so ignore guides that still use them. The time depends on database size and your test run: the copy runs in the background, and the outage is only the final switch.

RECOMMENDED ROUTE

Azure SQL Managed Instance with the Managed Instance link (from Azure Arc)

Keeps instance features such as SQL Agent while Microsoft runs the platform, and the link replicates changes in near real time, so downtime is short.

▼ **Other options and when to use them**

- **Log Replay Service or native backup and restore to Managed Instance**

  : Your SQL Server version or edition does not support the link, or you want more control over the restore.

- **Azure SQL Database with Azure DMS or a SqlPackage BACPAC**

  : The app uses a single database and no instance features such as SQL Agent or cross-database queries. Offline only.

- **SQL Server on an Azure VM**

  : The assessment finds blockers, or you need operating system access or third-party agents.

TOOLS FOR THIS PLAN

Not sure which route? Answer 6 quick questions	**Are you ready?** 22-question readiness check	**Plan your dates** Dated countdown you can add to your calendar
Compare options Cloud platforms compared	**Check your DNS** MX, SPF, DKIM and DMARC lookup	**New to the jargon?** Plain-English migration glossary

ORDER TO MIGRATE IN, AND WHY

Assess first, build the target and the server-level settings next, prove the copy on one database, then move the rest in waves. Applications switch last, one application at a time, because the Azure copy stays read-only until you cut over.

1. **Inventory and the Azure Arc assessment**

   Version, edition and blockers decide which target and which copy method each database can use. Every later step depends on that choice.

2. **Target Managed Instance, networking and backup storage**

   Managed Instance needs its own subnet and a network path from your server before any copy can start.

3. **Logins, SQL Agent jobs, linked servers and credentials on the target**

None of these move with the databases. If logins are missing on cutover day, applications cannot connect.

4 **Pilot: one non-critical database**

Shows the real copy time, lag and problems before you commit databases the business depends on.

5 **Remaining databases in waves, grouped by the applications that use them**

An application that reads several databases, or queries across them, must switch all of them in the same window.

6 **Cutover per application: stop writes, cut over, change connection strings**

The link cuts over only when the copy is fully in sync, so writes must stop first. Connection strings change last so nothing writes to the old server afterwards.

7 **Validation, sign-off, then the source server**

The read-only source is your fastest check and fallback, so it goes only after every application is signed off.

SETTINGS CHECKLIST: DECIDE THESE BEFORE YOU COPY

Built-in and third-party tools treat these differently. Tick each one off once you have decided.

Download this checklist (Excel)

- [ ] **Which copy method your SQL Server allows**

  For each instance, whether it can use the Managed Instance link or needs Log Replay Service, native backup and restore, or another target.

  ▼ How each tool handles it

Built-in method	Managed Instance link (started from Azure Arc): needs SQL Server 2016 or later on Windows Server 2016 or later, in Enterprise, Standard or Developer edition. Log Replay Service, the other Arc method, supports SQL Server 2012 and later in all editions. Neither method migrates a failover cluster instance (a SQL Server installation shared across clustered servers).

  Tip: Record version, edition and Windows Server version per instance in your inventory before you choose dates.

  Microsoft Learn: Migrate to Azure SQL Managed Instance with Azure Arc

- [ ] **Target service for each database**

Whether each database lands on Azure SQL Managed Instance, Azure SQL Database (a single managed database without server features such as SQL Agent) or SQL Server on an Azure VM.

▼ **How each tool handles it**

Built-in method	Managed Instance link: replicates to Azure SQL Managed Instance only. The Arc assessment reports readiness for each target and suggests a size from 30 days of performance history.
Azure Database Migration Service	Migrates to Azure SQL Database (offline only), Azure SQL Managed Instance or SQL Server on an Azure VM.
SqlPackage	Exports a database to a BACPAC file (a single-file copy of the schema and data) and imports it into Azure SQL Database. Offline only.

Tip: Check the assessment against your month-end peaks, not just the 30-day average.

Microsoft Learn: Assess migration readiness with Azure Arc

- [ ] **Downtime: online or offline copy**

How long the business can stop writing to each database, which decides whether you need a copy that keeps itself up to date.

▼ **How each tool handles it**

Built-in method	Managed Instance link: replicates changes in near real time and gives a read-only copy you can query during the migration. Cutover needs the copy fully in sync, so the outage is the final switch. Log Replay Service: no readable copy until you complete it, and it is not recommended for Business Critical targets.
Azure Database Migration Service	To Azure SQL Database: offline only, so downtime starts when the migration starts.
SQL Server transactional replication	Keeps Azure SQL Database in sync while the source stays in production. Setup is more complex.
SqlPackage	Offline: changes made after the export are not in the BACPAC.

Tip: Time a full test copy of the largest database before you promise an outage window.

Microsoft Learn: Migrate to Azure SQL Managed Instance with Azure Arc

- [ ] **Logins and database users**

Which logins (the server-level sign-ins for people and applications) to recreate, and how each stays matched to its database user.

▼ **How each tool handles it**

Built-in method	Managed Instance link: does not replicate logins, SQL Agent jobs or other server-level objects. Database users arrive inside the database, so a login created without the original SID (the ID that links a login to its users) leaves those users orphaned.
Azure Database Migration Service	DMS PowerShell has a login migration command, New-AzDataMigrationLoginsMigration, that moves selected logins with their server roles and permissions. Which target services it supports: not stated in Microsoft's documentation for the command: test it in your pilot.

Tip: Script logins with their SIDs before the pilot, and after it check that no database user is left without a login.

Microsoft Learn: Managed Instance link overview

- [ ] **SQL Agent jobs, linked servers and credentials**

Which scheduled jobs, linked servers and stored credentials are still needed, and where each will run.

▼ **How each tool handles it**

Built-in method	Managed Instance link: none of these are replicated. Managed Instance runs SQL Agent, so jobs can be recreated from your scripts.

Tip: If a database goes to Azure SQL Database, which has no SQL Agent, plan a replacement such as elastic jobs before cutover.

Microsoft Learn: Managed Instance link overview

- [ ] **Database features the target cannot hold**

What to change or remove in databases that use features the target does not support.

▼ **How each tool handles it**

Built-in method	Managed Instance link: replicates user databases only, not system databases. It cannot replicate a database with more than one log file, or with FileTable or FILESTREAM

data. Databases using In-Memory OLTP replicate only to the Business Critical tier.

Tip: Fix or record every blocker the Arc assessment lists, such as CLR assemblies or cross-database queries, before the pilot.

Microsoft Learn: Managed Instance link overview

- [ ] **Encrypted databases (TDE)**

Which databases use TDE (Transparent Data Encryption, which encrypts the database files on disk), and who holds the keys.

▼ How each tool handles it

Built-in method Managed Instance link: you export the database encryption key from SQL Server, upload it to Azure Key Vault and set the Managed Instance to use your own TDE key (Microsoft calls this BYOK) before you create the link.

Tip: Find encrypted databases early with SELECT name, is_encrypted FROM sys.databases; and book time with whoever manages your key vault.

Microsoft Learn: Managed Instance link overview

- [ ] **Backups used for the copy**

If you use Log Replay Service or native restore: how backups are taken, where they are uploaded and how long the restore may run.

▼ How each tool handles it

Built-in method Log Replay Service: restores your full, differential and log backups from Azure Blob Storage. Take them WITH CHECKSUM. The SAS token (a time-limited key to the storage) must have only Read and List permissions. A job is cancelled after 30 days, and stopping a job deletes the database being restored.

Tip: Keep the full backup chain until sign-off, so you can start again if a restore fails.

Microsoft Learn: Migrate databases with Log Replay Service

- [ ] **Data that a copy can change or skip**

Which tables, columns and defaults you check row by row after the test run.

▼ How each tool handles it

Built-in method Managed Instance link: replicates the whole user database, so tables arrive as they are.

Azure Database Migration Service	To Azure SQL Database: computed columns and empty tables are not migrated, NULLs in columns with default constraints become the default value, and double-byte table names are not supported. Disable change data capture on the source before the schema migration.
SqlPackage	A BACPAC creates the whole schema from the source definition. Use local storage for BACPAC files over 200 GB.

Tip: Compare row counts for every table and checksums for key tables, and keep them in the data validation report.

Microsoft Learn: Managed Instance link overview

- [ ] **How many databases run at once**

How many databases go in each wave, and whether the Arc extension needs updating first.

▼ **How each tool handles it**

Built-in method	Managed Instance link in Arc: up to 10 databases at once with Azure Extension for SQL Server 1.1.3348.364 or later, and only one with earlier versions. Each link carries one database. General Purpose and Business Critical instances hold up to 100 databases. Log Replay Service has no limit on simultaneous migrations.

Tip: Update the extension on every host before the first wave.

Microsoft Learn: Migrate to Azure SQL Managed Instance with Azure Arc

- [ ] **Network path and sign-in between the servers**

How your server reaches Azure: private network, VPN or ExpressRoute, and which ports the firewall allows.

▼ **How each tool handles it**

Built-in method	Managed Instance link: Managed Instance sits in a dedicated subnet, and the link authenticates with certificates exchanged between the two servers. Windows authentication cannot be used for the link.
Azure Database Migration Service	Needs its self-hosted integration runtime (a small agent) installed on a machine that can reach the source.

Tip: Test the connection from the source host to the target before the pilot, not on the night.

Microsoft Learn: Managed Instance link overview

- [ ] **Rollback and the update policy**
  How you would go back to SQL Server, how long that path stays open, and who can approve it.

  ▼ **How each tool handles it**

Built-in method	Arc migration pane: no reverse migration. You can go back manually with native backup and restore or a link set up in SSMS (SQL Server Management Studio), depending on the Managed Instance's update policy (the setting that decides which SQL Server version it matches). Switching that policy to Always-up-to-date permanently removes the option to restore back to SQL Server 2022 or 2025.

  Tip: Write the rollback method and its time into the cutover runsheet, and leave the update policy alone until the rollback window closes.

  Microsoft Learn: Migrate to Azure SQL Managed Instance with Azure Arc

THE PLAN, PHASE BY PHASE

1 Assess and choose

Before you start

List every SQL Server instance with its version, edition and databases in an inventory.

Connect each SQL Server host

to Azure Arc (Microsoft's service for managing your own servers from the Azure portal).

Run the Arc assessment and note the recommended target and readiness for each database.

Capture CPU, memory, IO and storage figures over a full business cycle, including

month-end.

Fix every blocker, such as linked servers, CLR assemblies or cross-database queries, or record a decision.

Choose a target per database: Managed Instance, Azure SQL Database or a SQL Server VM.

> **Watch out:** Azure Data Studio and its Azure SQL Migration extension were retired on 28 February 2026.
>
> **WAYS AROUND IT**
>
> - Use the Azure Arc migration pane, Azure DMS in the portal, or DMS PowerShell and Azure CLI instead.

> **Watch out:** In Arc, the Managed Instance link needs SQL Server 2016 or later, Enterprise, Standard or Developer edition, on Windows Server 2016 or later. Neither Arc method migrates a failover cluster instance.
>
> **WAYS AROUND IT**
>
> - Record version and edition per instance, and use Log Replay Service (all editions) or native backup and restore where the link does not fit.

Show the detailed steps (5) →

2 Prepare Azure

Create the Managed Instance (or database or VM) in the region nearest your application servers, such as Australia East.

Set up private networking (Managed Instance needs a dedicated subnet) and

test that the source can reach the target.

Script logins with their SIDs (the IDs that link them to database users), plus SQL Agent jobs, linked servers and credentials.

Create an Azure Storage container if you will restore from backups.

Update the Azure Extension for SQL Server on each host before the migration window.

> **Watch out:** The Managed Instance link and backup restores copy databases only. Logins, SQL Agent jobs, linked servers and other server-level objects do not move.
>
> **WAYS AROUND IT**
>
> - Run your saved scripts on the target. On Azure SQL Database, which has no SQL Agent, use a replacement such as elastic jobs.

> **Watch out:** Managed Instance link in Arc: with Azure Extension for SQL Server versions before 1.1.3348.364, only one database migrates at a time (up to 10 with later versions).
>
> **WAYS AROUND IT**
>
> - Update the extension first, then plan waves of no more than 10 databases per instance.

Show the detailed steps (3) →

3 Copy to Managed Instance

Start with one non-critical database as a

test, and time it.

In the Arc resource, open Migration > Database migration and choose Migrate using real-time replication (online).

Watch Monitor migrations until each database shows little or no lag.

Ask testers to check the read-only copy

on the Managed Instance.

For Log Replay Service, take backups WITH CHECKSUM, upload them and start it in continuous mode.

> **Watch out:** Log Replay Service will not start unless the SAS token (a time-limited key to the storage) has only Read and List permissions.
>
> **WAYS AROUND IT**
>
> - Create a dedicated token for the backup container, paste it without the leading question mark, and set its expiry after cutover.

> **Watch out:** A Log Replay Service job is cancelled after 30 days, and stopping a job deletes the database being restored.
>
> **WAYS AROUND IT**
>
> - Start it close enough to cutover to finish within 30 days, never stop a job you want to keep, and keep the full backup chain.

Show the detailed steps (3) →

4 Or copy to SQL Database

Only if you chose Azure SQL Database

Create an Azure

Database Migration Service and install its integration runtime (a small agent) on a machine that can reach the source.

Disable change data capture on the source before the schema migration.

Run a full test migration to measure how long the

copy takes.

Use a SqlPackage BACPAC export and import (a single-file copy of a database) for smaller databases.

Watch out: Azure DMS to Azure SQL Database is offline only, so downtime starts when the migration starts.

WAYS AROUND IT

- Size the change window from the test run, or choose Managed Instance if the window is too short.

TOOLS THAT CAN DO IT

SQL Server transactional replication
Keeps Azure SQL Database in sync while the source stays in production. Setup is more complex.

Watch out: Azure DMS to Azure SQL Database skips computed columns and empty tables, turns NULLs in columns with defaults into the default, and does not support double-byte table names.

WAYS AROUND IT

- Check the source for these cases before the test run, and include them in your data validation report.

TOOLS THAT CAN DO IT

SqlPackage
A BACPAC creates the whole schema from the source definition. Offline only; use local storage for BACPACs over 200 GB.

Show the detailed steps (4) →

5 Switch over

In an agreed change window

Stop application writes and disable the SQL Agent jobs on the source.

For the link, wait for Ready for cutover with no lag, then select Cutover. Otherwise complete Log Replay Service or DMS.

Update connection strings, DNS aliases and firewall rules to the new endpoint.

Set the source database to read-only, and keep it until sign-off.

Run the application smoke tests.

> **Watch out:** Log Replay Service databases cannot be read until you complete the migration, and no more backups can be restored after that.
>
> **WAYS AROUND IT**
>
> - Upload the last log backup before you run the complete command.

Turn this phase into dated tasks with the cutover planner

Check your DNS records before and after the change

Show the detailed steps (3) →

6 Check and sign off

Straight after the switch

Compare row counts and checksums for key tables, and record them in the data validation report.

Sign in with every application login and check no database user is orphaned (left without a login).

Check SQL Agent jobs, or their

replacements, run at their next schedule.

Compare performance with your baseline during a busy period, and confirm backups and alerts.

Decommission the source server only after sign-off. That is the final point of no return.

> **Watch out:** The Arc migration pane has no reverse migration. Once users write to Azure, going back means native backup and restore or a link set up in SSMS, if the Managed Instance update policy allows it, plus moving those new changes yourself.

WAYS AROUND IT

- Write the rollback method and its time into the cutover runsheet before you switch.
- Keep the source read-only and the full backup chain until sign-off.

Watch out: The update policy is the Managed Instance setting that decides which SQL Server version it matches. Switching it to Always-up-to-date permanently removes the option to restore back to SQL Server 2022 or 2025.

WAYS AROUND IT

- Leave the update policy unchanged until the rollback window has closed.

Show the detailed steps (3) →

You are finished when

- Every database shows ONLINE on the target and its migration job completed with no errors.
- Row counts match for every table, and checksums match for key tables.
- Applications connect to the new endpoint, smoke tests pass, and no sessions remain on the source.
- Backups, alerts and SQL Agent jobs (or their replacements) run on the target.

Tools named in this plan are independent third-party products, not endorsed by the platform vendor and not paid placements. Check the tool's own documentation and run a pilot first.

CLICK BY CLICK

What does each step involve?

Open a part when you reach it in the plan. Each step says where to go, what to do and what you should see.

About this migration — What it covers, limits at a glance —

Microsoft retired Azure Data Studio, and with it the Azure SQL Migration extension, on 28 February 2026. Guides that tell you to install that extension are out of date. Microsoft now points to SQL Server enabled by Azure Arc for assessment and Managed Instance migration, the Azure Database Migration Service (DMS) in the portal and its PowerShell and Azure CLI interfaces, and native options such as the Managed Instance link.
This guide follows the order of work: choose a target, assess, prepare, migrate, cut over.

AZURE DATA STUDIO
Retired 28 February 2026, including the Azure SQL Migration extension

ASSESSMENT
SQL Server enabled by Azure Arc, Azure Migrate, or DMS PowerShell and Azure CLI

TO MANAGED INSTANCE
Managed Instance link, Log Replay Service, or native backup and restore

TO AZURE SQL DATABASE
Azure DMS (offline), SqlPackage BACPAC, or transactional replication

TO SQL SERVER ON AZURE VM
Azure DMS, backup and restore, or Azure Migrate rehost of the whole server

1 What you need before step one

Need	Details	Where to get it
Azure roles	Contributor on the target resource group; Owner or Contributor on the subscription to create a Database Migration Service instance. The Microsoft.DataMigration resource provider must be registered.	Azure portal, https://portal.azure.com
SQL Server permissions	sysadmin on the source for assessment and backups. For DMS to Azure SQL Database the source login needs db_datareader and VIEW ANY DEFINITION; the target login needs db_owner.	Your DBA team

Need	Details	Where to get it
Azure Arc (assessment and MI migration)	The SQL Server host connected to Azure Arc with the latest Azure extension for SQL Server. The Arc assessment runs on Windows only.	Azure portal > Machines - Azure Arc > Onboard/Create > Onboard existing machines
Azure CLI	Version 2.42.0 or later for the Log Replay Service commands.	Install the Azure CLI
Self-hosted integration runtime (DMS)	Version 5.37 or later for schema migration, on a machine that can reach the source SQL Server.	https://www.microsoft.com/download/details.aspx?id=39717
Azure Storage account	A Blob container for backup-based methods (Log Replay Service, backup to URL).	Copy files with AzCopy for uploading existing backups
SQL Server Management Studio	To script logins and jobs and run the T-SQL in this guide.	Microsoft Learn: Download SQL Server Management Studio

2 Choose the Azure SQL target

Target	What you manage	Good fit	Watch for
Azure SQL Database	Databases only; Microsoft runs the engine	New or modern apps using a single database	No SQL Agent, cross-database queries or instance features; needs app changes in some cases

Target	What you manage	Good fit	Watch for
Azure SQL Managed Instance	Instance-level settings; Microsoft patches and backs up	Most lift-and-shift of existing SQL Server apps	Network design (dedicated subnet) and feature differences to check in assessment
SQL Server on Azure VM	Everything inside the VM	Full compatibility, third-party agents, or OS access	You still patch, back up and run high availability yourself

Choose the region. Pick the same region as the application servers, close to your users (for example Australia East if you are in Australia), to keep latency low and meet any data residency rules.

3 Assess compatibility and sizing

5 steps

1

Connect SQL Server hosts to Azure Arc

Where: Azure portal > Machines - Azure Arc > Onboard/Create > Onboard existing machines, then an elevated 64-bit PowerShell window on each SQL Server host

On **Basics** choose the subscription, resource group, region, **Windows** and the connectivity method, choose **Authenticate machines manually**, add tags, then **Download** the script and run it on the server as a local administrator. SQL Server instances on the machine are then registered with the Azure extension for SQL Server.

```
./OnboardingScript.ps1
```

You should see: The machine shows status Connected under Machines - Azure Arc, and the SQL Server instance appears as a SQL Server - Azure Arc resource.

2

Assess with SQL Server enabled by Azure Arc

Where: Azure portal > your SQL Server - Azure Arc resource > Migration > Assessments

The SQL Server migration experience in Azure Arc assesses readiness, recommends a right-sized Azure SQL target and keeps the assessment current. Microsoft says the assessment is generated automatically and refreshed every Sunday at 23:00 local time; select **Run assessment** for a fresh result, and **Assessment settings** to change the strategy or pricing options. It needs Azure Extension for SQL Server version 1.1.2594.118 or later.

✓ **You should see:** A Recommended Target banner, and Ready, Not ready or Unknown readiness for Azure SQL Managed Instance, SQL Server on Azure VM and Azure SQL Database.

3

Assess at scale with Azure Migrate

Where: Azure Migrate project, with an appliance discovering your servers (set up the Azure Migrate appliance)

If you are moving whole data centres, Azure Migrate discovers SQL Server instances alongside servers and gives Azure SQL deployment recommendations and target sizing. Add SQL Server credentials in the appliance configuration manager so it can connect to the instances.

✓ **You should see:** SQL Server instances appear in the project inventory and can be included in an assessment.

4

Capture a performance baseline

Where: Each source SQL Server (Performance Monitor data collector set), or the Arc assessment's source properties page

Collect CPU, memory, IO and storage figures over a full business cycle. Microsoft provides Logman-based Perfmon collection for this. The Arc assessment also shows the last 30 days of performance history.

✓ **You should see:** Peak and average CPU, memory, IOPS and storage for each instance, covering month-end.

5

Resolve blockers

Where: Azure portal > SQL Server - Azure Arc resource > Migration > Assessments > Compatibility tab > each Findings link

Fix issues such as unsupported features, cross-database dependencies, linked servers, CLR assemblies and logins before you migrate. Follow the remediation guidance behind each finding.

✓ **You should see:** No blocking issues remain for your chosen target, and warnings have a documented decision.

4 Prepare the target

3 steps −

1

Create the target

Where: Azure Cloud Shell or a terminal with the Azure CLI (install the Azure CLI). Managed Instance and SQL VMs: Azure portal > Azure SQL > Create

Create the logical server and database, the Managed Instance, or the SQL Server VM in your target region and virtual network. The Arc migration pane can also create a Managed Instance for you (Step 4a).

```
az group create --name rg-sql-aue --location australiaeast
az sql server create --name sql-contoso-aue --resource-group rg-sql-aue --location australiaeast
--admin-user sqladmin --admin-password <password>
az sql db create --resource-group rg-sql-aue --server sql-contoso-aue --name SalesDb --service-
objective S3
```

✓ **You should see:** Each command returns JSON for the new resource, and the database shows status Online in the portal.

2

Plan connectivity

Where: Azure portal > the logical server or Managed Instance > Networking, and your firewall

Use private endpoints or VNet integration, and confirm the source can reach the target (or Azure Storage, for backup-based methods).

```
Test-NetConnection sql-contoso-aue.database.windows.net -Port 1433
```

✓ **You should see:** TcpTestSucceeded : True from the source server or the integration runtime machine.

3

Recreate logins and jobs

Where: SQL Server Management Studio connected to the source: Security > Logins and SQL Server Agent > Jobs (right-click > Script ... as > CREATE To)

Script server logins, SQL Agent jobs, linked servers and credentials. Database-level objects move with the data; instance-level objects usually do not. Script SQL logins with their SIDs so database users stay mapped.

✓ **You should see:** A saved script for every login, job, linked server and credential, ready to run on the target.

5 Migrate to Azure SQL Managed Instance

3 steps

1

Managed Instance link for minimum downtime

Where: Azure portal > your SQL Server - Azure Arc resource > Migration > Database migration

The Managed Instance link uses distributed availability groups to replicate changes in near real time, with read-only access on the Managed Instance during migration. Microsoft lists SQL Server 2016 to 2022 as supported sources, and the Arc experience needs Enterprise, Standard or Developer edition on Windows Server 2016 or later. Select **Select target** (or create one), then **Migrate data** > **Migrate using real-time replication (online)**. Tick the databases, select **Next: Settings**, enter the link name, optionally **Test connection**, then **Next: Review + create** and **Start data migration**.

✓ **You should see:** Under Monitor migrations each database appears with its status and a Lag column. Microsoft says creating each link can take up to 10 minutes.

2

Log Replay Service for more control

Where: Azure Storage container (one folder per database), then Azure Cloud Shell or the Azure CLI 2.42.0 or later

Log Replay Service restores full, differential and log backups that you place in Azure Storage. Databases stay in restoring mode and cannot be read until the migration completes. Microsoft lists SQL Server 2008 to 2022 and Amazon RDS for SQL Server among the sources. Take backups WITH CHECKSUM, upload them to a folder per database, and generate a SAS token with **Read** and **List** permissions only. Pass the token without the leading question mark. Use continuous mode while backups are still being taken; you complete it at cutover (Step 5).

```
BACKUP DATABASE [SalesDb] TO DISK = 'C:\BACKUP\SalesDb_full.bak' WITH INIT, COMPRESSION, CHECKSUM;
az sql midb log-replay start -g rg-sql-aue --mi sqlmi-contoso-aue -n SalesDb --storage-uri
"https://stcontosomig.blob.core.windows.net/backups/SalesDb" --storage-sas "<sas-token-without-
question-mark>"
az sql midb log-replay show -g rg-sql-aue --mi sqlmi-contoso-aue -n SalesDb
```

✓ **You should see:** log-replay show reports the restore progress, and the database appears on the Managed Instance in Restoring state.

3

Native backup and restore for simple cases

Where: SSMS query window on the source SQL Server, then an SSMS query window connected to the Managed Instance

Back up to Azure Blob Storage with a SAS credential and restore on the Managed Instance. Downtime covers the full backup and restore.

```
-- On the source SQL Server
CREATE CREDENTIAL [https://stcontosomig.blob.core.windows.net/backups]
  WITH IDENTITY = 'SHARED ACCESS SIGNATURE', SECRET = '<sas-token-without-leading-question-mark>';
BACKUP DATABASE SalesDb TO URL = 'https://stcontosomig.blob.core.windows.net/backups/SalesDb.bak'
WITH COPY_ONLY, COMPRESSION, CHECKSUM;
-- On the Managed Instance (create the same credential first)
RESTORE DATABASE SalesDb FROM URL =
'https://stcontosomig.blob.core.windows.net/backups/SalesDb.bak';
```

✓ **You should see:** BACKUP and RESTORE both report that they processed the pages successfully, and the database is online on the Managed Instance.

6 Migrate to Azure SQL Database

4 steps

1

Set up DMS and the integration runtime

Where: Azure portal > Azure Database Migration Services > Create; then your DMS > Settings > Integration runtime > Configure integration runtime

Choose the source and target server types and **Database Migration Service**, then the subscription, resource group, location and name, and select **Review + Create**. In **Configure integration runtime** use the **Download and install integration runtime** link on a machine that can reach the source, paste an **Authentication key** into Microsoft Integration Runtime Configuration Manager and select **Register**.

✓ **You should see:** After a few minutes the integration runtime node shows under Settings > Integration runtime.

2

Azure DMS offline migration

Where: Azure portal > Azure Database Migration Services > your DMS > Overview > New migration

Azure Database Migration Service migrates schema and data offline using Azure Data Factory and a self-hosted integration runtime. Choose the source, target and migration mode, then in the wizard enter source details, connect to the source, select databases, connect to the target, map databases, tick **Migrate Missing schema** if the tables do not exist yet, select tables, and choose **Start migration**. DMS PowerShell and Azure CLI can run the same migrations at scale.

✓ **You should see:** Under Monitor migrations the status moves from In progress to Succeeded, with per-table statuses such as Copying and Rebuilding indexes.

3

SqlPackage for smaller databases

Where: Command prompt on a machine with SqlPackage installed and network access to both servers

Export a BACPAC from the source and import it into Azure SQL Database. This needs downtime because the data is exported and then imported.

```
SqlPackage /Action:Export /SourceServerName:sql01.contoso.com /SourceDatabaseName:SalesDb
/SourceTrustServerCertificate:True /TargetFile:C:\Migration\SalesDb.bacpac
SqlPackage /Action:Import /SourceFile:C:\Migration\SalesDb.bacpac /TargetServerName:sql-contoso-
aue.database.windows.net /TargetDatabaseName:SalesDb /TargetUser:sqladmin /TargetPassword
:<password>
```

✓ **You should see:** Both commands end with a message that the operation completed successfully, and the database appears in the portal.

4

Continuous replication options

Where: SSMS on the source: Replication > Local Publications > New Publication (transactional replication)

Transactional replication can keep Azure SQL Database in sync without taking the source offline, but setup is more complex. Microsoft also partners with Striim for online migration; contact Microsoft to access it.

✓ **You should see:** The subscription on Azure SQL Database shows as synchronising in Replication Monitor.

7 Migrate to SQL Server on Azure VM

3 steps

1

Azure DMS in the portal

Where: Azure portal > Azure Database Migration Services > your DMS > Overview > New migration (target: SQL Server on Azure Virtual Machine)

DMS supports online and offline migrations to SQL Server on Azure VMs, reading backups from Azure Blob Storage or an SMB network share. Use the same integration runtime as Step 4b.

✓ **You should see:** The migration appears under Monitor migrations and, for online mode, waits for you to cut over.

2

SSMS migration component

Where: SQL Server Management Studio connected to both the source and the Azure VM

If both servers are reachable from SQL Server Management Studio, its migration component can move databases to SQL Server on Azure VM.

✓ **You should see:** The databases are online on the Azure VM.

3

Rehost the whole server

Where: Azure Migrate project > Execute > Migration > Start execution

To move the entire VM, including SQL Server, use Azure Migrate. See the VMware to Azure guide.

✓ **You should see:** The whole server runs as an Azure VM with SQL Server unchanged.

8 Cut over

3 steps

1

Freeze and final sync

Where: The application (stop writes), then the method's console: Arc > Database migration > Monitor migrations > Monitor and cutover; the Azure CLI for Log Replay Service; DMS > Monitor migrations

Stop application writes, let the final log backup or replication catch up, then complete the migration. For the MI link, wait for status **Ready for cutover** with no lag (two dashes in the Lag column) and select **Cutover**. For Log Replay Service in continuous mode, upload the last log backup and run the complete command. For DMS, finish the migration from Monitor migrations.

```
az sql midb log-replay complete -g rg-sql-aue --mi sqlmi-contoso-aue -n SalesDb --last-backup-name "SalesDb_log_final.trn"
```

✓ **You should see:** The target database comes online and is writable.

2

Switch connection strings

Where: Application configuration files or app settings, DNS aliases and firewall rules

Update application connection strings, DNS aliases and firewall rules to the new endpoint.

✓ **You should see:** The application connects to the new endpoint and no sessions remain on the source (check with sp_who2 on the source).

3

Validate

Where: SSMS query windows on the source and the target

Compare row counts and checksums for key tables, run application smoke tests, check SQL Agent jobs and logins, and confirm backups and alerts on the target. Record the results in the data validation report.

```
SELECT t.name, SUM(p.rows) AS row_count
FROM sys.tables t JOIN sys.partitions p ON t.object_id = p.object_id AND p.index_id IN (0,1)
GROUP BY t.name ORDER BY t.name;
```

✓ **You should see:** Row counts match between source and target for every key table.

Rollback. Keep the source database available read-only until sign-off. Once users write to Azure, rolling back means moving those changes back.

WORTH KNOWING

What can go wrong, and how do you avoid it?

The detail behind the plan: options compared, and what experienced engineers check first. Tap any item.

Compare all the ways to do this 3 options –

Azure SQL Database

MEDIUM EFFORT

Best for: New or modern apps using a single database

+ Microsoft runs the engine, so there is no instance or OS to manage
+ Move with Azure DMS (offline), a SqlPackage BACPAC for smaller databases, or transactional replication
– No SQL Agent, cross-database queries or instance features, so some apps need changes
– DMS to Azure SQL Database is offline only

Read more →

Azure SQL Managed Instance

MEDIUM EFFORT

Best for: Most lift-and-shift moves of existing SQL Server apps

+ Instance-level settings, with Microsoft handling patching and backups
+ Minimum downtime with the Managed Instance link, more control with Log Replay Service, or native backup and restore
– Needs network design, including a dedicated subnet
– Feature differences to check in the assessment

Read more →

SQL Server on an Azure VM

LOW EFFORT

Best for: Apps needing full compatibility, third-party agents or operating system access

+ The same SQL Server you run today
+ Move with Azure DMS, backup and restore, or by rehosting the whole server with Azure Migrate
– You still patch, back up and run high availability yourself
– Fewer of the managed-service benefits that motivate most moves

Read more →

OUR ADVICE

Start with Managed Instance for existing apps, because it keeps instance features while Microsoft runs the platform. Choose Azure SQL Database when the app uses a single database and no instance features, and a VM only when the assessment finds blockers or you need OS access. Let the Azure Arc or Azure Migrate assessment confirm the choice per database.

Ignore guides that still use Azure Data Studio

SAVES TIME

Microsoft retired Azure Data Studio and the Azure SQL Migration extension on 28 February 2026. Many blog posts and scripts still depend on it.

WHAT TO DO

Use the Azure Arc migration experience, Azure DMS in the portal or its PowerShell and CLI interfaces instead, and update any internal runbooks that mention the extension.

Microsoft Learn: What is happening to Azure Data Studio

Check which method your version and edition allow

REDUCES RISK

In the Arc experience the Managed Instance link needs SQL Server 2016 or later on Windows Server 2016 or later, in Enterprise, Standard or Developer edition. Log Replay Service supports SQL Server 2012 and later in all editions.

WHAT TO DO

Record version and edition per instance in your inventory and pick the method from that, before you design the cutover. Neither Arc method supports migrating from a failover cluster instance.

Microsoft Learn: Migrate to Azure SQL Managed Instance with Azure Arc

Update the Arc extension for parallel MI link migrations

SAVES TIME

Microsoft says the Arc experience can run up to 10 Managed Instance link migrations at once with Azure Extension for SQL Server 1.1.3348.364 or later, but only one with earlier versions.

WHAT TO DO

Update the extension on each host before the migration window, and plan waves of no more than 10 databases per instance.

Microsoft Learn: Migrate to Azure SQL Managed Instance with Azure Arc

Log Replay Service SAS tokens must be Read and List only

SAVES TIME

Microsoft says LRS will not start if the SAS token has any permission other than Read and List, and the token must be passed without the leading question mark.

WHAT TO DO

Generate a dedicated SAS for the backup container with only Read and List, copy only the part after the question mark, and set an expiry beyond your planned cutover date.

Microsoft Learn: Migrate databases with Log Replay Service

Log Replay Service jobs have a 30-day limit and no restart

REDUCES RISK

Microsoft says a single LRS job is cancelled after 30 days, stopping a job deletes the database being restored, and after completion no more backups can be restored into it.

WHAT TO DO

Start LRS close enough to cutover to finish within 30 days, never stop a job you want to keep, and keep the full backup chain until sign-off in case you must start again.

Microsoft Learn: Log Replay Service overview

Take every migration backup WITH CHECKSUM

REDUCES RISK

Microsoft says CHECKSUM is the only way to be sure a backup restored to SQL Managed Instance is not corrupt, and it also speeds up the restore.

WHAT TO DO

Add WITH CHECKSUM (and COMPRESSION) to the full, differential and log backups used for the migration, including any scheduled log backup jobs that feed LRS.

Microsoft Learn: Migrate databases with Log Replay Service

DMS to Azure SQL Database is offline only

USER IMPACT

Microsoft says online migrations to Azure SQL Database are not available in DMS, and for an offline migration application downtime starts when the migration starts.

WHAT TO DO

Run a full test migration to measure the copy time, then size the change window from that. If the window is too short, consider transactional replication or Managed

Instance instead.

TOOLS THAT CAN GET AROUND THIS

Transactional replication keeps an Azure SQL Database copy in sync while the source stays in production, so the outage shrinks to changing connection strings.

SQL Server transactional replication

Microsoft lists it as the method to use when you cannot take the source out of production during the migration.

Third-party tools, not endorsed by the platform vendor. Confirm in the tool's documentation and run a pilot first.

Microsoft Learn: Tutorial: Migrate SQL Server to Azure SQL Database (offline)

Know what DMS does not copy to Azure SQL Database

REDUCES RISK

Microsoft lists that computed columns are not migrated, NULLs in columns with default constraints become the default value, tables with zero rows are skipped, and double-byte table names are not supported.

WHAT TO DO

Check the source for these cases before the test run, disable change data capture before schema migration, and include computed columns and defaults in your data validation report.

TOOLS THAT CAN GET AROUND THIS

A BACPAC holds the metadata and data of the whole database, so the schema, including computed columns, defaults and empty tables, is created from the source definition rather than table by table.

SqlPackage

Export and import a BACPAC. Offline only: stop writes during export for a consistent copy, and use local storage for BACPACs over 200 GB.

Third-party tools, not endorsed by the platform vendor. Confirm in the tool's documentation and run a pilot first.

Microsoft Learn: Tutorial: Migrate SQL Server to Azure SQL Database (offline)

Plan rollback before cutover: Arc has no reverse migration

REDUCES RISK

Microsoft says reverse migration is not available in the Arc migration experience. Once users write to Azure, going back needs native backup and restore or a link configured in SSMS.

WHAT TO DO

Write the rollback method and its time into the cutover runsheet, keep the source read-only until sign-off, and test the reverse path if the database is critical.

Microsoft Learn: Migrate to Azure SQL Managed Instance with Azure Arc

Use the Arc assessment to right-size, not the old server

COST LEVER

On-premises SQL Servers are often sized for hardware refresh cycles. The Arc assessment uses 30 days of performance history to recommend a tier and configuration.

WHAT TO DO

Check the SKU Recommendation tab, compare it with your month-end peaks, and set Azure Hybrid Benefit in Assessment settings if you have SQL Server licences with Software Assurance. Check the vendor's current pricing.

Microsoft Learn: Assess migration readiness with Azure Arc

FINISH SAFELY

How do you check it worked, or roll back?

Full validation checklist

8 checks

Run these before you declare the migration complete. Record the results in the data validation report.

Check	How to check	What good looks like
Every database in scope is online on the target	In SSMS connected to the target, run `SELECT name, state_desc FROM sys.databases;`. For Log Replay Service, run `az sql midb log-replay show` first; for the MI link, check Arc > Database migration > Monitor migrations.	Every migrated database shows ONLINE; the migration job shows completed with no errors.

Check	How to check	What good looks like
Row counts and checksums match	Run the row count query from the Validate step on the source and the target, and compare checksums for key tables. Record both in the data validation report.	Row counts match exactly for every table at the cutover point and checksums match for key tables, including computed columns and defaults DMS may not copy.
Logins and database users are mapped	Sign in to the target with each application login. In each database, run `SELECT name, type_desc FROM sys.database_principals;` and check for users without a matching login.	Every application login connects and no database user is orphaned.
Jobs, linked servers and credentials work	On Managed Instance or a SQL Server VM, check SQL Server Agent > Jobs > Job Activity Monitor in SSMS. On Azure SQL Database, check the replacement you chose (for example elastic jobs) has run.	Every job from your scripted list ran successfully at its next schedule, and linked servers you kept test successfully.
Applications connect to the new endpoint	From each application server, run `Test-NetConnection sql-contoso-aue.database.windows.net -Port 1433`, then run the application smoke tests.	TcpTestSucceeded is True from every application server and the smoke tests pass.
Performance matches the baseline	Run the same workload you captured in the performance baseline and review Query Performance Insight (Azure SQL Database) or Query Store and Azure Monitor metrics (Managed Instance) for CPU, IO and log usage.	Key transactions run within the baseline and the service tier is not at its limits during month-end peaks.

Check	How to check	What good looks like
Backups and alerts are in place on the target	In the Azure portal open the database or Managed Instance > Backups (or Data management > Backups) and check the retention settings. Check alert rules under Monitoring > Alerts.	Point-in-time restore is available with the retention you agreed and alerts reach a monitored mailbox.
The source no longer receives writes	On the source SQL Server, run `sp_who2` in SSMS and check the database is set to read-only.	No application sessions on the source after cutover and the database is read-only.

Rollback plan

If it goes wrong

ROLL BACK IF

- Row counts or checksums do not match after the final sync
- Application smoke tests fail against the new endpoint and cannot be fixed in the window
- A blocker missed in the assessment appears in production (for example a cross-database query or CLR assembly)
- Performance under real load is far below the baseline and scaling up does not fix it

POINT OF NO RETURN

Until applications write to Azure, rollback is simple because the source stays intact and read-only. After that, the Arc migration methods have no reverse migration, and Log Replay Service and native restore only run into Azure. For Managed Instance, Microsoft documents restoring a database back to SQL Server 2022 or 2025, or failing back through the link, only when the instance uses the matching update policy; switching to the Always-up-to-date policy removes that option permanently (see Microsoft Learn: Update policy in Azure SQL Managed Instance). Decommissioning the source server is the final point of no return.

1

Stop writes to Azure

Where: The application (maintenance mode or stopped services)

Stop the application so no more data is written to the Azure target.

✓ **You should see:** No application sessions on the target apart from admin connections.

2

Make the source writable again

Where: SSMS query window connected to the source SQL Server

Run `ALTER DATABASE SalesDb SET READ_WRITE;` and re-enable the SQL Agent jobs you disabled at cutover.

✓ **You should see:** The source accepts writes from a test login and jobs show as enabled.

3

Switch connection strings back

Where: Application configuration files or app settings, DNS aliases and firewall rules

Restore the old connection strings and DNS aliases, and reopen any firewall rules you closed at cutover.

✓ **You should see:** Application servers connect to the source and pass the smoke tests.

4

Bring back changes made in Azure SQL Database

Where: Command prompt on a machine with SqlPackage and access to both servers

Export a BACPAC from Azure SQL Database with `SqlPackage /Action:Export` (source server `sql-contoso-aue.database.windows.net`) and import it to a new database on the source server, then copy back the rows you need. Transactional consistency needs no writes during the export.

✓ **You should see:** The source holds the changes made in Azure, checked with row counts.

5

Bring back changes made in Managed Instance

Where: SSMS connected to the Managed Instance, then the source SQL Server

Check the update policy with `SELECT SERVERPROPERTY('ProductUpdateType');`. If it returns CU and your source is SQL Server 2022 or 2025, take a copy-only backup to URL from the Managed Instance and restore it on the source. Otherwise use a data export such as BACPAC or transactional replication you planned in advance.

✓ **You should see:** The source database contains the transactions made on the Managed Instance.

6

Record what cannot be undone

Where: Your Rollback plan and Decommission checklist

Note that a Log Replay Service job that was stopped cannot be restarted, that the Always-up-to-date update policy cannot be reversed, and that a decommissioned source server must be rebuilt from backup. Keep the source read-only and the full backup chain until sign-off.

✓ **You should see:** Decommission and any update policy change are scheduled after the rollback window with sign-off.

Agree the triggers and who decides before the go / no-go meeting. Use the rollback plan template.

Common problems and fixes

5 errors

`Cannot find the Azure SQL Migration extension`

Cause: Azure Data Studio and the extension were retired on 28 February 2026.

Fix: Use SQL Server enabled by Azure Arc, Azure DMS in the portal, or DMS PowerShell and Azure CLI instead.

`Log Replay Service database stuck in restoring state`

Cause: The database stays restoring until you complete the migration.

Fix: Upload the final log backup, then run the complete step for Log Replay Service.

`RESTORE FROM URL fails with access denied`

Cause: The SAS token is missing permissions, expired, or includes the leading question mark.

Fix: Generate a new SAS with read and list permissions and recreate the credential without the leading question mark.

`BACPAC import fails on unsupported objects`

Cause: The database contains features Azure SQL Database does not support, such as cross-database references.

Fix: Fix the blockers found in assessment, or choose Managed Instance as the target.

`Application cannot log in after cutover`

Cause: Server logins or contained users were not recreated, or SIDs do not match.

Fix: Script logins with their SIDs from the source, or create contained database users.

Search the full error library →

FAQ

Frequently asked questions

Is Azure Data Studio still the tool to use? –

No. Azure Data Studio retired on 28 February 2026. Microsoft recommends Azure Arc, Azure DMS and the target-specific tools instead.

Which target is closest to on-premises SQL Server? –

SQL Server on Azure VM gives full compatibility. Azure SQL Managed Instance is the closest platform-as-a-service option.

How do I migrate with minimal downtime? –

Use the Managed Instance link for Managed Instance targets, or an online DMS migration for SQL Server on Azure VM.

Can I migrate from Amazon RDS for SQL Server to Azure? –

Yes. Microsoft lists Amazon RDS for SQL Server as a supported source for Log Replay Service into Azure SQL Managed Instance.

What is the best tool for assessment now? −

SQL Server enabled by Azure Arc provides readiness and right-sized target recommendations. Azure Migrate suits whole data centre assessments.

SOURCES

Official sources

11 official pages this guide is based on −

Vendors change limits and features, so confirm details on the source before you rely on them. Last checked 4 Nov 2026.

- Microsoft Learn: What is happening to Azure Data Studio
- Microsoft Learn: SQL Server to Azure SQL Managed Instance migration overview
- Microsoft Learn: SQL Server to Azure SQL Database migration overview
- Microsoft Tech Community: Alternatives after the deprecation of the Azure SQL Mig...
- Microsoft Learn: Azure SQL Migration extension (previous versions)
- Microsoft Learn: Migrate to Azure SQL Managed Instance with Azure Arc
- Microsoft Learn: Assess migration readiness with Azure Arc
- Microsoft Learn: Migrate databases with Log Replay Service
- Microsoft Learn: Log Replay Service overview
- Microsoft Learn: Tutorial: Migrate SQL Server to Azure SQL Database (offline)
- Microsoft Learn: Connect hybrid machines with Azure Arc-enabled servers