



Workshop: SQL Ground-to-Cloud

(This e-book is valid as of 09/27/2019)

A Microsoft Workshop from the SQL Server team



About this Workshop

Welcome to this Microsoft solutions workshop on SQL Server in on-premises, in-cloud and hybrid solutions. In this one-day workshop, you'll learn how SQL Server 2019 and Azure SQL help you solve real-world challenges.

The Modules in this workshop lead you through conceptual and hands-on topics ranging from the newest technical features in SQL Server to its implementation in all the platforms it runs on. You'll learn not only specific technologies, but how to assemble them into a complete solution based on customer needs and requests.

You'll start by learning about the latest improvements in SQL Server 2019, work with the Big Data Clusters configuration, and then learn about the ways you can leverage SQL in Microsoft Azure (and how to get there) - all with a focus on how to extrapolate what you have learned to create other solutions for your organization. You'll end the day with a "What to Use When" module explaining how to create your own solutions.

This Workshop contains lecture and hands on lab work, and is particularly useful for Solution Architects, Data Architects, Application Architects, Technical Sellers, and Application Developers. A laptop, Microsoft Azure account, and experience with SQL is considered a prerequisite.

This README.MD file explains how the workshop is laid out, what you will learn, and the technologies you will use in this solution.

(You can view all of the [source files for this workshop on this GitHub site, along with other workshops as well. Open this link in a new tab to find out more.](#))



Learning Objectives

In this day-long, hands-on Workshop you'll learn how to:

- Articulate the key differentiators between SQL Server on-prem, in Azure VM, in Azure SQL, and hybrid configurations
- Explain the different service tiers within Azure SQL, and what to choose when
- Understand how Azure SQL is secured and address security concerns
- Explain hybrid configurations for Azure SQL and SQL Server in common examples

- Experiment with various features of SQL Server and Azure SQL
- Understand additional Azure platform features for data pipelines and security
- Understand what services are available to migrate and modernize your entire SQL Server stack
- Make informed decisions about how your business or customers should modernize their data estate
- Use Big Data solutions

Role Applications

The concepts and skills taught in this workshop form the starting points for:

- Solution Architects and Developers, to understand how to put together an end to end solution.
- Data Professionals and DevOps teams, to implement and operate SQL Server systems on premises and in the cloud.
- Data Scientists, to understand the environment used to analyze and solve specific predictive problems.



Business Applications of this Workshop

Businesses require near real-time insights from ever-larger sets of data from a variety of sources. Many have not explored the improvements made in the latest versions of SQL Server, and some are only now exploring the cloud as a computing platform. As time has progressed, a more dramatic upgrade process may be required.

In addition to traditional Online Transaction Processing (OLTP) and Online Analytic Processing (OLAP) workloads, some industry examples of data processing from multiple sources of data at scale are:

Industry	Example
Retail	Demand Prediction Market-Basket Analysis

Finance	Fraud detection customer segmentation
Healthcare	Fiscal control analytics Disease Prevention prediction and classification Clinical Trials optimization
Public Sector	Revenue prediction Education effectiveness analysis
Manufacturing	Predictive Maintenance Anomaly Detection
Agriculture	Food Safety analysis Crop forecasting



Technologies and Topics covered in this Workshop

The information covered in this workshop includes the following technologies and topics - although you are not limited to these, they form the basis of the workshop. At the end of the workshop you will learn how to extrapolate these components into other solutions. You will cover these at an overview level, with references to much deeper training provided.

Technology/Concept	Description
SQL Server 2019 improvements (on-prem and in-cloud)	Covers challenges and solutions in performance, security, availability, data virtualization, and Linux and containers in SQL Server 2019 spanning on-premise, containers, Kubernetes, and cloud platforms.

Big Data Clusters for SQL Server (on-prem and in-cloud)	Explains the architecture for Big Data Clusters for SQL Server in on-premises installations, containers, Kubernetes, and on the Microsoft Azure platform and in hybrid configurations
SQL Server Virtual Machines in Microsoft Azure	Covers the tools, processes and procedures for SQL Server Virtual Machines on the Microsoft Azure platform
Azure SQL	Covers the tools, processes and procedures for Azure SQL (Managed Instance, Single Database, and Elastic Pool)
Migrating SQL Server installations to Microsoft Azure	Explains the tools and processes to migrate on-premises SQL Server installations to the Microsoft Azure platform
"What to Use When"	Teaches a complete set of tools and processes used to determine the best architecture for a given customer scenario on the Microsoft Azure platform



Before Taking this Workshop

You'll need a local system or Virtual Machine that you are able to install software on. The workshop demonstrations use Microsoft Windows as an operating system and all examples use Windows for the workshop. Optionally, you can use a Microsoft Azure Virtual Machine (VM) to install the software on and work with the solution.

You must have a Microsoft Azure account with the ability to create assets, specifically the Azure Kubernetes Service (AKS).

This workshop expects that you understand data structures and working with SQL Server and computer networks. This workshop does not expect you to have any prior data science knowledge, but a basic knowledge of statistics and data science is helpful in the Data Science sections. Knowledge of SQL Server, Azure Data and AI services, Python, and Jupyter Notebooks is helpful for the Big Data Clusters feature. AI techniques are implemented in Python packages. Solution templates are implemented using Azure

services, development tools, and SDKs. You should have experience working with the Microsoft Azure Platform.

If you are new to these, here are a few references you can complete prior to class:

- [Microsoft SQL Server](#)
- [Microsoft Azure](#)

Setup

[A full prerequisites document is located here](#). These instructions should be completed before the workshop starts, since you will not have time to cover these in class. *Remember to "Stop" any Virtual Machines from the Azure Portal when not taking the class so that you do incur charges (shutting down the machine in the VM itself is not sufficient).*

Intended Audience

The following roles will find this workshop useful. Others may also attend, as described in the Secondary Audience section.

Primary Audience:	Solution Architects and Data Professionals tasked with implementing modern Data Systems, Big Data, Machine Learning and AI solutions
Secondary Audience:	Security Architects, Developers, and Data Scientists
Level:	300
Type:	In-Person or Self-Paced

Length:	8-9 hours
---------	-----------



Workshop Modules and In-Class Agenda

This is a modular workshop, and in each section, you'll learn concepts, technologies and processes to help you complete the solution. The times shown below are for an instructor-led course, you may also take the modules in a self-paced fashion.

Module	Time	Topics
<u>01 - Introduction and Workshop Methodology</u>	9:00AM-9:15AM	Workshop introduction, logistics, setup check
<u>02 - Modernizing Your Data Estate with SQL Server 2019</u>	9:15AM-11:15AM	<p>This module covers challenges and solutions using the latest version of SQL Server including:</p> <ul style="list-style-type: none"> • Overall SQL Server 2019 • Intelligent Performance • What's New in Security • Mission Critical Availability • Data Virtualization • SQL Server Linux and Containers • What Else, Migration and Next Steps
<u>03 - Working with Big Data and Data Science (Big Data Clusters for SQL Server 2019)</u>	11:30AM-12:30PM	Abstraction levels, frameworks, architectures and components within SQL Server big data clusters

<u>04 - SQL Server on the Microsoft Azure Platform</u>	1:30PM-2:30PM	Covers the multiple ways to use SQL Server technologies on the Microsoft Azure Platform, along with the fundamentals of SQL in Azure with additional deeper resources provided. Topics covered include: <ul style="list-style-type: none"> • Azure SQL: SQL VMs, Single instance, Single database, Elastic pools • Fundamentals of SQL in Azure • Migration process
<u>05 - Migrating SQL Server to Azure</u>	2:45PM-3:45PM	Covers the migration workflow and tools for assessing, planning, and migrating SQL workloads to Azure that meets the business requirements. Some of the tools and topics (not exhaustive) covered are: <ul style="list-style-type: none"> • Azure Migrate • Data Migration Assistant • Azure Database Migration Service • Post-migration operations
<u>06 - What to use When</u>	3:45PM-5:00PM	Covers the decision process and provides tools for deciding on the proper technologies on-premises and in-cloud for a solution based on requirements and constraints.



Related Workshops

- [Other SQL Workshops by Microsoft](#)



Next Steps

Next, Continue to [Prerequisites](#)



Workshop: SQL Ground-to-Cloud

A Microsoft workshop from the SQL Server team



01 - Introduction and Workshop Methodology

In this workshop you'll cover using the latest SQL Server features, from on-premises to the cloud, for various solutions. You'll work through a series of modules that will lead you from working with the latest improvements in SQL Server, working with large sets of data (in on-premises, cloud, and hybrid configurations), to working with the SQL options in the Microsoft Azure cloud platform. You'll close out the workshop with a set of scenarios to discover "what to use when", with a complete set of resources to help you in your real-world implementations. Along the way you'll have Lab exercises walking you through a complete assessment and migration of an on-premises database to Azure SQL.

In each module you'll get more references, which you should follow up on to learn more. Also watch for links within the text - click on each one to explore that topic.



Workshop Scenario

The Workshop follows the process of evaluating a solution from "Ground To Cloud" - using various phases in a defined process:

- **01 Discovery:** The original statement of the problem from the customer
- **02 Envisioning:** A "blue-sky" description of what success in the project would look like. Often phrased as "*I can...*" statements
- **03 Architecture Design Session (ADS):** An initial layout of the technology options and choices for a preliminary solution

- **04 Proof-Of-Concept** (POC): After the optimal solution technologies and processes are selected, a POC is set up with a small representative example of what a solution might look like, as much as possible. If available, a currently-running solution in a parallel example can be used
- **05 Implementation:** Implementing a phased-in rollout of the completed solution based on findings from the previous phases
- **06 Handoff:** A post-mortem on the project with a discussion of future enhancements

In these modules you'll work through each of these phases for a fictional company called [Wide World Importers](#) (WWI) - a wholesale novelty goods importer and distributor operating from the San Francisco bay area in the United States. WWI will need to integrate their operations with other companies, such as [The Contoso company](#), a multi-national business with headquarters in Paris, France, and [Adventure Works Cycles](#) - a large, multinational company that manufactures and sells metal and composite bicycles to North American, European and Asian commercial markets. Another company, [Tailspin Toys](#), is the developer of several popular toys and online video games. Founded in 2010, and acquired shortly after by Wide World Importers, the company has experienced exponential growth since releasing the first installment of their most popular game franchise to include online multiplayer gameplay. They have since built upon this success by adding online capabilities to the majority of their game portfolio.

Each of these companies have different challenges, and you'll get hands-on experience in the Labs with each of them.

At the end of the Workshop, you'll be presented with additional needs from each of these companies, and get tools, processes and other assets that you can use to design your own unique solution. By the end of this Workshop, you'll have everything you need to repeat this process in real-world production scenarios.



Workshop Methodology

This is a modular workshop, designed to be used in a linear fashion - you'll start at Module 1 and work your way through to the last one. You can, however, use these modules independently in many cases, and the Workshop exercises point off to Labs, which can often be taken independently as well.

If you are working through the Workshop with an instructor, you'll get additional instructions on the Workshop and you may be provided with a special environment for

the class. You'll be notified if that's the case when you sign up for the Workshop. Otherwise, if you are following along on your own, [Make sure you check out the prerequisite page before you start](#). You'll need all of the items loaded there before you can proceed with the Workshop Labs. If you plan to just read through the Workshop, the Labs are optional.



Workshop Modules and Agenda

This is a modular workshop, and in each section, you'll learn concepts, technologies and processes to help you complete the solution. The times shown below are for an instructor-led course, you may also take the modules in a self-paced fashion.

Module	Time	Topics
01 - Introduction and Workshop Methodology	9:00AM-9:15AM	Workshop introduction, logistics, setup check
02 - Modernizing Your Data Estate with SQL Server 2019	9:15AM-11:15AM	<p>This module covers challenges and solutions using the latest version of SQL Server including:</p> <ul style="list-style-type: none">• Overall SQL Server 2019• Intelligent Performance• What's New in Security• Mission Critical Availability• Data Virtualization• SQL Server Linux and Containers• What Else, Migration and Next Steps
03 - Working with Big Data and Data Science (Big Data Clusters for SQL Server 2019)	11:30AM-12:30PM	Abstraction levels, frameworks, architectures and components within SQL Server big data clusters

<u>04 - SQL Server on the Microsoft Azure Platform</u>	1:30PM-2:30PM	Covers the multiple ways to use SQL Server technologies on the Microsoft Azure Platform, along with the fundamentals of SQL in Azure with additional deeper resources provided. Topics covered include: <ul style="list-style-type: none"> • Azure SQL: SQL VMs, Single instance, Single database, Elastic pools • Fundamentals of SQL in Azure • Migration process
<u>05 - Migrating SQL Server to Azure</u>	2:45PM-3:45PM	Covers the migration workflow and tools for assessing, planning, and migrating SQL workloads to Azure that meets the business requirements. Some of the tools and topics (not exhaustive) covered are: <ul style="list-style-type: none"> • Azure Migrate • Data Migration Assistant • Azure Database Migration Service • Post-migration operations
<u>06 - What to use When</u>	3:45PM-5:00PM	Covers the decision process and provides tools for deciding on the proper technologies on-premises and in-cloud for a solution based on requirements and constraints.



Next Steps

Next, Continue to [02 - Modernizing Your Data Estate with SQL Server 2019.](#)



Workshop: SQL Ground-to-Cloud

A Microsoft workshop from the SQL Server team



02 - Modernizing Your Data Estate with SQL Server 2019

SQL Server 2019 is a modern data platform designed to tackle the challenges of today's data professional. SQL Server 2019 is not just an upgrade option from previous releases of SQL Server.

The WideWorldImporters company has been evaluating upgrading several SQL Server 2008 instances along with some instances of SQL Server 2012, 2014, and even 2016. WideWorldImporters wants to know whether SQL Server 2019 will solve some of the challenges they face including:

- Improving query performance without having to make application changes
- Classifying key data columns and being able to audit access to these columns to meet the needs of GDPR compliance.
- Providing better database availability when applications use long-running transactions.
- Allowing the WideWorldImporters team to access data from sources like Oracle, Azure SQL Database, and Azure CosmosDB without having to develop expensive ETL jobs.
- The database team at WideWorldImporters has evaluated SQL Server on Linux but SQL Server 2017 didn't include Replication, a feature they need for their application. They would like to evaluate how SQL Server Replication on Linux works. They also want to understand more about container technology and how it can be used with SQL Server
- WideWorldImporters also would like to know other capabilities exist in SQL Server 2019 that might help them before more efficient and extend the capabilities of T-SQL.
- WideWorldImporters also would like to learn more about how they can plan and execute a migration to SQL Server 2019 and reduce their risk for upgrades.

- WideWorldImporters is also evaluating Azure SQL so wants to know what features in SQL Server 2019 also work in Azure.

In this workshop module, you will see the capabilities in SQL Server 2019 that can provide solutions for these challenges.

NOTE: [Make sure you check out the prerequisite page before you start.](#) You'll need all of the items loaded there before you can proceed with the workshop. Instructor led workshops may have provided all the resources in the prerequisites.



Activity: SQL Server 2019 Lab

In this activity, you will complete a series of modules from the **SQL Server 2019 Lab**.

Instructor led workshops will use modules from the SQL Server 2019 lab along with the PowerPoint slides [SQL Server 2019 Modern Data Platform](#). Instructor led labs may not go through all modules in the SQL Server 2019 lab.



Activity Steps

To complete this module, go

to <https://github.com/microsoft/sqlworkshops/blob/master/sql2019lab/README.md>.

When you are done with the SQL Server 2019 lab you have completed Module 02 of the SQL Server Ground to Cloud Workshop! Proceed to the next module using the link in Next Steps below.



For Further Study

- [What's new in SQL Server 2019](#)
- [The SQL Server 2019 Lab](#)
- [Download and try SQL Server 2019](#)



Next Steps

Next, Continue to [03 - Working with Big Data Clusters on SQL Server 2019](#).

Lab: SQL Server 2019 (RC)

A Microsoft Course from the SQL Server team



About this Workshop

Welcome to this Microsoft solutions workshop on *SQL Server 2019 Lab*.

SQL Server 2019 is a modern data platform designed to tackle the challenges of today's data professional. SQL Server 2019 is not just an upgrade option from previous releases of SQL Server. Consider today's modern data challenges for a company called WideWorldImporters

Modernizing the WideWorldImporters Company

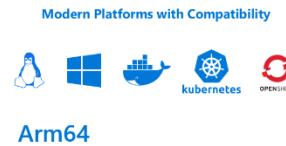
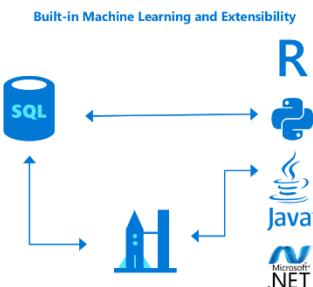
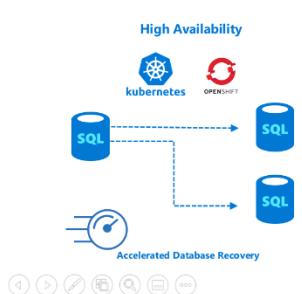
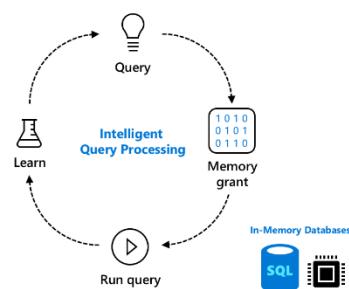
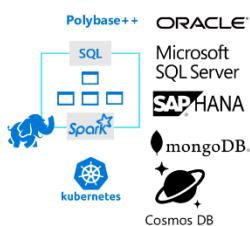
- Mixture of SQL Server 2008, 2014, and 2016 servers
- Expensive ETL applications for data sources outside of SQL Server
- Should we use "Big Data" technologies?
- Query performance tuning expensive
- Some database applications suffer from I/O performance
- Need data secure end-to-end with classification and auditing
- Applications need better availability and less downtime
- Our company uses a mixture of different operating system platforms
- Popularity of containerized applications growing
- What, when, and how should I migrate to the cloud?
- Desire to build more intelligent applications with AI and Machine Learning



SQL Server 2019 includes new capabilities that can solve these challenges. Consider the following infographic

SQL Server 2019

Solving Modern Data Challenges



SQL Server 2019 provides many new capabilities including:

- Data Virtualization with Polybase and Big Data Clusters to reduce the need for data movement
- Intelligent Performance to boost query performance with no application changes
- Security enhancements such as Always Encrypted and Data Classification
- Mission Critical Availability including enhancements for Availability Groups, built-in HA from Kubernetes and OpenShift, and Accelerated Database Recovery
- Modern Development capabilities including Machine Learning Services and Extensibility with Java and the language of your choice
- SQL Server on the platform of your choice with compatibility including Windows, Linux, Docker, Kubernetes, and Arm64 (Azure SQL Database Edge)

In this course you will learn some of the concepts and how to solve modern data challenges using a hands-on lab approach.

This course is intended to be taken as a self-paced lab in 2 to 4 hours but can be instructor-led to broaden the discussion or go deeper into specific subjects. A supplement slide deck is available for this course in the [slides](#) folder.

This course is designed for data professionals who have a basic working knowledge of SQL Server and the T-SQL language.

This **README.MD** file explains how the lab is structured, what you will learn, and the technologies you will use in this solution.



Learning Objectives

When you complete this course, you will be able to:

- Understand and use **Intelligent Performance** features to boost query performance with application changes
- Understand and use **Data Classification and Auditing** to meet the needs of compliance and regulation standards.
- Understand and use **Accelerated Database Recovery** to increase database availability.
- Understand and use **Polybase** to connect and query other data sources with no data movement.
- Understand and use **Docker containers** to deploy **SQL Server Replication on Linux**.
- Learn more about **additional capabilities** of SQL Server 2019, **Migration** tools, and **Database Compatibility**.

As part of taking this lab you are also learning about new capabilities in **Azure SQL Database**.

The following features in this lab also exist in Azure SQL Database:

- Intelligent Query Processing
- Data Classification and Auditing
- Accelerated Database Recovery

In addition, SQL Server Polybase allows you to connect to Azure SQL Database, Azure SQL Data Warehouse, and Azure CosmosDB.



Business Applications of this Workshop

- Boosting Database Performance with no application changes
- Classifying data for industry or regulatory compliance
- Ensuring data is highly available

- Reducing costs of expensive data movement applications
- Choosing different platforms for SQL Server and taking advantage of containerized applications.
- Learning tools and techniques to modernize and migrate to SQL Server 2019



Technologies used in this Workshop

Technology	Description
<i>SQL Server</i>	Database Platform produced by Microsoft
<i>SQL Server 2019</i>	Most current release of SQL Server currently in preview
<i>Intelligent Query Processing</i>	Automated query processing enhancements in SQL Server 2019
<i>Query Store</i>	Built-in query performance execution statistics stored in a user database
<i>Data Classification</i>	Built-in data information classification with SQL Server with auditing
<i>Accelerated Database Recovery</i>	Turbocharged Recovery, fast rollback, and aggressive transaction log truncation
<i>Polybase</i>	Data Virtualization through external tables
<i>Linux</i>	Operating system used in Containers and Container Orchestration
<i>Docker</i>	Engine for running and manage containers

<i>SQL Server Management Studio (SSMS)</i>	Graphical User Interface Management and Query Tool
<i>Azure Data Studio</i>	Graphical User Interface to execute T-SQL queries, notebooks, and manage SQL Server



Before Taking this Workshop

To complete this workshop you will need the following:

- Clone the workshop using git from <https://github.com/microsoft/sqlworkshops.git>. All the scripts and files in the labs are found in the **sql2019lab** folder.
- On Windows systems, you should use the following git syntax

```
git clone --config core.autocrlf=false
https://github.com/microsoft/sqlworkshops.git
```
- Install the software as listed in the **Setup** section below

Each module of this workshop can be studied and used independently of each other or taken all as a single lab. The Modules are designed in a sequence but you can use each of them one at a time at your own pace.



Setup

In order to complete this workshop you need to install the following software:

- SQL Server 2019 Release Candidate or later. You can run all of the activities from this workshop on an installed SQL Server on Windows, Linux, or Containers. You can use the client tools on a separate computer or VM provided it has access to connect to SQL Server.
 - For **Modules 1, 2, and 3** you only need the database engine installed
 - Module 3** requires disk space to hold a database with a 10Gb data and 10Gb or 40Gb log file.
 - Module 4** requires you to install and enable Polybase (you don't need the Java option and you can choose a stand-alone Polybase.)

- **Docker is required only for Module 5.** You can use Docker for Windows, Linux, or MacOS.
- Install SQL Server Management Studio (SSMS) 18.2 or higher from <https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms>. Several of the modules require features built only into SSMS.
- Install Azure Data Studio June 2019 or higher from <https://docs.microsoft.com/en-us/sql/azure-data-studio/download>. T-SQL notebooks are used extensively in this course.
- The WideWorldImporters sample backup from <https://github.com/Microsoft/sql-server-samples/releases/download/wide-world-importers-v1.0/WideWorldImporters-Full.bak>. You can find example scripts to restore the WideWorldImporters database in the **sql2019lab** folder.
- Access to an **Azure SQL Database** is required for **Module 4**.

This workshop was built and designed for a computer or VM to run SQL Server with at least 8Gb RAM and 4 CPUs. You may be able to run this lab with less compute resources.

NOTE: If you run this lab from a virtual machine in Azure running Windows, and you want to use Module 5, you will need to enable nested virtualization. Read more at <https://docs.microsoft.com/en-us/azure/virtual-machines/windows/nested-virtualization>



Workshop Details

This workshop uses SQL Server 2019, SQL Server Management Studio, Azure Data Studio, Docker containers, and Azure SQL Database for you to learn how you can solve modern data challenges with SQL Server 2019.

Primary Audience:	Data professionals looking to understand and use new capabilities of SQL Server 2019
Secondary Audience:	Developers, Architects, and IT Pros

Level:	300
Type:	Self-Paced or Instructor Led
Length:	1-2 hours (self-paced) 2-4 hours (Instructor Led)



Related Workshops

- [SQL Server Big Data Clusters Architecture](#)
- [SQL Server 2019 on OpenShift](#)



Workshop Modules

This is a modular workshop, and in each section, you'll learn concepts, technologies, and processes to help you complete the solution. This table is provided for you to see the list of modules in the workshop. You can use any module in any order you like but the preferred method is to **proceed to Next Steps below to start the workshop.**

Module	Topics
01 - Intelligent Performance	Learn the how SQL Server can boost your performance with no application changes
02 - Security	Learn new security features of SQL Server 2019 such as Data Classification and Auditing
03 - Availability	Learn new capabilities to make your SQL Server more available such as Accelerated Database Recovery
04 - Data Virtualization	Learn how to use SQL Server as a data hub and reduce data movement using Polybase++

<u>05 - Linux and Containers</u>	Learn how to use Docker containers to deploy SQL Server Replication on Linux.
<u>06 - Additional Capabilities, Migration, and Next Steps</u>	Learn more about Additional Capabilities in SQL Server 2019, Migration Tools, Database Compatibility, and Next Steps



Next Steps

Next, Continue to [Intelligent Performance](#)



Workshop: SQL Server 2019 Lab (RC)

A Microsoft workshop from the SQL Server team



SQL Server Intelligent Performance

SQL Server 2019 includes new capabilities designed to enhance your performance with no application changes. These enhancements include:

- Intelligent Query Processing
- Lightweight Query Profiling
- Sequential Key Insert Performance
- In-Memory Database
 - Hybrid Buffer Pool
 - Memory Optimized Tempdb Metadata
 - Persistent Memory Support

You can learn more about all of these enhancements at <https://docs.microsoft.com/en-us/sql/sql-server/what-s-new-in-sql-server-ver15?view=sqlallproducts-allversions>.

You will cover the following topics in this Module:

1.0 SQL Server Intelligent Query Processing

1.1 Using Query Store for performance analysis



1.0 SQL Server Intelligent Query Processing

In this module you will learn about the Intelligent Query Processing capabilities in SQL Server 2019.

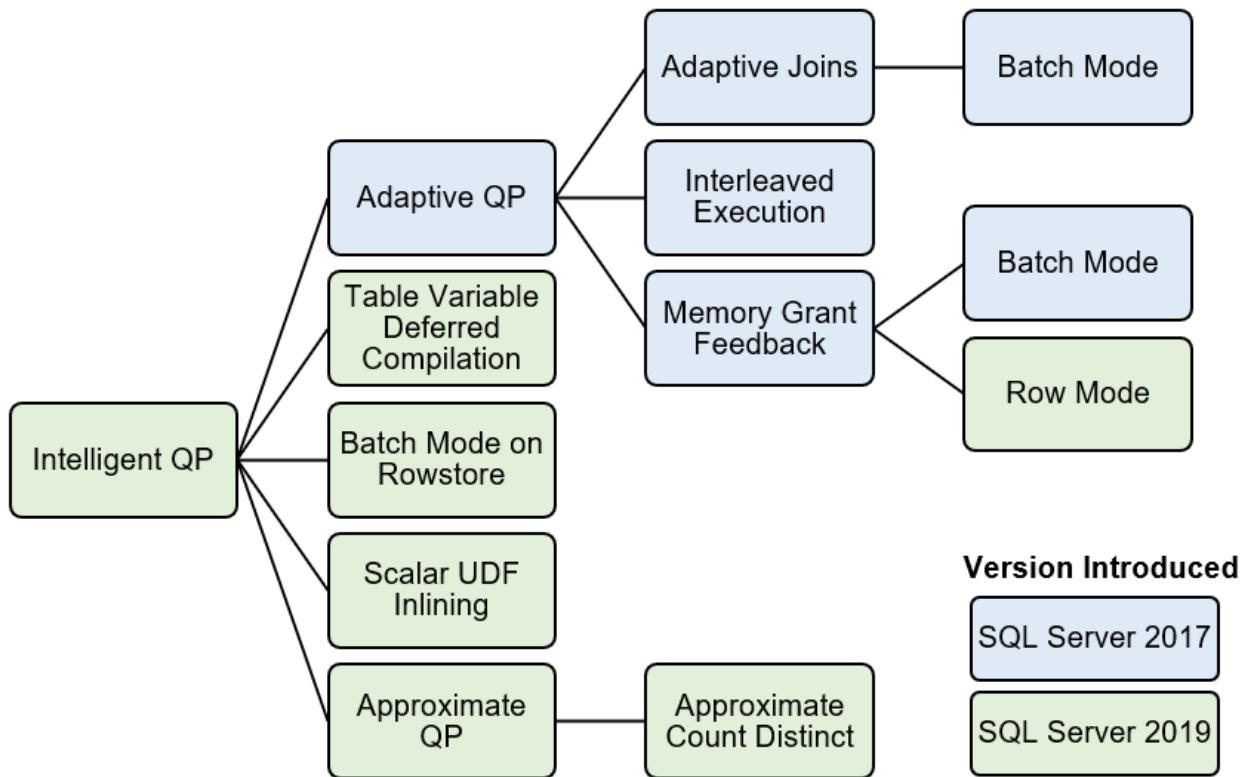
The Challenge

Application developers and DBAs want to gain performance for queries without making application changes. Query tuning can be an expensive undertaking so they want the query processor to adapt to their workload needs vs having to use options and flags to gain performance.

The Solution

Intelligent Query processing is a suite of features built into the query processor for SQL Server 2019 allowing developers and data professionals to accelerate database performance automatically **without application changes**. T-SQL queries simply need to be run with a database compatibility level of 150 to take advantage of these enhancements.

You can read more about database compatibility at <https://docs.microsoft.com/en-us/sql/t-sql/statements/alter-database-transact-sql-compatibility-level#compatibility-levels-and-sql-server-upgrades>. The following is a diagram showing the features of Intelligent Query Processing including capabilities from SQL Server 2017 and SQL Server 2019:



Intelligent Query Processing is a feature that exists for both SQL Server 2019 and Azure SQL Database. You can read the documentation for a description and example of all of these features at <https://docs.microsoft.com/en-us/sql/relational-databases/performance/intelligent-query-processing>.

NOTE: One of the features of Intelligent Query Processing, approximate count distinct, does not require database compatibility of 150

Now proceed to the Activity to learn an example of how Intelligent Query Processing can accelerate query performance automatically with no application changes.



Activity: SQL Server Intelligent Query Processing

In this activity, you will learn how to use the built-in capabilities of Intelligent Query Processing in SQL Server 2019 simply by changing the database compatibility of WideWorldImporters to version 150 with no application changes.

NOTE: If at anytime during the Activities of this Module you need to "start over" you can go back to the first Activity in 1.0 and run through all the steps again.

You have been provided a stored procedure called **CustomerProfits** which you will deploy in the **Facts** schema of the **WideWorldImporters** database. The stored procedure uses a *table variable* to store interim results from a user table and then uses that table variable to join with other data in the **WideWorldImporters** database. In previous releases of SQL Server, this design pattern could cause an issue, since SQL Server would always estimate that the table variable only contains 1 row of data. This can cause issues with building the optimal query plan for the best performance.

SQL Server 2019 Intelligent Query Processing includes a capability called *table variable deferred compilation* to improve the performance of T-SQL code that uses table variables. The application simply needs to change the database compatibility level to 150, which is the default for SQL Server 2019, and execute the T-SQL statements with table variables to see a gain in performance.

The **WideWorldImporters** database example was created using SQL Server 2016 which has a default database compatibility level of 130. When a database is restored from a previous version of SQL Server, the compatibility level of the database is preserved to help reduce the risk of upgrades.

You will observe the performance of the **CustomerProfits** stored procedure with database compatibility level of 130 on SQL Server 2019. You will then compare the performance of the same procedure with no changes with a database compatibility of 150 which will enable the query processor to use table variable deferred compilation.

✓ Activity Steps

All scripts for this activity can be found in the **sql2019lab\01_IntelligentPerformance\iqp** folder.

Follow these steps to observe performance differences with table variable deferred compilation.

STEP 1: Restore the WideWorldImporters backup

NOTE: If you have restored the WideWorldImporters database backup in other modules, you can skip this step.

Use a tool like SQL Server Management Studio (SSMS) or Azure Data Studio (ADS) to execute the T-SQL script **restorewwi.sql** as found in the **sql2019lab\01_IntelligentPerformance\iqp** folder to restore the WideWorldImporters backup. The script assumes a specific path for the backup and

database/log files. You may need to edit this depending on your installation. *Remember for Linux installations, the default path is /var/opt/mssql/data.* Your instructor may have provided this backup for you but if necessary you can download it from <https://github.com/Microsoft/sql-server-samples/releases/download/wide-world-importers-v1.0/WideWorldImporters-Full.bak>.

STEP 2: Use a T-SQL notebook to complete the rest of the activity.

T-SQL notebooks provide a very nice method to execute T-SQL code with documentation in the form of markdown code. All the steps and documentation to complete the rest of the activity for Module 1.0 can be found in the T-SQL notebook **iqp_tablevariabledeferred.ipynb** which can be found in the **sql2019lab\01_IntelligentPerformance\iqp** folder.

NOTE: A T-SQL script **iqp_tablevariabledeferred.sql** is also provided if you want to go through the same steps as the notebook but use a tool like SQL Server Management Studio.

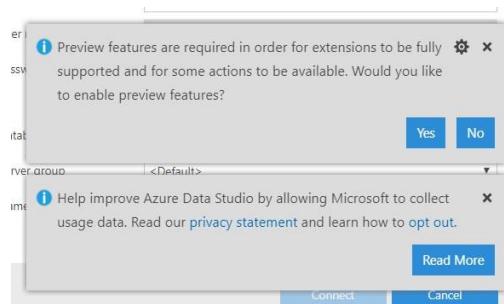
T-SQL notebooks can be executed with Azure Data Studio. If you are familiar with using Azure Data Studio and T-SQL notebooks open up the **iqp_tablevariabledeferred.ipynb** notebook and go through all the steps. When you are done proceed to the **ActivitySummary** section for the Activity below.

If you have never opened a T-SQL notebook with Azure Data Studio, use the following instructions:

Launch the Azure Data Studio application. Look for the icon similar to this one:

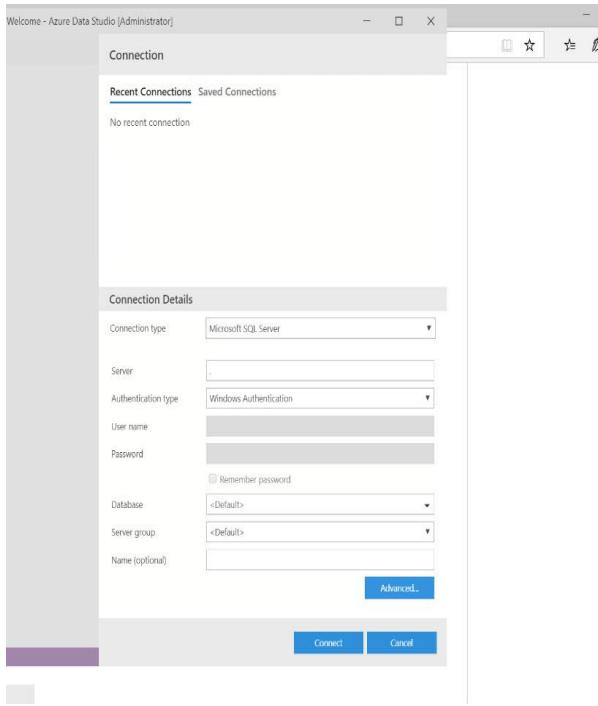


The first time you launch Azure Data Studio, you may see the following choices. For the purposes of this workshop, You can select No to loading 2019 preview extensions and click X to not read the usage data statement.

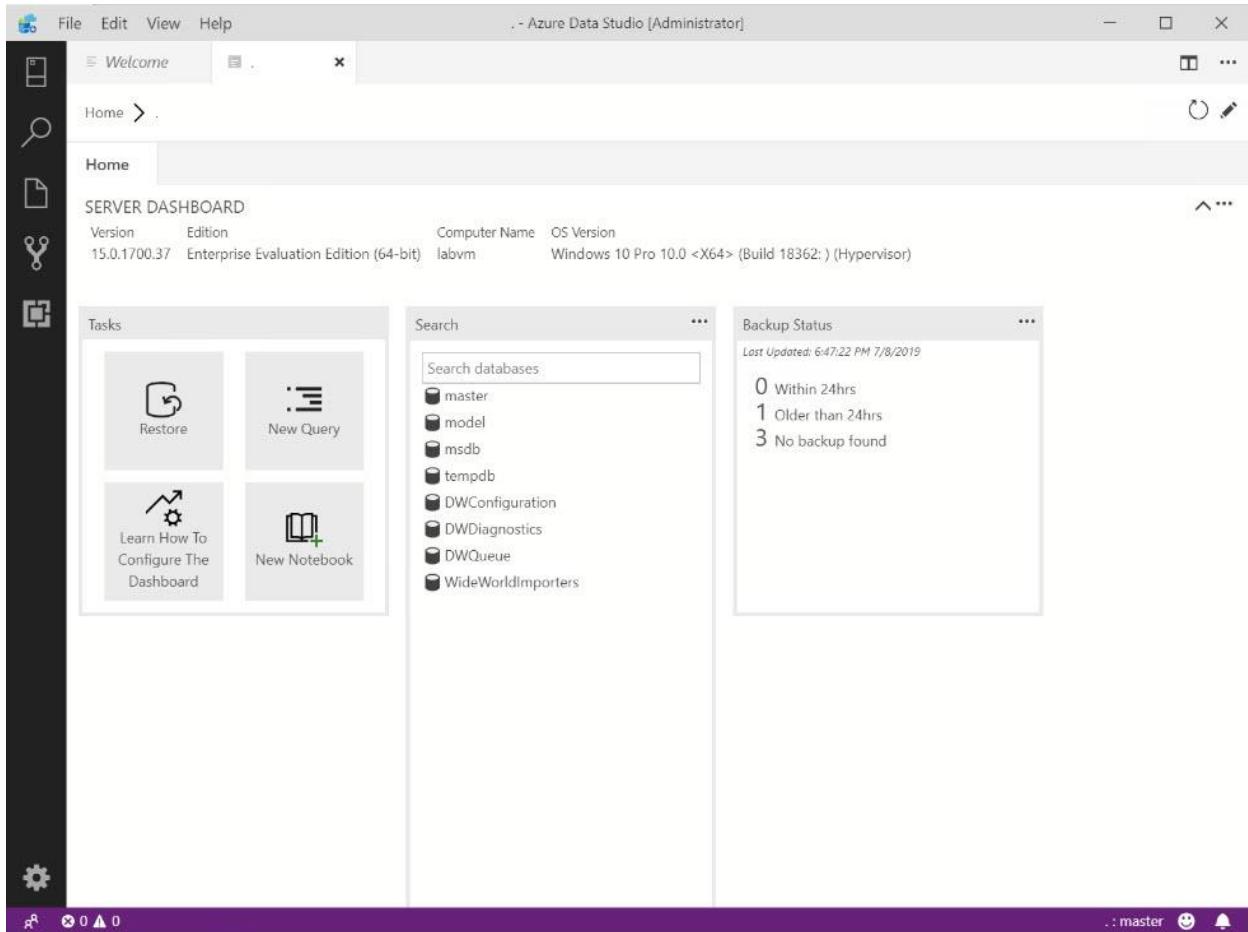


You will now be presented with the following screen to enter in your connection details for SQL Server. Use connection details as provided by your instructor to connect to SQL Server or the connection you have setup yourself for your SQL Server instance.

Now click the **Connect** button to connect. An example of a connection looks similar to this graphic (your server, Auth type, and login may be different):



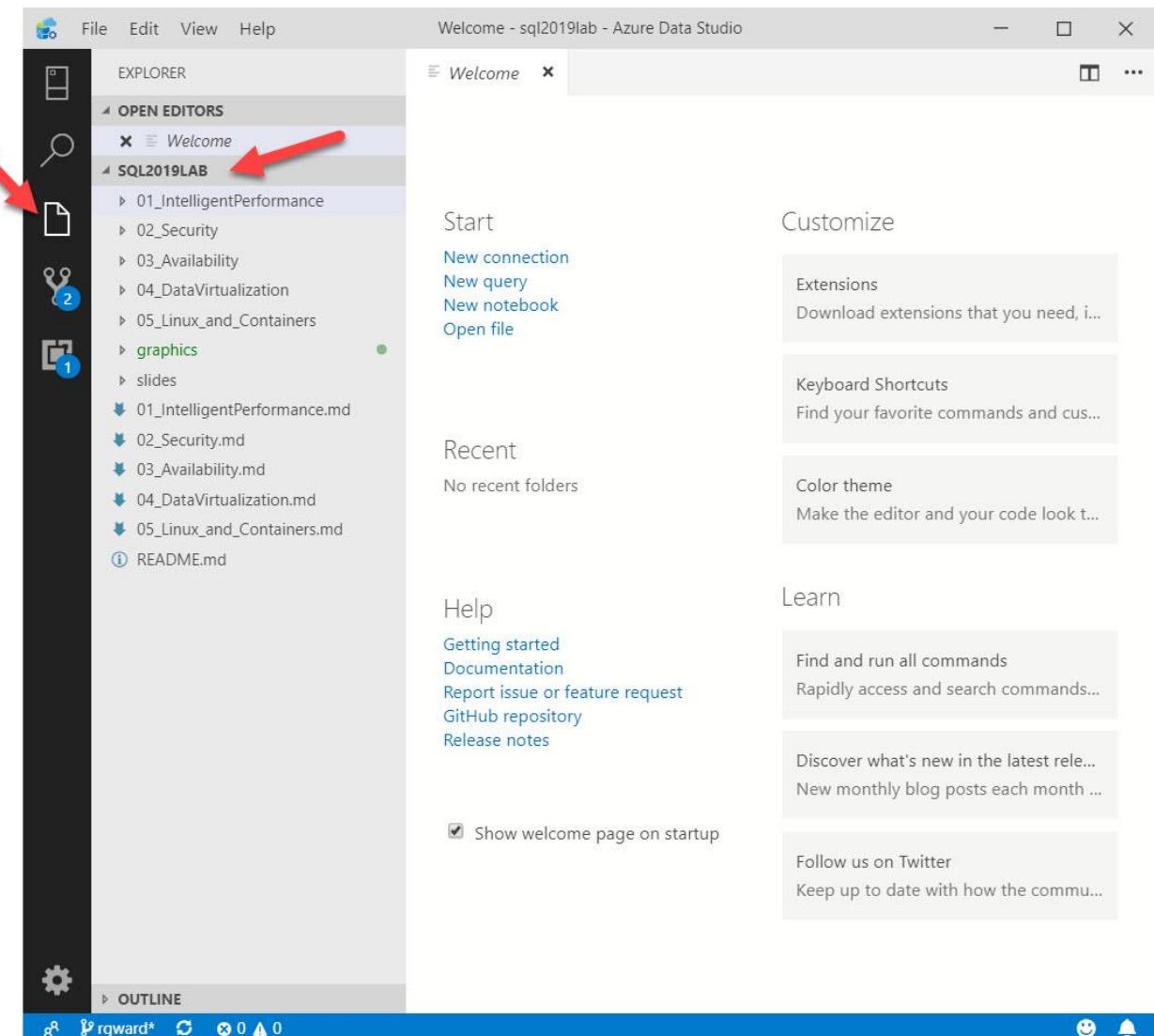
A successful connection looks similar to this (your server may be different):



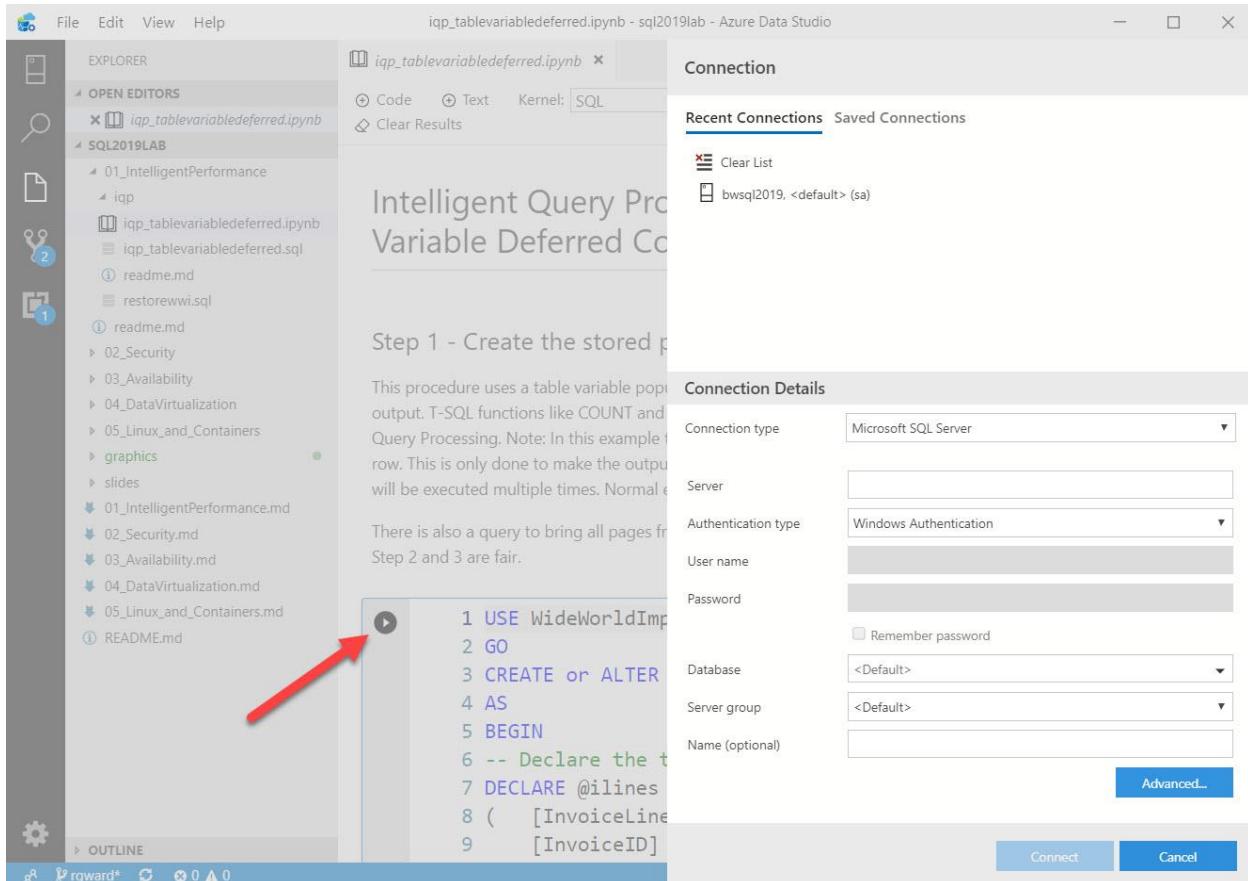
Use the power of Azure Data Studio Explorer to open up any file including notebooks. Use the File/Open Folder menu to open up the **sqlworkshops\sql2019lab** folder. Now click the Explorer icon on the left hand side of Azure Data Studio to see all files and directories for the lab. Navigate to the **01_IntelligentPerformance\iqp** folder, open up the **iqp_tablevariabledeferred.ipynb** notebook and go through all the steps.

NOTE: Be sure to only run one notebook cell at a time for the lab.

You can now use Azure Data Studio explorer to open up a notebook or script without exiting the tool.



When you start using a notebook and use the "Play" button of a cell, you may get prompted for the connection. Choose the connection you used when you first opened up Azure Data Studio.



There is additional documentation on how to use SQL notebooks at <https://docs.microsoft.com/en-us/sql/azure-data-studio/sql-notebooks>. When you are done proceed to the **Activity Summary** section for the Activity below.

Activity Summary

In this activity you have learned the powerful capabilities of Intelligent Query Processing by looking at an example where no application changes were required to boost performance. All that was required was to simply change the database compatibility level to 150 for the database. This example used a concept called table variable deferred compilation. You can read more about this capability at <https://docs.microsoft.com/en-us/sql/relational-databases/performance/intelligent-query-processing#table-variable-deferred-compilation>.



1.1 Using Query Store for Performance Analysis

In this module you will learn how to use the Query Store, a built-in performance analysis feature of SQL Server, to analyze the performance differences of the queries run in Module 1.0. Even though query store is not new to SQL Server 2019, it is an excellent feature to help analyze performance differences with Intelligent Query Processing.

The Challenge

Developers and DBA need to track query performance execution over time without having to "poll" Dynamic Management Views and save it to permanent storage. Data professionals need to easily be able to compare query execution for different query plans associated with the same query text.

The Solution

The Query Store is built into the query processing engine, enabled using an option for each database in SQL Server. Once enabled, performance statistics for queries are cached and stored in the SQL user database so they are persisted across server restarts.

In addition, the Query Store comes with a series of catalog and dynamic management views to gain insight into recorded query performance, including the ability to view execution details for different query plans over time associated with the same query.

You can read more about the Query Store at <https://docs.microsoft.com/en-us/sql/relational-databases/performance/monitoring-performance-by-using-the-query-store>. In addition, there were enhancements in Query Store for SQL Server 2019 for "capture policy" which you can read at <https://docs.microsoft.com/en-us/sql/t-sql/statements/alter-database-transact-sql-set-options?view=sqlallproducts-allversions>.



Activity: Using Query Store for Performance Analysis

The **WideWorldImporters** sample database that you restored in Module 1.0 has the Query Store feature enabled. If you performed the Activities in Module 1.0, the Query Store recorded performance information about each query execution.

NOTE: *If at anytime during the Activities of this Module you need to "start over" you can go back to the first Activity in 1.0 and run through all the steps again.*

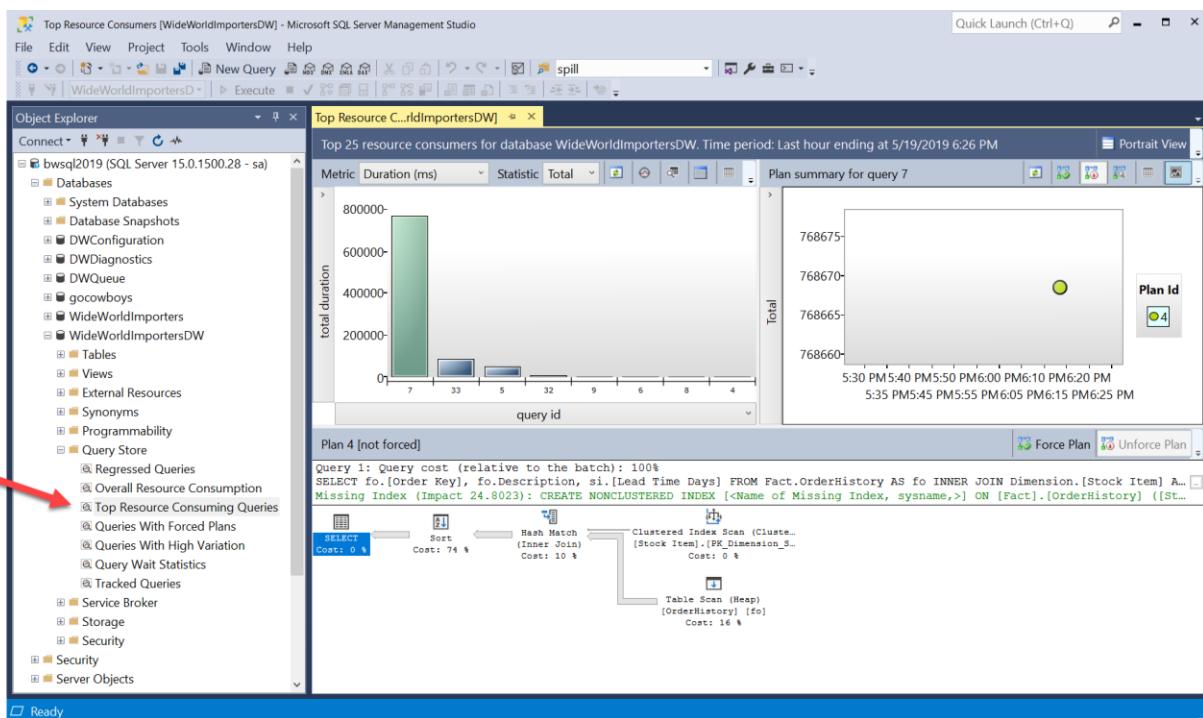
✓ Activity Steps

Work through the following steps to use the Query Store to examine the query performance differences for the CustomerProfits stored procedure when executed with database compatibility 130 versus 150.

STEP 1: Find Query Store Reports

Open SQL Server Management Studio (SSMS). Using Object Explorer, navigate to the WideWorldImporters database. Find the Query Store folder and select the **Top Resource Consuming Queries** report.

Your screen should look similar to the following



STEP 2: Find the query from the CustomerProfits procedure

There could be other queries in the Query Store. Click on the bars in the graph starting from left to right until you find the query for the query text

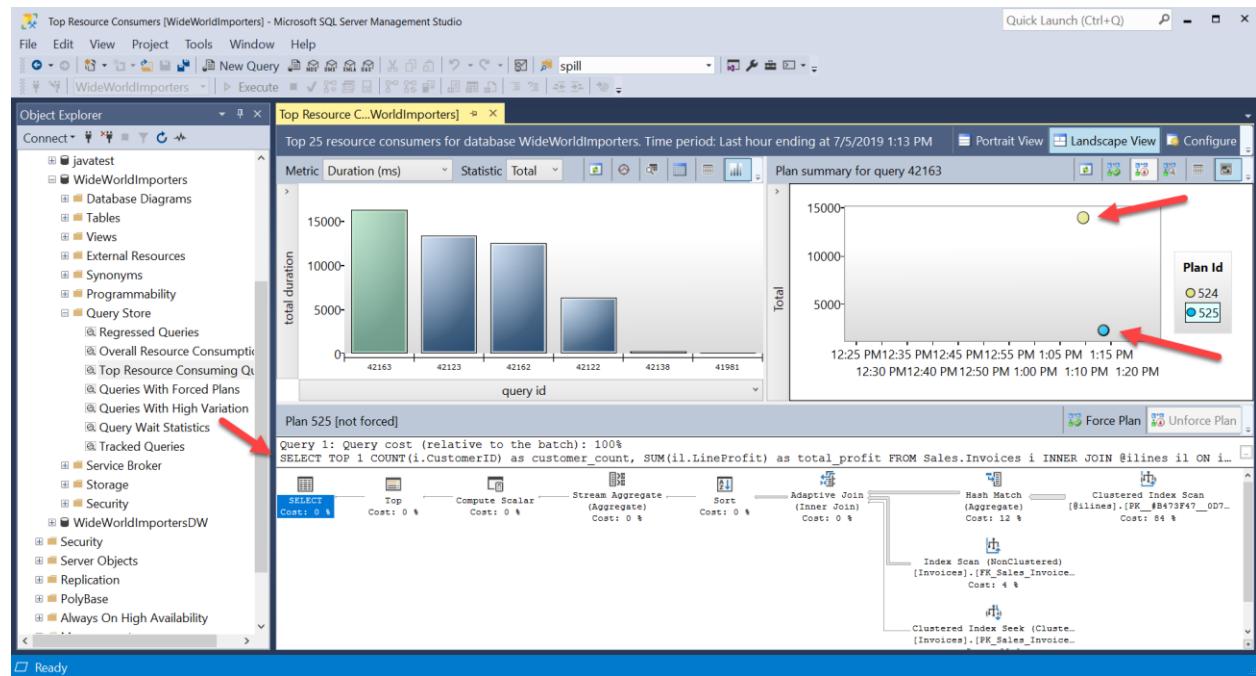
```
SELECT TOP 1 COUNT(i.CustomerID) as customer_count, SUM(il.LineProfit) as total_profit  
FROM Sales.Invoices i  
INNER JOIN @ilines il
```

```

ON i.InvoiceID = il.InvoiceID
GROUP By i.CustomerID
END
GO

```

When you find this query your screen should look similar to the following with the chart in the right hand corner showing two "dots" representing the variance in performance for two query plans for the same query text.



The "higher" the dot in the chart, the longer the average duration is for that query plan. Query store knows how to store changes in query plans for the same query text. In the example you executed, you ran the same stored procedure but with different dbcompat levels. A different query plan was generated when you switched to dbcompat 150 but didn't change the stored procedure. This is because the query processor used the table variable deferred compilation technique when building the new plan.

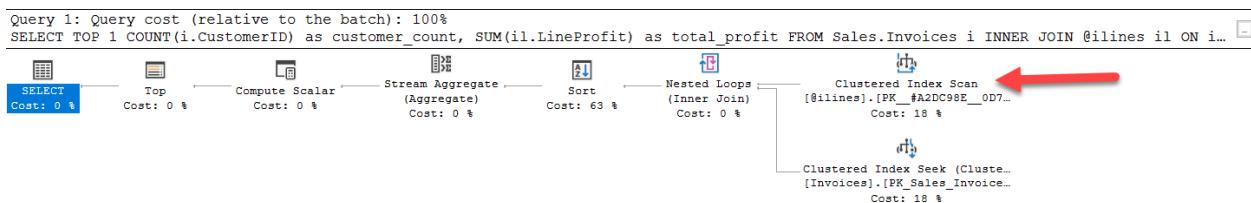
STEP 3: Observe query duration and plan differences

Click on the higher dot in the chart. Observe the plan in the lower hand window showing a Clustered Index Scan for the table variable. This is the query plan with dbcompat of 130. Move your cursor over the higher dot to show the query execution numbers for this plan.

Notice the average duration is less than 1 second but the total duration for all executions is ~13 seconds (your times may vary depending on compute resources for SQL Server). This may not seem all that bad but the business requires faster query execution.

Plan Id	524
Execution Type	Completed
Plan Forced	No
Interval Start	2019-07-05 13:09:00.000 -05:00
Interval End	2019-07-05 13:10:00.000 -05:00
Execution Count	25
Total Duration (ms)	13950.53
Avg Duration (ms)	558.02
Min Duration (ms)	491.68
Max Duration (ms)	1601.34
Std Dev Duration (ms)	214.5
Variation Duration (ms)	0.38

In the query plan window move your cursor over the Clustered Index Scan operator for the table variable and observe how SQL Server only estimates 1 row for the scan (even though the scan is 200K+ rows). A Nested Loops join is used with the Clustered Index Scan. That wouldn't be a problem if the table variable truly only had 1 row. But that is not a good plan choice if there are some 200K+ rows in the table variable.



Clustered Index Scan	
Scanning a clustered index, entirely or only a range.	
Physical Operation	Clustered Index Scan
Logical Operation	Clustered Index Scan
Estimated Execution Mode	Row
Storage	RowStore
Estimated I/O Cost	0.003125
Estimated Operator Cost	0.0032831 (18%)
Estimated CPU Cost	0.0001581
Estimated Subtree Cost	0.0032831
Estimated Number of Executions	1
Estimated Number of Rows	1
Estimated Number of Rows to be Read	1
Estimated Row Size	20 B
Ordered	False
Node ID	5
Object	
[@ilines].[PK__#A2DC98E_0D760AF93BF11711] [ii]	
Output List	
@ilines.InvoiceID, @ilines.LineProfit	

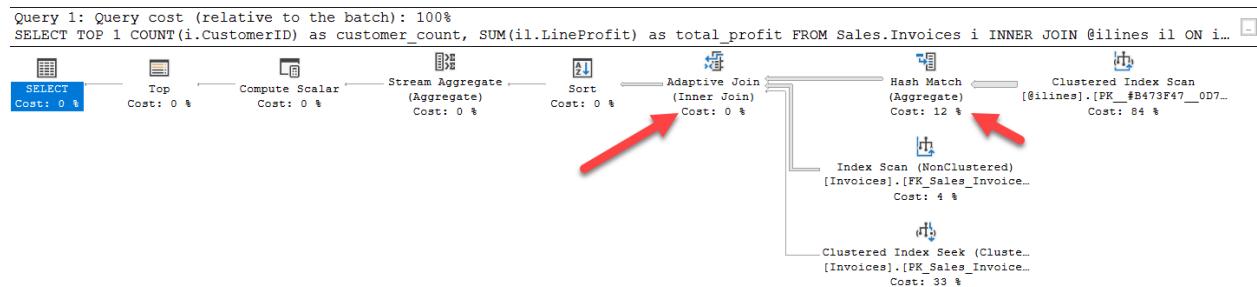
STEP 4: Observe plan and stats for the faster plan using table variable deferred compilation

Repeat the same process for the lower dot which represents the query plan that is faster using dbcompat of 150.

Move your cursor over the lower dot and observe the execution statistics. See how the average and total duration are significantly lower than before.

Plan Id	525
Execution Type	Completed
Plan Forced	No
Interval Start	2019-07-05 13:13:00.000 -05:00
Interval End	2019-07-05 13:14:00.000 -05:00
Execution Count	25
Total Duration (ms)	2284.7
Avg Duration (ms)	91.39
Min Duration (ms)	82.88
Max Duration (ms)	108.71
Std Dev Duration (ms)	5.93
Variation Duration (ms)	0.06

Click on the dot and observe the new query plan.



You will notice a clustered index scan is still used but this time with a Hash Join. Additionally, the query processor has used a concept called an Adaptive Join, which was introduced in SQL Server 2017. An adaptive join allows the query processor to choose the join method at execution time based on the number of rows from input of the operator. In this case, a Hash Join is the better choice. You can read more about Adaptive Joins at <https://docs.microsoft.com/en-us/sql/relational-databases/performance/intelligent-query-processing>.

Move your cursor over the Clustered Index Scan for the table variable. Notice the estimated rows is more accurate and the query processor has also chosen batch mode

for rowstore processing. This is another Intelligent Query Processing feature in SQL Server 2019 to make queries faster with no application changes.

Clustered Index Scan	
Scanning a clustered index, entirely or only a range.	
Physical Operation	Clustered Index Scan
Logical Operation	Clustered Index Scan
Estimated Execution Mode	Batch
Storage	RowStore
Estimated I/O Cost	3.6935
Estimated Operator Cost	3.94474 (84%)
Estimated CPU Cost	0.251249
Estimated Subtree Cost	3.94474
Estimated Number of Executions	1
Estimated Number of Rows	228265
Estimated Number of Rows to be Read	228265
Estimated Row Size	20 B
Ordered	False
Node ID	8
Object	
[@ilines].[PK_#B473F47_0D760AF915A5C3B3] [il]	
Output List	
@ilines.InvoiceID, @ilines.LineProfit	



In this example, the query processor used three different Intelligent Query Processing features to make this query faster with no application changes:

- Table Variable Deferred Compilation
- Adaptive Join
- Batch mode for rowstore

Activity Summary

In this activity you have seen how to use the Query Store for performance insights including the ability to see differences for the same query text of different query plans, including those that benefit from Intelligent Query Processing. You observed the SQL Server query processor using multiple techniques to make your queries faster with no changes to the stored procedure or application.

Proceed to the next module to learn about new **security** capabilities in SQL Server 2019.



For Further Study

- [Intelligent Query Processing in SQL Server](#)
- [Q&A about Intelligent Query Processing](#)
- [Monitoring performance of SQL Server using the Query Store](#)
- [What is Azure Data Studio?](#)
- [How to use Notebooks in Azure Data Studio](#)



Next Steps

Next, Continue to [*Security*](#).



Workshop: SQL Server 2019 Lab (RC)

A Microsoft workshop from the SQL Server team



SQL Server 2019 Security

SQL Server 2019 has new security enhancements including:

- Always Encrypted with Secure Enclaves
- Data Classification and Auditing
- Transparent Data Encryption (TDE) Suspend and Resume
- Improved Certificate Management
- Feature Restrictions

You can read more details about all of these enhancements at <https://docs.microsoft.com/en-us/sql/sql-server/what-s-new-in-sql-server-ver15?view=sqlallproducts-allversions>.

You will cover the following topics in this Module:

2.0 Data Classification

2.1 Auditing Data Classification



2.0 Data Classification

In this module you will learn about the Data Classification capabilities in SQL Server 2019.

The Challenge

Many organization face the challenge of classifying their data and auditing access to data that is classified. Regulations such as General Data Protection Regulation (GDPR) have requirements for business to be able to easily find and report on data access. Having the ability to classify data at the column level with SQL Server can assist to meet these type of regulations. Furthermore, auditing of data that is marked as classified can also assist with regulations and compliance from government agencies or industry standards.

The Solution

In SQL Server 2017, SQL Server Management Studio (SSMS) was enhanced to include the ability to classify data based on *labels* and *types* you could associate with columns in a table. This solution was built into the SSMS tool.

SQL Server 2019 includes data classification capabilities built-in to the SQL Server Engine through the new T-SQL statement

ADD SENSITIVITY CLASSIFICATION

Now classification information is stored directly with columns in metadata in system tables. There are two separate pieces of metadata for the classification feature

label - This represents the sensitivity of data but can be any string you choose. An example of a label would be *GDPR*.

information_type - This represents the type of data that is being classified. This can be any string you choose. An example of an information_type would be *Financial*

The added benefit of built-in classification is that now auditing of data classification is included. SSMS starting with version 18 (you should use 18.2 or greater) has been enhanced to take advantage of this new T-SQL feature.

NOTE: *The use of data classification and auditing with SQL Server does not imply an organization has met requirements like GDPR. SQL Server is providing some of the capabilities needed by organizations to meet certain regulations and compliance standards. It is up to a business or organization to use these tools to meet their requirements or regulations.*

Data Classification and Auditing is a feature that exists for both SQL Server 2019 and Azure SQL Database. You can read the documentation for data discovery and classification for SQL Server, Azure SQL Database, and Azure SQL Data Warehouse at <https://docs.microsoft.com/en-us/azure/sql-database/sql-database-data-discovery-and-classification>.

The best way to see how data classification and auditing can help an organization is to see it in action.

Proceed to the Activity to learn an example of how to use Data Classification with SSMS and with T-SQL.



Activity: Data Classification

In this activity, you will learn how to use SQL Server Management Studio (SSMS) and T-SQL to classify columns of tables in a database. You will use the **WideWorldImporters** sample database to classify specific columns in tables of that database.

NOTE: *If at anytime during the Activities of this Module you need to "start over" you can go back to the first Activity in 2.0 and run through all the steps again.*



Activity Steps

Follow these steps to classify certain columns in the WideWorldImporters database using SSMS and T-SQL. All scripts for this activity can be found in the **sql2019lab\02_Security\dataclassification** folder.

NOTE: SSMS 18.1 has a bug where data classification through the tools has an issue if the database compatibility level is NOT 150. Therefore, you must have SSMS 18.2 (or greater) to go through this activity or set the dbcompat of WideWorldImporters to 150 if using a version < SSMS 18.2.

STEP 1: Restore the WideWorldImporters backup.

NOTE: If you have restored the WideWorldImporters database backup in other modules, you can skip this step.

Use a tool like SQL Server Management Studio (SSMS) or Azure Data Studio (ADS) to execute the T-SQL script **restorewvi.sql** as found in the **sql2019lab\02_Security\dataclassification** folder to restore the WideWorldImporters backup. The script assumes a specific path for the backup and database/log files. You may need to edit this depending on your installation. *Remember for Linux installations, the default path is /var/opt/mssql/data.* Your instructor may have provided this backup for you but if necessary you can download it from <https://github.com/Microsoft/sql-server-samples/releases/download/wide-world-importers-v1.0/WideWorldImporters-Full.bak>.

STEP 2: Setup the activity

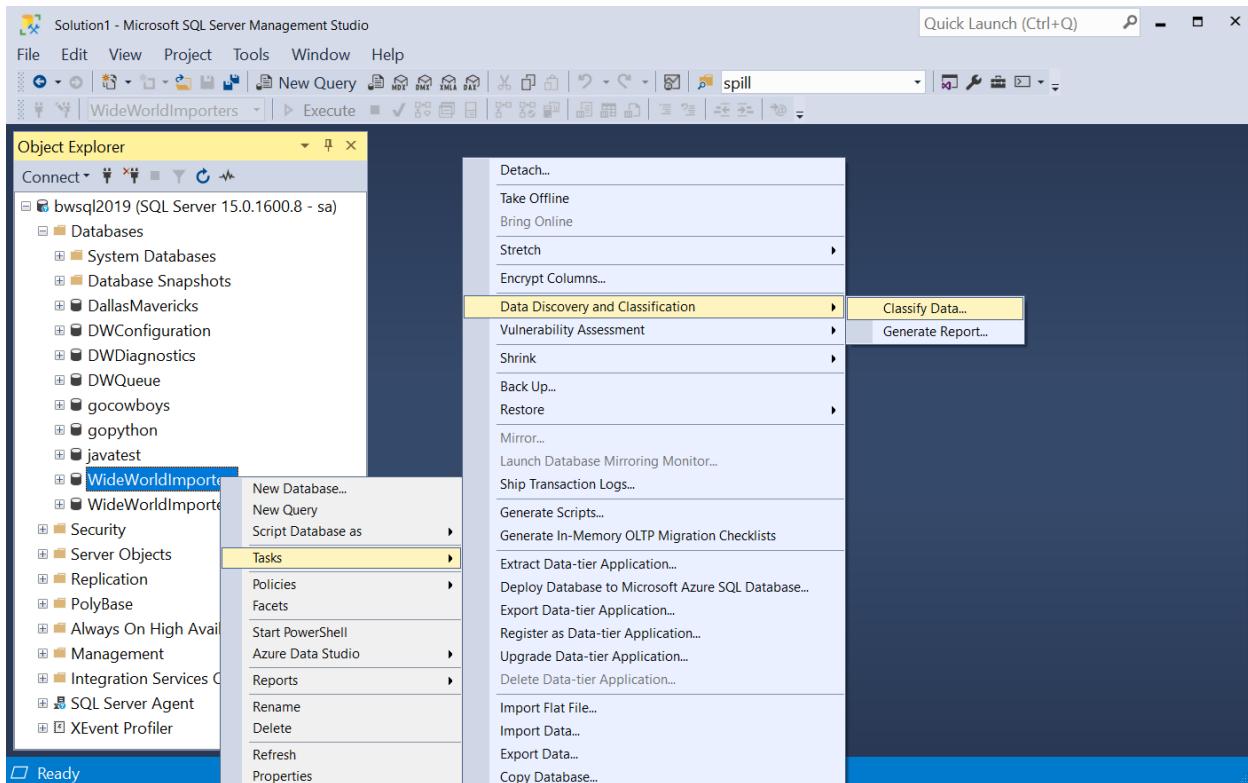
Open up the script **setup_classification.sql** in SQL Server Management Studio (SSMS) and execute each step in the script or copy and paste these T-SQL commands to execute them

```
-- Step 1: In case you have run these demos before drop existing classifications
USE WideWorldImporters
GO
IF EXISTS (SELECT * FROM sys.sensitivity_classifications sc WHERE
object_id('[Application].[PaymentMethods]') = sc.major_id)
BEGIN
    DROP SENSITIVITY CLASSIFICATION FROM
[Application].[PaymentMethods].[PaymentMethodName]
END
GO
IF EXISTS (SELECT * FROM sys.sensitivity_classifications sc WHERE
object_id('[Application].[People]') = sc.major_id)
BEGIN
    DROP SENSITIVITY CLASSIFICATION FROM [Application].[People].[FullName]
    DROP SENSITIVITY CLASSIFICATION FROM [Application].[People].[EmailAddress]
END
GO
```

STEP 3: Add classifications using the wizard in SSMS

NOTE: SSMS still supports using Data Discover and Classification against older versions of SQL Server but will use the older technique built into the tool. Auditing will not be available for these scenarios.

- Launch SSMS and select the Data Discovery and Classification option as a Task from WideWorldImporters in Object Explorer



- View recommendations from the tool

SSMS analyzes column names in the database and creates recommendations for data classification for labels and information_type.

Select "click to view" in the gray bar to see the recommendations for WideWorldImporters

66 columns with classification recommendations (click to view)

0 classified columns

Schema	Table	Column	Information Type	Sensitivity Label

- Select a few columns based on recommendations

The recommendations provided by the tool are fixed and cannot be customized (you will see later how to add your own). To see how this information is saved, click on the columns **PaymentMethodName** and **FullName**. Then click on **Accept selected recommendations**.

66 columns with classification recommendations (click to minimize)

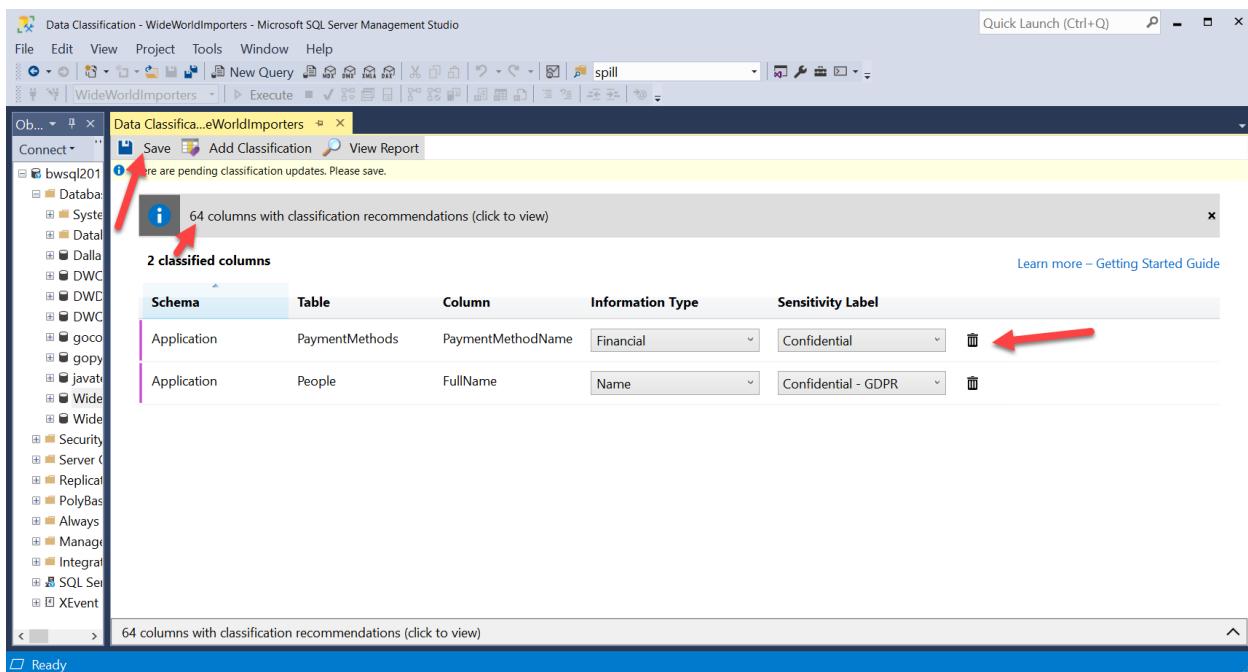
Accept selected recommendations

Schema	Table	Column	Information Type	Sensitivity Label
Application	Cities	CityName	Contact Info	Confidential
Application	PaymentMethods	PaymentMethodID	Financial	Confidential
<input checked="" type="checkbox"/> Application	PaymentMethods	PaymentMethodName	Financial	Confidential
<input type="checkbox"/> Application	People	EmailAddress	Contact Info	Confidential
<input checked="" type="checkbox"/> Application	People	FullName	Name	Confidential - GDPR
<input type="checkbox"/> Application	People	HashedPassword	Credentials	Confidential

- Save the recommendations

The tool shows you the recommendations you accepted. You could delete these and choose others at this point (do not do this for this lab). Notice there are 2 less recommendations.

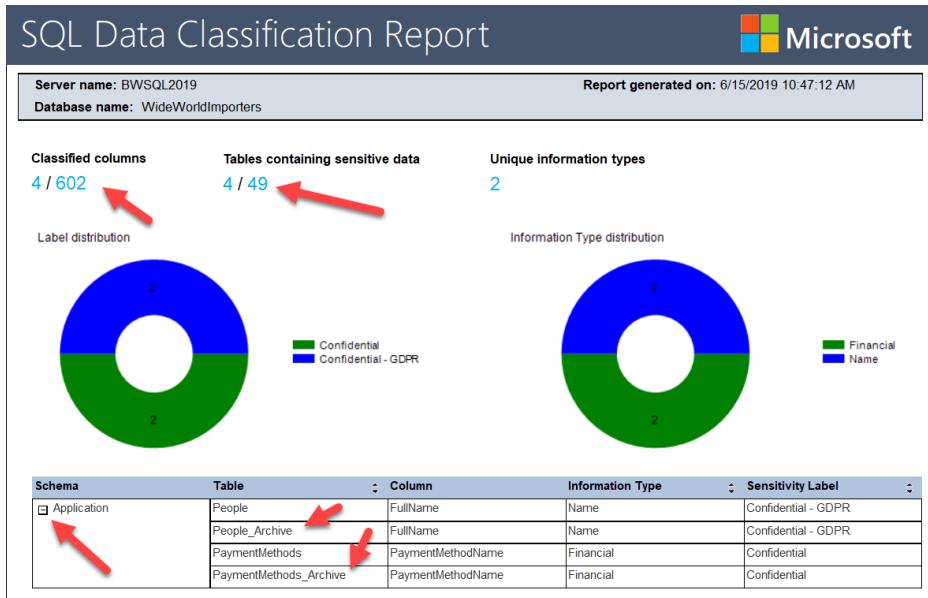
Click the save button to save your recommendations. The tool will run the corresponding ADD SENSITIVITY CLASSIFICATION T-SQL statements when you click Save.



- View a report of classifications

SSMS has a report that shows classifications saved in the database both with the tool and through any ADD SENSITIVITY CLASSIFICATION execution.

On the window where you clicked the Save button, click the View Report button. A new tab will be created with a report that looks like the following



The report shows details of what is marked for classification and a total of how many columns and tables are marked with classifications based on totals in the database.

Click the + in the table on the Application schema to see all columns marked for classification.

You will notice in the report that tables are columns for *Archive* tables are listed. This is because you saved classification on tables that have temporal tables in WideWorldImporters. Any classification added to a table with temporal tables automatically has columns in those temporal tables marked for classification. You don't need to take any action, this happens behind the scenes (When you drop a classification any classification for associated temporal tables is also dropped).

STEP 4: View classifications using system catalog views.

SQL Server provides system catalog views to view all classification metadata in a database.

Execute the script **findclassifications.sql** to view classifications added to the database. You can also execute the T-SQL statements from the script

```
USE WideWorldImporters
GO
SELECT o.name AS table_name, c.name AS column_name, sc.information_type,
sc.information_type_id, sc.label, sc.label_id
FROM sys.sensitivity_classifications sc
JOIN sys.objects o
ON o.object_id = sc.major_id
JOIN sys.columns c
```

```
ON c.column_id = sc.minor_id
AND c.object_id = sc.major_id
ORDER BY sc.information_type, sc.label
GO
```

Your results should look the same as the table at the bottom of the report from the previous step.

STEP 5: Add your own classifications

The wizard in SSMS allows you to add your own classifications but you must pick the list of label and information_type values built into the tool. What if you wanted to add your own values for labels and information_type? You can use the T-SQL command **ADD SENSITIVITY CLASSIFICATION**.

The **addclassification.sql** script contains an example to add a new classification for the **[Application].[People].[EmailAddress]** column. This example uses an information_type of **Email** and a label of **PII** (which stands for Personably Identifiable Information). Execute this script to add the new classification and look at the system catalog view to see all classifications.

```
-- Step 1: Add the classification
ADD SENSITIVITY CLASSIFICATION TO
[Application].[People].[EmailAddress]
WITH (LABEL='PII', INFORMATION_TYPE='Email')
GO

-- Step 2: View all classifications
USE WideWorldImporters
GO
SELECT o.name AS table_name, c.name AS column_name, sc.information_type,
sc.information_type_id, sc.label, sc.label_id
FROM sys.sensitivity_classifications sc
JOIN sys.objects o
ON o.object_id = sc.major_id
JOIN sys.columns c
ON c.column_id = sc.minor_id
AND c.object_id = sc.major_id
ORDER BY sc.information_type, sc.label
GO
```

Your results from the catalog view query should look like the following

	table_name	column_name	information_type	information_type_id	label	label_id
1	People	EmailAddress	Email	NULL	PII	NULL
2	People_Archive	EmailAddress	Email	NULL	PII	NULL
3	PaymentMethods	PaymentMethodName	Financial	c44193e1-0e58-4b2a-9001-f7d6e7bc1373	Confidential	331f0b13-76b5-2f1b-a77b-def5a73c73c2
4	PaymentMethods_Archive	PaymentMethodName	Financial	c44193e1-0e58-4b2a-9001-f7d6e7bc1373	Confidential	331f0b13-76b5-2f1b-a77b-def5a73c73c2
5	People_Archive	FullName	Name	57845286-7598-22f5-9659-15b24aeb125e	Confidential - GDPR	989adc05-3f3f-0588-a635-f475b994915b
6	People	FullName	Name	57845286-7598-22f5-9659-15b24aeb125e	Confidential - GDPR	989adc05-3f3f-0588-a635-f475b994915b

Notice the results have columns for **information_type_id** and **label_id**. These are GUID values that you can use instead of just string values. Your organization's data catalog system may require a unique ID for tracking classification metadata. Remember the **NEWID()** T-SQL function can be used to generate unique GUID values. The SSMS tool generates **information_type_id** and **label_id** values.

Activity Summary

In this activity you have learned how to use SSMS and T-SQL to create data classification label and information_type metadata for columns in a database. You have also learned how to use a report in SSMS to view classification metadata or system catalog views to see more details.

Armed with this knowledge, proceed to the next activity to learn how auditing is integrated with data classification in SQL Server 2019.



2.1 Auditing Data Classification

In this module you will learn how to use the SQL Server Audit capability that comes with SQL Server to audit who, what, and when attempted to view columns that are marked for classification.

The Challenge

Classifying data is only the first step. Organizations need to track and audit details of users that attempt to view data marked with classification labels and types.

The Solution

SQL Server Audit is a feature of SQL Server to audit all types of access to the SQL Server instance and databases. You can read more about SQL Server Audit

at <https://docs.microsoft.com/en-us/sql/relational-databases/security/auditing/sql-server-audit-database-engine>.

SQL Server 2019 includes a new property with audit called *data_sensitivity_information*. Now when a user attempts to view a column, for example with a SELECT statement, that is marked for classification, this property will contain the label and information_type details from the classification.

Let's use an activity to see how audit and classification work together.



Activity: Using SQL Server Audit with Data Classification

In this activity you will learn how to audit users trying to view columns that were marked for data classification.

IMPORTANT: *This activity assumes you have completed the steps in the Activity for Module 2.0.*

NOTE: *If at anytime during the Activities of this Module you need to "start over" you can go back to the first Activity in 2.0 and run through all the steps again.*



Activity Steps

Work through the following steps to enable SQL Server Audit and view auditing details when a user attempts to view columns that are associated with data classification.

All scripts can be found in the **sql2019lab\02_Security\dataclassification** directory. You can use any T-SQL tool to run these scripts such as SQL Server Management Studio (SSMS) or Azure Data Studio.

STEP 1: Cleanup any previous examples

If you have never run this module activity on your SQL Server, you can skip this step. Otherwise, execute all the steps in the script **dropsqaudit.sql**

NOTE: *For Linux installations you will need to change the path when deleting past audits to /var/opt/mssql/data.*

```
-- Step 1: Disable the audits and drop them  
USE WideWorldImporters  
GO
```

```

IF EXISTS (SELECT * FROM sys.database_audit_specifications WHERE name = 
'People_Audit')
BEGIN
    ALTER DATABASE AUDIT SPECIFICATION People_Audit
    WITH (STATE = OFF)
    DROP DATABASE AUDIT SPECIFICATION People_Audit
END
GO
USE master
GO
IF EXISTS (SELECT * FROM sys.server_audits WHERE name = 'GDPR_Audit')
BEGIN
    ALTER SERVER AUDIT GDPR_Audit
    WITH (STATE = OFF)
    DROP SERVER AUDIT GDPR_Audit
END
GO

-- Step 2: Remove the .audit files from default or your path
-- Note: Remember for Linux installations, the default path is /var/opt/mssql/data
-- del C:\program files\microsoft sql
server\mssql15.mssqlserver\mssql\data\GDPR*.audit

```

STEP 2: Setup an audit to track SELECT statements against the table

Use the T-SQL script **setupsqlaudit.sql** to create and enable a new SQL Audit to track SELECT statements against the **[Application].[People]** table in the **WideWorldImporters** database

NOTE: For Linux installations change the path to /var/opt/mssql/data

```

USE master
GO
-- Create the server audit.
-- Note: Remember for Linux installations, the default path is /var/opt/mssql/data
CREATE SERVER AUDIT GDPR_Audit
    TO FILE (FILEPATH = 'C:\program files\microsoft sql
server\mssql15.mssqlserver\mssql\data')
GO
-- Enable the server audit.
ALTER SERVER AUDIT GDPR_Audit
WITH (STATE = ON)
GO
USE WideWorldImporters
GO
-- Create the database audit specification.
CREATE DATABASE AUDIT SPECIFICATION People_Audit
FOR SERVER AUDIT GDPR_Audit
ADD (SELECT ON Application.People BY public )
WITH (STATE = ON)
GO

```

This module will not go into the details of how SQL Server Audit works. You can get more information on SQL Server Audit at <https://docs.microsoft.com/en-us/sql/relational-databases/security/auditing/sql-server-audit-database-engine>.

STEP 3: Scan all columns of the table

Use the script **findpeople.sql** to scan all columns for all rows.

```
-- Scan the table and see if the sensitivity columns were audited
USE WideWorldImporters
GO
SELECT * FROM [Application].[People]
GO
```

STEP 4: Check the audit

Use the script **checkaudit.sql** to see if anything was audited.

NOTE: For Linux installations change the path to /var/opt/mssql/data

```
-- Check the audit
-- The audit may now show up EXACTLY right after the query but within a few seconds.
-- Note: Remember for Linux installations, the default path is /var/opt/mssql/data
SELECT event_time, session_id, server_principal_name,
database_name, object_name,
cast(data_sensitivity_information as XML) as data_sensitivity_information,
client_ip, application_name
FROM sys.fn_get_audit_file ('C:\program files\microsoft sql
server\mssql15.mssqlserver\mssql\data\*.sqlaudit',default,default)
GO
```

The system function **fn_get_audit_file()** can be used to obtain results from SQL Server Audit files (which are extended event files)

Your results should look like the following:

	event_time	session_id	server_principal_name	database_name	object_name	data_sensitivity_information	client_ip	application
1	2019-06-15 17:42:01.6702752	53	sa				10.0.0.50	Microsoft S
2	2019-06-15 17:43:29.7592877	53	sa	WideWorldImporters	People	<sensitivity_attributes><sensitivity_attribute la...	10.0.0.50	Microsoft S



The first row is a record that the audit has started. The second row is an audit of the SELECT statement. The **data_sensitivity_information** column contains an XML record of the **label** and **information_type** values associated with columns that have data classifications. This includes the information to look up what columns are affected through the **sys.sensitivity_columns** catalog view.

Keep the **checkaudit.sql** query tab available in SSMS as you will use it again over the next several steps.

STEP 5: SELECT one column from the table

Use the script **findpeopleonecolumn.sql** to query only one column that is marked for classification.

```
-- What if I access just one of the columns directly?  
SELECT FullName FROM [Application].[People]  
GO
```

STEP 6: Check the audit again

Use the script **checkaudit.sql** to see if anything was audited.

NOTE: For Linux installations change the path to /var/opt/mssql/data

```
-- Check the audit  
-- The audit may now show up EXACTLY right after the query but within a few seconds.  
-- Note: Remember for Linux installations, the default path is /var/opt/mssql/data  
SELECT event_time, session_id, server_principal_name,  
database_name, object_name,  
cast(data_sensitivity_information as XML) as data_sensitivity_information,  
client_ip, application_name  
FROM sys.fn_get_audit_file ('C:\program files\microsoft sql  
server\mssql15.mssqlserver\mssql\data\*.sqlaudit',default,default)  
GO
```

Your results should now look like the following:

	event_time	session_id	server_principal_name	database_name	object_name	data_sensitivity_information	client_ip	application
1	2019-06-15 17:42:01.6702752	53	sa				10.0.0.50	Microsoft S
2	2019-06-15 17:43:29.7592877	53	sa	WideWorldImporters	People	<sensitivity_attributes> <sensitivity_attribute la...	10.0.0.50	Microsoft S
3	2019-06-15 17:53:45.6213307	53	sa	WideWorldImporters	People	<sensitivity_attributes> <sensitivity_attribute la...	10.0.0.50	Microsoft S

The third row is for the new SELECT statement and the data_sensitivity_information contains an XML record of data classification for only one column.

STEP 7: Use a column marked for classification in a WHERE clause

Use the script **findpeoplewhereclause.sql** to query only one column that is marked for classification.

```
-- What if I reference a classified column in the WHERE clause only?  
SELECT PreferredName FROM [Application].[People]  
WHERE EmailAddress LIKE '%microsoft%'  
GO
```

This query should return no rows.

STEP 8: Check the audit again

Auditing for columns with data classification only apply to queries where columns are in the SELECT "list" of the query. Use the script **checkaudit.sql** to see if anything was audited.

NOTE: For Linux installations change the path to `/var/opt/mssql/data`

```
-- Check the audit
-- The audit may now show up EXACTLY right after the query but within a few seconds.
-- Note: Remember for Linux installations, the default path is /var/opt/mssql/data
SELECT event_time, session_id, server_principal_name,
database_name, object_name,
cast(data_sensitivity_information as XML) as data_sensitivity_information,
client_ip, application_name
FROM sys.fn_get_audit_file ('C:\program files\microsoft sql
server\mssql15.mssqlserver\mssql\data\*.sqlaudit',default,default)
GO
```

Your results should now look like the following:

	event_time	session_id	server_principal_name	database_name	object_name	data_sensitivity_information	client_ip	application
1	2019-06-15 17:42:01.6702752	53	sa				10.0.0.50	Microsoft S
2	2019-06-15 17:43:29.7592877	53	sa	WideWorldImporters	People	<sensitivity_attributes><sensitivity_attribute la...	10.0.0.50	Microsoft S
3	2019-06-15 17:53:45.6213307	53	sa	WideWorldImporters	People	<sensitivity_attributes><sensitivity_attribute la...	10.0.0.50	Microsoft S
4	2019-06-15 18:01:18.8555534	53	sa	WideWorldImporters	People		10.0.0.50	Microsoft S



Notice in this results for the new row the `data_sensitivity_information` columns is NULL. This is because auditing for data classification only apply where columns are listed in the SELECT "list" of a query.

STEP 9: Cleanup audits and classifications

Use the script **cleanup.sql** to disable and drop audits and delete classifications.

NOTE: For Linux installations change the path in the script to delete audit files to `/var/opt/mssql/data`

```
USE WideWorldImporters
GO
IF EXISTS (SELECT * FROM sys.database_audit_specifications WHERE name =
'People_Audit')
BEGIN
    ALTER DATABASE AUDIT SPECIFICATION People_Audit
    WITH (STATE = OFF)
    DROP DATABASE AUDIT SPECIFICATION People_Audit
END
GO
USE master
```

```

GO
IF EXISTS (SELECT * FROM sys.server_audits WHERE name = 'GDPR_Audit')
BEGIN
    ALTER SERVER AUDIT GDPR_Audit
    WITH (STATE = OFF)
    DROP SERVER AUDIT GDPR_Audit
END
GO

-- Remove the .audit files from default or your path
-- Remember for Linux installations, the default path is /var/opt/mssql/data.
-- del C:\program files\microsoft sql
server\mssql15.mssqlserver\mssql\data\GDPR*.audit

ALTER DATABASE WideWorldImporters SET COMPATIBILITY_LEVEL = 130
GO

USE WideWorldImporters
GO
IF EXISTS (SELECT * FROM sys.sensitivity_classifications sc WHERE
object_id('[Application].[PaymentMethods]') = sc.major_id)
    DROP SENSITIVITY CLASSIFICATION FROM
[Application].[PaymentMethods].[PaymentMethodName]
GO
IF EXISTS (SELECT * FROM sys.sensitivity_classifications sc WHERE
object_id('[Application].[People]') = sc.major_id)
    DROP SENSITIVITY CLASSIFICATION FROM [Application].[People].[FullName]
    DROP SENSITIVITY CLASSIFICATION FROM [Application].[People].[EmailAddress]
GO

```

Activity Summary

In this activity you have seen how SQL Server Audit uses the new property **data_sensitivity_information** to track users who list columns associated with data classifications in a SELECT query. However, audit does not apply to all "usage" of a column that is marked for classification.

Proceed to the next module learn about new mission critical **availability** features in SQL Server 2019.



For Further Study

- [SQL Server and Azure Data Discovery and classification](#)
- [ADD SENSITIVITY CLASSIFICATION](#)
- [sys.sensitivity_classifications catalog view](#)

- [SQL Server Audit](#)



Next Steps

Next, Continue to [Availability](#).



Workshop: SQL Server 2019 Lab (RC)

A Microsoft workshop from the SQL Server team



SQL Server 2019 Mission Critical Availability

SQL Server 2019 includes new capabilities to keep your database and application highly available:

- Online Index Enhancements
- Enhancements to Availability Groups
- Taking advantage of built-in HA for Kubernetes
- Accelerated Database Recovery

You can read more details about all of these enhancements at <https://docs.microsoft.com/en-us/sql/sql-server/what-s-new-in-sql-server-ver15?view=sqlallproducts-allversions>.

You will cover the following topics in this Module:

[**3.0 Accelerated Database Recovery**](#)



3.0 Accelerated Database Recovery

In this module you will learn about a new capability in SQL Server 2019 to solve problems caused by long running transactions. This enhancement to SQL Server 2019 is called Accelerated Database Recovery.

Accelerated Database Recovery started years ago as a project within Microsoft called Constant Time Recovery (CTR). The idea was to enhance the database engine so that the time it took to recover a database was constant instead of based on the length of the oldest active transaction as recorded in the transaction log. You can read more about the project in this detailed paper at <https://www.microsoft.com/en-us/research/publication/constant-time-recovery-in-azure-sql-database>.

The Challenge

Long running transactions can take the following forms:

- A transaction that does not make a lot of modifications but is held open for a long period of time
- A transaction that makes a lot of data modifications (think deleting all rows in a 1 billion row table)

Both of these scenarios can lead to the following problems:

- A **long** time to **recover** the database should it be taken offline or SQL Server shutdown while a long running transaction is active.
- **Transaction rollback** for queries with a lot of modifications can take a **long time** holding locks.
- The **transaction log** may grow unexpectedly because it **cannot be truncated** due to an active transaction.

The Solution

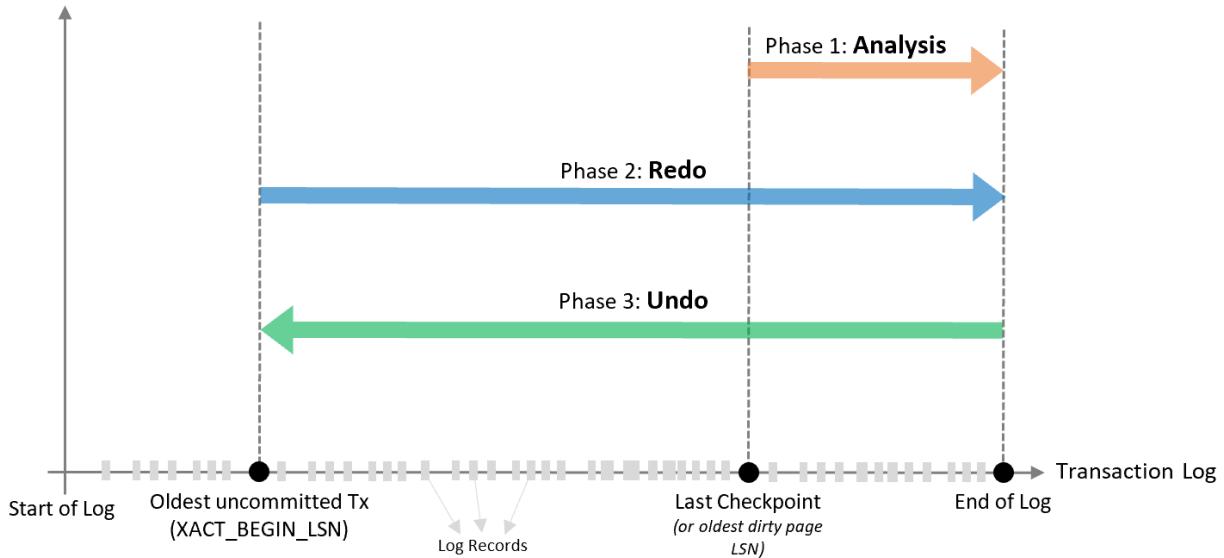
Accelerated Database Recovery (ADR) attempts to solve these problems by using a concept called the **Persistent Version Store (PVS)**. This is not the same version store that is kept in tempdb for snapshot isolation. The PVS is stored in the user database inside the rows of a page or in an off-row store internal table. Because it is persistent (i.e. survives restarts) it can be used for recovery purposes.

Now the **redo** and **undo** phases of recovery can be significantly faster (hence the term accelerated) because versions can be used to determine the state of a transaction (vs having to logically undo uncommitted transactions).

These diagrams from the documentation show the recovery process with and without ADR.

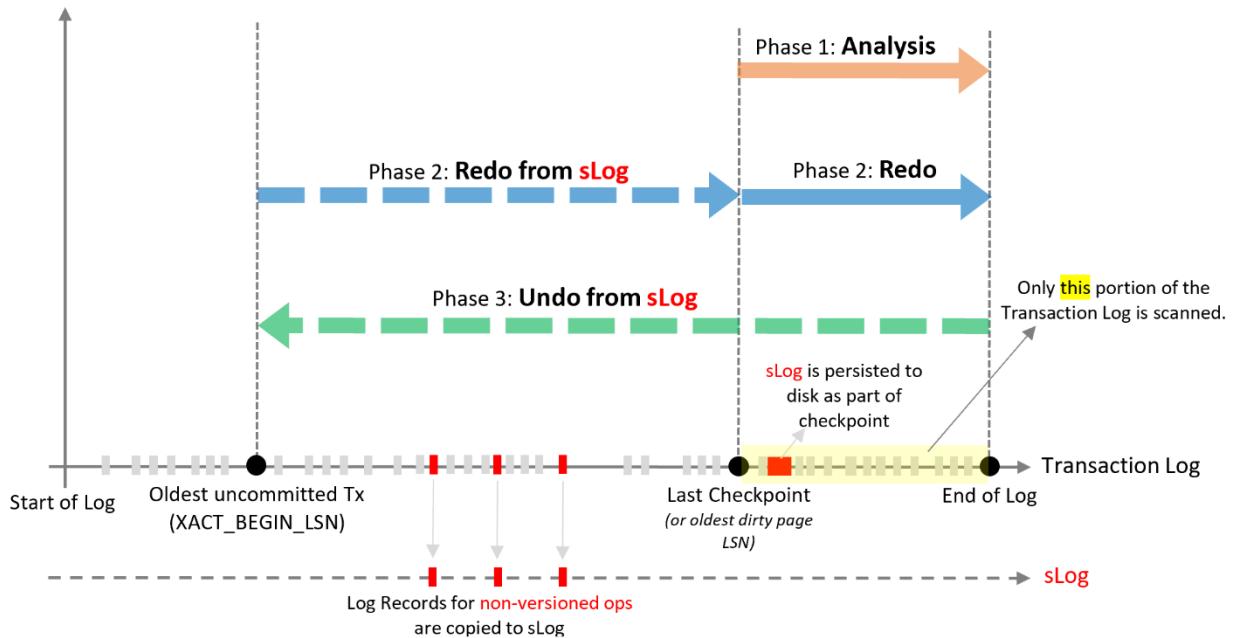
Recovery without ADR (this is the default)

Recovery Phase / Transaction Log (**without ADR**)



Recovery with ADR (a new option with ALTER DATABASE)

Recovery Phase / Transaction Log / sLog (**with ADR**)



Both the documentation at <https://docs.microsoft.com/en-us/azure/sql-database/sql-database-accelerated-database-recovery> and the paper at <https://www.microsoft.com/en-us/research/publication/constant-time-recovery-in->

[azure-sql-database](#) have detailed explanations of how ADR works including the PVS and a concept called the SLog (for system transactions).

Because of this design, not only is recovery must faster and no longer affected by a long running transaction but ADR provides two other benefits:

- **rollback is instant.** Because versions are tracked rollback involves simply marking a transaction as aborted
- **Log truncation** is no longer dependent on an active transaction.

Here are some frequently asked questions about ADR:

- **Will my database be larger with the PVS?**

It will be larger than without ADR. However, it may not grow extremely large due to concepts like logical revert and cleanup processes designed with ADR. Testing is the only way to know but the paper has some observations from the engineering team.

- **Will it affect performance?**

As with any feature performance impact will vary. However, extremely "write-heavy" applications may see some effect. Typically those applications don't use long running transactions so they may not benefit from ADR. The paper has testing observations using benchmarks derived from TPC.

- **How is the Persistent Version Store different than the Version Store in TempDB?**

SQL Server uses TempDB to store versions for rows for queries that use SNAPSHOT transaction isolation levels (or the database option READ_COMMITTED_SNAPSHOT_ISOLATION). If you enable ADR for a database, any queries using SNAPSHOT transaction isolation levels will use the persistent version store instead.

- **How does this affect Always On Availability Groups?**

Secondary replicas receive all changes to the persistent version store and log records. ADR can help speed up failover times because undo processing can be extremely fast. In addition read-only queries on secondary replicas will now use the persistent version store instead of TempDB on the secondary. Secondary replicas signal the primary replica which versions need to be kept to service read queries on secondaries. If a secondary

replica needs to be restarted, the persistent version store can easily be used where as with TempDB, the version store is lost and read queries may be held up for transactions to commit or undo.

Accelerated Database Recovery is a feature that exists for both SQL Server 2019 and Azure SQL Database.

Proceed to the Activity to learn an example of how Accelerated Database Recovery works in SQL Server 2019.



Activity: Accelerated Database Recovery

In this activity, you will see how Accelerated Database Recovery affects log truncation and the speed of rollback.

NOTE: If at anytime during the Activities of this Module you need to "start over" you can go back to the first Activity in 3.0 and run through all the steps again.



Activity Steps

All scripts for this activity can be found in the **sql2019lab\03_Availability\adr** folder. The database will be created as part of this activity. There is no need to restore a separate database. The scripts will create a database with a data file of 10Gb and transaction log of 10Gb so there is plenty of space for the activity.

NOTE: For Linux installations change the path to /var/opt/mssql/data in the T-SQL notebook and T-SQL scripts.

STEP 1: Use a T-SQL notebook to complete the rest of the activity.

T-SQL notebooks provide a very nice method to execute T-SQL code with documentation in the form of markdown code. All the steps and documentation to complete the rest of the activity for Module 3.0 can be found in the T-SQL notebook **adr.ipynb** which can be found in the **sql2019lab\03_Availability\adr** folder.

NOTE: A T-SQL script **adr.sql** is also provided if you want to go through the same steps as the notebook but use a tool like SQL Server Management Studio.

T-SQL notebooks can be executed with Azure Data Studio. If you are familiar with using Azure Data Studio and T-SQL notebooks open up the **adr.ipynb** notebook and go

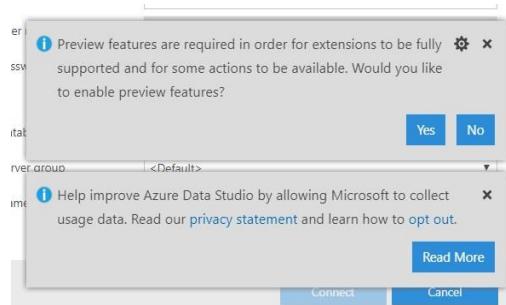
through all the steps. When you are done proceed to the **Activity Summary** section for the Activity below.

If you have never opened a T-SQL notebook with Azure Data Studio, use the following instructions:

Launch the Azure Data Studio application. Look for the icon similar to this one:

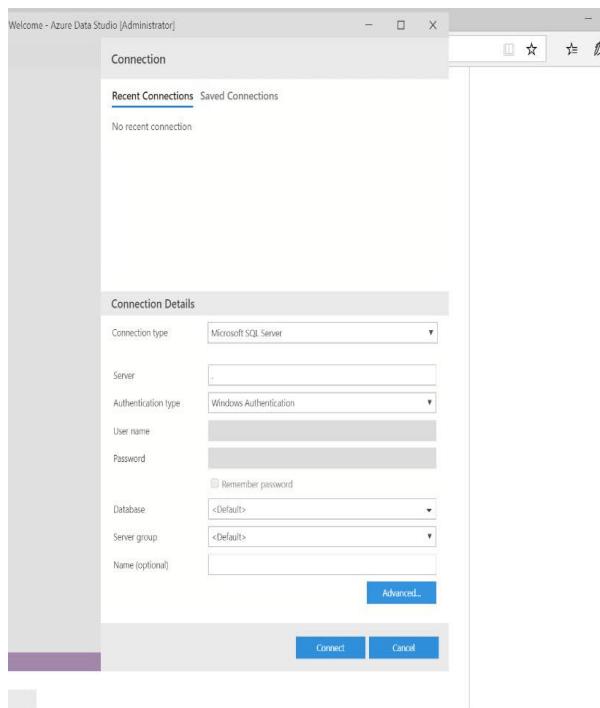


The first time you launch Azure Data Studio, you may see the following choices. For the purposes of this workshop, select No to not load the preview feature and use x to close out the 2nd choice to collect usage data.

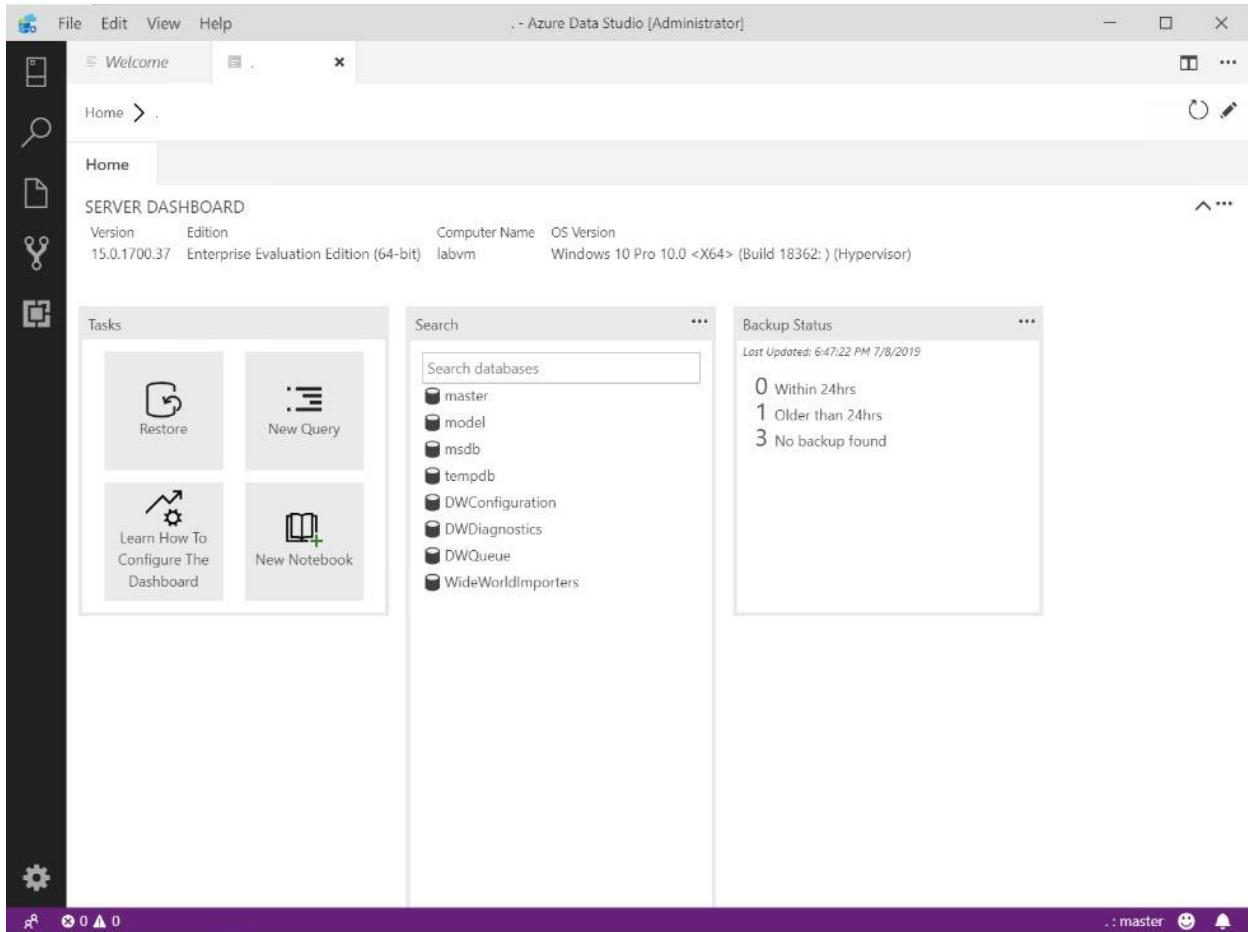


You will now be presented with the following screen to enter in your connection details for SQL Server. Use connection details as provided by your instructor to connect to SQL Server or the connection you have setup yourself for your SQL Server instance.

Now click the **Connect** button to connect. An example of a connection looks similar to this graphic (your server, Auth type, and login may be different):



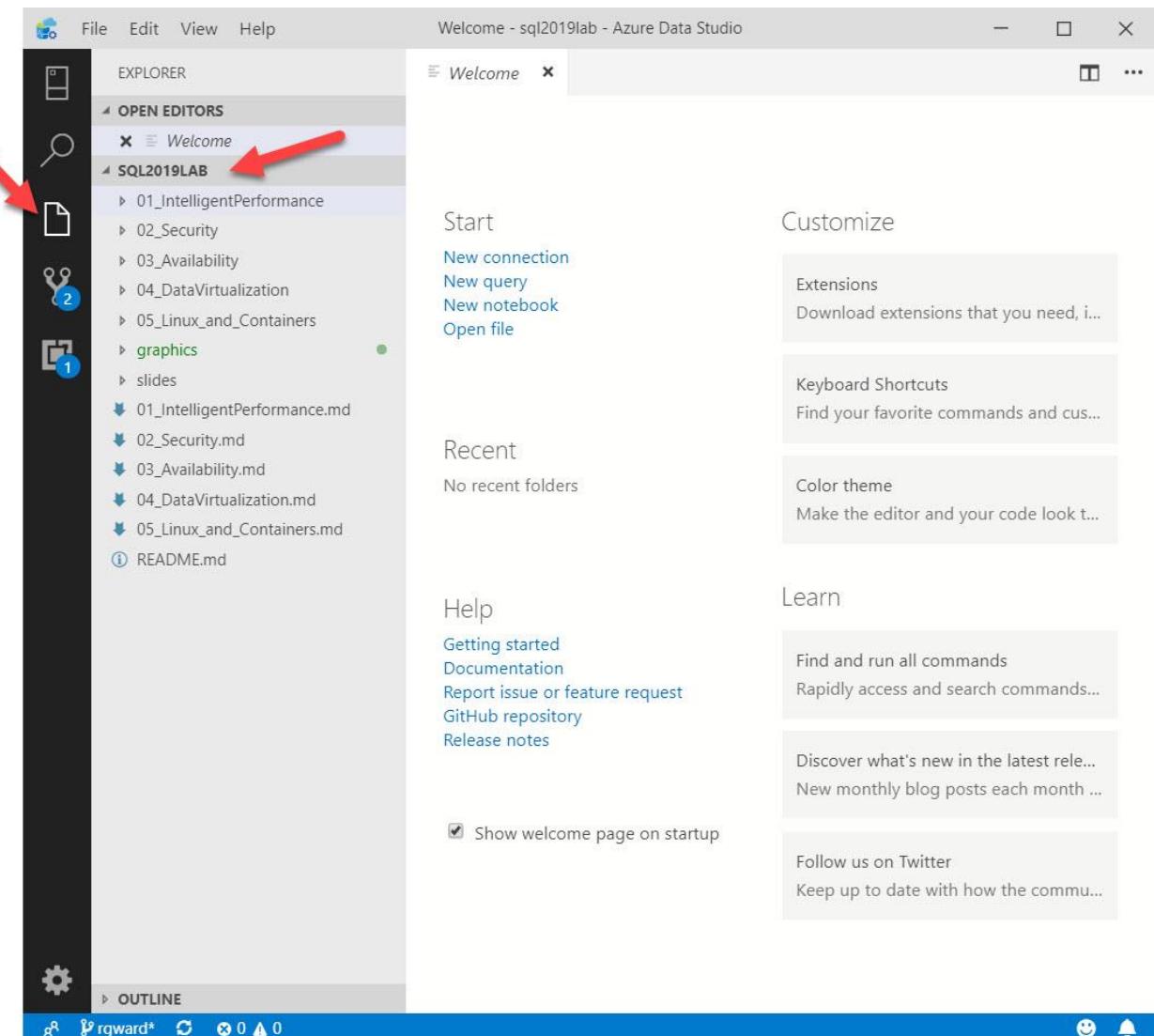
A successful connection looks similar to this (your server may be different):



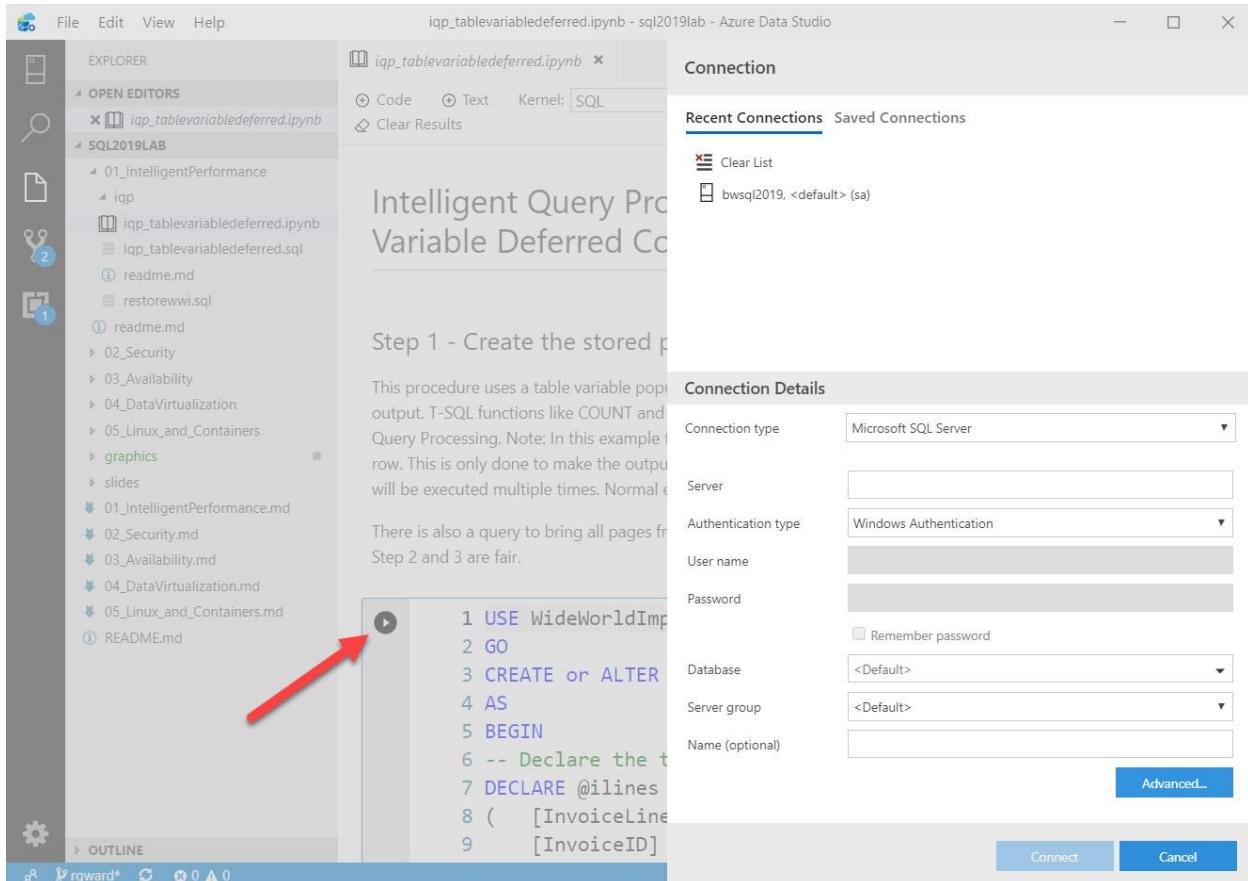
If you haven't already used Explorer in Azure Data Studio, it can be used to explore files. Use the power of Azure Data Studio Explorer to open up any file including notebooks. Use the File/Open Folder menu to open up the **sqlworkshops\sql2019lab** folder. Now click the Explorer icon on the left hand side of Azure Data Studio to see all files and directories for the lab. Navigate to the **03_Availability\adr** folder, open up the **adr.ipynb** notebook and go through all the steps.

NOTE: Be sure to only run one notebook cell at a time for the lab.

You can now use Azure Data Studio explorer to open up a notebook or script without exiting the tool.



When you start using a notebook and use the "Play" button of a cell, you may get prompted for the connection. Choose the connection you used when you first opened up Azure Data Studio.



There is additional documentation on how to use SQL notebooks at <https://docs.microsoft.com/en-us/sql/azure-data-studio/sql-notebooks>.

When you are done proceed to the **Activity Summary** section for the Activity below.

Activity Summary

In this activity you have learned Accelerated Database recovery can speed up transaction rollback significantly. You have also learned how transaction log truncation is no longer affected by long running transactions.

Bonus Activity

NOTE: Close out any outstanding scripts or notebooks before running this bonus activity.

If you have time use the T-SQL notebook **adr_recovery.ipynb** or T-SQL script **adr_recovery.sql** to see how ADR affects the recovery process. This bonus activity can take some time as a large number of modifications are needed to see a bigger

effect on recovery. The scripts will create a database with a data file of 10Gb and transaction log of 40Gb so there is plenty of space for the activity.

Armed with this knowledge, proceed to the next activity to learn how **data virtualization** makes SQL Server 2019 the new "data hub".



For Further Study

- [Accelerated Databased Recovery](#)
- [The Constant Time Recovery Paper](#)
- [What is Azure Data Studio?](#)
- [How to use Notebooks in Azure Data Studio](#)



Next Steps

Next, Continue to [Data Virtualization](#).



Workshop: SQL Server 2019 Lab (RC)

A Microsoft workshop from the SQL Server team



SQL Server 2019 Data Virtualization

SQL Server 2019 includes new capabilities for data virtualization by extending the Polybase (nicknamed it Polybase++) feature introduced in SQL Server 2016 with the ability to connect to other data sources with **client drivers installed**:

- SQL Server
- Oracle
- Teradata
- MongoDB

In addition for SQL Server 2019 on Windows, Polybase now supports the ability to connect to any ODBC data source such as MySQL, PostgreSQL, SAP HANA, IBM DB2, or even Excel.

SQL Server 2019 Polybase still supports connectivity to Hadoop systems as it did in SQL Server 2016 and 2017. In addition, SQL Server 2019 **Big Data Clusters** provide an integrated data virtualization solution in a Kubernetes cluster.

You will cover the following topics in this Module:

[4.0 SQL Server 2019 Polybase](#)



4.0 SQL Server 2019 Polybase

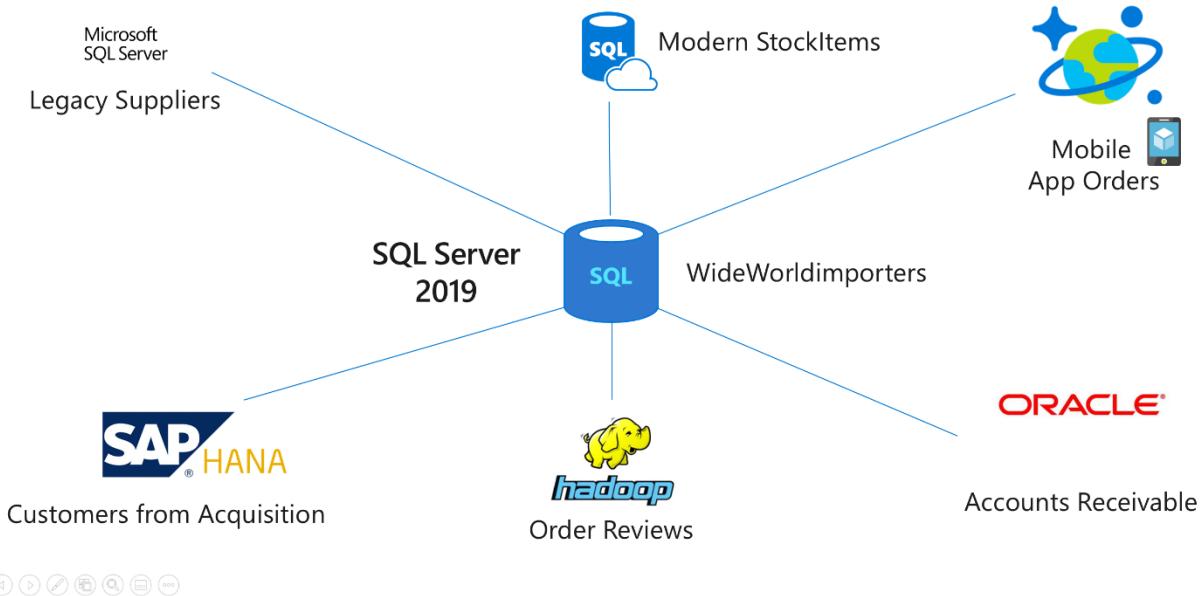
In this module you will learn about the enhanced Polybase feature of SQL Server 2019 designed to meet the needs of the modern data platform.

The Challenge

Organizations often today have multiple database providers and sources that they must integrate together. SQL Server is a leading industry data platform so many customers move data from other data sources into SQL Server often using Extract, Transform, and Load (ETL) applications. ETL applications can be costly to build and maintain and often results in latencies to access data external to SQL Server.

Consider all the data sources facing a company like WideWorldImporters

SQL Server 2019: Data Virtualization



Traditionally, WideWorldImporters would have to write ETL jobs to move data from all these sources into SQL Server. While that solution still may be best for this company, what if SQL Server could act as a "data hub" for all of these sources? What if you could develop T-SQL queries and execute them against SQL Server 2019, and SQL Server would query the external data sources to bring back the results? Imagine if you could join local SQL Server tables with external data sources or perhaps join all of these together!

The Solution

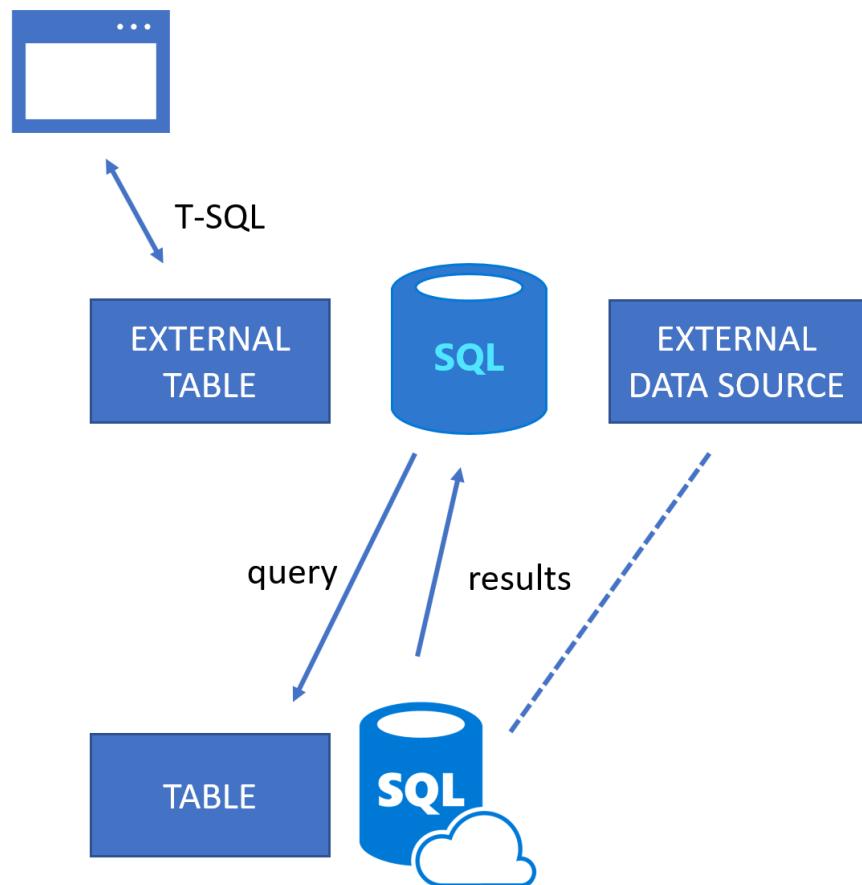
The questions just asked are in a nutshell what Polybase provides in SQL Server 2019 through a concept called an **EXTERNAL TABLE**.

External tables are just like SQL Server tables except SQL Server only stores the metadata of the table definition. The data stays where it lives from the external data source. Think of an external table like a view on top of a data source outside of SQL Server. Since an external table is like a table, you can use SQL Server securables to control access to external data sources like you would a local table.

Polybase uses ODBC drivers to connect to sources such as Oracle, Teradata, MongoDB, and SQL Server. And the driver to connect to other SQL Server data sources can be used to connect to Azure SQL Database and Azure SQL Data Warehouse. With SQL Server 2019 on Windows, you can "bring your own driver" and connect to almost any ODBC

data source in your environment. Polybase is integrated with the SQL Server engine so the query processor can make smart decisions to "push-down" query predicates instead of streaming back an entire remote table and filtering it locally.

Consider this simple diagram for how external tables work



An application runs a T-SQL query against an EXTERNAL TABLE. The EXTERNAL TABLE is mapped to an EXTERNAL DATA SOURCE which points to the location of the external data source (in this diagram imagine this is Azure SQL Database). SQL Server will take the query against the EXTERNAL TABLE and produce a new query native to the external data source. Results are brought back to SQL Server and produced back to the application.

Polybase has advantages over features like linked servers such as:

- External tables are defined in the database so are naturally part of an Availability Group

- External tables allow you to define your table names per your organization standard vs using object names from the external data source.
- Polybase has built-in scalability through a concept called a scale-out group.
- Polybase has built-in client driver support for Hadoop, Oracle, Teradata, and MongoDB.

You can read more about a comparison of Polybase to Linked Servers at <https://docs.microsoft.com/en-us/sql/relational-databases/polybase/polybase-faq?view=sql-server-ver15>.

Proceed to the Activity to learn an example of how to use Polybase with SQL Server 2019 and Azure SQL Server Database.



Activity: Using Polybase with Azure SQL Server Database

In this activity, you will learn how to build an external data source and table to query a table in Azure SQL Database without connecting directly yourself.

NOTE: If at anytime during the Activities of this Module you need to "start over" you can go back to the first Activity in 4.0 and run through all the steps again.

NOTE: There are example scripts in **sql2019lab\04_DataVirtualization** for other data sources for you to use at a later time for Hadoop, Oracle, CosmosDB (through MongoDB), SQL Server 2008R2, and SAP HANA. For these examples, you will need to create or use your own external data source. The scripts and directions to create table definitions, data, and external tables are included.



Activity Steps

Go through the following steps to learn how to query tables in Azure SQL Database using Polybase.

All scripts for this activity can be found in the **sql2019lab\04_DataVirtualization\sqldatahub\azuredb** folder.

IMPORTANT: This activity assumes the following:

- You have installed and enabled Polybase (stand-alone is acceptable) per the documentation at <https://docs.microsoft.com/en-us/sql/relational-databases/polybase/polybase-installation?view=sqlallproducts-allversions> or for Linux at <https://docs.microsoft.com/en-us/sql/relational-databases/polybase/polybase-linux-setup>.
- You have access to an Azure SQL Database database or Managed Instance. The T-SQL script **createazuredtable.sql** as found in the **sql2019lab\04_DataVirtualization\sqldatahub\azuredb** directory contains the target schema and data. The scripts in the activity have a specific Azure SQL Database connection string, login, password, and database name (wwiazure). You may need to modify the scripts to match your Azure SQL Database connection, login, password, and db name.

STEP 1: Restore the WideWorldImporters backup.

NOTE: If you have restored the WideWorldImporters database backup in other modules, you can skip this step.

Use a tool like SQL Server Management Studio (SSMS) or Azure Data Studio (ADS) to execute the T-SQL script **restorewwi.sql** as found in the **sql2019lab\04_DataVirtualization** folder to restore the WideWorldImporters backup. The script assumes a specific path for the backup and database/log files. You may need to edit this depending on your installation. *Remember for Linux installations, the default path is /var/opt/mssql/data*. Your instructor may have provided this backup for you but if necessary you can download it from <https://github.com/Microsoft/sql-server-samples/releases/download/wide-world-importers-v1.0/WideWorldImporters-Full.bak>

STEP 2: Clean up any previous execution

Use the T-SQL script **cleanup.sql** from the **sql2019lab\04_DataVirtualization\sqldatahub\azuredb** folder to clean up any previous execution of this activity. If you get errors that the objects don't exist you can safely ignore them because it means you have not run the activity before.

```
USE [WideWorldImporters]
GO
DROP EXTERNAL TABLE azuresqlDb.ModernStockItems
GO
DROP SCHEMA azuresqlDb
GO
DROP EXTERNAL DATA SOURCE AzureSQLDatabase
GO
DROP DATABASE SCOPED CREDENTIAL AzureSQLDatabaseCredentials
```

```
GO  
DROP MASTER KEY  
GO
```

STEP 3: Use a T-SQL notebook to complete the rest of the activity.

T-SQL notebooks provide a very nice method to execute T-SQL code with documentation in the form of markdown code. All the steps and documentation to complete the rest of the activity for Module 4.0 can be found in the T-SQL notebook **azuredbexternaltable.ipynb** which can be found in the **sql2019lab\04_DataVirtualization\sqldatahub\azuredb** folder.

NOTE: A T-SQL script **azuredbexternaltable.sql** is also provided if you want to go through the same steps as the notebook but use a tool like SQL Server Management Studio.

IMPORTANT: For instructor led courses, your instructor will provide you the details of the following authentication and Azure SQL Database Server details

```
CREATE DATABASE SCOPED CREDENTIAL AzureSQLDatabaseCredentials  
WITH IDENTITY = '<login>', SECRET = '<password>'  
GO  
  
CREATE EXTERNAL DATA SOURCE AzureSQLDatabase  
WITH (  
LOCATION = 'sqlserver://<azure sql database server URI>',  
PUSHDOWN = ON,  
CREDENTIAL = AzureSQLDatabaseCredentials  
)  
GO
```

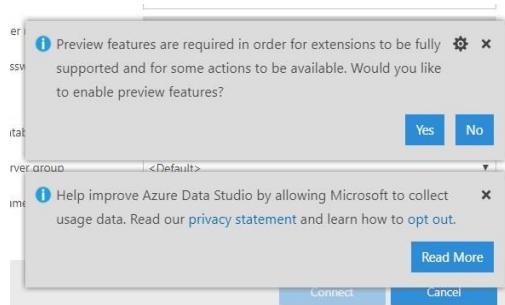
T-SQL notebooks can be executed with Azure Data Studio. If you are familiar with using Azure Data Studio and T-SQL notebooks open up the **azuredbexternaltable.ipynb** notebook and go through all the steps. When you are done proceed to the **Activity Summary** section for the Activity below.

If you have never opened a T-SQL notebook with Azure Data Studio, use the following instructions:

Launch the Azure Data Studio application. Look for the icon similar to this one:

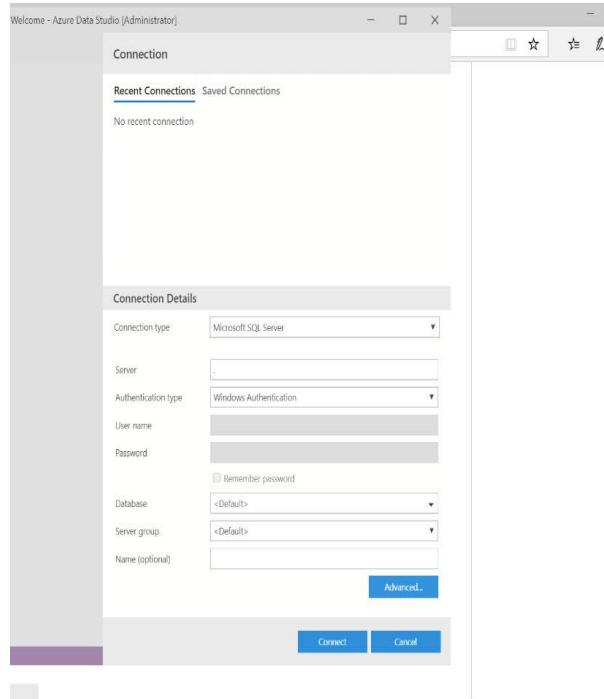


The first time you launch Azure Data Studio, you may see the following choices. For the purposes of this workshop, select No to not load the preview feature and use x to close out the 2nd choice to collect usage data.

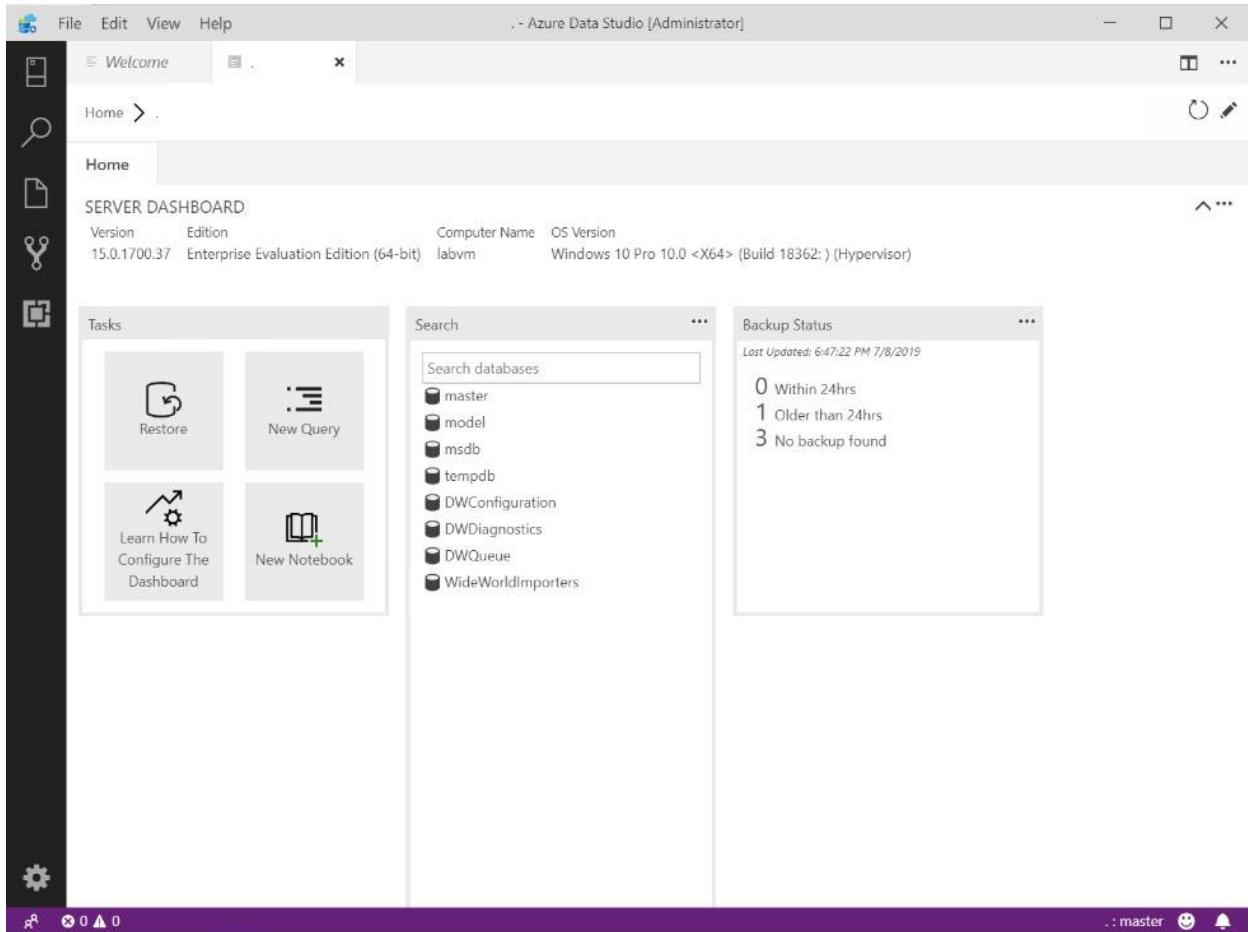


You will now be presented with the following screen to enter in your connection details for SQL Server. Use connection details as provided by your instructor to connect to SQL Server or the connection you have setup yourself for your SQL Server instance.

Now click the **Connect** button to connect. An example of a connection looks similar to this graphic (your server, Auth type, and login may be different):



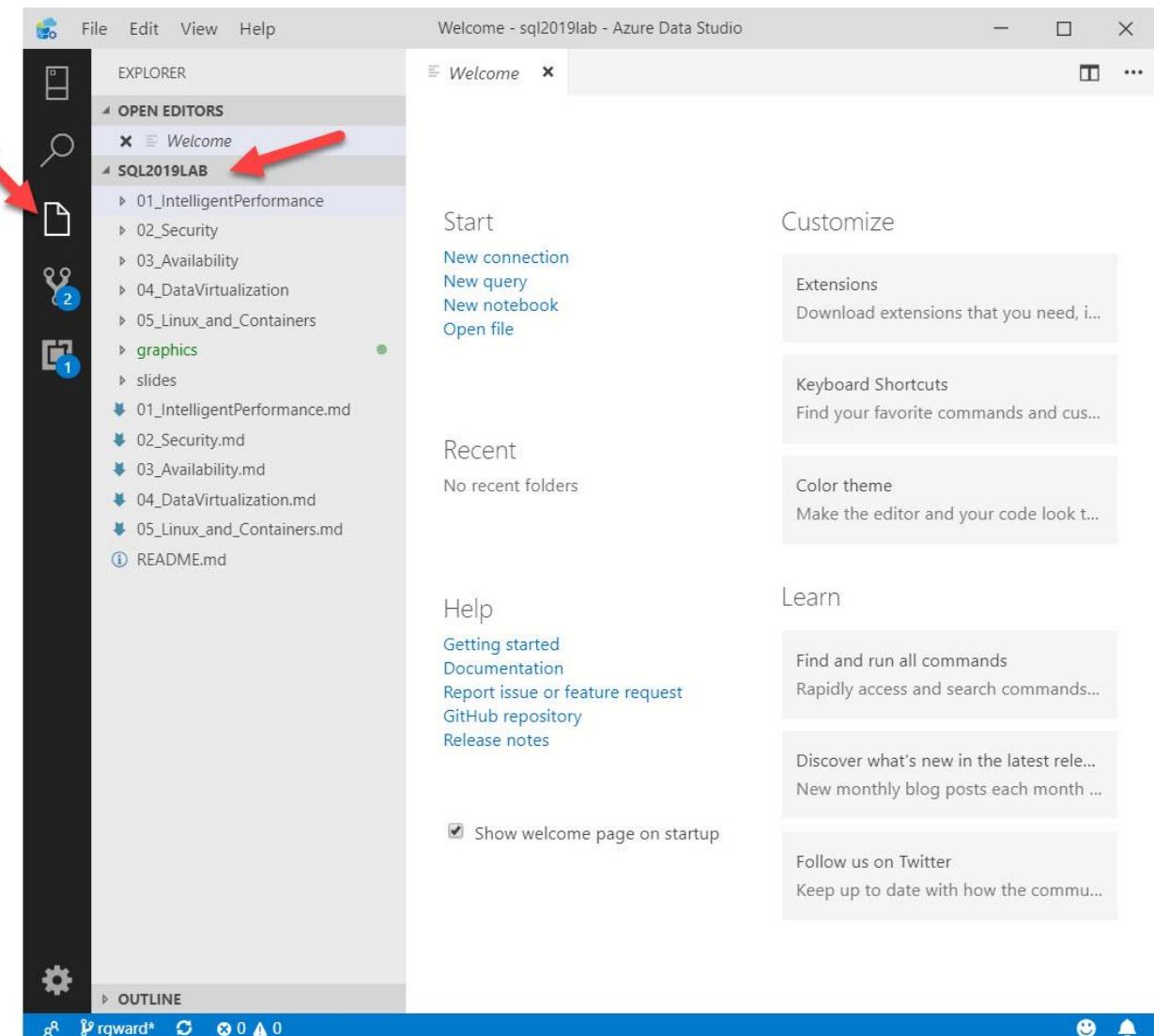
A successful connection looks similar to this (your server may be different):



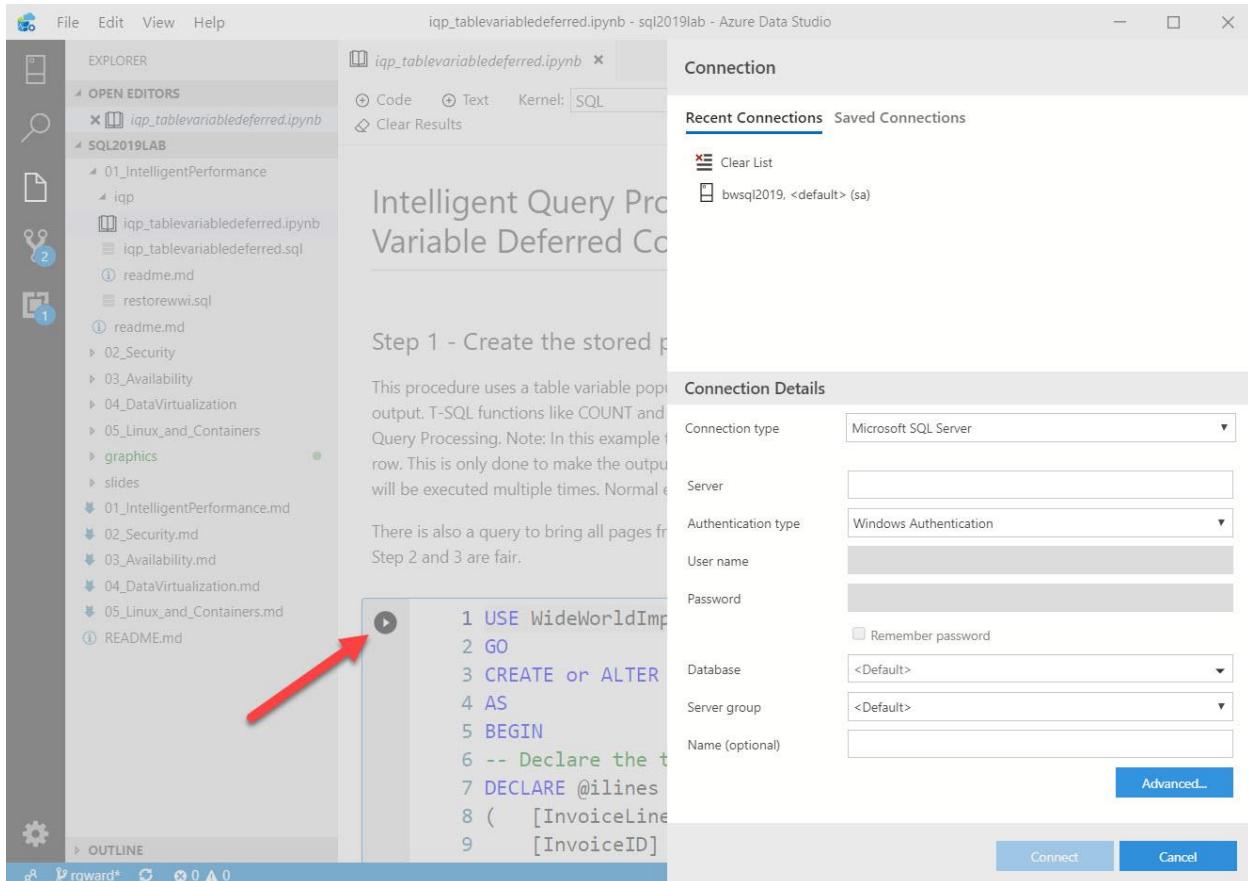
If you haven't already used Explorer in Azure Data Studio, it can be used to explore files. Use the power of Azure Data Studio Explorer to open up any file including notebooks. Use the File/Open Folder menu to open up the **sqlworkshops\sql2019lab** folder. Now click the Explorer icon on the left hand side of Azure Data Studio to see all files and directories for the lab. Navigate to the **04_DataVirtualization\sqldatahub\azuredb** folder, open up the **azuredbexternaltable.ipynb** notebook and go through all the steps.

NOTE: Be sure to only run one notebook cell at a time for the lab.

You can now use Azure Data Studio explorer to open up a notebook or script without exiting the tool.



When you start using a notebook and use the "Play" button of a cell, you may get prompted for the connection. Choose the connection you used when you first opened up Azure Data Studio.



There is additional documentation on how to use SQL notebooks at <https://docs.microsoft.com/en-us/sql/azure-data-studio/sql-notebooks>.

✓ Bonus Activity Steps

If you have time, look over the scripts and T-SQL notebooks (with saved results) for other data sources in the **04_DataVirtualization\sqldatahub** folder. There are subfolders for examples for these data sources:

- Azure CosmosDB (using MongoDB)
- HDFS
- Oracle
- SAP HANA
- SQL Server

When you are done proceed to the **Activity Summary** section for the Activity below.

Activity Summary

In this activity you have learned how Polybase in SQL Server 2019 allows you to connect and query data from external data sources without moving data using ETL applications. In the Activity you learned how to create and query an EXTERNAL DATA SOURCE and EXTERNAL TABLE mapped to an Azure SQL Database table.

Proceed to the next module to learn about enhancements to **SQL Server Linux and Containers** in SQL Server 2019.



For Further Study

- [What is Polybase?](#)
- [CREATE EXTERNAL TABLE](#)
- [What are SQL Server Big Data Clusters?](#)
- [What is Azure Data Studio?](#)
- [How to use Notebooks in Azure Data Studio](#)



Next Steps

Next, Continue to [*Linux and Containers*](#).



Workshop: SQL Server 2019 Lab (RC)

A Microsoft workshop from the SQL Server team



SQL Server 2019 Linux and Containers

SQL Server 2017 introduced the world to SQL Server on Linux and Containers. SQL Server on Linux is completely compatible with SQL Server on Windows due to the unique design of the SQL Platform Abstraction Layer (SQLPAL). The core engine features

of SQL Server on Linux work exactly as SQL Server on Windows because the engine codebase is the same on both platforms.

However, there were some features that come with SQL Server that were not included in SQL Server 2017 on Linux. SQL Server 2019 shores up these gaps by including the following new enhancements for Linux:

- Replication and Change Data Capture (CDC)
- Distributed Transactions
- Machine Learning Services and Extensibility
- Polybase
- Tempdb file auto-config

Containers were also introduced with SQL Server 2017. Containers provide portability, consistent, reduced footprint, and increased availability due to their unique ability for updates.

SQL Server 2019 introduces the following new enhancements for containers:

- Microsoft Container Registry
- Red Hat Images
- Non-root Containers
- Windows Containers (Currently Private Preview)

For this module, we will join forces to show you how to implement SQL Server Replication with Linux Containers.

[**5.0 SQL Server 2019 Replication on Linux using Containers**](#)

5.0 SQL Server 2019 Replication on Linux using Containers

In this module you will learn how to deploy SQL Server Replication using Containers.

The Challenge

Customers who need or desire to use Linux as a platform now have a choice with SQL Server. Even though SQL Server 2017 is compatible with SQL Server on Windows, some features like Replication were not included in the release.

In addition, SQL Server 2017 released SQL Server on containers only based on Ubuntu images. Some customers prefer Red Hat Enterprise Linux (RHEL) and in fact Red Hat requires container images to be based on RHEL to be officially supported on platforms like OpenShift.

The Solution

SQL Server 2019 on Linux now supports Replication. Since SQL Server supports containers on Linux, it is possible to create a SQL Server replication solution using containers. In addition, SQL Server 2019 now supports container images based on Red Hat Enterprise Linux (RHEL). These forces can be combined to build a solution using Docker. Anywhere Docker runs (Windows, Linux, MacOS, Kubernetes), you can build a SQL Server Replication solution with a publisher, distributor, and subscriber all with docker containers.

SQL Server container images include SQL Server "pre-installed" with the SQL Server Engine, SQL Server Agent, and tools. You can customize the SQL Server container image to include your own files. One scenario would be to include in the customized container images scripts to create database objects. Since Replication can all be deployed using T-SQL, you can use a customized solution to deploy replication across multiple containers. This customized solution is called the "Vin Yu" method named after the lead program manager within the SQL Server team. You can then bring in a tool like **docker-compose** to deploy this solution all with a single command.

The Further Reading section at the end of this module provides reference information about SQL Server 2019 on Linux, Docker containers, and Replication on Linux.

Proceed to the activity to explore how this method can allow you to deploy SQL Server 2019 on Linux with Replication with a single command.



Activity: Deploying SQL Server Replication on Linux using Containers

In this activity, you will learn how to deploy a SQL Server replication solution with a publisher, distributor, and subscriber using two SQL Server Linux containers.

NOTE: If at anytime during the Activities of this Module you need to "start over" you can go back to the first Activity in 5.0 and run through all the steps again.



Activity Steps

All scripts for this activity can be found in the **sql2019lab\05_Linux_and_Containers\replication** folder. These scripts and this activity is based on the sample Vin Yu built as found at <https://github.com/microsoft/sql-server-samples/tree/master/samples/containers/replication>.

NOTE: This activity assumes the following:

- **Docker** is installed. You can use Docker Desktop for Windows or macOS or Docker for Linux to run this activity.
- You have the **docker-compose** tool installed. In many docker installations, docker-compose comes with the install.
- You have internet access to pull SQL Server images from mcr.microsoft.com
- You have **SQL Server Management Studio** (SSMS) 18.1 or higher installed to use as a client. If you are running this example on Linux or MacOS you can still view the results of the replication setup using T-SQL commands. SSMS provides graphical navigation that is not currently available in other tools.

STEP 1: Explore how the solution works

It will help to describe how scripts and files are provided to make the solution work. Examine the following files described in this step to see the flow.

In the **sql2019lab\05_Linux_and_Containers\replication** folder there are two directories and a file:

- **db1** - Scripts and files to build a custom container image for the publisher and distributor.
- **db2** - Scripts and files to build a customer container for the subscriber.
- **docker-compose.yml** - A text file that describes how docker-compose should build and run containers for the solution. Notice in this file are specifics for running the container including the SA password and enabling SQL Agent (which is required for replication)

When docker-compose is run it will use the docker-compose.yml file to build two container images and run them based on files in the db1 and db2 directories.

In each **db1** and **db2** directory are the following files:

- **Dockerfile** - This tells docker how to build a custom image based on the SQL Server 2019 RHEL image
- **entrypoint.sh** - This is a Linux shell script included in the custom image which will be the main "program" for the container. This script will run a script called **db-init.sh** and then run the sqlservr program
- **db-init.sh** - This shell script will sleep for a period of time waiting for sqlservr to start up. Then it will use sqlcmd inside the container to run a T-SQL script called **db-init.sql**
- **db-init.sql** - This could be any set of T-SQL statements to run after sqlservr has started. For the db1 directory this will create a database, a table, insert data, and then run T-SQL scripts to configure a publisher, distributor, subscriber, publication, article, and agents to run a snapshot. This script for the db2 directory only creates an empty database to hold the table in the publication.

So the sequence will be:

1. docker-compose up
2. Build the custom image in the db1 directory and run that container
3. Build the custom image in the db2 directory and run that container.

Once the container db1 has been run, docker-compose will shift and build and run db2. db1 will be running its db-init.sql script while db2 starts up so the output from the command line may look interleaved.

Step 2: Deploy SQL Server Replication Containers with docker-compose

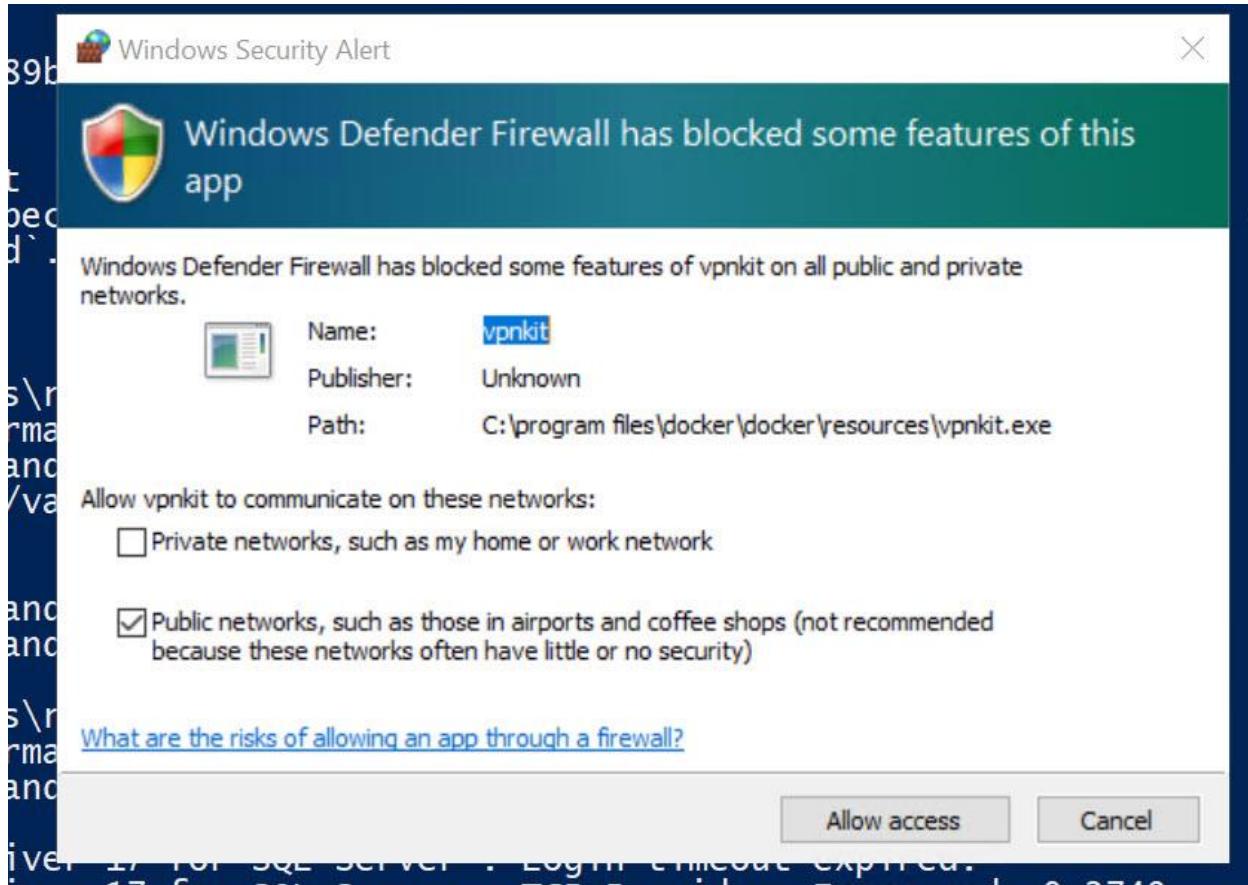
To run the solution, simple type the following from the command line (you must be in the **replication** directory)

```
docker-compose up
```

When you run this you will see alot of information scroll by on the command line. If the docker images for the lab have not been pulled it will take a few minutes for docker to pull them into local storage.

Since docker containers run sqlservr from the command line, part of the output is the ERRORLOG from each server startup being written to stdout.

If you are using Docker for Windows, you may also see the following firewall pop-up from Windows Defender if you are running this lab on Windows. Please click Allow Access. vpnkit is a gateway used by Docker on Windows to communicate with the VM hosting Linux containers.



You will also see output of the db-init.sql execution mostly from the db1 directory. If you have not previously pulled the images for SQL Server 2019 for RHEL, that will be done first and you will see that output.

The replication solution should be ready to go when you see this output from the command line. It may take a few minutes to see this as the containers are pausing for enough time for SQL Server to startup.

```
db1 | Creating Snapshot...
db1 | Job 'db1-Sales-SnapshotRepl-DB2-1' started successfully.
```

The command line will appear to "hang" at this point. This is normal as both programs are running. Once you see this output leave the command line alone. If you attempt <Ctrl>+<C> or exit the command line it would stop the 2 containers.

NOTE: You can use add the **-d** parameter to docker-compose -up to run the containers in the background. But then you would need to monitor SQL Server replication to see when the snapshot has been applied.

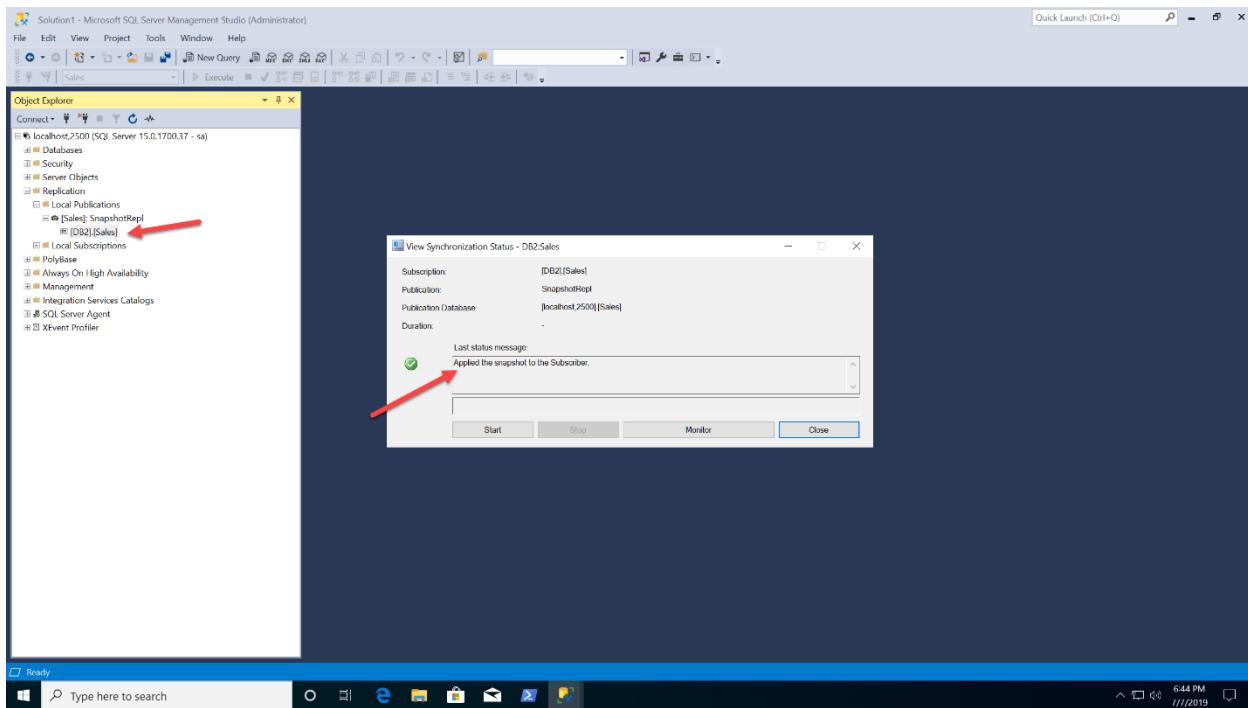
Step 3: Verify the Replication Deployment

The docker-compose.yml file contains key information to connect to both SQL Server containers.

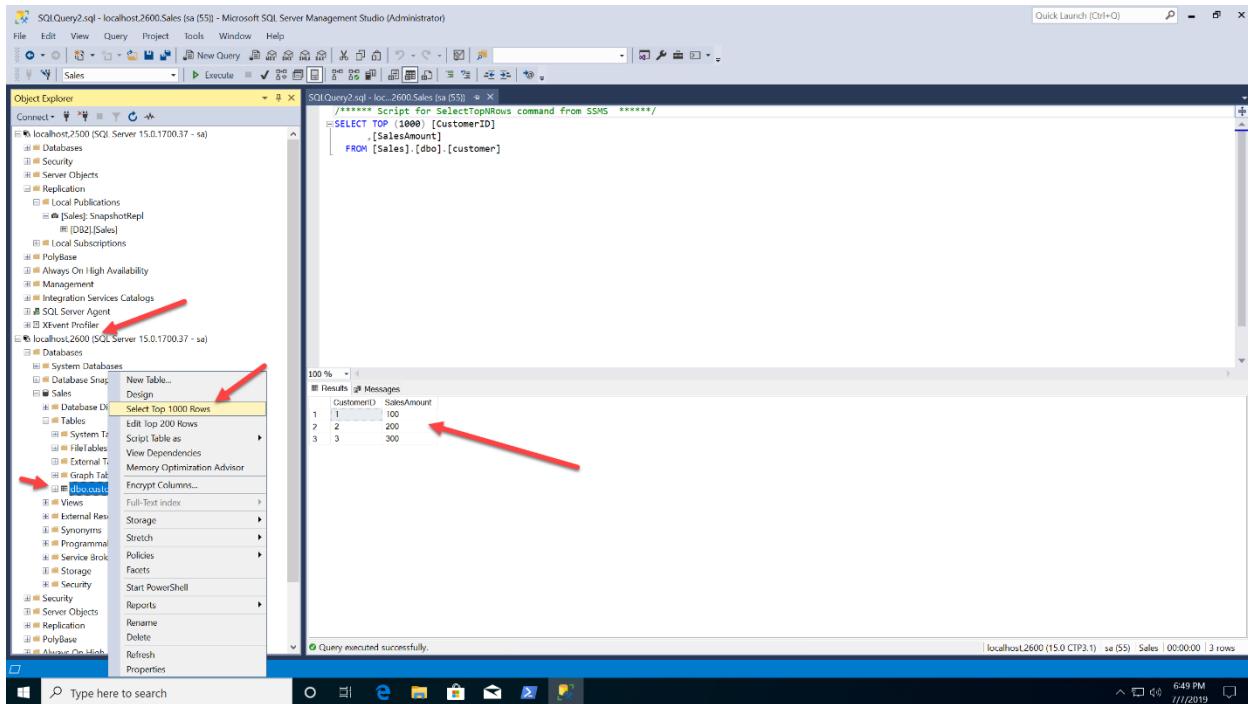
- port 2500 is mapped to the publisher
- port 2600 is mapped to the subscriber

Connect to localhost,2500 (or <server>,2500) for the SQL Server publisher. The sa password can be found in the **docker-compose.yml** file.

Now navigate using SSMS to check the status of the replication snapshot. Use the figure below as a guide on how to navigate Object Explorer to check this status



Now connect to the subscriber (port 2600) to check that the new table, customers in the Sales database, and data are present. Use the following figure as a guide to navigate and check the table and data. The sa password can be found in the **docker-compose.yml** file.



Step 4: Shutdown and remove the containers

If you open up a new command line shell or Powershell window, you can simply run the following command:

```
docker-compose down
```

and the containers will be stopped and removed.

You can also hit **<Ctrl>+<C>** from the command line where you run docker-compose up. This will stop the containers. You will then need to run

```
docker-compose down
```

to remove the containers. The images that you built with docker-compose will still be cached on the machine.

When you are done proceed to the **Activity Summary** section for the Activity below.

Activity Summary

In this activity, you learned how to deploy a SQL Server replication topology with a single command. You learned that SQL Server on Linux is completely compatible with SQL Server Replication on Windows. You used SQL Server 2019 on Linux combined with containers to easily deploy a publisher, distributor, and subscriber. You learned some of the details behind how to build custom images and add your own T-SQL scripts when a SQL Server container is started.



For Further Study

- [SQL Server on Linux Overview](#)
- [Running SQL Server container images with Docker](#)
- [SQL Server Replication on Linux](#)
- [Dockerfile reference](#)
- [Docker compose](#)



Next Steps

Next, Continue to [Additional Capabilities, Migration, and Next Steps](#).



Workshop: SQL Server 2019 Lab (RC)

A Microsoft workshop from the SQL Server team



Additional Capabilities in SQL Server 2019 and Migration

This module is designed for you to read and learn about additional capabilities in SQL Server 2019, tools to assist Migration, using Database Compatibility, and Next Steps for your journey with SQL Server 2019. There are no activities for this module.

Use the Powerpoint slides the [slides](#) folder as you read through this section for more details.

You will cover the following topics in this Module:

[6.0 Additional Capabilities in SQL Server 2019](#)

[6.1 Migration and Database Compatibility](#)



6.0 Additional Capabilities in SQL Server 2019

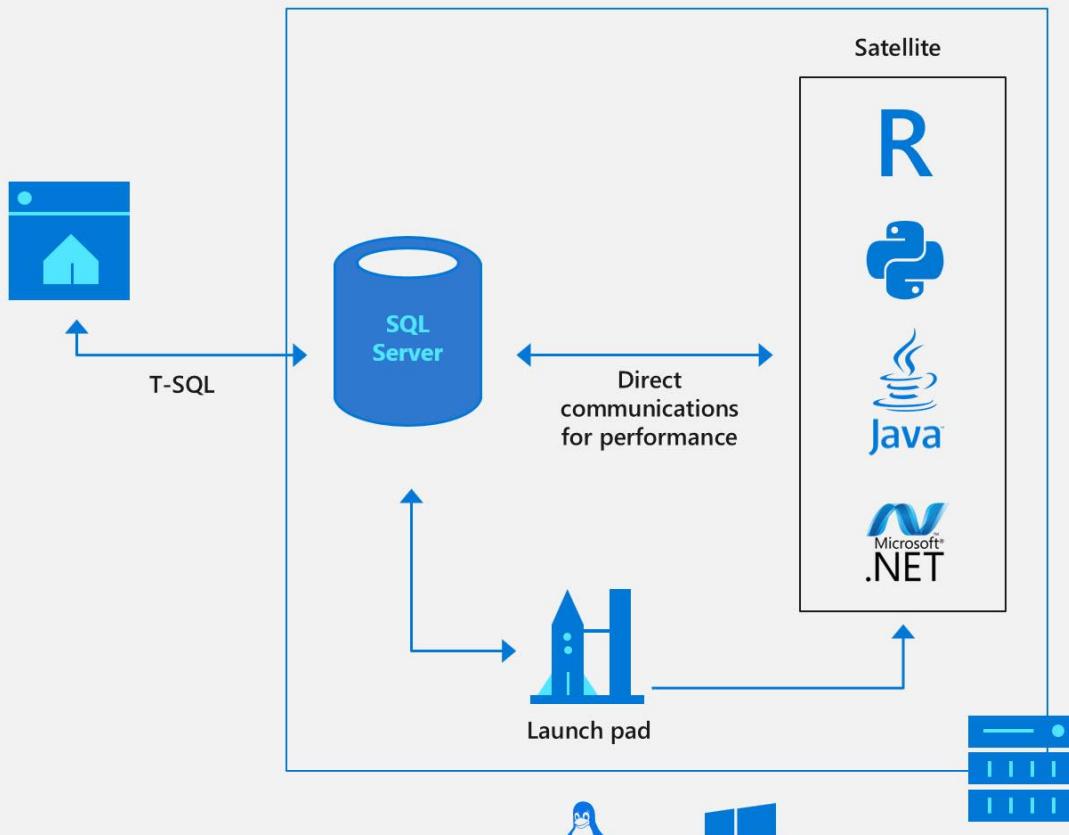
In this section of the module you will learn about additional capabilities in SQL Server 2019 including:

- SQL Server Language Extensions
- Features in the Database Engine based on the Voice of the Customer

SQL Server Language Extensions

To meet the challenge of extending the T-SQL language for requirements not met with T-SQL today we have built a framework called SQL Server Language Extensions. Using the same architecture as SQL Server Machine Learning Services, we now allow for new languages to be accessed via T-SQL as seen in the following diagram:

Extend T-SQL with R, Python, and Java



You can read more about SQL Server Language Extensions at <https://docs.microsoft.com/en-us/sql/language-extensions/language-extensions-overview>.

In SQL Server 2019, we have shipped Java as an example of using SQL Server Language Extensions. As part of the Java installation, we now include an open-source, full-supported version of Java from Azul.

After taking this workshop you can try the Java tutorial to implement Regular Expressions at <https://docs.microsoft.com/en-us/sql/language-extensions/tutorials/search-for-string-using-regular-expressions-in-java>.

The Voice of the Customer

All the capabilities you see in SQL Server 2019 are based on some type of feedback from our customers. While you have seen major capabilities in SQL Server 2019, there are other new features we have added to this release all based on the voice of the customer and customer feedback.

This list includes but is not limited to:

- Columnstore stats in DBCC CLONEDATABASE
- Estimate compression for Columnstore indexes
- Diagnostics for auto stats blocking
- The #1 voted customer feedback item of all time: String Truncation Error Message
- Troubleshoot page resource waits with new built-in T-SQL
- Custom capture policy for the Query Store
- MAXDOP and Memory Config during Setup
- Reduce recompiles for tempdb workloads
- Indirect Checkpoint Scalability

You can get more details about all of these in our documentation at <https://docs.microsoft.com/en-us/sql/sql-server/what-s-new-in-sql-server-ver15>.



6.1 Migration and Database Compatibility

As work towards a migration to SQL Server 2019, you need to consider a planned process for pre-migration, migration, and post-migration. We have provided a series of tools to help with this process.

In addition, database compatibility can become an important solution to make migrations easier and reduce the risk of upgrades.

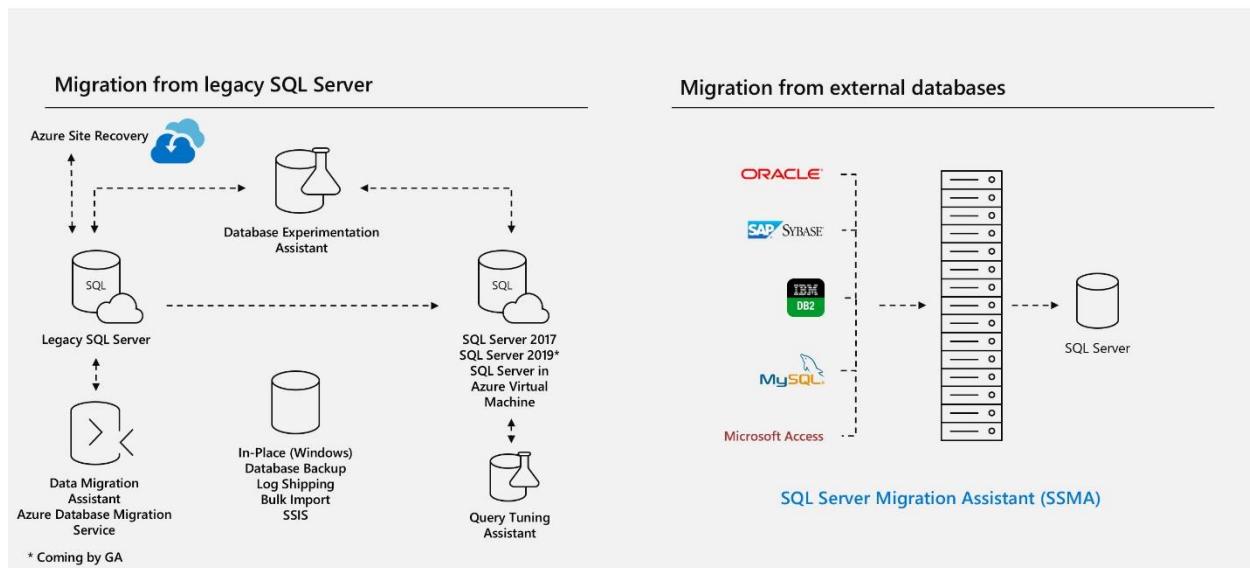
Consider watching the following video called **Modernizing SQL Server** by Pam Lahoud and Pedro Lopes to get the complete picture of migration and database compatibility.



Migration Tools

As you plan for a migration to SQL Server 2019, consider looking at the process to perform a pre-migration assessment, your options to migrate, and post-migration optimizations. The following diagram outlines tools we provide for these scenarios:

Migrate to the Modern SQL Server



The following tools are available to help you in this process:

- **Data Migration Assistant (DMA)**

Download and use this free tool to assess the configuration of your SQL Server and database. You can read more about DMA at <https://docs.microsoft.com/en-us/sql/dma/dma-overview>.

- **Database Experimentation Assistant (DEA)**

Download and use this free tool to assess the performance of your queries using a replay/compare method. Read more about DEA at <https://docs.microsoft.com/en-us/sql/dea/database-experimentation-assistant-overview>.

- **Query Tuning Assistant (QTA)**

Use the power of Query Store along with the "knowledge base" of the SQL Engineering team to optimize queries after you migrate and want to look at moving to a newer database compatibility level. You can read more about QTA at <https://docs.microsoft.com/en-us/sql/relational-databases/performance/upgrade-dbcompat-using-qta>.

- **SQL Server Migration Assistant (SSMA)**

Migrate your schema and data from other database products into SQL Server using SQL Server Migration Assistant. You can read more about SSMA at <https://docs.microsoft.com/en-us/sql/ssma/sql-server-migration-assistant>.

In addition, if you need to lift and shift a SQL Server Virtual Machine directly to Azure take a look at the **Azure Site Recovery** Service at <https://docs.microsoft.com/en-us/azure/site-recovery/migrate-tutorial-on-premises-azure>.

Database Compatibility

Database Compatibility Levels can be a powerful solution to reduce the risk of upgrades to the latest releases of SQL Server. This includes protection of functionality and query plans. Microsoft is encouraging developers to use database compatibility as a method to *certify* an application vs a version of SQL Server.

Read all the details of using Database Compatibility Levels for upgrades at <https://aka.ms/dbcompat>.



For Further Study

- [SQL Server Language Extensions](#)
- [What's New in SQL Server 2019](#)
- [Data Migration Assistant](#)

- [Database Experimentation Assistant](#)
- [Query Tuning Assistant](#)
- [SQL Server Migration Assistant](#)
- [Database Compatibility](#)



Next Steps

Use the following resources to learn more on your journey to SQL Server 2019:

- Take advantage of all the free training at <https://aka.ms/sqlworkshops>
- Learn from videos and demos at <https://aka.ms/sqlchannel>
- Download and try SQL Server 2019 at <http://aka.ms/ss19>
- Watch the video of Modernizing SQL Server at <https://www.youtube.com/watch?v=5RPkuQHcxxs> to plan your migration
- Read the what's new for SQL 2019 documentation at <https://docs.microsoft.com/en-us/sql/sql-server/what-s-new-in-sql-server-ver15>
- Sign-up for the EAP program at <https://aka.ms/eapsignup>



Workshop: SQL Ground-to-Cloud

A Microsoft workshop from the SQL Server team



03 - Working with Big Data and Data Science - Big Data Clusters for SQL Server 2019

In this Module of the Workshop you'll cover using SQL Server on-premises and in-cloud configurations, as well as hybrid applications as a solution for data processing. In each section you'll get more references, which you should follow up on to learn more. Also

watch for links within the text - click on each one to explore that topic. The end of this module contains several helpful references you can use in this course and in production.

([Make sure you check out the prerequisite page before you start](#). You'll need all of the items loaded there before you can proceed with the workshop.)

In this module you'll cover working with Data Science workloads with a focus on larger sets of data. Starting in SQL Server 2019, big data clusters allows for large-scale, near real-time processing of data over the HDFS file system and other data sources. It also leverages the Apache Spark framework which is integrated into one environment for management, monitoring, and security of your environment. This means that organizations can implement everything from queries to analysis to Machine Learning and Artificial Intelligence within SQL Server, over large-scale, heterogeneous data. SQL Server big data clusters can be implemented fully on-premises, in the cloud using a Kubernetes service such as Azure's AKS, and in a hybrid fashion. This allows for full, partial, and mixed security and control as desired.

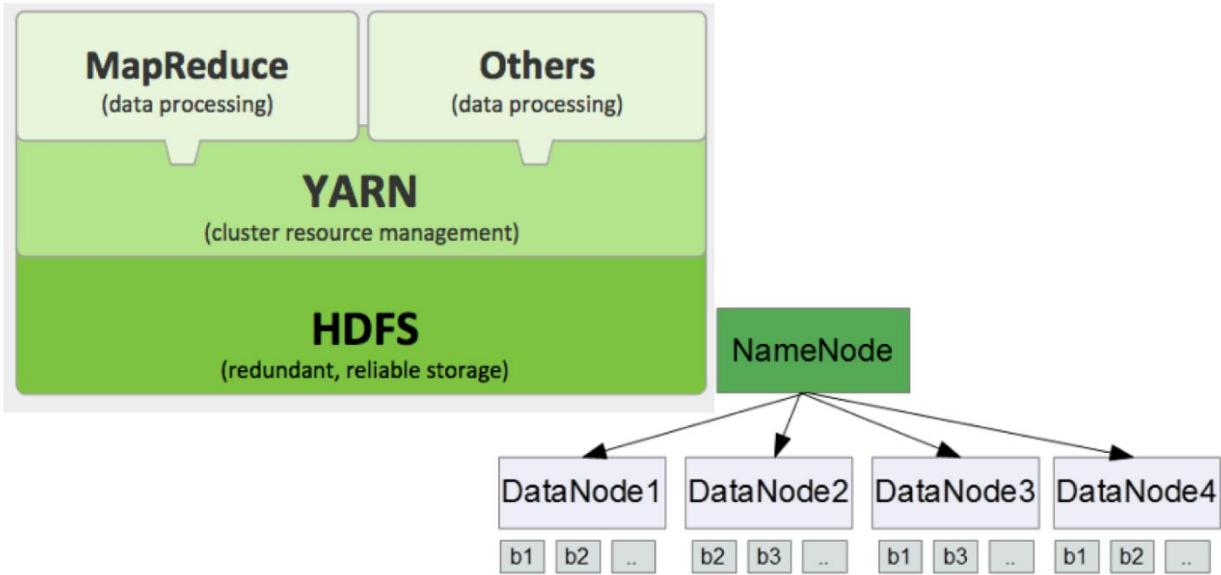
Note: This is a complex topic, so you can find a [complete workshop on this topic here](#).



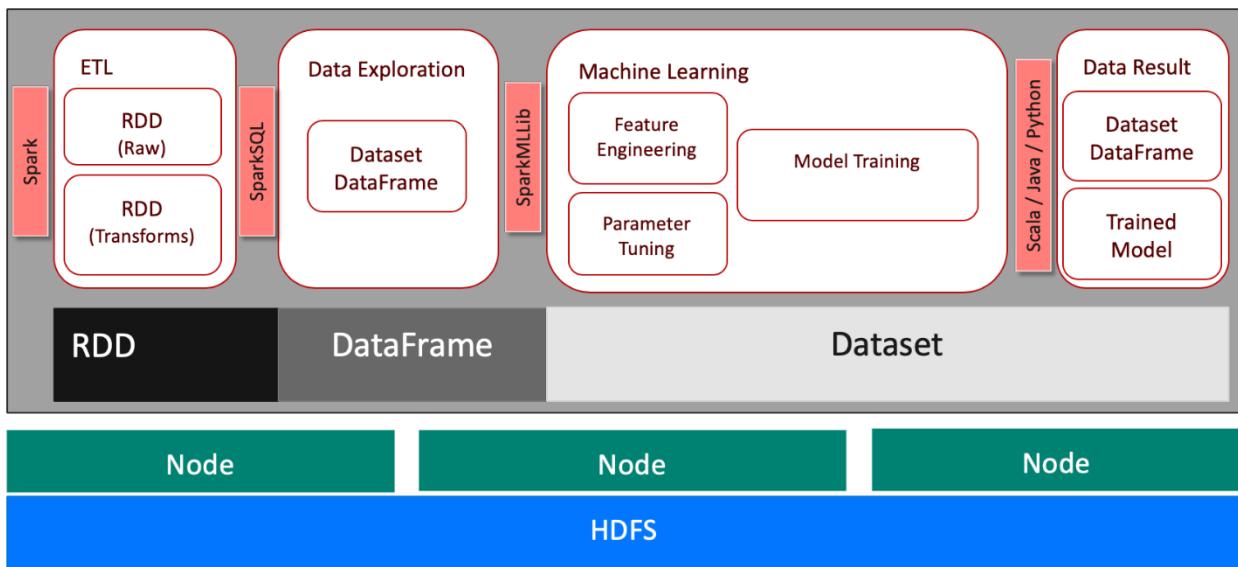
3.1 Data Science and Big Data Processing

Businesses require near real-time insights from ever-larger sets of data from a variety of sources. Large-scale data ingestion requires scale-out storage and processing in ways that allow fast response times. In addition to simply querying this data, organizations want full analysis and even predictive capabilities over their data. Machine Learning, Artificial Intelligence, and Deep Learning techniques all require large sets of data for training their models to be effective. Two technologies have emerged as the primary methods for processing large sets of data, using a scale-out paradigm - Hadoop and Spark.

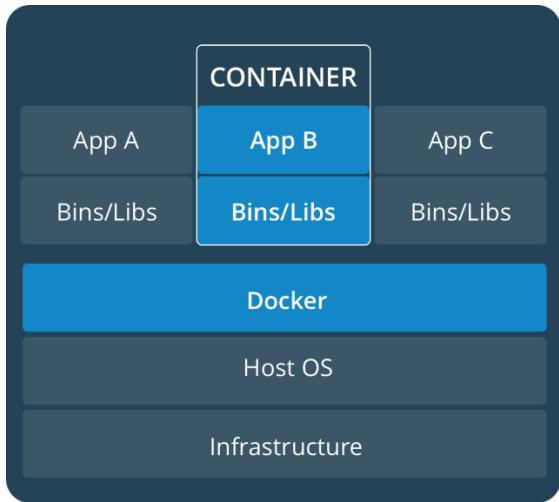
Hadoop uses a set of computing nodes that position the workload over distributed data nodes:



Spark is a technology that uses various libraries to make processing over distributed storage more efficient and faster:



Both of these technologies assume many nodes (computers), and since you only need the computation elements (not all the drivers and other components of a full computer or Virtual Machine), Container technologies work well for this solution.

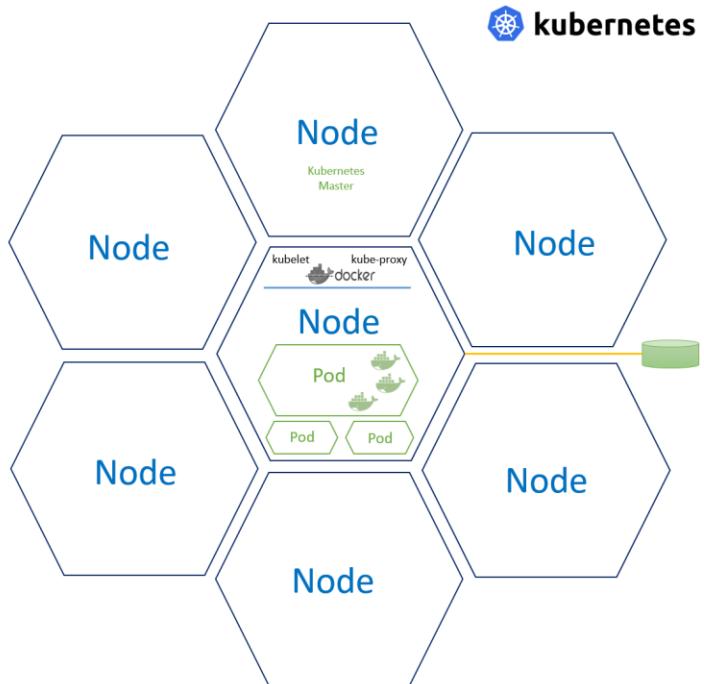


To control containers, a technology called *Kubernetes* is used for deployment, management and storage of a grouping of containers, called a *Cluster*.

Container Orchestration

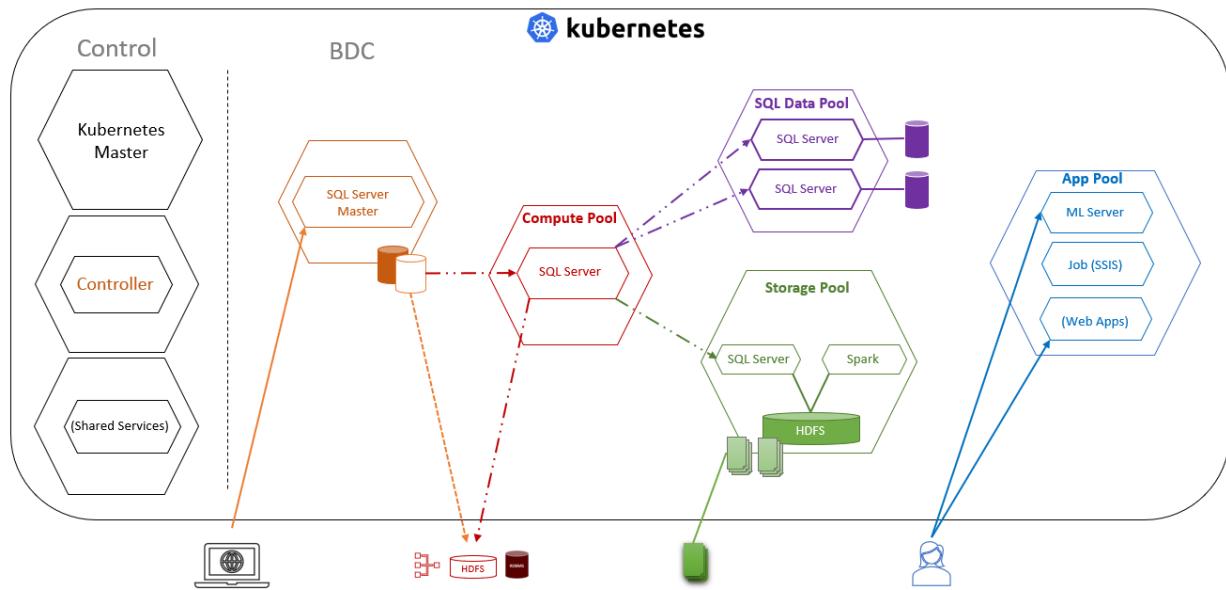
Containers at Scale

- Container(s) live in *Pods*
- Pod(s) are abstractions within *Nodes*
- Node(s) are PC's or VM's
- Cluster(s) are groups of *Nodes*
- Storage is by means of Volume(s) mounted through a *Claim*



3.2 SQL Server 2019 Big Data Clusters Architecture

Using the technologies described above, SQL Server uses a Kubernetes Cluster to deploy multiple components for SQL Server processing, distributed queries using PolyBase, Spark, and Storage on HDFS for a complete environment to work with large sets of data, and includes Machine Learning, Artificial Intelligence and Deep Learning capabilities.



A SQL Server Big Data Cluster (BDC) can be deployed to multiple environments, in-cloud and on-premises:

- In a Cloud Service (Such as the Azure Kubernetes Service or AKS)
- On premises (using KubeADM)

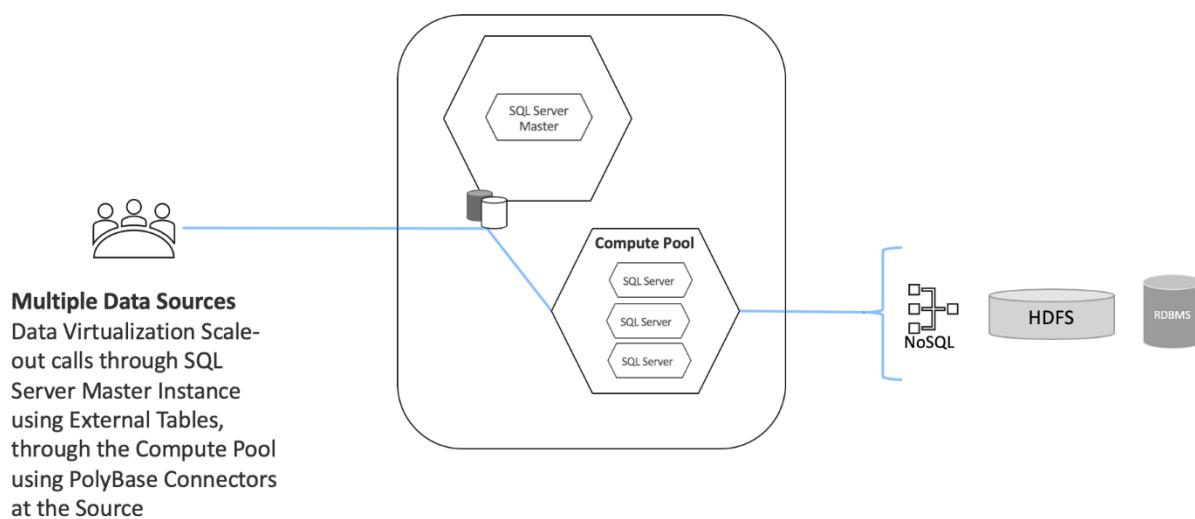
These architectures are not mutually exclusive - you can install some components on-premises, and others as a service. Your connections can interconnect across these environments.

3.3 SQL Server 2019 Big Data Clusters Programming

In addition to traditional OLTP workloads, the SQL Server Big Data Cluster has three other uses that work with data at scale.

Data Virtualization with SQL Server Big Data Clusters

Using the Data Virtualization capability of PolyBase in SQL Server Big Data Clusters you create External Tables for data outside the Cluster. These External Table definitions are stored in the database on the SQL Server Master Instance within the cluster. When queried by the user, the queries are engaged from the SQL Server Master Instance through the Compute Pool in the SQL Server BDC, which holds Kubernetes Nodes containing the Pods running SQL Server Instances. These Instances send the query to the PolyBase Connector at the target data system, which processes the query based on the type of target system. The results are processed and returned through the PolyBase Connector to the Compute Pool and then on to the Master Instance, and then on to the user.

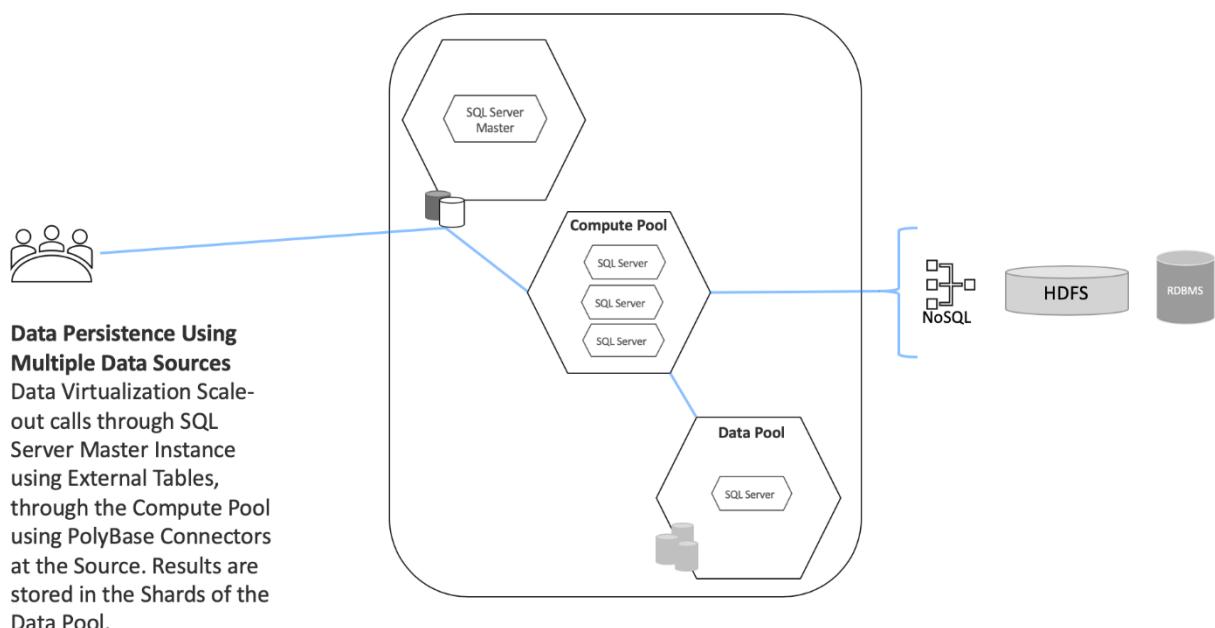


This process allows not only a query to disparate systems, but also those remote systems can hold extremely large sets of data. Normally you are querying a subset of that data, so the results are all that are sent back over the network. These results can be joined with internal tables for a single view, and all from within the same Transact-SQL statements.

Data Marts with SQL Server Big Data Clusters

Ad-hoc queries over large sets of normalized data are very useful for many scenarios. There are times when you would like to bring the data into storage, so that you can create denormalized representations of datasets, aggregated data, and other purpose-specific data tasks.

Using the Data Virtualization capability (PolyBase), the IT team creates External Tables using PolyBase statements. These External Table definitions are stored in the database on the SQL Server Master Instance within the cluster. When queried by the user, the queries are executed from the SQL Server Master Instance through the Compute Pool in the SQL Server BDC, which holds Kubernetes Nodes containing the Pods running SQL Server Instances. These Instances send the query to the PolyBase Connector at the target data system, which processes the query based on the type of target system. The results are processed and returned through the PolyBase Connector to the Compute Pool and then on to the Master Instance, and the PolyBase statements can specify the target of the Data Pool. The SQL Server Instances in the Data Pool store the data in a distributed fashion across multiple databases, called *Shards*.

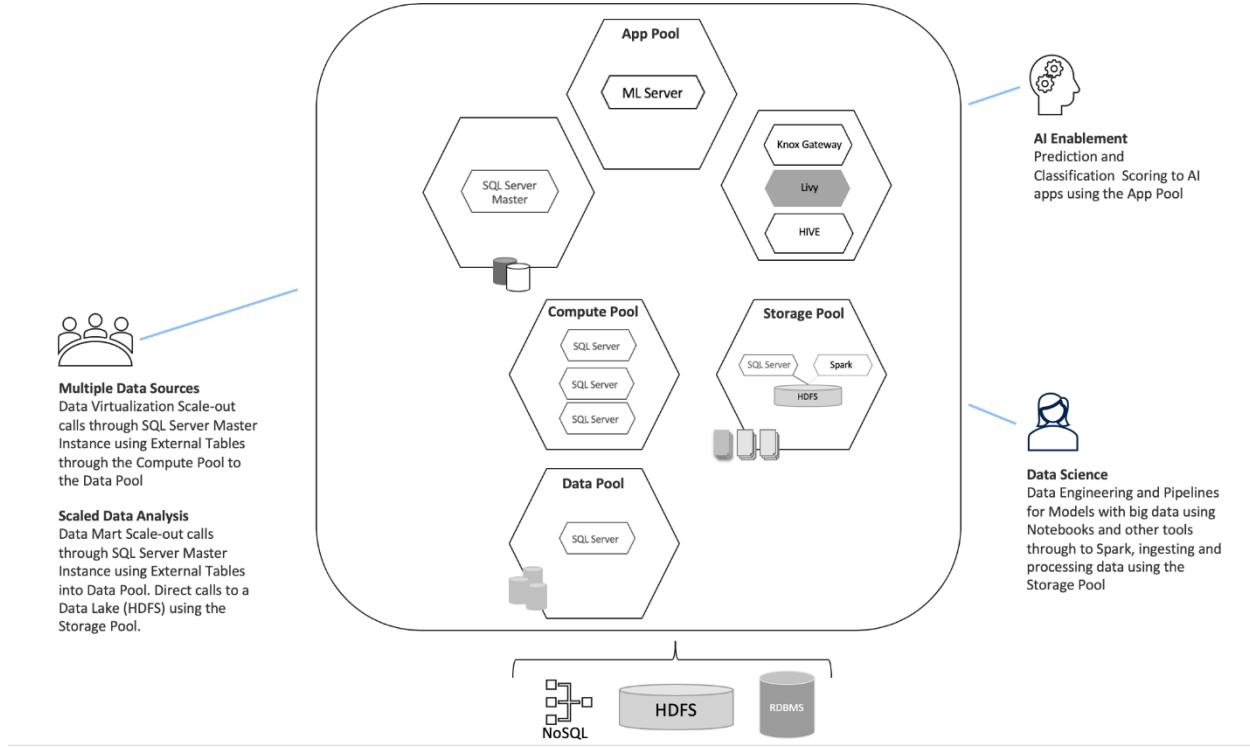


Data Science and Apache Spark with SQL Server Big Data Clusters

There are various uses for a large cluster of data processing systems for Machine Learning and AI applications. One primary use-case is the creation of the [Features and Labels used in various ML and AI algorithms](#). Large sets of data stored in the Data Pool allow Python, R and Spark calls and libraries are available for this task in BDC. This process is often called *Data Engineering*.

The SQL Server Master Instance in the BDC installs with [Machine Learning Services](#), which allow creation, training, evaluation and persisting of Machine Learning Models. Data from all parts of the BDC are available, and Data Science oriented languages and libraries in R, Python and Java are enabled. In this scenario, the Data Scientist creates the R or Python code, and the Transact-SQL Developer wraps that code in a Stored Procedure. This code can be used to train, evaluate and create Machine Learning Models. The Models can be stored in the Master Instance for scoring, or sent on to the App Pool where the Machine Learning Server is running, waiting to accept REST-based calls from applications.

The Data Scientist has another option to create and train ML and AI models. The Spark platform within the Storage Pool is accessible through the Knox gateway, using Livy to send Spark Jobs. This gives access to the full Spark platform, using Jupyter Notebooks (included in *Azure Data Studio*) or any other standard tools that can access Spark through REST calls.



Activity: Big Data Clusters for SQL Server Lab Review

In this lab, you will review a series of Jupyter Notebooks you can open in Azure Data Studio to work with a Big Data Cluster. It demonstrates loading data, querying SQL Server databases, accessing HDFS using PolyBase, loading data into the Data Pool, working with Spark, and implementing a Data Science Machine Learning Model in Python.

Note: This lab assumes a completely installed and function SQL Server Big Data Cluster if you wish to run the Notebooks, otherwise review the completed Notebooks at the reference.



Steps

[Open this reference and review the notebooks you find there.](#)



For Further Study

- [Understanding the Big Data Landscape](#)

- [Linux for the Windows Admin](#)
- [Docker Guide](#)
- [Video introduction to Kubernetes](#)
- [Complete course on Azure Kubernetes Service \(AKS\)](#)
- [Working with Spark](#)
- [Full tutorial on Jupyter Notebooks](#)



Next Steps

Next, Continue to [04 - SQL Server on the Microsoft Azure Platform](#).



Workshop: SQL Ground-to-Cloud

A Microsoft workshop from the SQL Server team



04 - SQL Server on the Microsoft Azure Platform

In the previous modules, you learned about SQL Server 2019, big data, and more. In this module, you'll learn more about Azure SQL, including the benefits, the options, and how to get there. Near the end of the module, you'll learn how to assess your on-premises estate, through an example with the Tailspin Toys Gaming division of Wide World Importers, with tools like [Azure Migrate](#) and [Data Migration Assistant](#). The module is broken up into the following sections:

- [4.1: Introduction to Azure SQL](#)
- [4.2: Azure SQL Platform Benefits](#)
- [4.3: Migrating to Azure SQL](#)
- [4.4: Database Discovery and Assessment](#)

([Make sure you check out the Prerequisites page before you start](#). You'll need all of the items loaded there before you can proceed with the workshop.)

SELF-PACED USERS ONLY: If you are using this module self-paced, carefully read through Module 4 of this workshop and the references provided before completing the lab in Module 4.4. You can also check out the [slides that will be presented in a live workshop](#). Then, continue to Module 5.

Note:

While all the deployment options will be referenced, the focus of this module is on **Azure SQL Managed Instance**, with some details that are also related to **Azure SQL Database (single database)**.



4.1 Introduction to Azure SQL

In this section, you'll learn about the different Azure SQL deployment options and service tiers available for Tailspin Toys Gaming, a division of Wide World Importers that focuses on online gameplay. You'll start with understanding Tailspin Toys Gaming's scenario deeper, so you can determine which options are best suited to their business needs.

Scenario Review: Tailspin Toys Gaming

Tailspin Toys is the developer of several popular online video games. Founded in 2010, and acquired shortly after by Wide World Importers, the company has experienced exponential growth since releasing the first installment of their most popular game franchise to include online multiplayer gameplay. They have since built upon this success by adding online capabilities to the majority of their game portfolio.

Adding online gameplay has greatly increased popularity of their games, but the rapid increase in demand for their services has made supporting the current setup problematic. To facilitate online gameplay, they host gaming services on-premises using rented hardware. For each game, their gaming services setup consists of three virtual machines running the gaming software and five game databases hosted on a single SQL Server 2008 R2 instance. In addition to the dedicated gaming VMs and databases, they also host authentication and gateway VMs and databases, which are shared by all their games. At its foundation, Tailspin Toys is a game development company, made up primarily of software developers. The few dedicated database and infrastructure resources they do have are struggling to keep up with their ever-increasing workload.

Tailspin Toys is hoping that migrating their services from on-premises to the cloud can help to alleviate some of their infrastructure management issues, while simultaneously helping them to refocus their efforts on delivering business value by releasing new and

improved games. They are looking for a proof-of-concept (PoC) for migrating their gamer information web application and database into the cloud. They maintain their gamer information database, TailspinToys, on an on-premises SQL Server 2008 R2 database. This system is used by gamers to update their profiles, view leader boards, purchase game add-ons and more. Since this system helps to drive revenue, it is considered a business-critical application, and needs to be highly-available. They are aware that SQL Server 2008 R2 has reached end of support, and are looking at options for migrating this database into Azure. They have read about some of the advanced security and performance tuning options that are available only in Azure and would prefer to migrate the database into a platform-as-a-service (PaaS) offering, if possible. Tailspin Toys is using the Service Broker feature of SQL Server for messaging within the TailspinToys database. This functionality is being used for several critical processes, and they cannot afford to lose this capability when migrating their operations database to the cloud. They have also stated that, at this time, they do not have the resources to rearchitect the solution to use an alternative message broker.

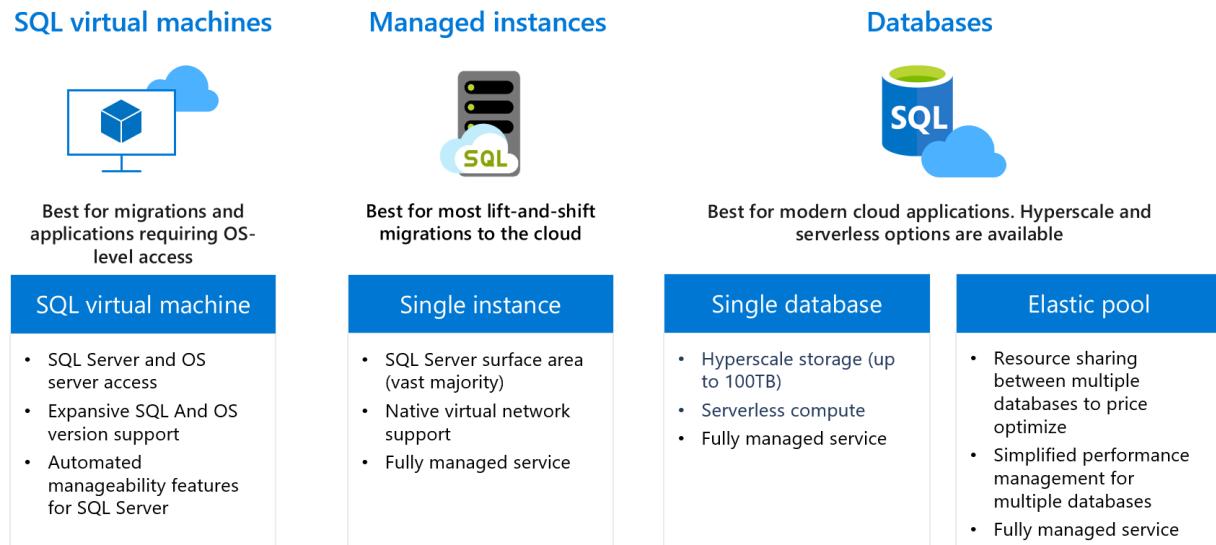
If you'd like to conduct a deep whiteboard design session for this scenario, it is available [here](#).

Azure SQL deployment options

Tailspin Toys has requested your help in determining where and how they should land their data in Azure. This is a decision that all businesses moving to the cloud have to make, and the result will depend on their unique business requirements. Microsoft has recently introduced Azure SQL, which brings all the SQL Server products in Azure under one suite. If you're already familiar with [Azure SQL Database](#), this slight shift means that Azure SQL also includes [SQL Server on Virtual Machines](#) (also referred to as Azure SQL VMs), which Microsoft is continuously investing in and enhancing the benefits associated with it.

There are several different options within Azure SQL, and one visual is shown below.

Azure SQL



Recently, we announced [Azure SQL Database Instance Pools in preview](#), which provide a convenient and cost-efficient way to migrate smaller SQL instances to the cloud at scale.

Every business is going to have unique requirements that make some options better for them than others. There won't necessarily be one 'correct' answer, but there are tradeoffs to consider. Let's look at some examples of Azure SQL customers and what they decided:

- **Allscripts:** Allscripts is a leading healthcare software manufacturer, serving physician practices, hospitals, health plans, and Big Pharma. To transform its applications frequently and host them securely and reliably, Allscripts wanted to move to Azure quickly. In just three weeks, the company lifted and shifted dozens of acquired applications running on 1,000 virtual machines to Azure with [Azure Site Recovery](#). After the migration to **Azure SQL VMs**, Allscripts began to evaluate and test **Azure SQL Managed Instance**, and started to move some workloads there.
- **Komatsu:** Komatsu is a manufacturing company that produces and sells heavy equipment for construction. They had multiple mainframe applications for different types of data, which they wanted to consolidate to get a holistic view. Additionally, they wanted a way reduce overhead. Because Komatsu uses a large surface area of SQL Server features, they chose to move to **Azure SQL Managed Instance**. They were able to move about 1.5 terabytes of data smoothly, and [start enjoying benefits like automatic patching and version updates, automated](#)

[backups, high availability, and reduced management overhead](#). After migrating, they reported ~49% cost reduction and ~25-30% performance gains.

- **AccuWeather:** AccuWeather has been analyzing and predicting the weather for more than 55 years. They wanted access to the rich and rapidly advanced platform of Azure that includes big data, machine learning, and AI capabilities. They want to focus on building new models and applications, not managing databases. They selected **Azure SQL Database (single database)** to use with other services, like [Azure Data Factory](#) and [Azure Machine Learning Services](#), to quickly and easily deploy new internal applications to make sales and customer predictions.
- **Paychex:** Paychex is a human capital management firm that serves more than 650,000 businesses across the US and Europe. They needed a way to separately manage the time and pay management for each of their businesses, and cut costs. They opted for **Azure SQL Database (elastic pools)**, which allowed them to simplify the management and enable resource sharing between separate databases to lower costs.

If you want to dive deeper into the deployment options and how to choose, check out the following resources:

- [Blog announcement for Azure SQL](#) which explains and walks through Azure SQL and some of the resulting views and experiences available in the Azure portal.
- [Microsoft Customer Stories](#) for many more stories similar to the ones above. You can use this to explore various use cases, industries, and solutions.
- [Choose the right deployment option in Azure SQL](#) is a page in the documentation regularly updated to help provide insight into making the decisions between the Azure SQL options.
- [Choosing your database migration path to Azure](#) is a white paper that talks about tools for discovering, assessing, planning and migrating SQL databases to Azure. This workshop will refer to it several times, and it's a highly recommended read. Chapter 5 deeply discusses choosing the right deployment option.
- [Feature comparison between SQL database, SQL managed instance, and SQL Server](#)
- There are also tools covered later in this module that can help in recommending a deployment option.

Azure SQL service tiers and other options

Once you have an understanding of the deployment options, it's time to consider some of the details within the service tiers that are available.

SQL Server on Virtual Machines options

Azure SQL VMs don't have specific service tiers, but there are decisions to be made around which images to choose, how to configure storage and what sizes, etc. This isn't the focus of this workshop, but if you're considering Azure SQL VMs, you'll want to review the [guidance on images to choose from](#), the [quick checklist](#) to obtain optimal performance of Azure SQL VMs, and the guidance for [storage configuration](#).

Recently, [Resource Provider](#) was [announced](#), which brings the functionality of Azure Marketplace images to SQL Server instances self-installed on Azure VMs.

Note: If you're specifically looking at SQL Server on RHEL Azure VMs, there's a full operations guide available [here](#).

Note: In an earlier module of this workshop, you learned about some of the problems SQL Server 2019 is solving. The same applies in an Azure SQL VM (if you choose 2019 as the target).

Azure SQL Database options

For Azure SQL Database, which is the focus of this module, there are several options and tiers available, and the choices will depend on the scenario.

There are a few main decisions to be made, which will be explored next.

Decision 1: Choose the purchasing model

You have two options, [virtual core \(vCore\)-based](#) (recommended) or [Database transaction unit \(DTU\)-based](#).

The vCore-based model is recommended because it allows you to independently choose compute and storage resources, while the DTU-based model is a bundled measure of compute, storage and I/O resources, which means you have less control over paying only for what you need. This model also allows you to use [Azure Hybrid Benefit for SQL Server](#) to gain cost savings. In the [vCore model](#), you pay for:

- Compute resources (the service tier + the number of vCores and the amount of memory + the generation of hardware).
- The type and amount of data and log storage.
- Backup storage ([read-access, geo-redundant storage \(RA-GRS\)](#)).

For the purposes of this workshop, we'll focus on the vCore purchasing model (recommended), but you can [compare vCores and DTUs here](#).

Decision 2: Choose service tier for performance and availability

There are three tiers available in the vCore model for Azure SQL Database:

- **General purpose**: Most business workloads. Offers budget-oriented, balanced, and scalable compute and storage options.
- **Business critical**: Business applications with low-latency response requirements. Offers highest resilience to failures by using several isolated replicas. This is the only tier that can leverage [in-memory OLTP](#) to improve performance.
- **Hyperscale**: Most business workloads with highly scalable storage and read-scale requirements. *Currently only available for single databases, not managed instances.*

A member of the Product Group recently released a [blog](#) and [video](#) explaining the SLA (service level agreements that set an expectation for uptime and performance). This resource will help you make an informed decision about which tier to move to.

For a deeper explanation between the three tiers (including scenarios), you can also refer to the [service-tier characteristics](#) in the documentation.

Decision 3: Provisioned or serverless compute?

If you choose **General Purpose within Single databases** and the **vCore-based model**, you have an additional decision to make regarding the compute that you pay for:

- **Provisioned compute** is meant for more regular usage patterns with higher average compute utilization over time, or multiple databases using elastic pools.
- **Serverless compute** is meant for intermittent, unpredictable usage with lower average compute utilization over time. Serverless has auto-pause and resume capabilities (with a time delay you set), meaning when your database is paused, you only pay for storage.

For a deeper explanation between the two compute options (including scenarios), you can refer to the detailed [comparison in the documentation](#).

If you're looking for compute cost saving opportunities, you can prepay for compute resources with [Azure SQL Database reserved capacity](#).

Decision 4: Choose hardware generation

The vCore model lets you choose the generation of hardware:

- **Gen4:** Up to 24 logical CPUs based on Intel E5-2673 v3 (Haswell) 2.4-GHz processors, vCore = 1 PP (physical core), 7 GB per core, attached SSD
- **Gen5:** Up to 80 logical CPUs based on Intel E5-2673 v4 (Broadwell) 2.3-GHz processors, vCore = 1 LP (hyper-thread), 5.1 GB per core, fast eNVM SSD

Basically, Gen4 hardware offers substantially more memory per vCore. However, Gen5 hardware allows you to scale up compute resources much higher.

Note: If you choose General Purpose within Single databases and want to use the serverless compute tier, Gen5 hardware is the only option.

Summary

As you've hopefully noticed, while there are a lot of options, Azure is able to provide flexibility so you get exactly what you need, nothing less. A summary of the service tier options with some additional considerations is included below, but be sure to check out [pricing information](#) for the latest details.

Service tier	General purpose		Business critical		Hyperscale
Best for	Most budget-oriented workloads		Critical business applications with high IO requirements.		VLDB OLTP and HTAP workloads with highly scalable storage and read-scale requirements
Deployment option	Single / Elastic Pools	Managed Instance	Single / Elastic Pools	Managed Instance	Single
Compute tiers	Gen4: 1 to 24 vCore Gen5: 2 to 80 vCore	Gen4: 4 to 24 vCore Gen5: 4 to 80 vCore	Gen4: 1 to 24 vCore Gen5: 2 to 80 vCore	Gen4: 4 to 24 vCore Gen5: 4 to 80 vCore	Gen4: 1 to 24 vCore Gen5: 2 to 80 vCore
Storage	Premium remote		Local SSD		Local SSD Cache
Storage	32GB – 8TB per instance	32GB – 8TB per instance	32GB – 4TB per instance	32GB – 8TB per instance	Scale from 5GB to 100TB of storage in 1GB increments
In-Memory	Not supported		Supported		Not supported
Read-write IO	~2ms for all data access		<0.5ms for all data access		<0.5ms for hot data access ~2ms otherwise
Availability	2 read replicas		3 read replicas, zone-redundant HA	3 replicas, 1 read-scale replica, zone-redundant HA	Primary read/write replica + up to 4 read replicas
Backups	RA-GRS, 7-35 days (7 days by default)		RA-GRS, 7-35 days (7 days by default)		LRS, ZRS, RA-GRS, 7-35 days (7 days by default)

Note: Data Migration Assistant (covered later in this module) runs scans that can help you choose some of the options as well as the SKU. [Learn more here](#).

Tailspin Toys Gaming is leaning towards Azure SQL Managed Instance, because it supports movement to Azure without changing the database features that are used (in this case, Service Broker). They think business critical is the right tier for them, as they

need to maximize performance and availability. By using the vCore-based model, they're able to leverage their existing investments with Azure Hybrid Benefits (1 Enterprise license core for SQL Server = 1 Business Critical vCore). Additionally, they want to leverage the three replicas provided free with business critical, where one can be used for [read scale-out](#), that is needed as the game experiences heavy read workloads.



4.2 Azure SQL Platform Benefits

Before Tailspin Toys Gaming commits to moving to Azure SQL Managed Instance, they want to understand some of the benefits that come with running their SQL workloads in Azure.

In this section, we will highlight some of the key benefits of choosing Azure for your SQL workloads. If you want to go deeper, additional reference information is available.

Azure SQL Database (including single databases, elastic pools, and managed instances) is a fully managed Database Engine that automates most of the database management functions such as upgrading, patching, backups, and monitoring. Some of the built-in capabilities include:

- [Business continuity](#) enables your business to continue operating in the face of disruption, particularly to its computing infrastructure.
- [High availability](#) of Azure SQL Database guarantees your databases are up and running 99.99% of the time, no need to worry about maintenance/downtimes.
- [Automated backups](#) are created and use Azure read-access geo-redundant storage (RA-GRS) to provide geo-redundancy.
- [Long term backup retention](#) enables you to store specific full databases for up to 10 years.
- [Geo-replication](#) by creating readable replicas of your database in the same or different data center (region).
- [Scale](#) by easily adding more resources (CPU, memory, storage) without long provisioning.
- Network Security
 - [Azure SQL Database \(single database and elastic pool\)](#) provides firewalls to prevent network access to the database server until access is explicitly granted based on IP address or Azure Virtual Network traffic origin.

- [Azure SQL Managed Instance](#) has an extra layer of security in providing native virtual network implementation and connectivity to your on-premises environment using [Azure ExpressRoute](#) or [VPN Gateway](#).
- [Advanced security](#) detects threats and vulnerabilities in your databases and enables you to secure your data.
- [Automatic tuning](#) analyzes your workload and provides you the recommendations that can optimize performance of your applications by adding indexes, removing unused indexes, and automatically fixing the query plan issues.
- [Built-in monitoring](#) capabilities enable you to get the insights into performance of your databases and workload, and troubleshoot the performance issues.
- [Built-in intelligence](#) automatically identifies the potential issues in your workload and provides you the recommendations that can [help you to fix the problems](#).

Note: Many benefits typically thought of as Platform as a Service (PaaS) are surfacing in Infrastructure as a Service (IaaS). You can learn more about automated updates, automated backups, high availability, and performance provided in Azure SQL VMs [here](#).

In addition to the resources linked to above, several [slide decks](#) are available (with notes and animations) that you can review to learn more. You can also check out the [Core Cloud Services - Azure architecture and service guarantees](#) module from Microsoft Learn.



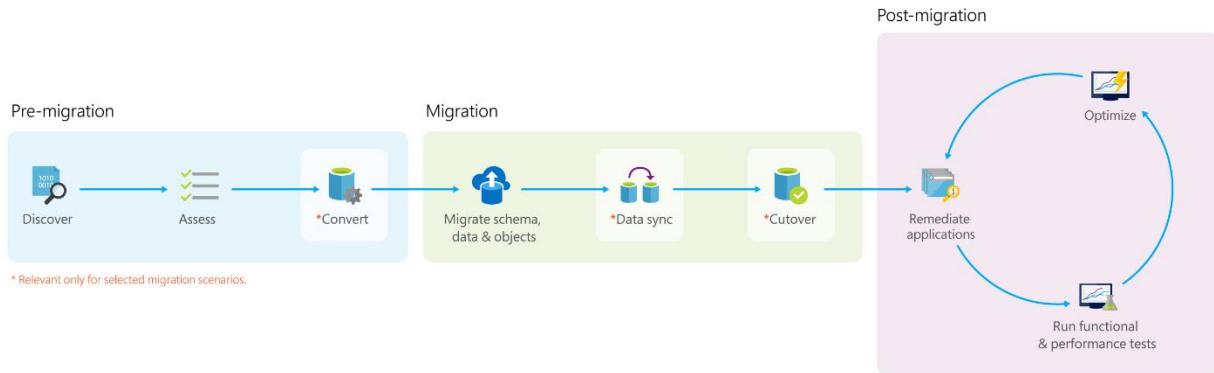
4.3 Migrating to Azure SQL

Tailspin Toys has spent some time with you learning more about the benefits of Azure SQL. Based on their research, they've decided to migrate the Tailspin Toys database to Azure as a PoC for migrating larger workloads.

The general process for [migrating assets to Azure](#) (whether it's apps, infrastructure, or data) is as follows:

- **Assess:** Evaluate an existing asset and establish a plan for migration of the asset.
- **Migrate:** Replicate the functionality of an asset in the cloud.
- **Optimize:** Balance the performance, cost, access, and operational capacity of a cloud asset.
- **Secure and manage:** Ensure a cloud asset is ready for ongoing operations.

In guiding Tailspin Toys through the *data* migration journey, it's helpful to leverage the following process that is specific to data:



Below you'll find a detailed survey of some of the resources and tips available for going through each phase of the migration journey to Azure (you can find a [full list of tools based on phase and target here](#)). The number of resources shows the dedication Microsoft and partners have to helping you move successfully to the cloud.

In an effort to simplify, we recommend using the [Data migration guide](#) which allows you to pick your source and target, and then guides you step-by-step through your specific migration journey.

Additionally, you can check out the [Azure migration center](#) which includes detailed information about the migration journey, and the [Azure Migrate Program \(AMP\)](#) where you can get support, migration partners, webinars, and more.

With your help, Tailspin Toys Gaming decided to apply for the Azure Migrate Program, to get support and guidance from Microsoft and partners. Then, they leveraged the integration of Azure Migrate, Data Migration Assistant, and Azure Database Migration Service to migrate their databases.

Once the Tailspin Toys Gaming division has migrated to Azure, the overall corporation may choose to scale the migration to Azure. In that case, [refer to guidance here](#).

Below, you'll find a reference chart that explains each step in the three-phase process following the diagram above. Each phase contains steps with descriptions and helpful resources. *This chart is not meant to be consumed in its entirety here, but feel free to follow what is interesting or relevant to your needs, or refer back to it later.*

STAGE	PHASE	DESCRIPTION	RESOURCES
Pre-migration	Discover	Inventory database assets, and application stack discovery.	<ul style="list-style-type: none"> • Azure Migrate for managing the whole migration process
	Assess	Assess workloads and fix recommendations.	<ul style="list-style-type: none"> • MAP Toolkit for inventory, assessment, and reporting • Data workload assessment model and tool • Database Experimentation Assistant (DEA) for upgrades • DMA SKU Recommender for Azure SQL
	Convert	Convert the source schema to work in the target environment. This is only relevant for heterogeneous (non-SQL Server) migrations.	<ul style="list-style-type: none"> • SQL Server Migration Assistant

STAGE	PHASE	DESCRIPTION	RESOURCES
Migration	Migrate schema, data, and objects	Migrate the source schema, and then migrate the source data to the target.	<ul style="list-style-type: none"> • Migrate with Azure Database Migration Services (DMS) (pull model) • Migrate to Azure with Data Migration Assistant (DMA) (push model) • Online migrations with DMS • Offline migrations with DMS • Utility to move on-premises SQL Server logins to Azure SQL MI • Bulk database creation with PowerShell • Bulk schema deployment with MSSQL-Scripter and PowerShell • Migrate SSIS packages to Azure • SSRS can be deployed using SQL Server on an Azure VM or rewritten in Power BI • Migrate SSAS to Azure Analysis Services

STAGE	PHASE	DESCRIPTION	RESOURCES
	Data Sync	Sync your target schema and data with the source. This is only relevant for minimal-downtime migrations.	<ul style="list-style-type: none"> • Azure Data Sync for syncing data bi-directionally on-prem and in Azure • Transactional replication for migrations by continuously publishing the changes
	Cutover	Cut over from the source to the target environment. This is only relevant for minimal-downtime migrations.	<ul style="list-style-type: none"> • Cutover with DMS
Post-migration	Remediate applications	Iteratively make any necessary changes to your applications.	<ul style="list-style-type: none"> • Application migration patterns and examples
	Perform tests	Iteratively run functional and performance tests.	<ul style="list-style-type: none"> • Data Quality Solution available from the partner QuerySurge
	Optimize	Based on the tests you performed, address any performance issues, and then retest to confirm the performance improvements.	<ul style="list-style-type: none"> • Post-migration Validation and Optimization Guide



4.4 Database discovery and assessment

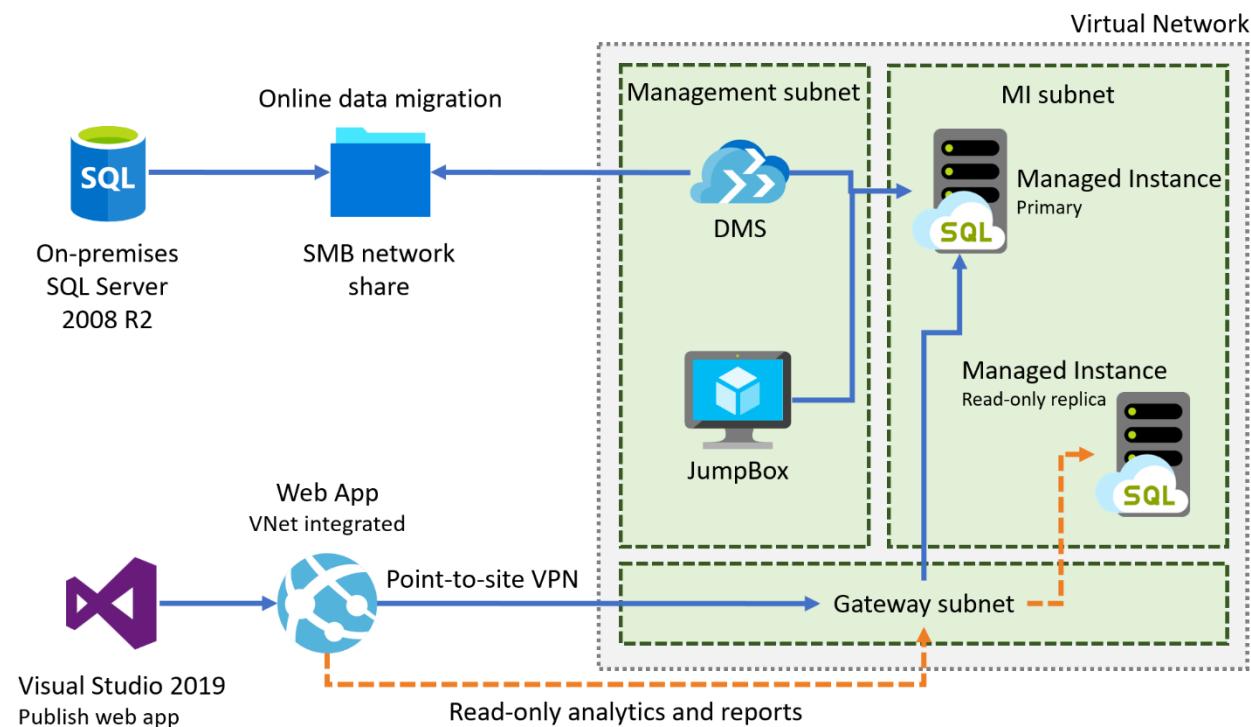
In this series of hands-on labs (throughout Modules 4 and 5), you will implement a proof-of-concept (PoC) for migrating an on-premises SQL Server 2008 R2 database into Azure SQL Database Managed Instance (SQL MI). You will perform assessments to reveal any feature parity and compatibility issues between the on-premises SQL Server 2008 R2 database and the managed database offerings in Azure. You will then migrate the customer's on-premises gamer information, web application and database into Azure, with minimal to no down-time. Finally, you will enable some of the advanced SQL features available in SQL MI to improve security and performance in the customer's application.

At the end of these hands-on labs, you will be better able to implement a cloud migration solution for business-critical applications and databases.

Note: These labs were modified from an existing day-long, hands-on-labs workshop to fit into this workshop. If you'd like to access the extended version of these labs refer to [MCW: Migrating SQL databases to Azure](#).

Solution architecture

Below is a diagram of the solution architecture that Tailspin Toys decided to go with, and what you will build in this lab. Please study this carefully, so you understand the whole of the solution as you are working on the various components.



Throughout the solution, you will use [Azure Migrate](#) as the central hub to track the discovery, assessment, and migration of Tailspin Toys. The solution begins with using the [Microsoft Data Migration Assistant](#) to perform assessments of feature parity and compatibility of the on-premises SQL Server 2008 R2 database against both Azure SQL Database (Azure SQL DB) and Azure SQL Database Managed Instance (SQL MI), with the goal of migrating the TailspinToys database into an Azure PaaS offering with minimal or no changes. After completing the assessments and reviewing the findings, the SQL Server 2008 R2 database is migrated into SQL MI using the Azure Database Migration Service's online data migration option. This allows the database to be migrated with little to no downtime, by using a backup and transaction logs stored in an SMB network share.

They'll also leverage their existing licenses to get [Azure Hybrid Benefits](#), and they'll prepay for [reserved capacity](#). This will help them save money from the start.

The web app is deployed to an Azure App Service Web App using Visual Studio 2019. Once the database has been migrated and cutover, the TailspinToysWeb application is configured to talk to the SQL MI VNet through a virtual network gateway using [point-to-site VPN](#), and its connection strings are updated to point to the new SQL MI database.

Note: Due to time constraints, the lab will deal with switching the app running on a Jumpbox VM locally from leveraging data in SQL Server 2008 to SQL MI (but not the deployment to Azure or integrating the App Service with the Virtual Network). In the [extended version of these labs](#), or if you have time at the end of the lab, you can do that.

Once in SQL MI, several features of Azure SQL Database are examined. [Advanced Data Security \(ADS\)](#) is enabled and [Data Discovery and Classification](#) is used to better understand the data and potential compliance issues with data in the database. The ADS [Vulnerability Assessment](#) is used to identify potential security vulnerabilities and issues in the database, and those finding are used to mitigate one finding by enabling [Transparent Data Encryption](#) in the database. [Dynamic Data Masking \(DDM\)](#) is used to prevent sensitive data from appearing when querying the database. Finally, [Read Scale-out](#) is used to point reports on the Tailspin Toys web app to a read-only secondary, allowing reporting, particularly for the Leaderboard statistics page, to occur without impacting the performance of the primary database.

Note:

If you are attending this lab as part of a day-long workshop, the activities below should be skipped (they will be demoed in class).

Activities

Note:

If you are attending this lab as part of a day-long workshop, the activities below should be skipped (they will be demoed in class), but they are available should you want to review them again.

In these activities, you'll set up Azure Migrate, and explore some of the new integrations between Microsoft's Data Migration Assistant (DMA) and Azure Database Migration Services (DMS). Before you begin the assessments, you need to restore a copy of the TailspinToys database in your SQL Server 2008 R2 instance. Finally, you will use the Microsoft Data Migration Assistant (DMA) to perform an assessment of the TailspinToys database against Azure SQL Database (Single database and Managed Instance). The assessment will provide a report about any feature parity and compatibility issues between the on-premises database and the various services.

To complete the activities for this module, proceed directly to the Activities in **Lab:**

Database Discovery and Assessment for Migrating to Azure for the hands-on exercises for Tailspin Toys Gaming. When are you done with these activities, come back here and continue to Module 5.



For Further Study

- [Azure Database Migration Guide](#) contains lots of resources that will help in guiding and supporting database migrations to Azure.
- [Azure Migrate Documentation](#) contains more information, guidance, and pointers on how to migrate your entire on-premises estate to Azure.
- [Data Migration Assistant Documentation](#) contains more information and best practices around the DMA tool explored in this module.
- [Choosing your database migration path to Azure](#) is a white paper created by Microsoft for deeper understanding of how to modernize and migrate on-premises SQL Server to Azure.
- [In one location, ready to use or review slides](#) about Azure SQL are hosted online, including a few design sessions that could be redelivered. Feel free to use these resources with your customers or at events.



Next Steps

Next, Continue to [05 - Migrating to Azure SQL](#).



Activity 1: Set up Azure Migrate

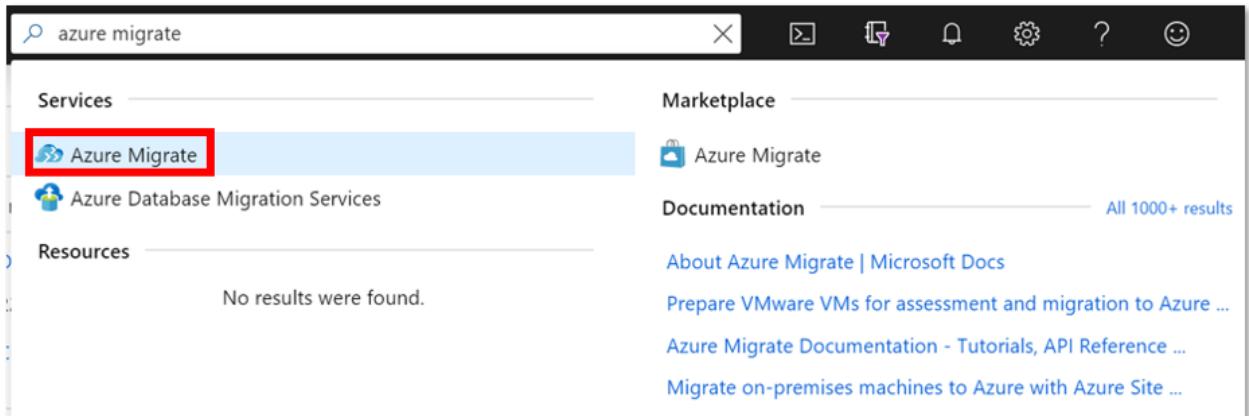
Note:

If you are attending this lab as part of a day-long workshop, you should skip this activity, it was demoed earlier. If you have time at the end of the day, feel free to return to it.

In this activity, you'll set up Azure Migrate, and explore some of the new integrations between Microsoft's Data Migration Assistant (DMA) and Azure Database Migration Services (DMS).

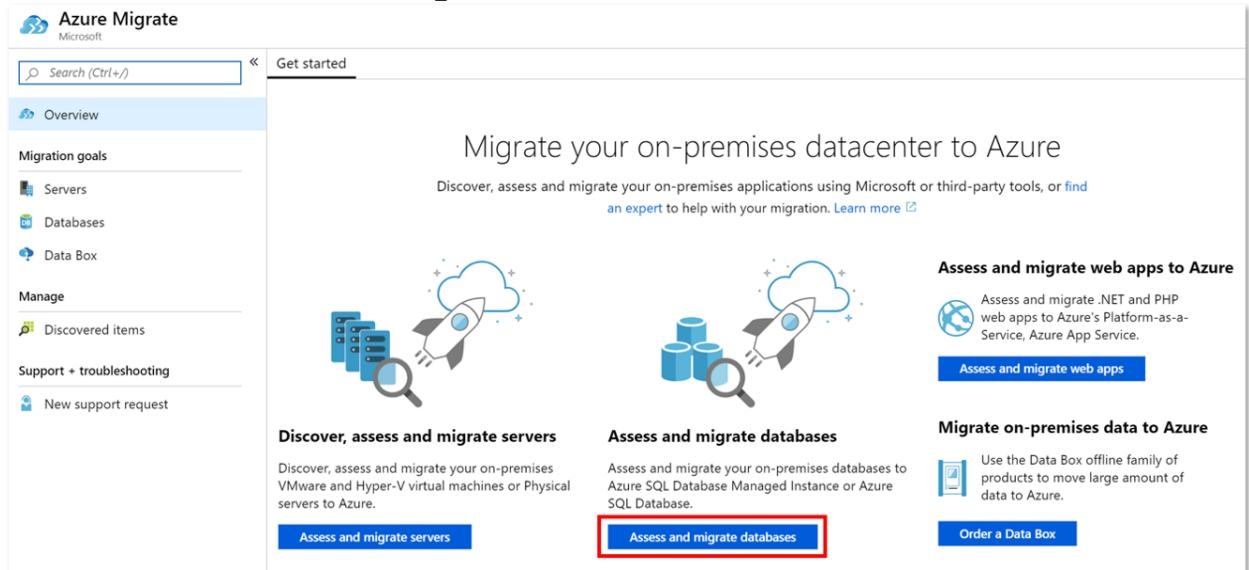
✓ Steps

1. Log in to the Azure portal (with the account you're using for this workshop), and search for **Azure Migrate**:



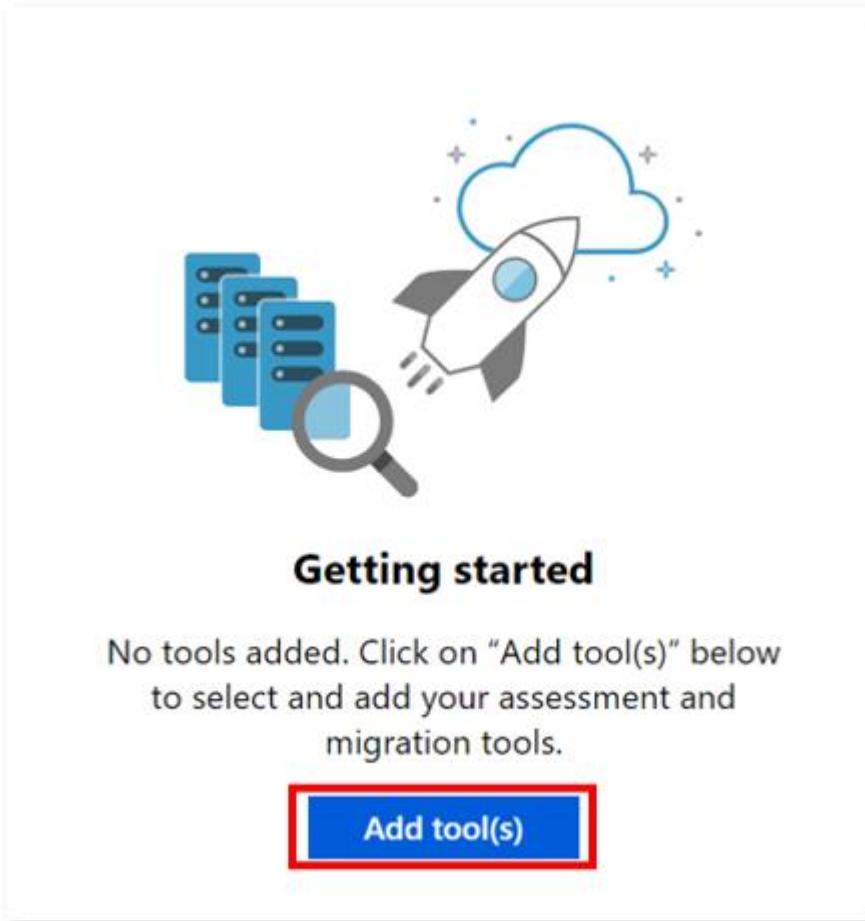
The screenshot shows the Azure portal search results for "azure migrate". The "Azure Migrate" service card is highlighted with a red box. The search bar at the top contains "azure migrate". Below the search bar, there are two main sections: "Services" and "Marketplace". In the "Services" section, "Azure Migrate" is the first item, followed by "Azure Database Migration Services". In the "Marketplace" section, "Azure Migrate" is listed under "Documentation" with a link to "About Azure Migrate | Microsoft Docs". There are also links to "Prepare VMware VMs for assessment and migration to Azure ...", "Azure Migrate Documentation - Tutorials, API Reference ...", and "Migrate on-premises machines to Azure with Azure Site ...".

2. As you can see, Azure Migrate can be used to migrate more than just databases. For now, select **Assess and migrate databases**:



The screenshot shows the Azure Migrate service overview page. On the left, there is a navigation sidebar with options like "Overview", "Migration goals", "Manage", "Support + troubleshooting", and "New support request". The "Overview" tab is selected. In the center, there is a "Get started" section with the heading "Migrate your on-premises datacenter to Azure". It includes a description: "Discover, assess and migrate your on-premises applications using Microsoft or third-party tools, or find an expert to help with your migration. Learn more". Below this, there are four main migration paths: "Discover, assess and migrate servers", "Assess and migrate databases", "Assess and migrate web apps to Azure", and "Migrate on-premises data to Azure". Each path has a corresponding icon and a "Get started" button. The "Assess and migrate databases" button is highlighted with a red box.

3. Select **Add tool(s)** under the *Assessment tools* section.



4. In *Migrate project* area select the subscription and resource group you're using for the workshops. Then, supply a Migration project name of **MigrateTailspinToys** and region. Then

select **Next**.

Add a tool

Migrate project [Select assessment tool](#) [Select migration tool](#) [Review + add tool\(s\)](#)

An Azure Migrate project is used to store the discovery, assessment and migration metadata reported by your on-premises environment. Select a subscription and resource group in your preferred geography to create the migrate project.

* Subscription 

 * Resource group 
 hands-on-lab-anthro [Create new](#)

PROJECT DETAILS

Specify the name of the migrate project and the preferred geography.

* Migrate project  ✓

* Geography 
 United States

Next 

Note:

for region, select the one closest to you. The geography specified for the project is only used to store the metadata gathered from on-premises VMs. You can select any target region for the actual migration.

5. For assessment tools, select **Azure Migrate: Database Assessment**, then select **Next**.
6. For migration tools, select **Azure Migrate: Database Migration**, and then select **Next**.
7. Finally, select **Add tool(s)**. Azure Migrate is now set up for you to use. In the next activity, you'll use some of the tools you added to assess and migrate Tailspin Toys to Azure.



Activity 2: Restore TailspinToys on the SQLServer2008 VM

Note:

If you are attending this lab as part of a day-long workshop, you should skip this activity, it was demoed earlier. If you have time at the end of the day, feel free to return to it.

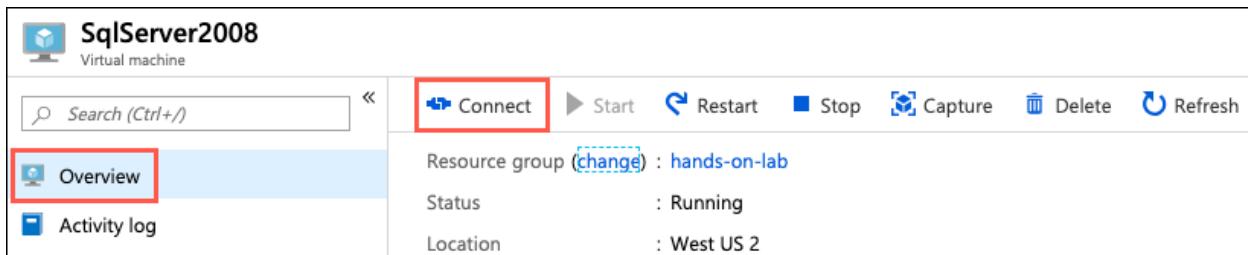
Before you begin the assessments, you need to restore a copy of the TailspinToys database in your SQL Server 2008 R2 instance. In this task, you will create an RDP connection to the SqlServer2008 VM and then restore the TailspinToys database onto the SQL Server 2008 R2 instance using a backup provided by Tailspin Toys.

✓ Steps

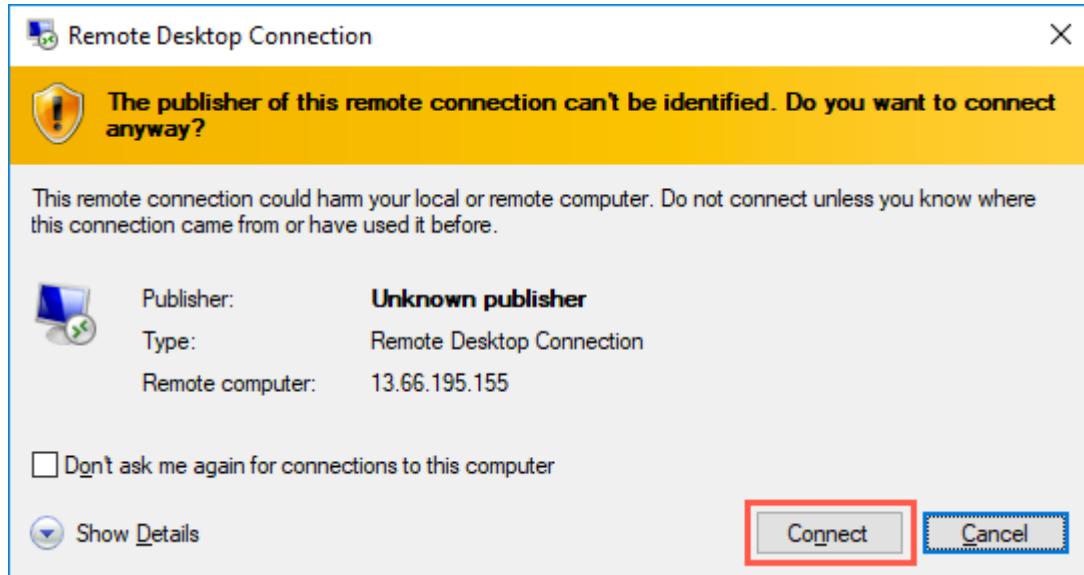
1. In the [Azure portal](#), navigate to your **SqlServer2008** VM by selecting **Resource groups** from the left-hand navigation menu, selecting the **hands-on-lab-SUFFIX** resource group, and selecting the **SqlServer2008** VM from the list of resources. On the SqlServer2008 Virtual Machine's *Overview* blade, select **Connect** on the top menu.

Note:

If you were provided with an environment for these labs, the "SUFFIX" part of the **hands-on-lab-SUFFIX** will be used in several portions of the lab. You should note this value from the credentials provided. Any time "SUFFIX" is referenced, use this number. **Do not use other "SUFFIX" values or you will disrupt another attendees' environment.**



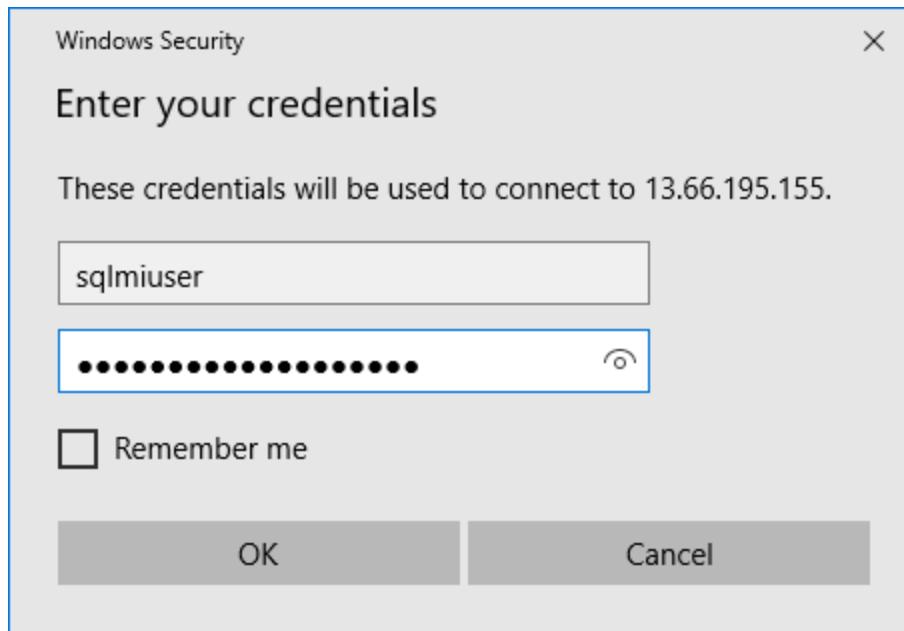
2. On the *Connect to virtual machine* blade, select **Download RDP File**, then open the downloaded RDP file.
3. Select **Connect** on the *Remote Desktop Connection* dialog.



4. Enter the following credentials when prompted, and then select **OK**:

- o *Username:* **sqlmiuser**

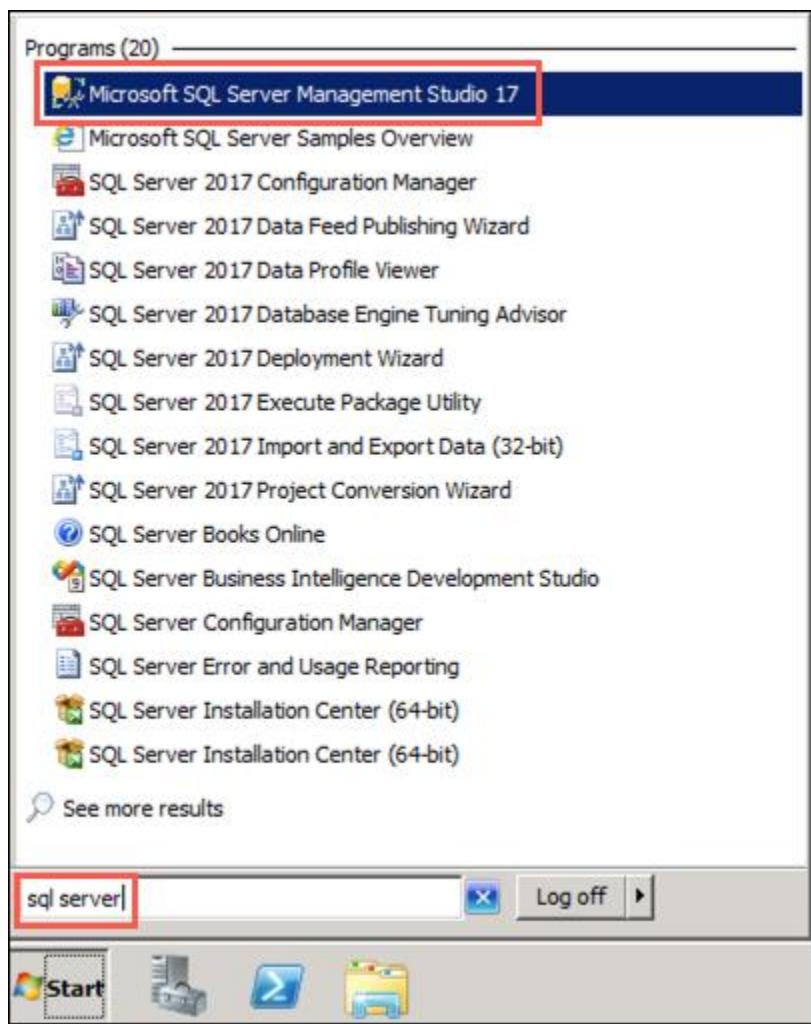
Note: Password should be consistent among all labs, ask your instructor for the password in an in-person Lab.



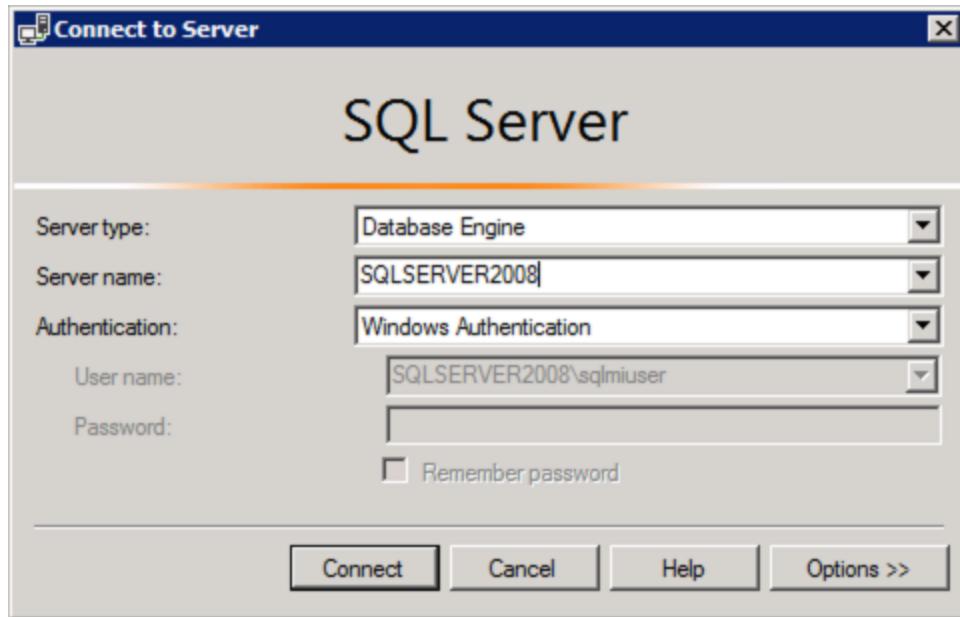
5. Select **Yes** to connect, if prompted that the identity of the remote computer cannot be verified.



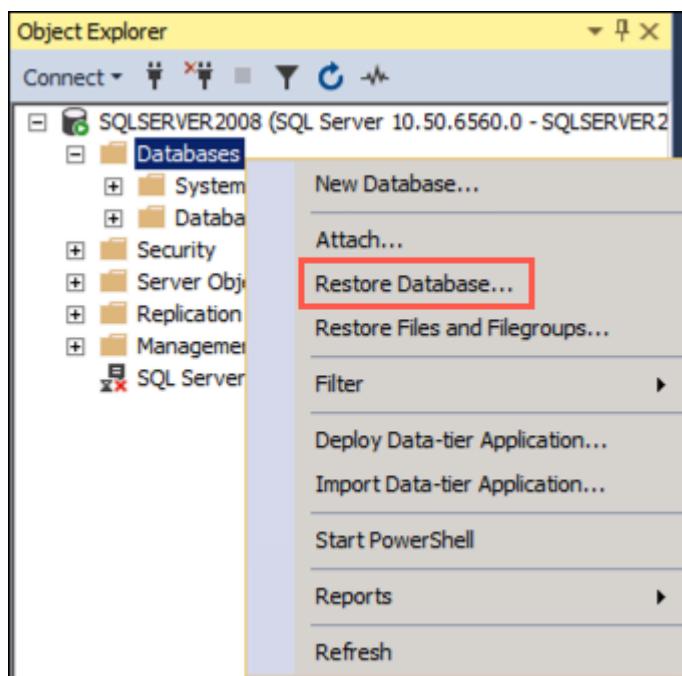
6. Once logged into the SqlServer2008 VM, download a [backup of the TailspinToys database](#), and save it to the c:\ drive of the VM.
7. Next, open *Microsoft SQL Server Management Studio 17* by entering **sql server** into the search bar in the Windows Start menu.



8. In the SSMS Connect to Server dialog, enter **SQLSERVER2008** into the *Server name* box, ensure **Windows Authentication** is selected, and then select **Connect**.



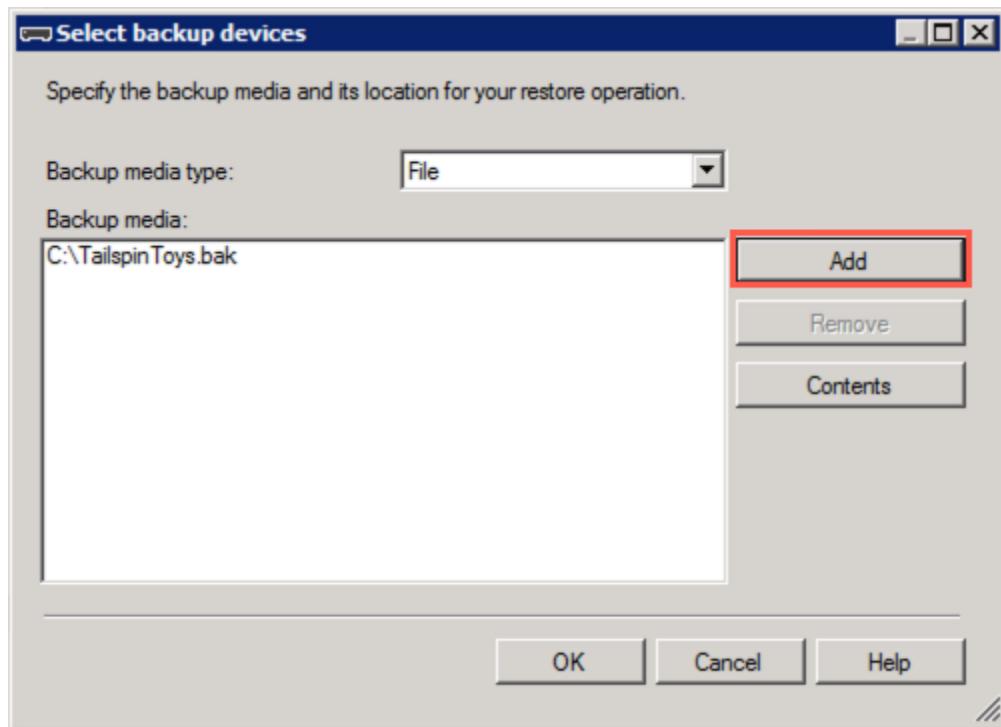
- Once connected, right-click *Databases* under *SQLSERVER2008* in the *Object Explorer*, and then select **Restore Database** from the context menu.



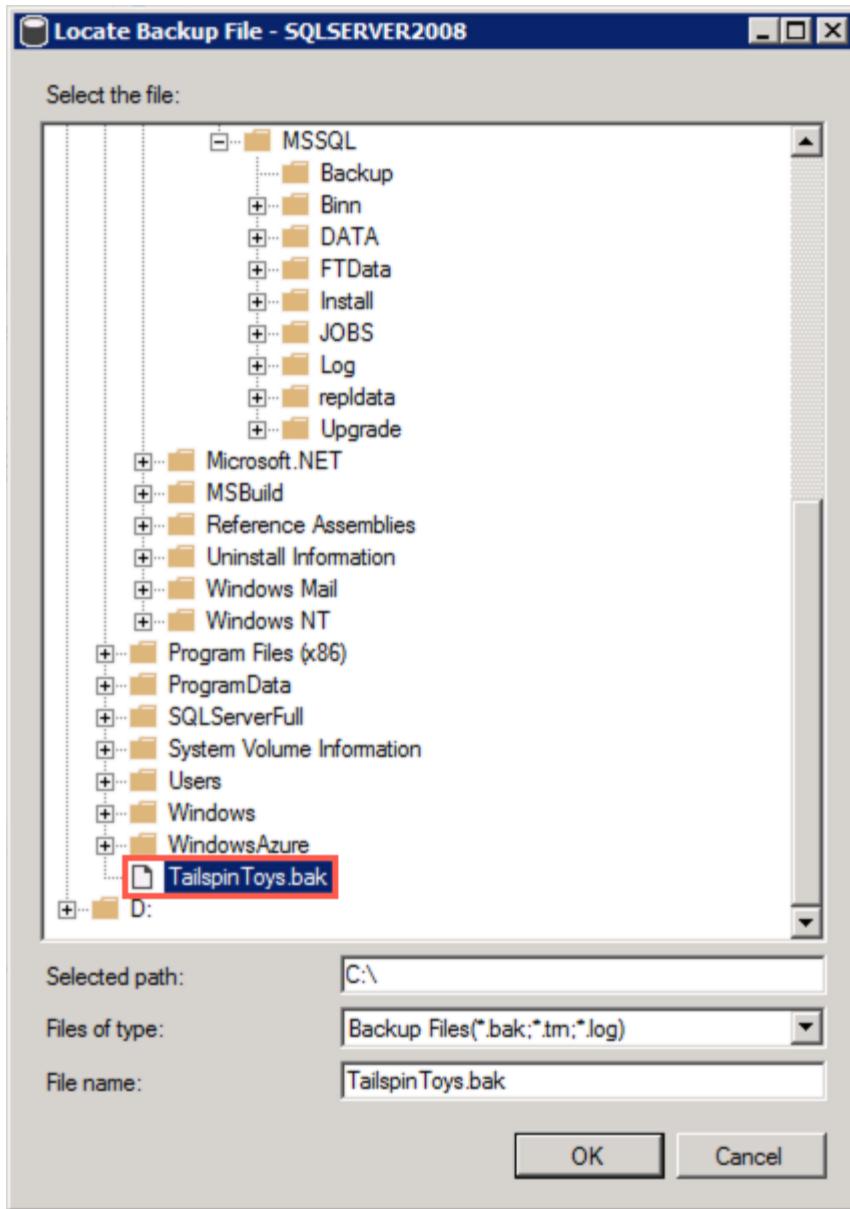
- You will now restore the *TailspinToys* database using the downloaded *TailspinToys.bak* file. On the *General* page of the *Restore Database* dialog, select **Device** under *Source*, and then select the **Browse (...)** button to the right of the *Device* box.



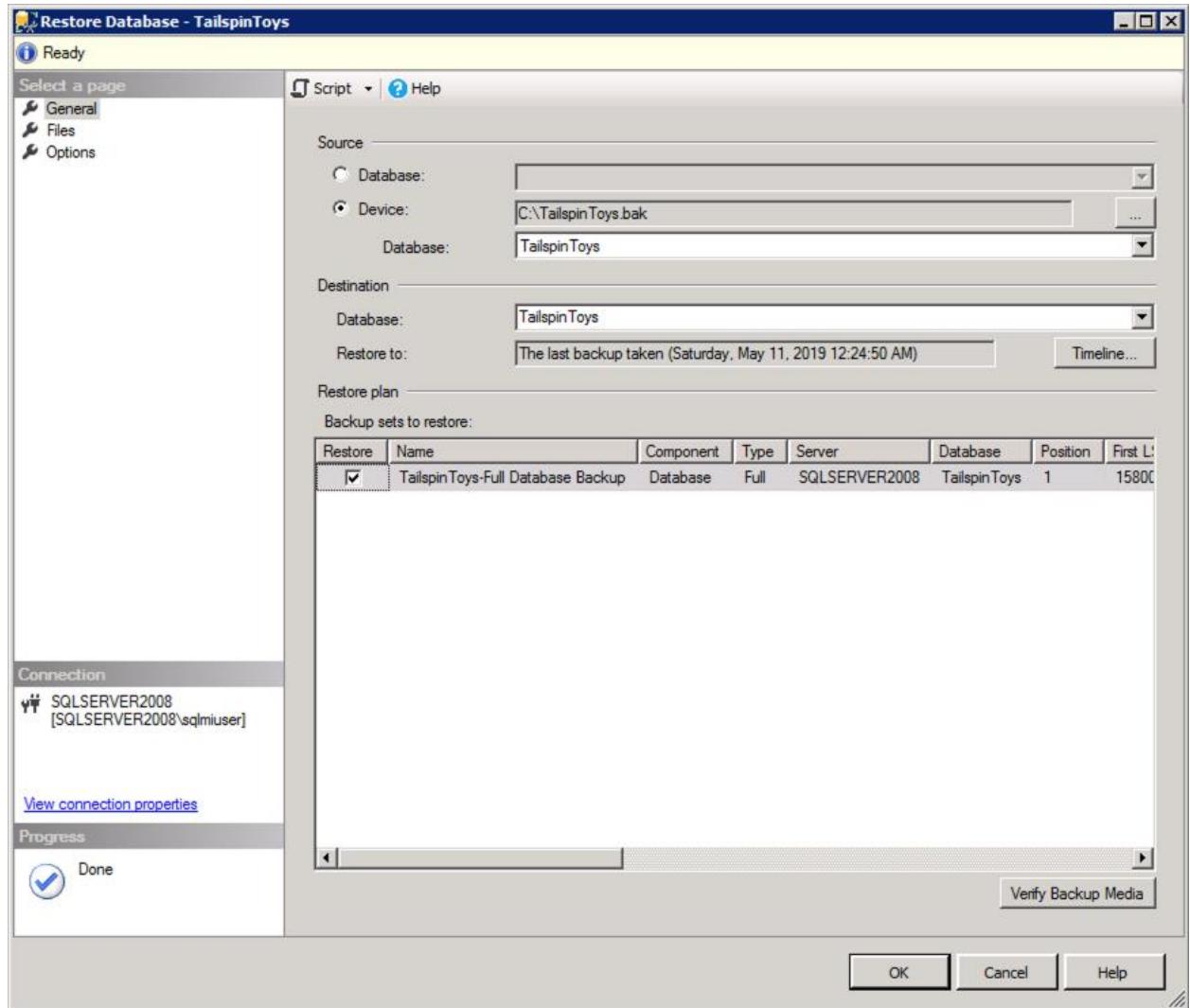
11. In the *Select backup devices* dialog that appears, select **Add**.



12. In the *Locate Backup File* dialog, browse to the location you saved the downloaded TailspinToys.bak file, **select that file**, and then select **OK**.

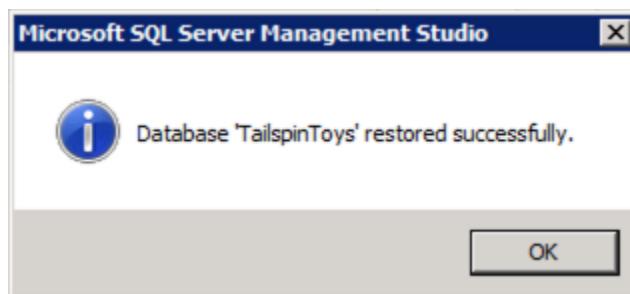


13. Select **OK** on the *Select backup devices* dialog. This will return you to the *Restore Database* dialog. The dialog will now contain the information required to restore the TailspinToys database.



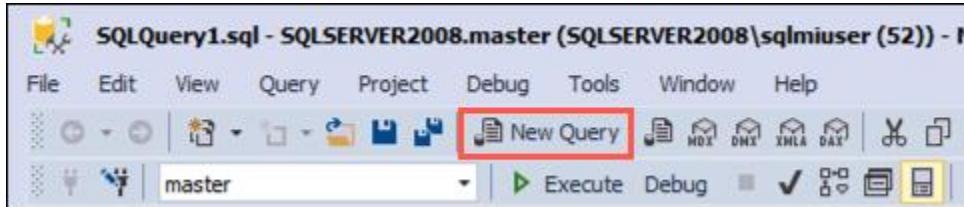
14. Select **OK** to start the restore.

15. Select **OK** in the dialog when the database restore is complete.



16. Next, you will execute a script in SSMS, which will reset the sa password, enable mixed mode authentication, enable Service broker, create the WorkshopUser account, and change the database recovery model to FULL.

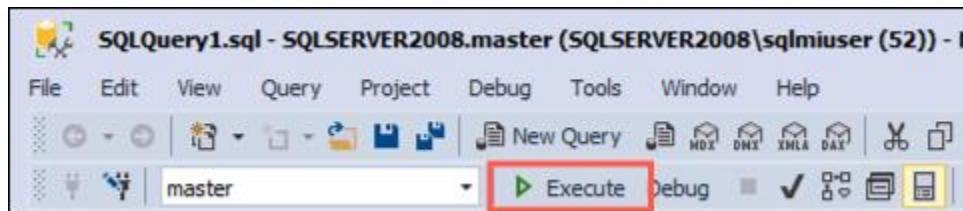
To create the script, open a new query window in SSMS by selecting **New Query** in the *SSMS toolbar*.



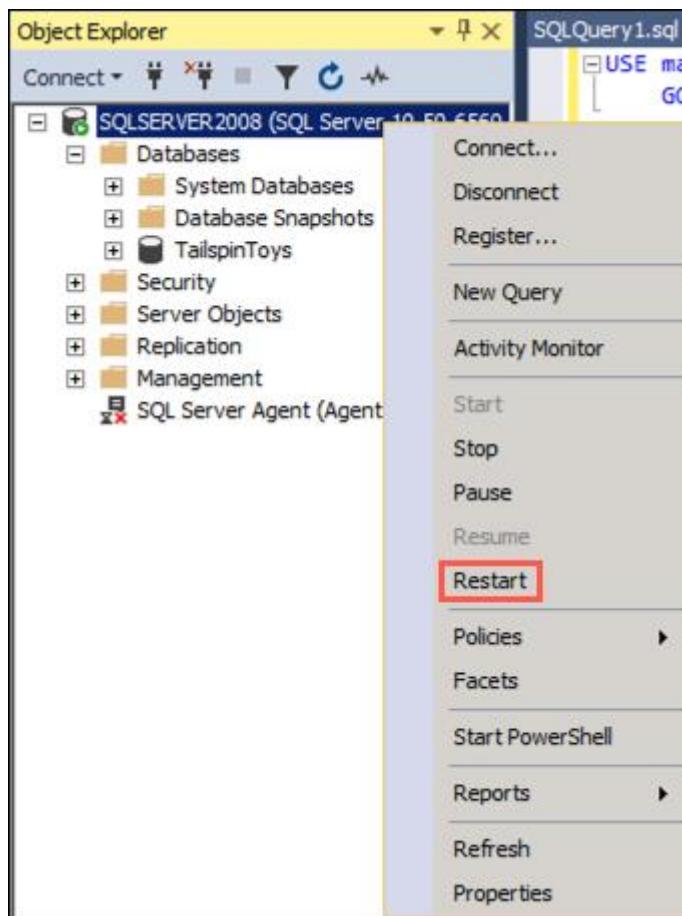
17. Copy and paste the SQL script below into the new query window (replacing <YourPasswordHere> with the same password as the SQL Server 2008 VM):

```
18. USE master;
19. GO
20.
21. -- SET the sa password
22. ALTER LOGIN [sa] WITH PASSWORD=N'<YourPasswordHere>';
23. GO
24.
25. -- Enable Service Broker on the database
26. ALTER DATABASE TailspinToys SET ENABLE_BROKER WITH ROLLBACK immediate;
27. GO
28.
29. -- Enable Mixed Mode Authentication
30. EXEC xp_instance_regwrite N'HKEY_LOCAL_MACHINE',
31. N'Software\Microsoft\MSSQLServer\MSSQLServer', N'LoginMode', REG_DWORD, 2;
32. GO
33.
34. -- Create a login and user named WorkshopUser
35. CREATE LOGIN WorkshopUser WITH PASSWORD = N'<YourPasswordHere>';
36. GO
37.
38. EXEC sp_addsrvrolemember
39.     @loginame = N'WorkshopUser',
40.     @rolename = N'sysadmin';
41. GO
42.
43. USE TailspinToys;
44. GO
45.
46. IF NOT EXISTS (SELECT * FROM sys.database_principals WHERE name =
    N'WorkshopUser')
47. BEGIN
48.     CREATE USER [WorkshopUser] FOR LOGIN [WorkshopUser]
49.     EXEC sp_addrolemember N'db_datareader', N'WorkshopUser'
50. END;
51. GO
52.
53. -- Update the recovery model of the database to FULL
54. ALTER DATABASE TailspinToys SET RECOVERY FULL;
GO
```

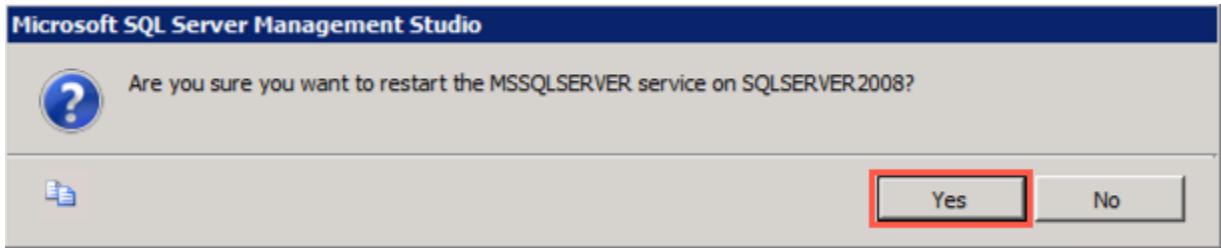
55. To run the script, select **Execute** from the *SSMS toolbar*.



56. For Mixed Mode Authentication and the new sa password to take effect, you must restart the *SQL Server (MSSQLSERVER)* Service on the SqlServer2008 VM. To do this, you can use SSMS. Right-click the *SQLSERVER2008* instance in the *SSMS Object Explorer*, and then select **Restart** from the context menu.



57. When prompted about restarting the *MSSQLSERVER* service, select **Yes**. The service will take a few seconds to restart.



Activity 3: Perform assessment for migration to Azure SQL Database

Note:

If you are attending this lab as part of a day-long workshop and were provided an environment to use, you should skip this activity, it was demoed earlier. If you have time at the end of the day, feel free to return to it.

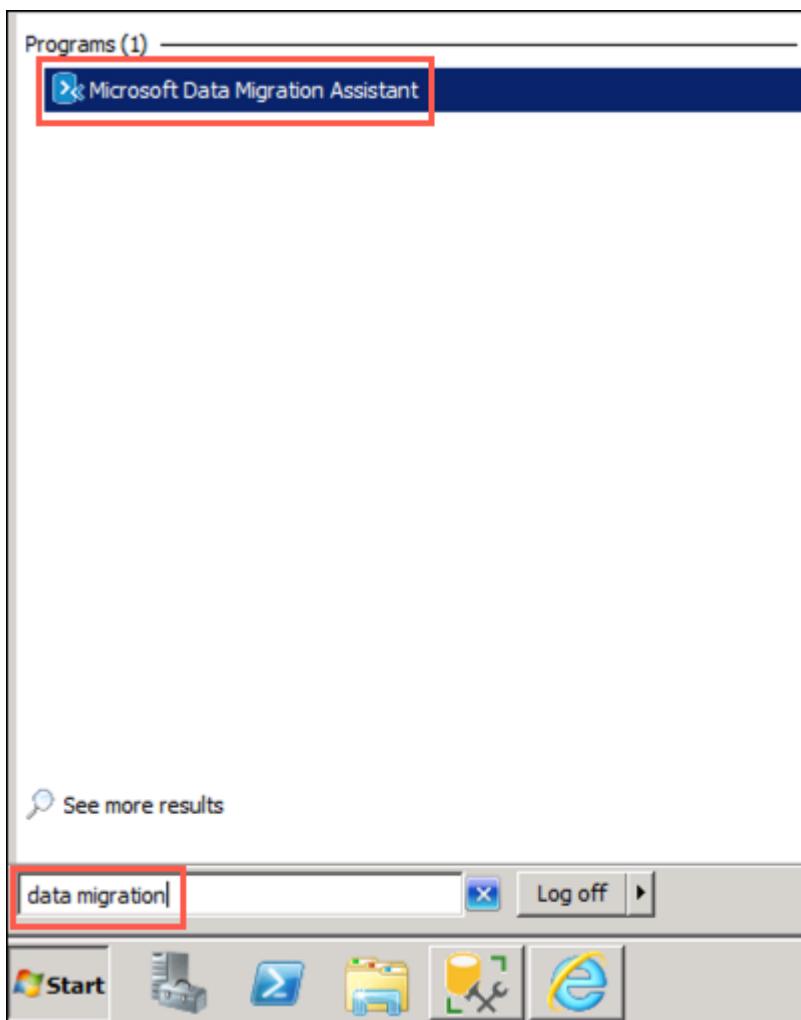
In this task, you will use the *Microsoft Data Migration Assistant* (DMA) to perform an assessment of the TailspinToys database against Azure SQL Database (Azure SQL DB). The assessment will provide a report about any feature parity and compatibility issues between the on-premises database and the Azure SQL DB service.

Note:

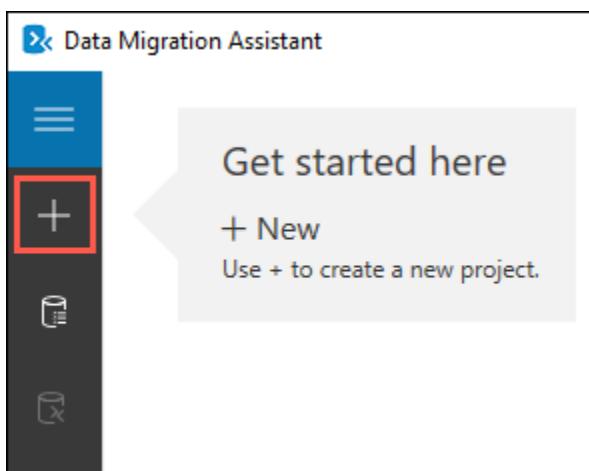
Tailspin Toys Gaming has already tentatively decided that they want to move to Managed Instance. However, this is a good exercise if they didn't know that Service Broker wasn't supported by Azure SQL DB and wanted to move to Azure SQL DB.

Steps

1. On the SqlServer2008 VM, launch DMA from the *Windows Start menu* by typing **data migration** into the *search bar*, and then selecting **Microsoft Data Migration Assistant** in the search results.

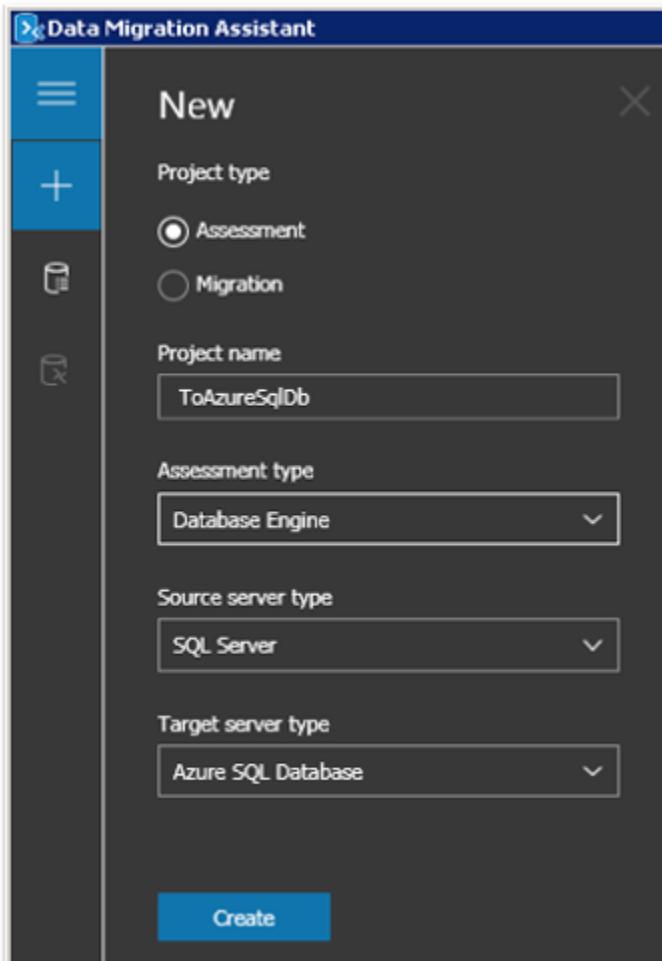


2. In the *DMA dialog*, select + from the left-hand menu to create a new project.



3. In the *New project* pane, set the following:

- *Project type:* Select **Assessment**
- *Project name:* Enter **ToAzureSqlDb**
- *Assessment type:* Select **Database Engine**
- *Source server type:* Select **SQL Server**
- *Target server type:* Select **Azure SQL Database**

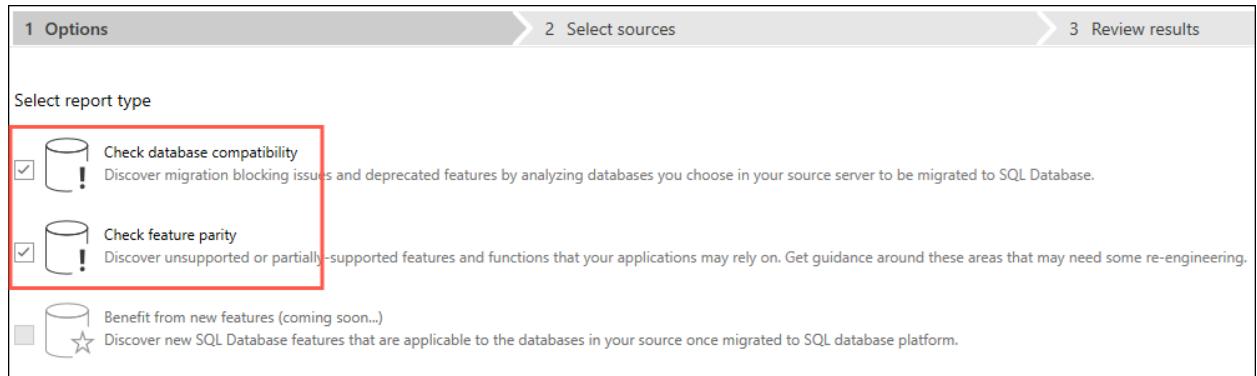


Note:

There's also an option to select "Migration" within DMA. You could use this option if you have restrictions around pushing versus pulling the data to Azure. Using DMA to migrate would be pushing data from the SQL Server 2008 VM into Azure. In this lab, we'll instead pull data from the SQL Server 2008 VM. You can learn more about migrating to Azure using DMA [here](#).

4. Select **Create**

5. On the *Options* screen, ensure **Check database compatibility** and **Check feature parity** are both checked, and then select **Next**:

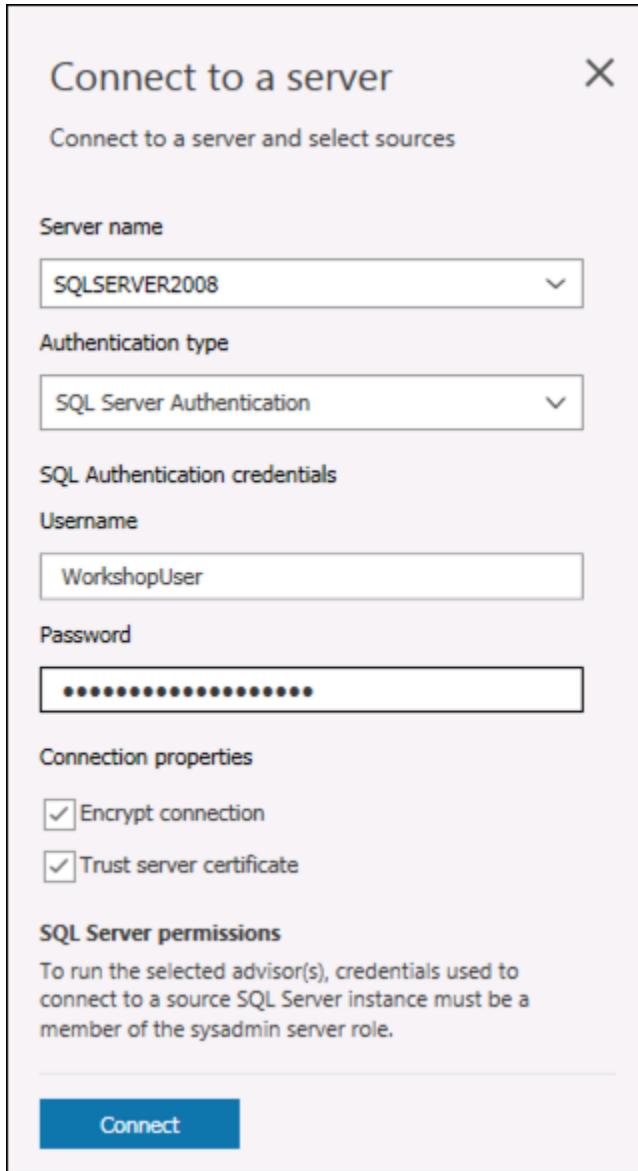


6. On the *Sources* screen, enter the following into the *Connect to a server* dialog that appears on the right-hand side:

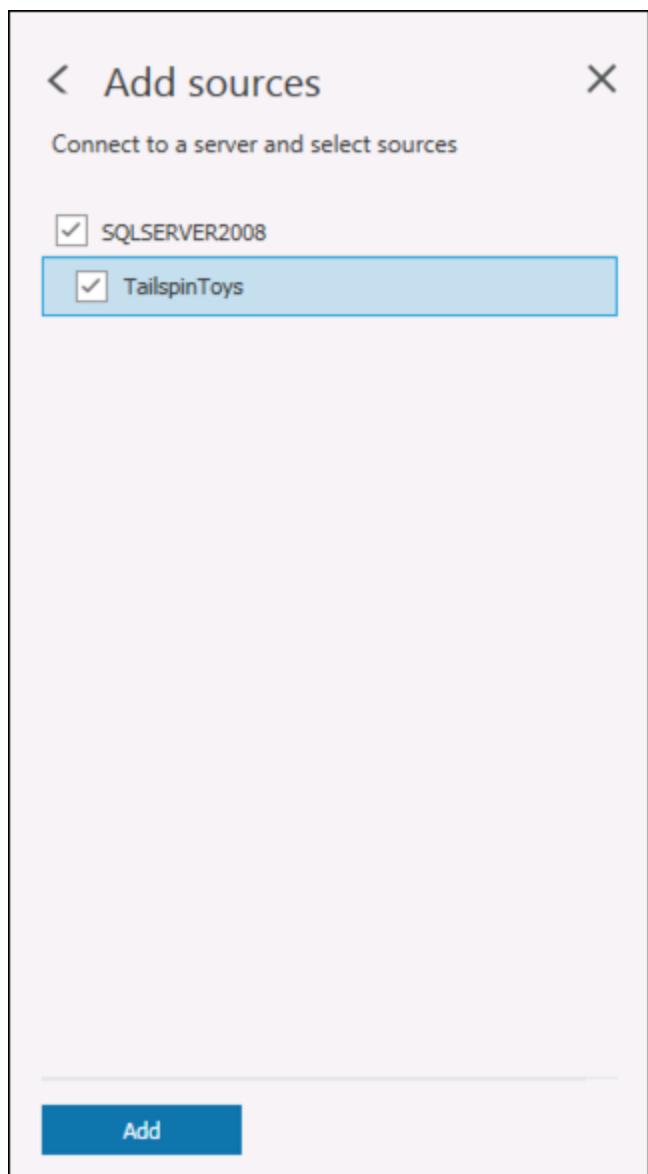
- **Server name:** Enter **SQLSERVER2008**
- **Authentication type:** Select **SQL Server Authentication**
- **Username:** Enter **WorkshopUser**
- **Password:** Enter your password
- **Encrypt connection:** Check this box
- **Trust server certificate:** Check this box

Note:

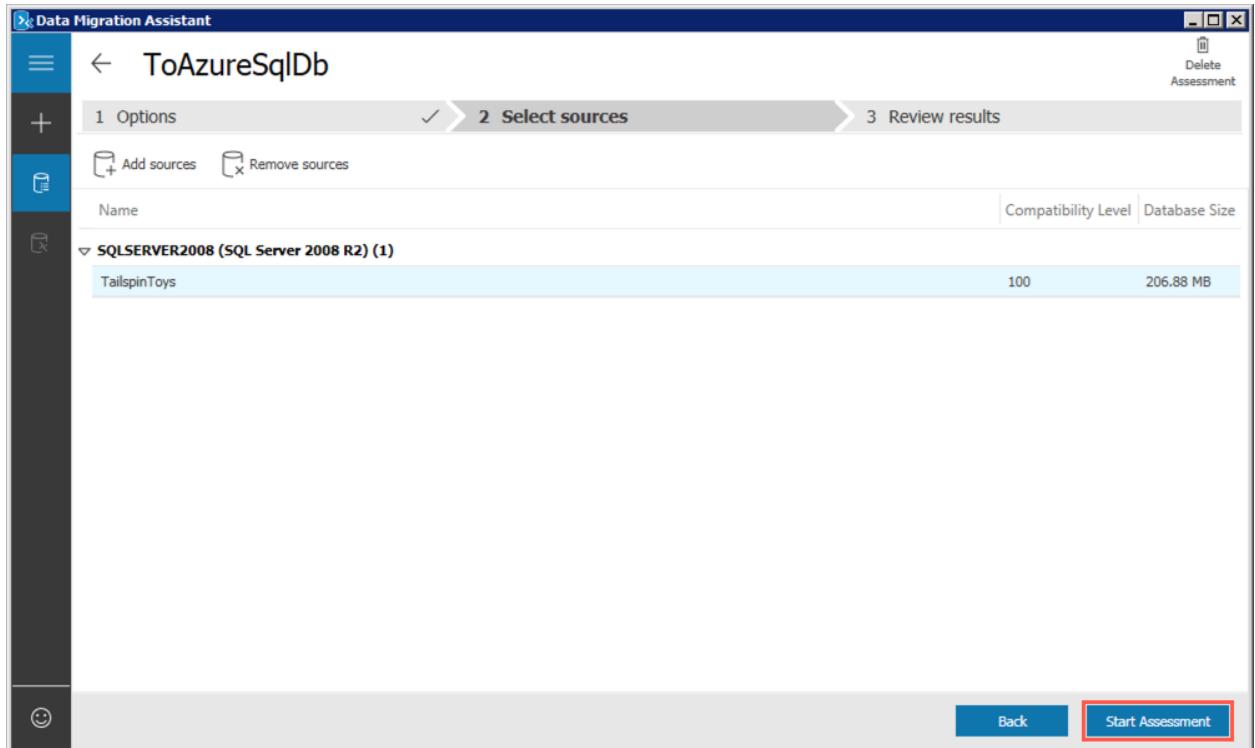
Password should be consistent among all labs, ask your instructor if you don't know what your password is.



7. Select **Connect**
8. On the *Add sources* dialog that appears next, check the box for **TailspinToys** and select **Add**:



9. Select **Start Assessment**:



10. Review the assessment of ability to migrate to Azure SQL DB.

Service Broker feature is not supported in Azure SQL Database

Object details:
Type: Database
Name: TailspinToys

This database has Service Broker enabled.

Upload to Azure Migrate

The DMA assessment for a migrating the TailspinToys database to a target platform of Azure SQL DB shows two features in use which are not supported. These features, cross-database references and Service Broker, will prevent

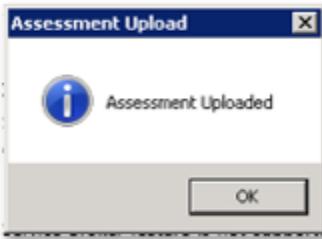
TailspinToys from being able to migrate to the Azure SQL DB PaaS offering without first making changes to their database.

11. In the bottom right (see above), select **Upload to Azure Migrate**. You'll be prompted to sign in (use the credentials you're using for this workshop). Then, select the subscription and **Azure Migrate Project** created earlier. This process will upload the summarized report to the Azure Migrate service. If you have multiple servers and/or databases, you'll be provided with a consolidated view of all the databases you scan and upload in the Azure Migrate portal.

Select **Upload**:



12. When it completes, you'll see the following message:



Activity 4: Perform assessment for migration to Azure SQL Database Managed Instance

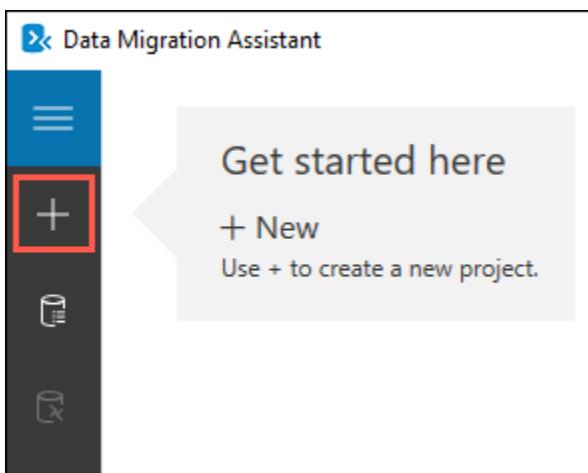
Note:

If you are attending this lab as part of a day-long workshop, you should skip this activity, it was demoed earlier. If you have time at the end of the day, feel free to return to it.

With one PaaS offering ruled out due to feature parity, you will now perform a second assessment. In this task, you will use DMA to perform an assessment of the TailspinToys database against Azure SQL Database Managed Instance (SQL MI). The assessment will provide a report about any feature parity and compatibility issues between the on-premises database and the SQL MI service.

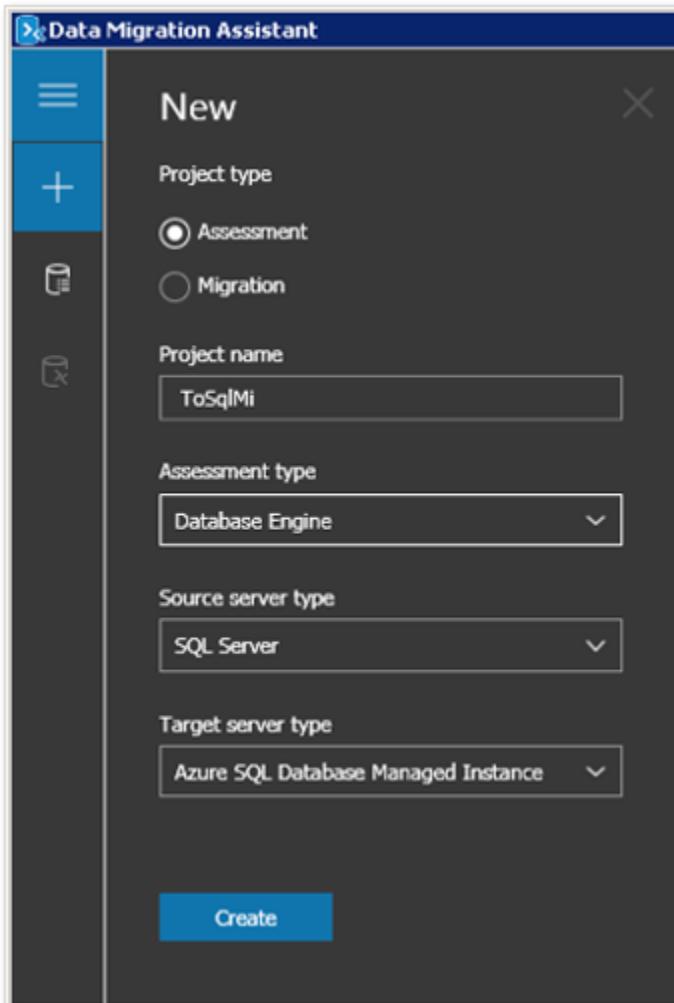
Steps

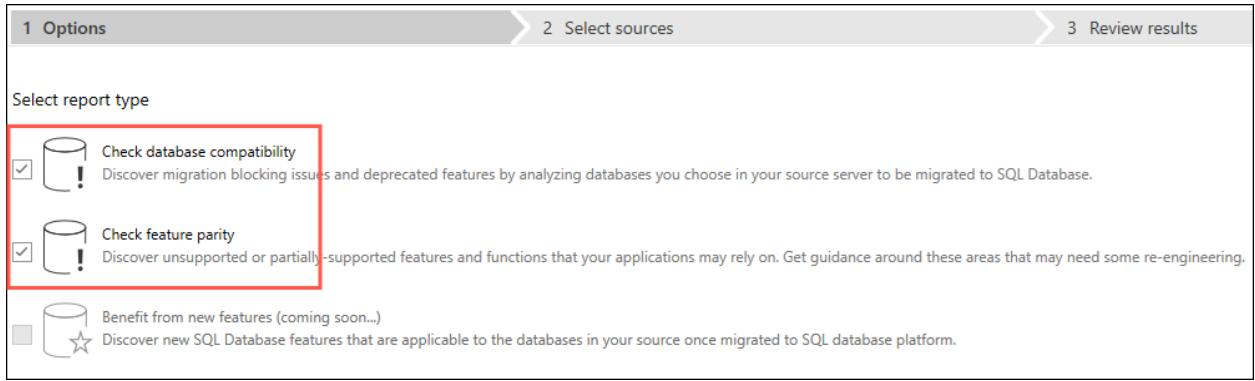
1. To get started, select + on the left-hand menu in DMA to create another new project:.



2. In the *New project* pane, set the following:

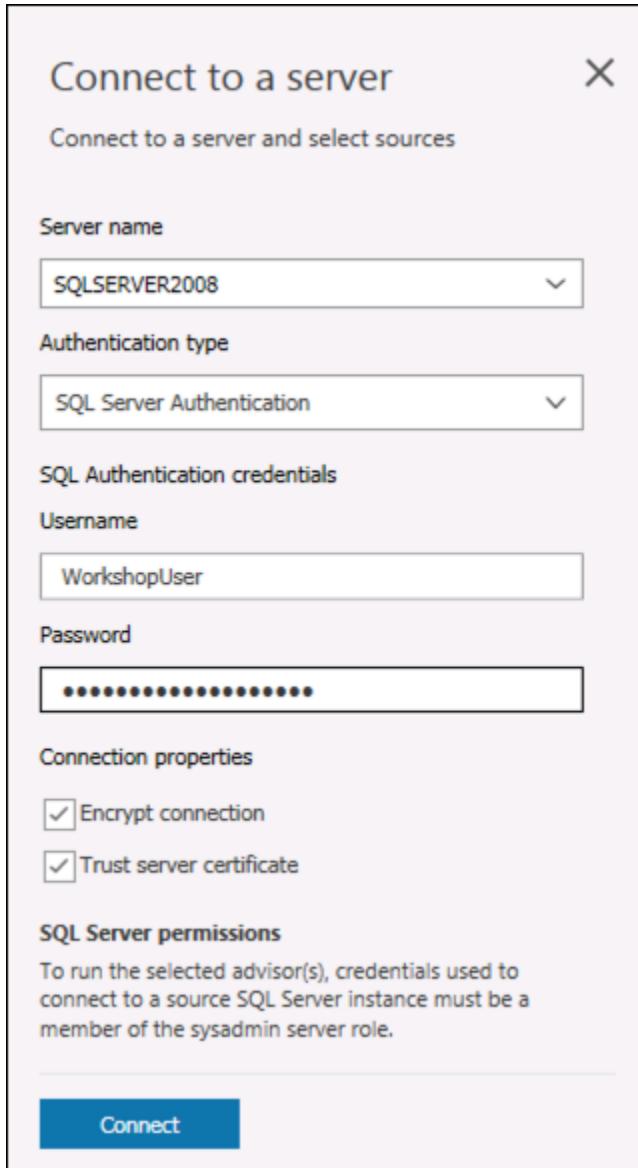
- *Project type:* Select **Assessment**
- *Project name:* Enter **ToSqlMi**
- *Assessment type:* Select **Database Engine**
- *Source server type:* Select **SQL Server**
- *Target server type:* Select **Azure SQL Database Managed Instance**



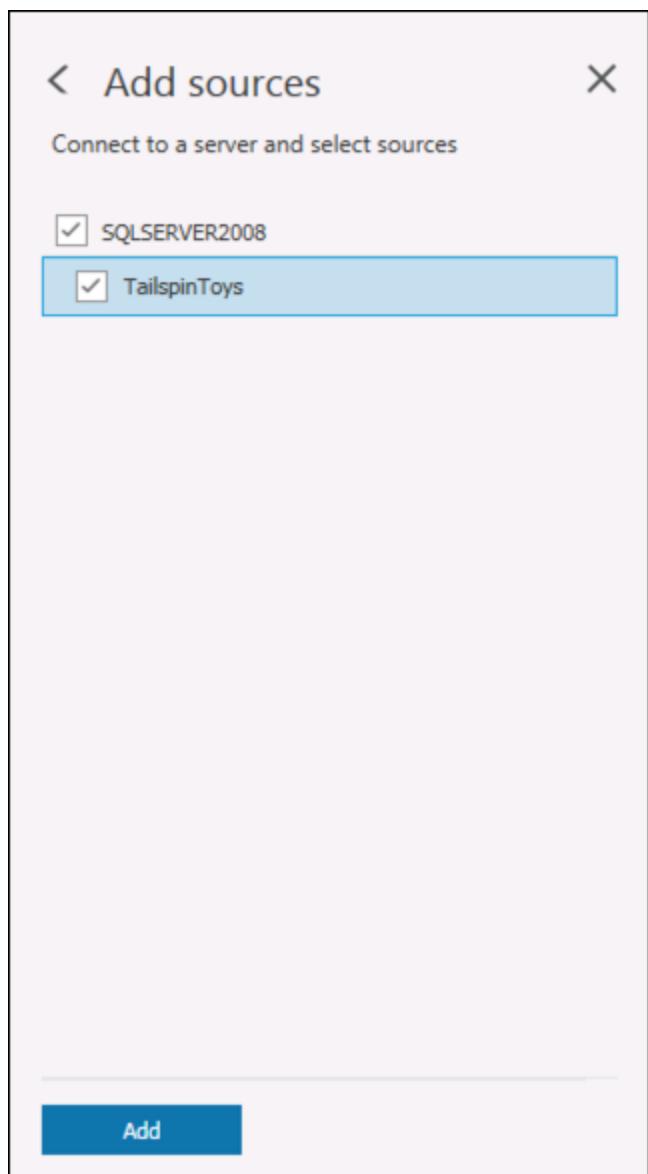


5. On the *Sources* screen, enter the following into the *Connect to a server* dialog that appears on the right-hand side:
 - *Server name*: Enter **SQLSERVER2008**
 - *Authentication type*: Select **SQL Server Authentication**
 - *Username*: Enter **WorkshopUser**
 - *Password*: Enter your password
 - *Encrypt connection*: Check this box
 - *Trust server certificate*: Check this box

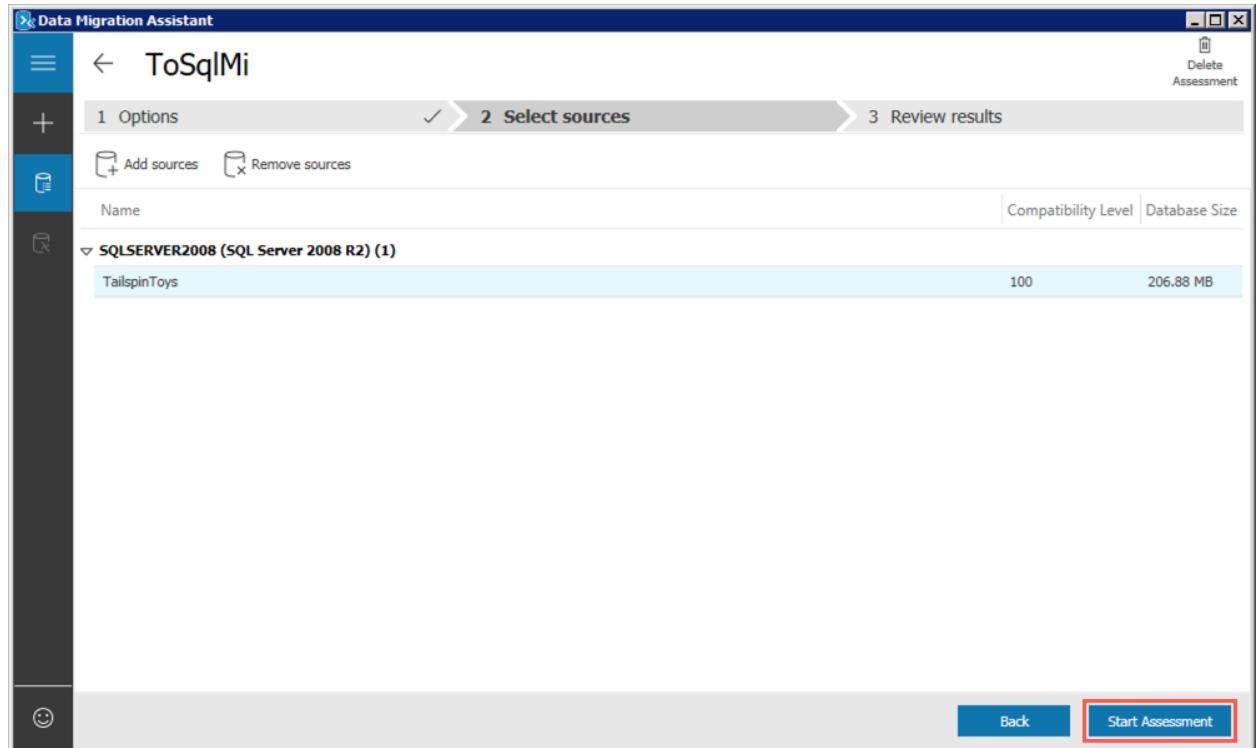
Note: Password should be consistent among all labs, ask your instructor if you don't know what your password is.



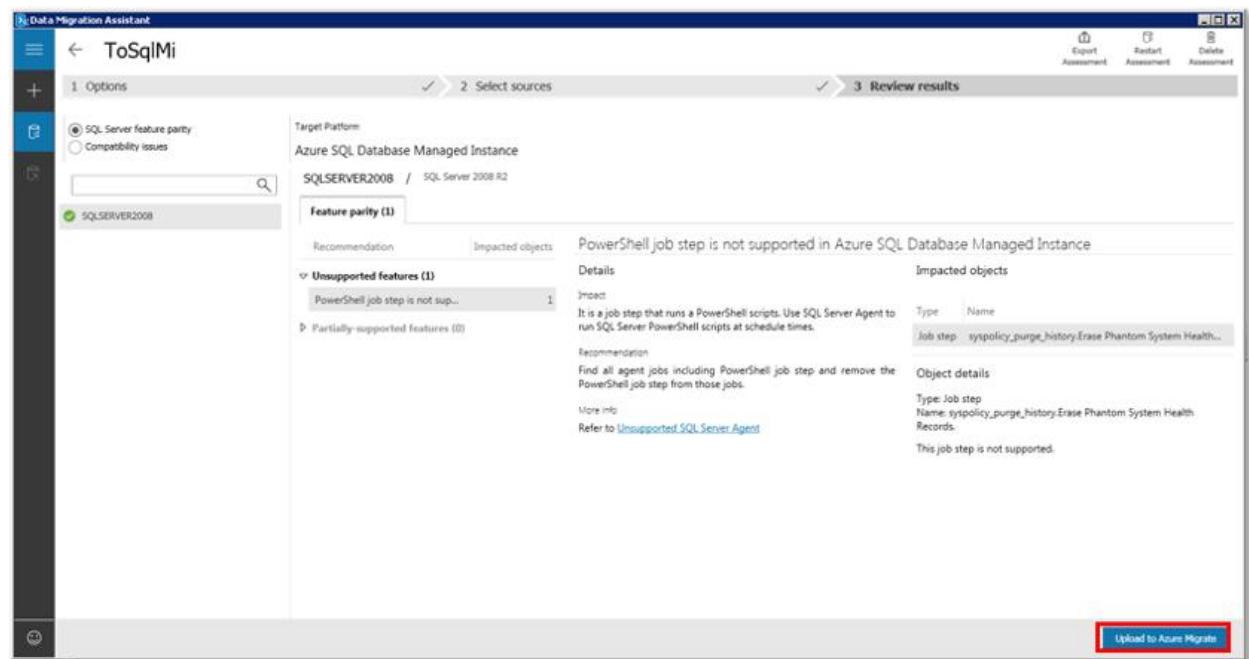
6. Select **Connect**
7. On the *Add sources* dialog that appears next, check the box for *TailspinToys* and select **Add**:



8. Select **Start Assessment**:



9. Review the assessment of ability to migrate to Azure SQL Database Managed Instance, then upload to Azure Migrate, as in the previous activity:



Note:

The assessment report for migrating the TailspinToys database to a target platform of SQL MI shows feature parity only with a PowerShell job step. In SQL MI, SQL Server

Agent is always running, and T-SQL and SSIS job steps are supported. However, PowerShell job steps are not yet supported. This serves as a warning, but it will not impact the migration of the TailspinToys database to SQL MI. Since SQL Server Agent Jobs are not supported in Azure SQL Database at all and TailspinToys doesn't leverage SQL Server Agent jobs, this is not mentioned in the assessment to move to Azure SQL Database. You can [review the full configuration of Azure SQL MI here](#).

10. The database, including the cross-database references and Service broker features, can be migrated as is, providing the opportunity for TailspinToys to have a fully managed PaaS database running in Azure. Previously, their options for migrating a database using features, such as Service Broker, incompatible with Azure SQL Database, were to deploy the database to a virtual machine running in Azure (IaaS) or modify their database and applications to not use the unsupported features. The introduction of Azure SQL MI, however, provides the ability to migrate databases into a managed Azure SQL database service with near 100% compatibility, including the features that prevented them from using Azure SQL Database.

Note:

For more information, check out the [feature comparison list](#) for SQL Server vs. Azure SQL Database vs. Azure SQL MI, and the [known T-SQL differences between Azure SQL MI and SQL Server](#).

11. Open the Azure portal, and navigate back to Azure Migrate. Select **Databases**. You should now see the results from the DMA scans. This will bring all of the databases and servers you scan and upload using DMA together, and provide a consolidated view.

The screenshot shows the Azure Migrate - Databases interface. On the left, there's a navigation sidebar with options like Overview, Migration goals, Servers, Databases (which is selected), Data Box, Manage, Discovered items, Support + troubleshooting, and New support request. The main area is titled 'Assessment tools' and contains a section for 'Azure Migrate: Database Assessment'. It shows the following data:

	Assessed database instances	1
DB	Assessed databases	1
Cloud	Databases ready for Azure SQL DB	1
Cloud	Databases ready for Azure SQL VM	1

A yellow lightning bolt icon with the text 'Next step: Start migration using Database Migration Service' is present.

12. You can click into **Assessed databases > TailspinToys** to see additional details and recommendations:

NAME	DATABASE SIZE (GB)	TARGET IN AZURE	MIGRATION BLOCKERS	BREAKING CHANGES COUNT	READY FOR AZURE SQL DB	READY FOR AZURE SQL VM	COMPATIBILITY LEVEL
TailspinToys	0.2	Azure SQL Database Managed Instance	0	0	Yes	Yes	100

Note:

If you are attending this lab as part of a day-long workshop, you have finished the activities for Module 4. [Return to it here](#), review, and refer to instructor guidance.



For Further Study

- [Azure Database Migration Guide](#) contains lots of resources that will help in guiding and supporting database migrations to Azure.
- [Azure Migrate Documentation](#) contains more information, guidance, and pointers on how to migrate your entire on-premises estate to Azure.

- [Data Migration Assistant Documentation](#) contains more information and best practices around the DMA tool explored in this module.
- [Choosing your database migration path to Azure](#) is a white paper created by Microsoft for deeper understanding of how to modernize and migrate on-premises SQL Server to Azure.



Next Steps

You can use this assessment to determine to move the database to Azure SQL Managed Instance. If you would like to complete a lab on that topic, navigate to [Lab - Migrating to Azure SQL Managed Instance](#).



Workshop: SQL Ground-to-Cloud

A Microsoft workshop from the SQL Server team



05 - Migrate to Azure SQL

In the previous module, you learned about Azure SQL, the benefits, the options, and how to get there. You reviewed how to assess your on-premises estate, and in this module you'll actually migrate to [Azure SQL Database Managed Instance](#).

([Make sure you check out the Prerequisites page before you start](#). You'll need all of the items loaded there before you can proceed with the workshop.)

In this module, you will use the [Azure Database Migration Service](#) (DMS) to migrate the TailspinToys database from the on-premises SQL 2008 R2 database to SQL MI. At the end of the module, you'll also explore some of the security and performance features available. The activities in this module include:

- [5.1: Migrate the database to SQL Managed instance](#)
- [5.2: Improve database security with Advanced Data Security \(Bonus\)](#)
- [5.3: Use online secondary for read-only queries \(Bonus\)](#)
- [5.4: After the Migration](#)

Note:

This module is mainly guided labs. Proceed as instructed in each section. You may not

have the time or desire to complete 5.2 and 5.3. They are optional, supplementary sections you can review at the end or at a later time.



5.1 Migrate the database to SQL Managed instance

In this section, you will use the [Azure Database Migration Service](#) (DMS) to migrate the TailspinToys database from the on-premises SQL 2008 R2 database to SQL MI. Tailspin Toys mentioned the importance of their gamer information web application in driving revenue, so for this migration you will target the [Business Critical service tier](#). The Business Critical service tier is designed for business applications with the highest performance and high-availability (HA) requirements. To learn more, read the [Managed Instance service tiers documentation](#).

This is the longest exercise in the module, so take your time (estimate about 30-45 minutes).

First, you will create a new SMB network share on the SqlServer2008 VM. This will be the folder used by DMS for retrieving backups of the TailspinToys database during the database migration process. Next, you will use the SQL Server Configuration Manager to update the service account used by the SQL Server (MSSQLSERVER) to the sqlmiuser account. This is done to ensure the SQL Server service has the appropriate permissions to write backups to the shared folder.

You'll then use the Azure Cloud shell to retrieve the information necessary to connect to your SQL MI and SqlServer2008 VM from DMS. You will also use the Azure Cloud Shell to create an Azure Active Directory (Azure AD) application and service principal (SP) that will provide DMS access to Azure SQL MI. You will grant the SP permissions to the hands-on-lab-SUFFIX resource group.

Now that your environment is prepared, you can create a new online data migration project in DMS and migrate the TailspinToys database. Since you perform the migration as an "online data migration," the migration wizard will continue to monitor the SMB network share for newly added log files. This allows for any updates that happen on the source database to be captured until you cut over to the SQL MI database. You'll then add a record to one of the database tables, backup the logs, and complete the migration of the TailspinToys database by cutting over to the SQL MI database. You will then be able to connect to the SQL MI database using SSMS, and quickly verify the migration.

With the TailspinToys database now running on SQL MI in Azure, the next step is to make the required modifications to the TailspinToys gamer information web application.



Activities

Complete activities 1-9 by following the links below. Once you're in Activity 1, you can continue to the next activity through Activity 9 (the links below all point to different parts of the same overall lab). When are you done with activities 1-9, come back here for the next section (5.2).

Note:

If you were provided an environment to do this workshop, you should review Activities 4 and 5, but a service principal has already been created (no action required).

[Activity 1](#): Create an SMB network share on the SQLServer2008VM

[Activity 2](#): Change MSSQLSERVER service to run under sqlmiuser account

[Activity 3](#): Create a backup of TailspinToys database

[Activity 4](#): Retrieve SQL MI, SQL Server 2008 VM, and service principal connection information (**review only - no action needed**)

[Activity 5](#): Create a service principal (**review only - no action needed**)

[Activity 6](#): Create and run an online data migration project

[Activity 7](#): Perform migration cutover

[Activity 8](#): Verify database and transaction log migration

[Activity 9](#): Update the application



Activity 1: Create an SMB network share on the SQLServer2008VM

In this task, you will create a new [SMB network share](#) on the SqlServer2008 VM. This will be the folder used by DMS for retrieving backups of the TailspinToys database during the database migration process. By creating the share, you're making it possible for services like DMS to access items in the share if you authenticate to it. You can read more about [the SMB protocol here](#).



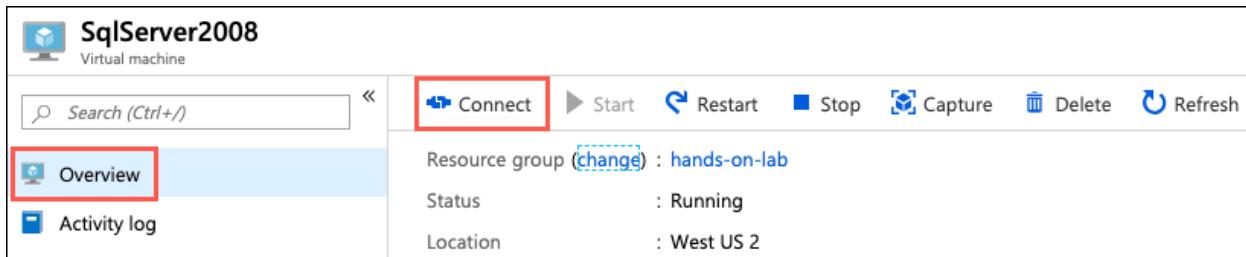
Steps

1. In the [Azure portal](#), navigate to your *SqlServer2008* VM by selecting **Resource groups** from the *left-hand navigation menu*, selecting the **hands-on-lab-SUFFIX** resource group, and selecting the **SqlServer2008** VM from the list of

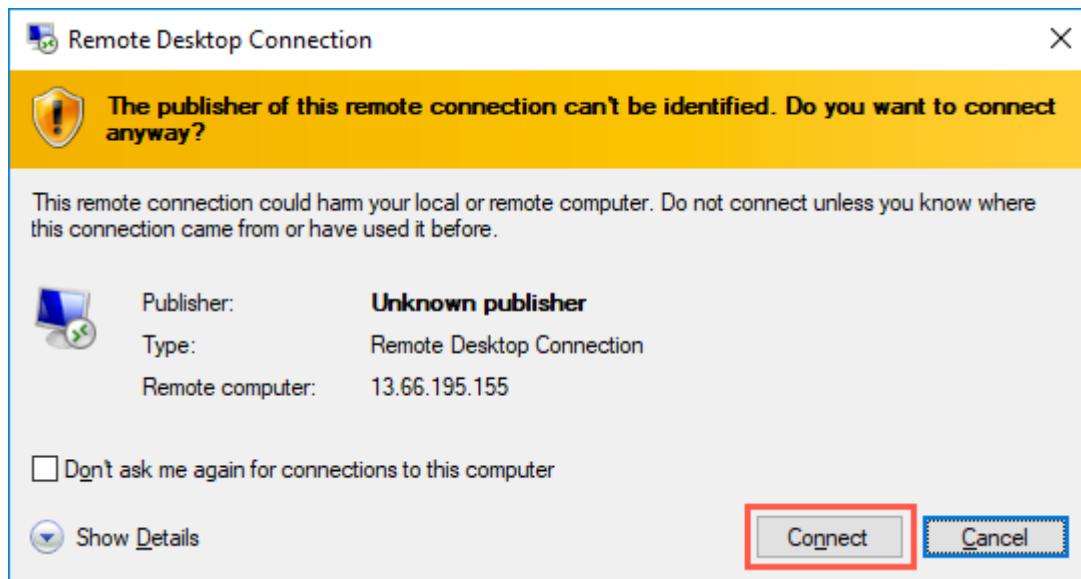
resources. On the SqlServer2008 Virtual Machine's *Overview* blade, select **Connect** on the top menu:

Note:

If you were provided with an environment for these labs, the "SUFFIX" part of the **hands-on-lab-SUFFIX** will be used in several portions of the lab. You should note this value from the credentials provided. Any time "SUFFIX" is referenced, use this number. **Do not use other "SUFFIX" values or you will disrupt another attendees' environment.**



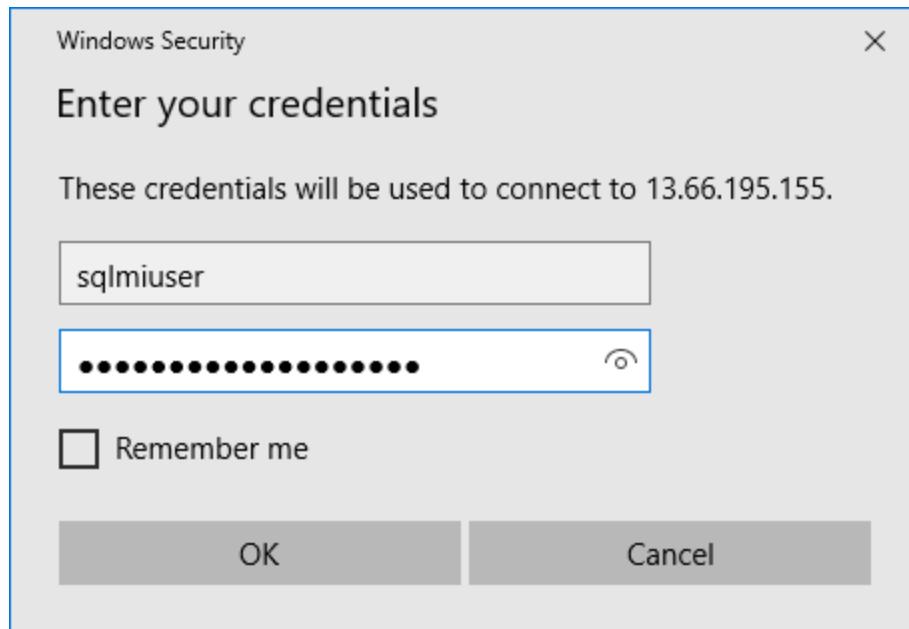
2. On the Connect to virtual machine blade, select **Download RDP File**, then open the downloaded RDP file.
3. Select **Connect** on the Remote Desktop Connection dialog:



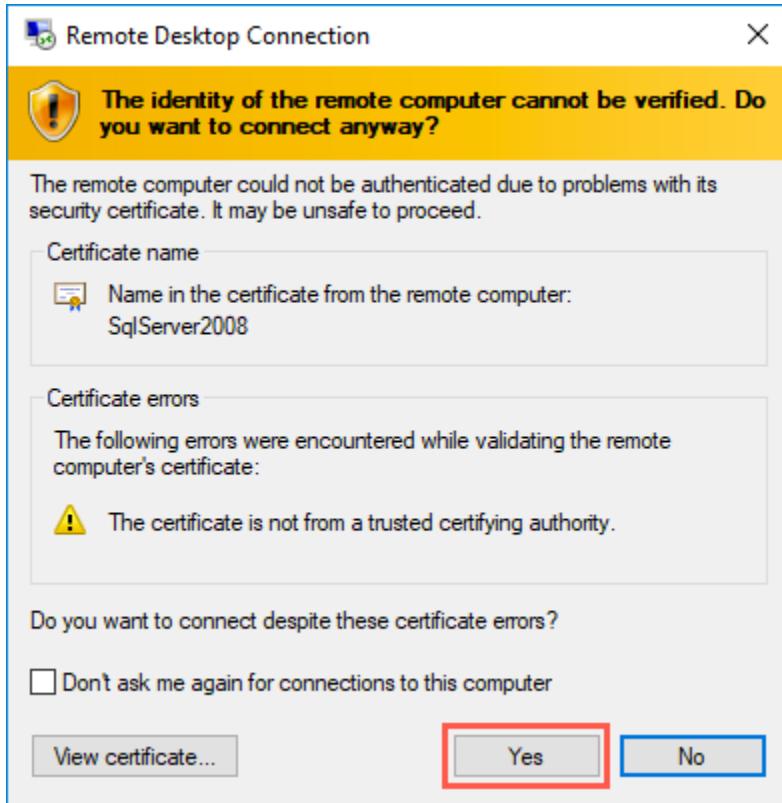
4. Enter the following credentials when prompted, and then select **OK**:
 - o Username: **sqlmiuser**

Note:

Password should be consistent among all labs. Your instructor will provide the password for in-class Labs.



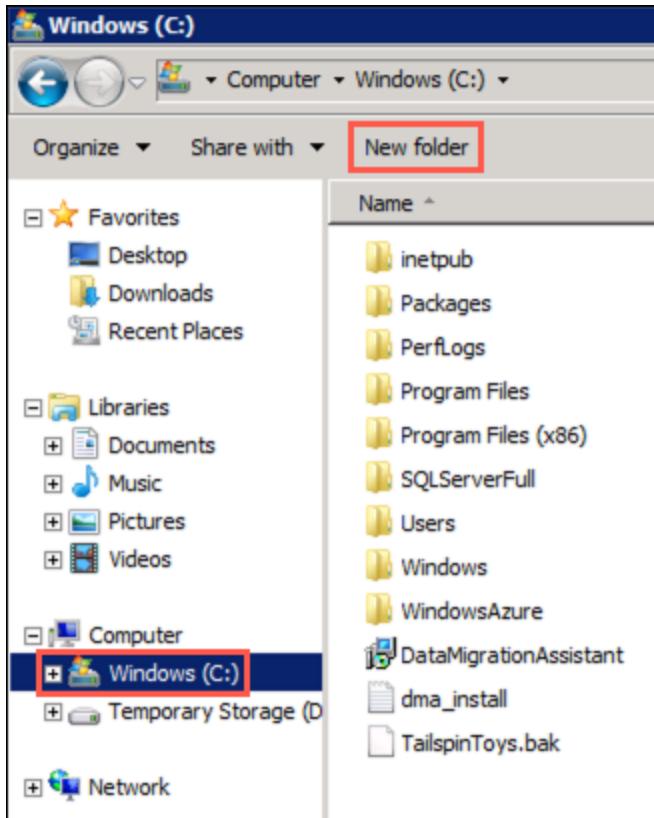
5. Select **Yes** to connect, if prompted that the identity of the remote computer cannot be verified:



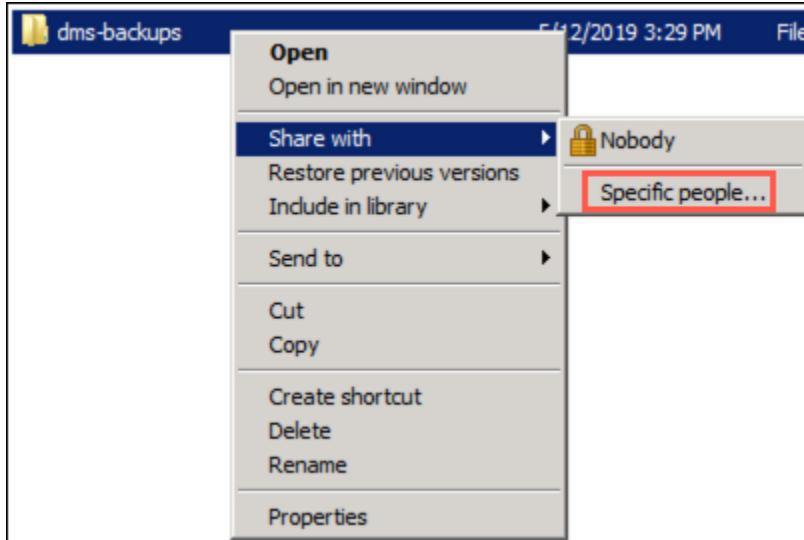
6. On the SqlServer2008 VM, open *Windows Explorer* by selecting its icon on the Windows Task bar:



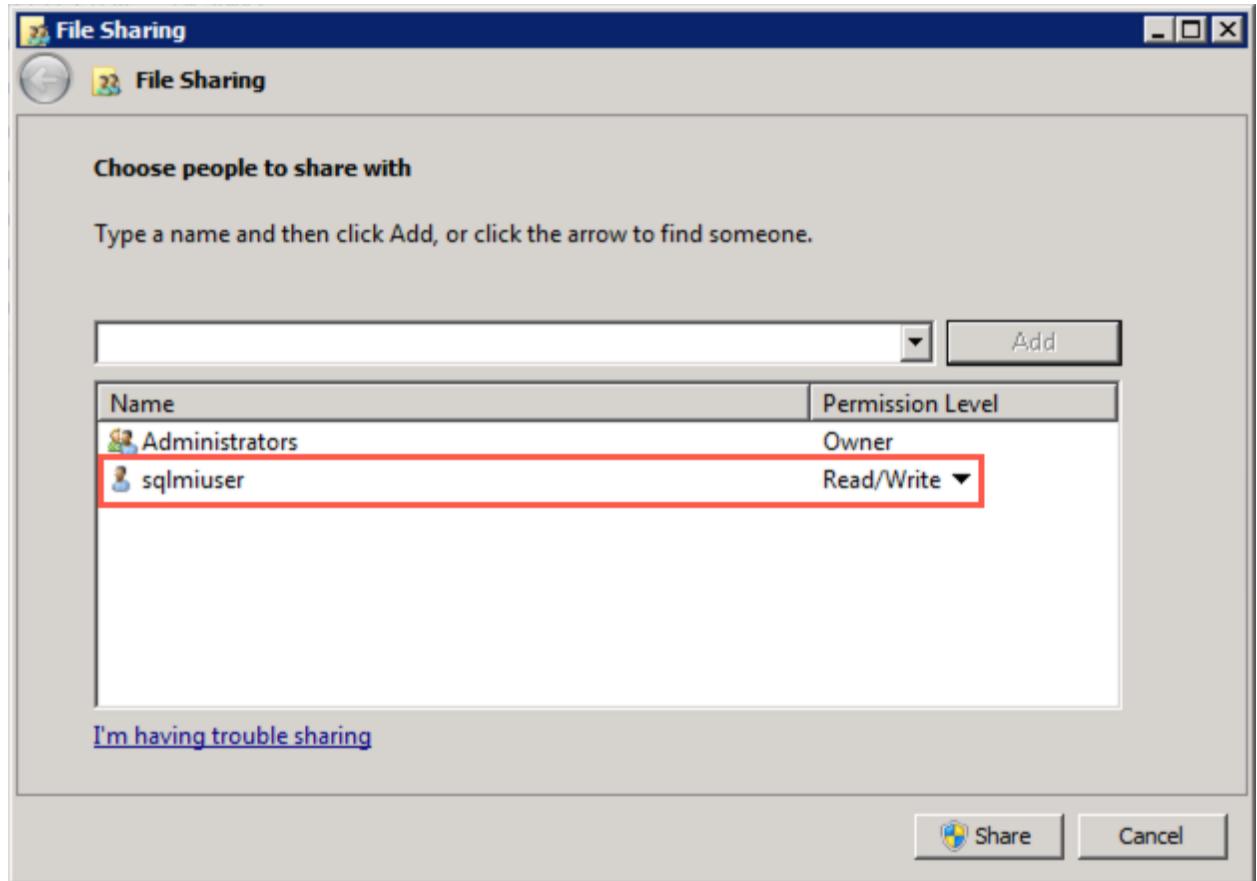
7. In the *Windows Explorer* window, expand **Computer** in the tree view, select **Windows (C:)**, and then select **New folder** in the top menu:



8. Name the new folder **dms-backups**, then right-click the folder and select **Share with** and **Specific people** in the context menu:



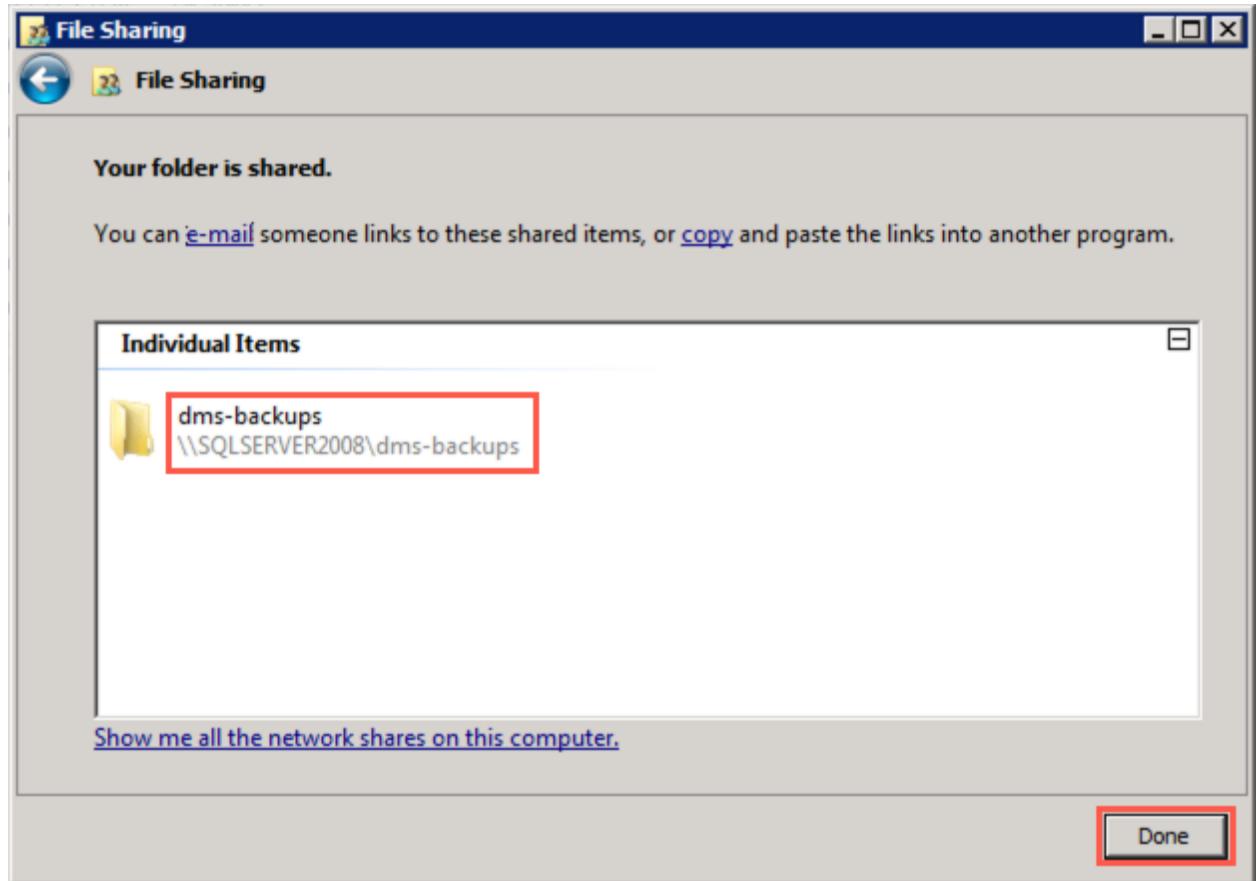
9. In the *File Sharing* dialog, ensure the **sqlmiuser** is listed with a **Read/Write** permission level, and then select **Share**:



10. In the **Network discovery and file sharing** dialog, select the default value of **No, make the network that I am connected to a private network**:



11. Back on the File Sharing dialog, note the path of the shared folder, \\SQLSERVER2008\dms-backups, and select **Done** to complete the sharing process.



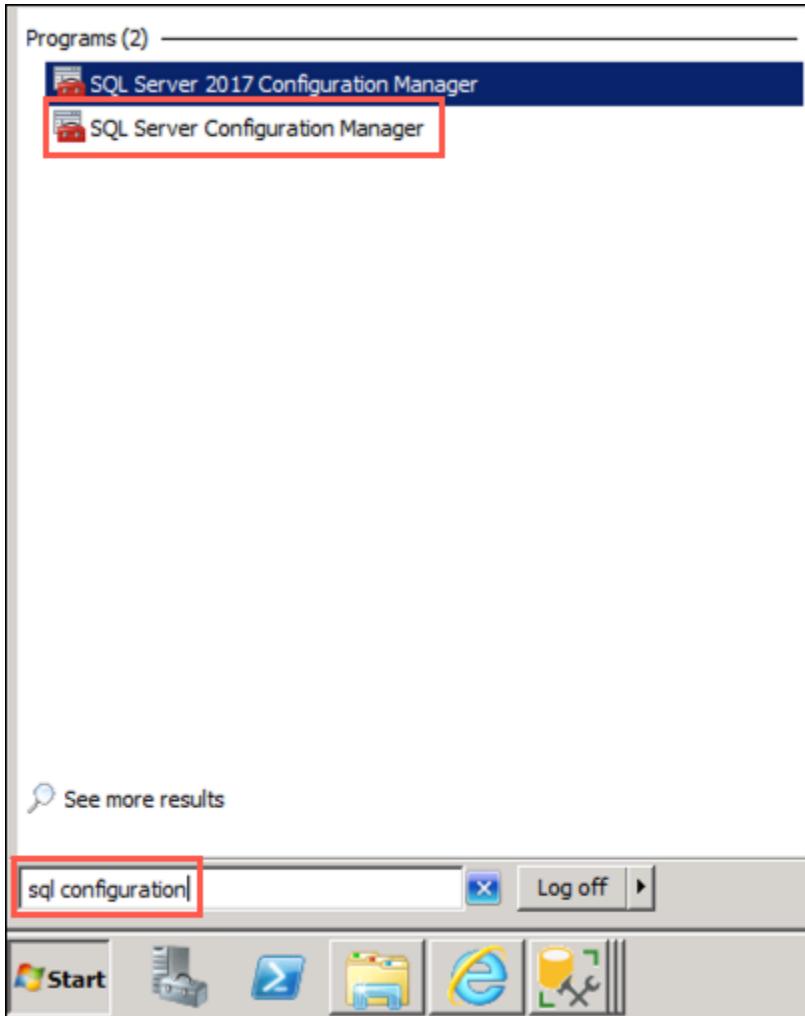
Activity 2: Change the MSSQLSERVER service to run under sqlmiuser account

In this task, you will use the SQL Server Configuration Manager to update the service account used by the *SQL Server (MSSQLSERVER)* service to the `sqlmiuser` account. This is done to ensure the SQL Server service has the appropriate permissions to write backups to the shared folder.



Steps

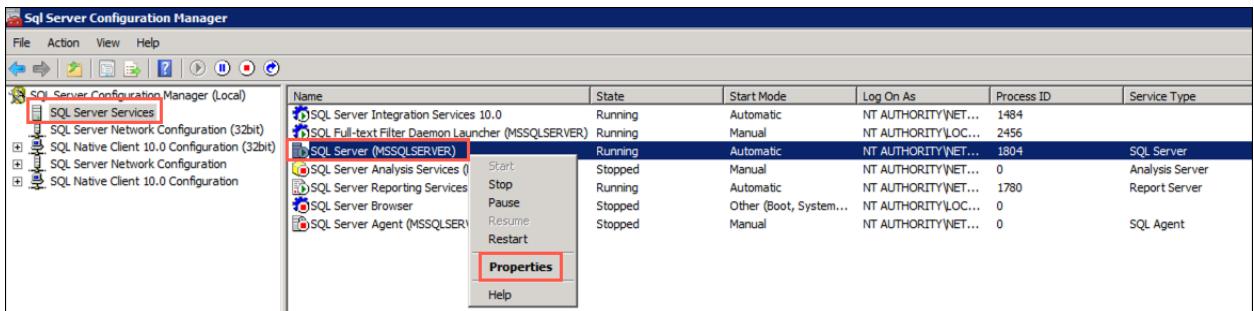
1. On your SqlServer2008 VM, select the **Start menu**, enter **sql configuration** into the search bar, and then select **SQL Server Configuration Managed** from the search results:



Note:

Be sure to choose **SQL Server Configuration Manager**, and not **SQL Server 2017 Configuration Manager**, which will not work for the installed SQL Server 2008 R2 database.

2. In the *SQL Server Configuration Manager* dialog, select **SQL Server Services** from the tree view on the left, then right-click **SQL Server (MSSQLSERVER)** in the list of services and select **Properties** from the context menu:

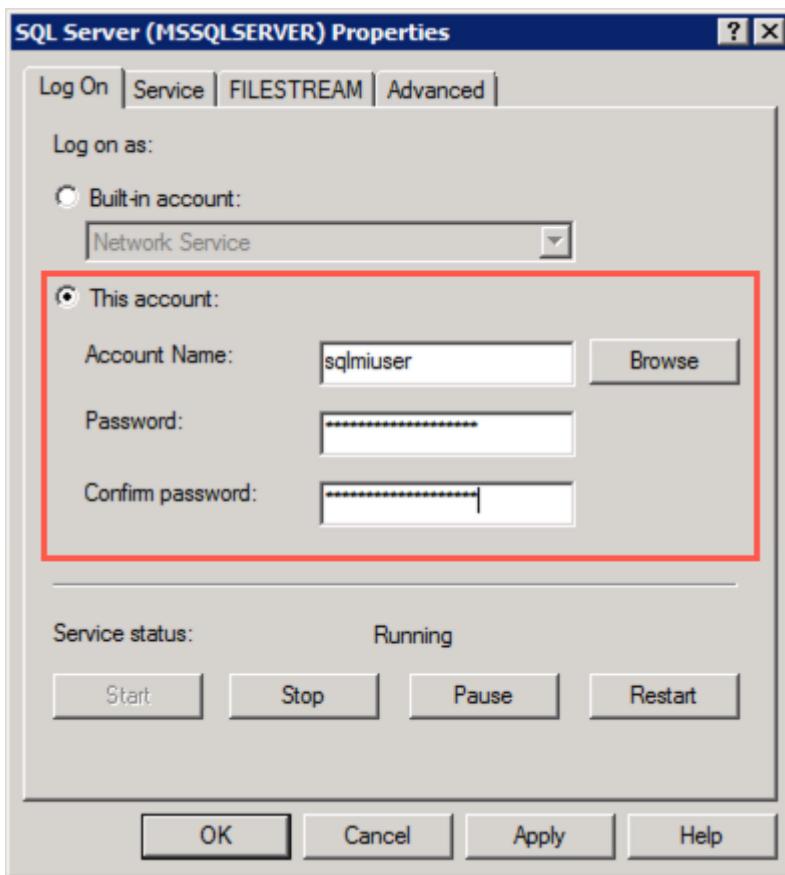


3. In the *SQL Server (MSSQLSERVER) Properties* dialog, select **This account** under *Log on as*, and enter the following:

- o Account name: **sqlmiuser**

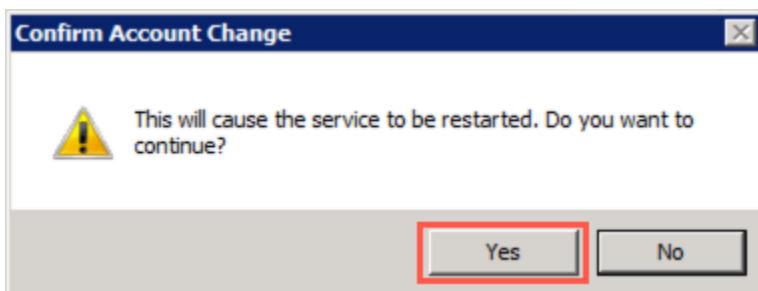
Note:

Password should be consistent among all labs, ask your instructor if you don't know what your password is.



4. Select **OK**

5. Select **Yes** in the *Confirm Account Change* dialog:



6. You will now see the *Log On As* value for the *SQL Server (MSSQLSERVER)* service changed to `./sqlmiuser`:

Name	State	Start Mode	Log On As	Process ID	Service Type
SQL Server Integration Services 10.0	Running	Automatic	NT AUTHORITY\NET...	1484	
SQL Full-text Filter Daemon Launcher (MSSQLSERVER)	Running	Manual	NT AUTHORITY\LOC...	2456	
SQL Server (MSSQLSERVER)	Running	Automatic	<code>.\sqlmiuser</code>	5100	SQL Server
SQL Server Analysis Services (MSSQLSERVER)	Stopped	Manual	NT AUTHORITY\NET...	0	Analysis Server
SQL Server Reporting Services (MSSQLSERVER)	Running	Automatic	NT AUTHORITY\NET...	1780	Report Server
SQL Server Browser	Stopped	Other (Boot, System...)	NT AUTHORITY\LOC...	0	
SQL Server Agent (MSSQLSERVER)	Stopped	Manual	NT AUTHORITY\NET...	0	SQL Agent



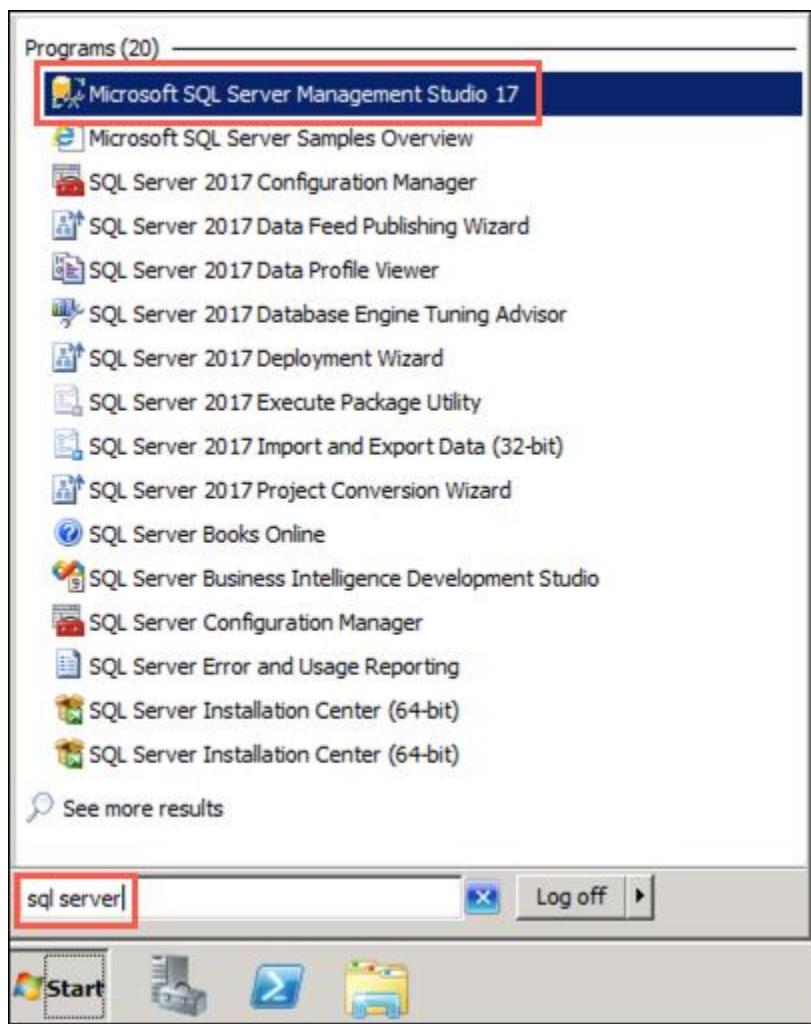
Activity 3: Create a backup of TailspinToys database

To perform online data migrations, DMS looks for backups and logs in the SMB shared backup folder on the source database server. In this task, you will create a backup of the `TailspinToys` database using SSMS, and write it to the SMB network share you created in the previous task. The backup file needs to include a checksum, so you will add that during the backup steps.

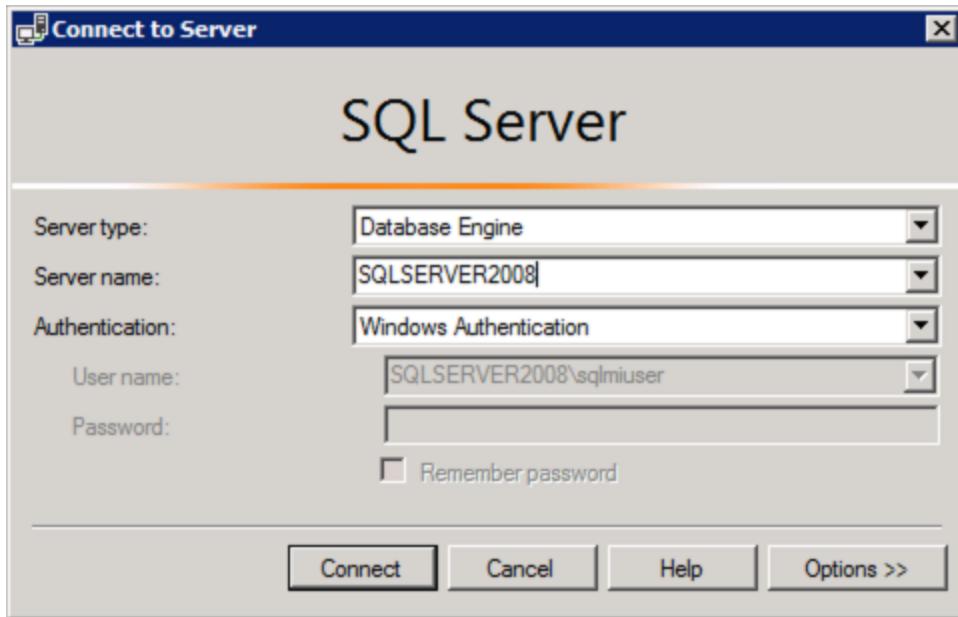


Steps

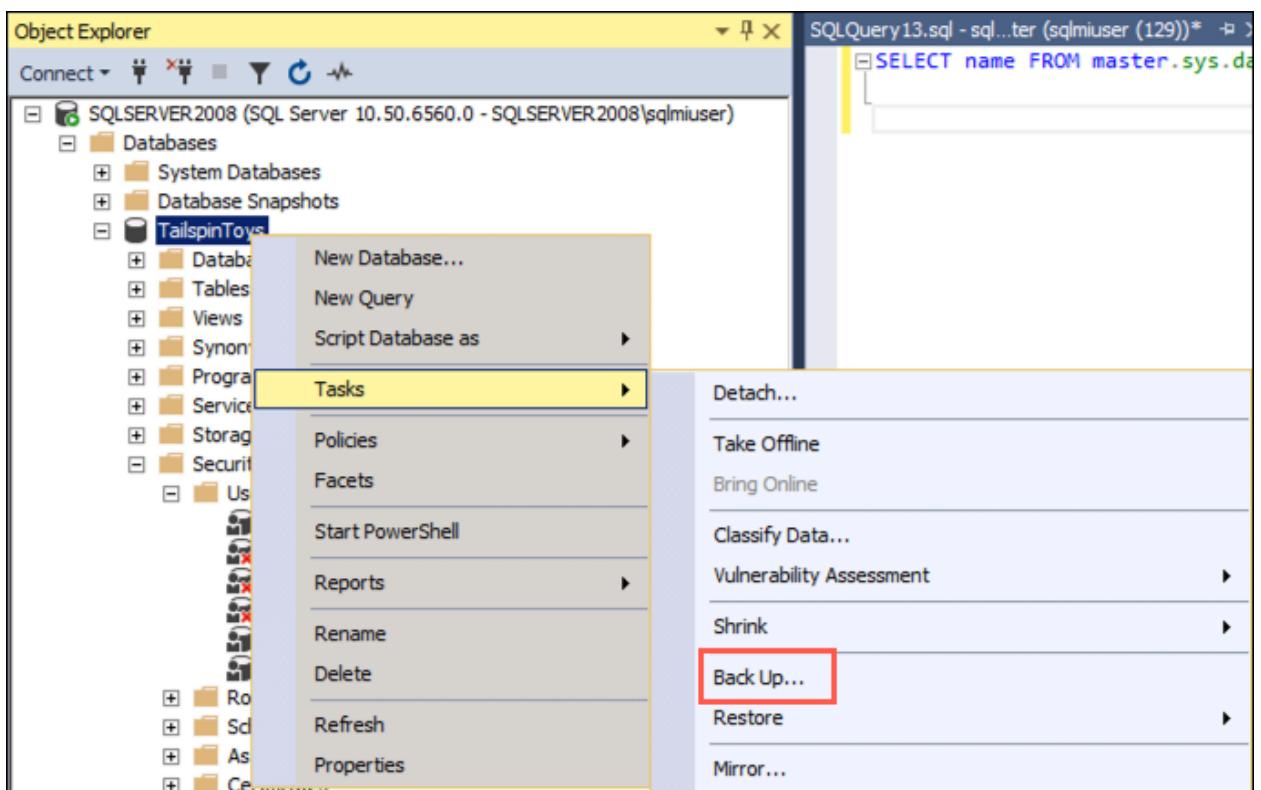
1. On the SqlServer2008 VM, open *Microsoft SQL Server Management Studio 17* by entering **sql server** into the search bar in the Windows Start menu:



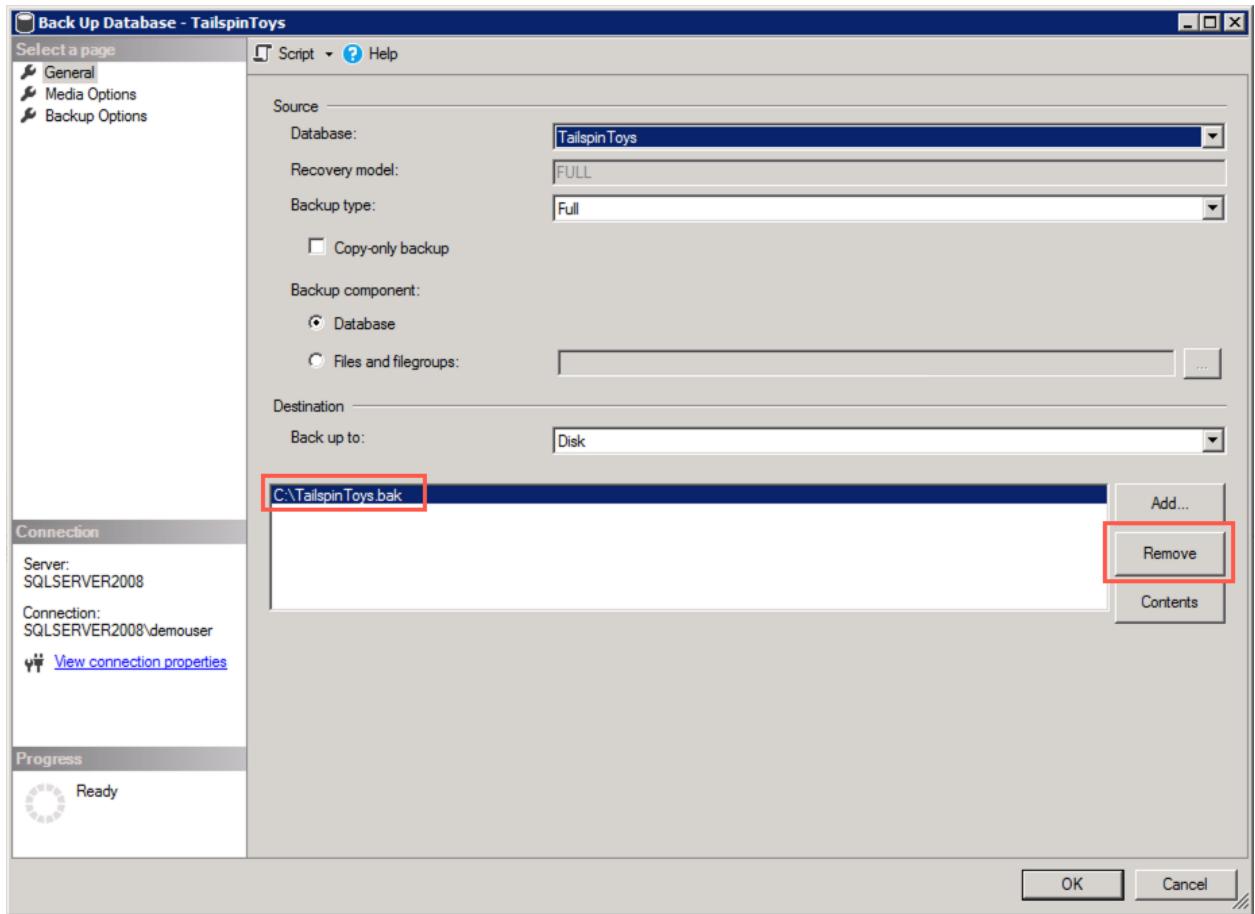
2. In the SSMS Connect to Server dialog, enter **SQLSERVER2008** into the *Server name* box, ensure **Windows Authentication** is selected, and then select **Connect**:



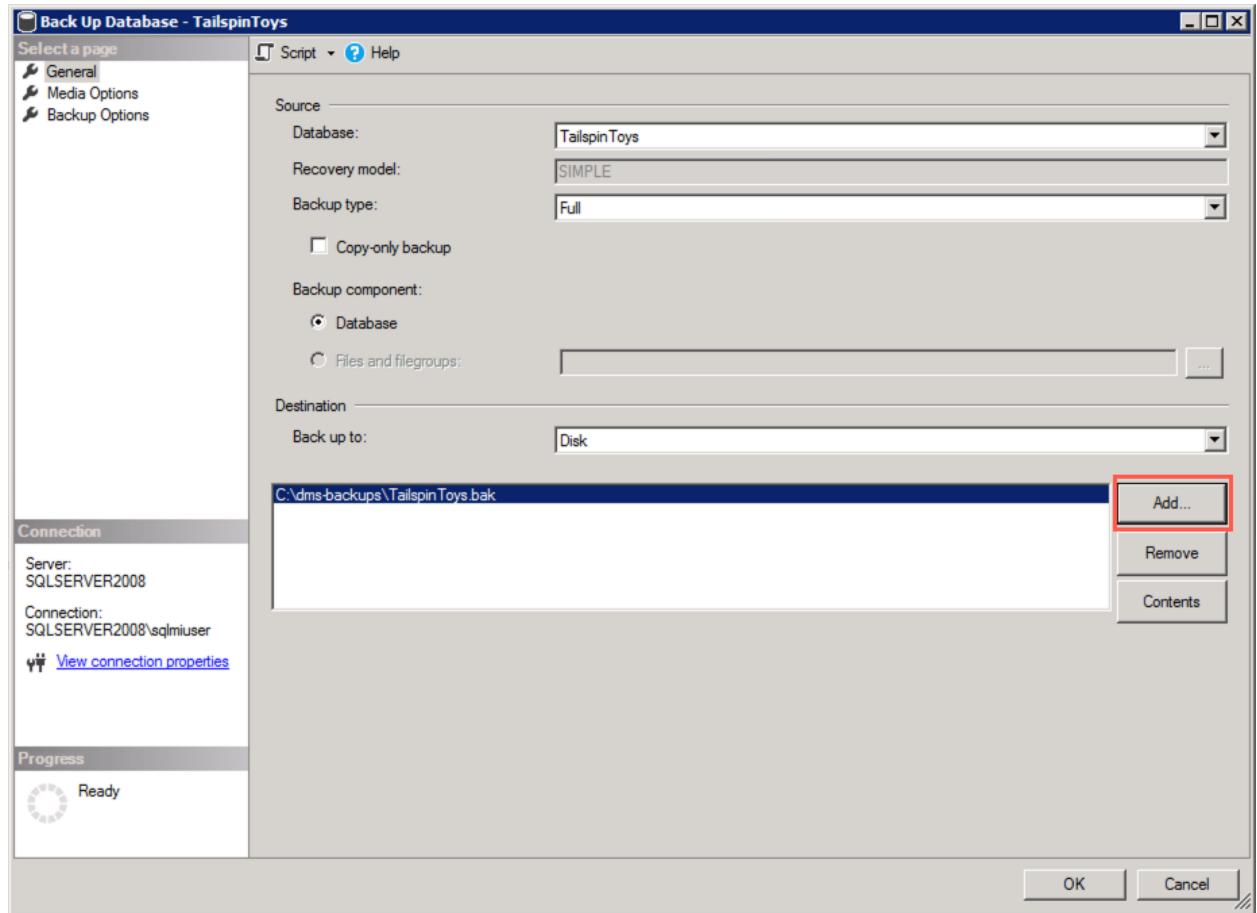
- Once connected, expand **Databases** under **SQLSERVER2008** in the *Object Explorer*, and then right-click the **TailspinToys** database. In the context menu, select **Tasks** and then **Back Up**:



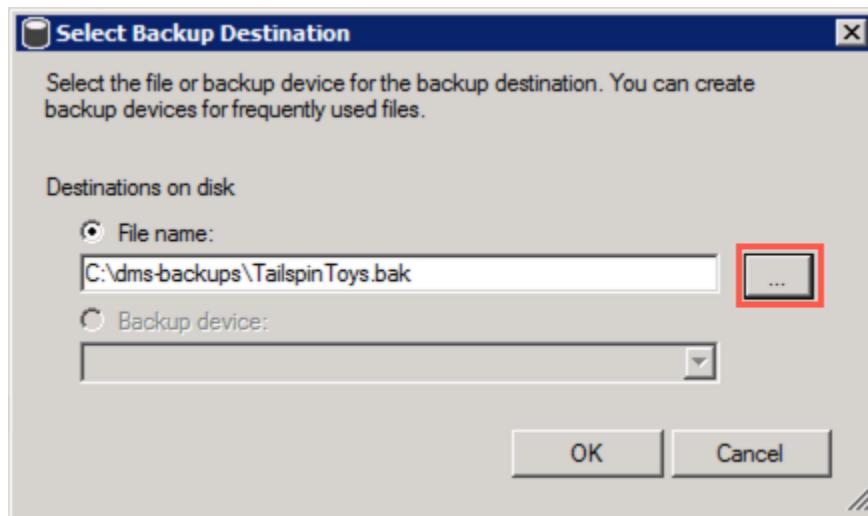
4. In the *Back Up Database* dialog, you will see C:\TailspinToys.bak listed in the *Destinations* box. This is no longer needed, so select it, and then select **Remove**:



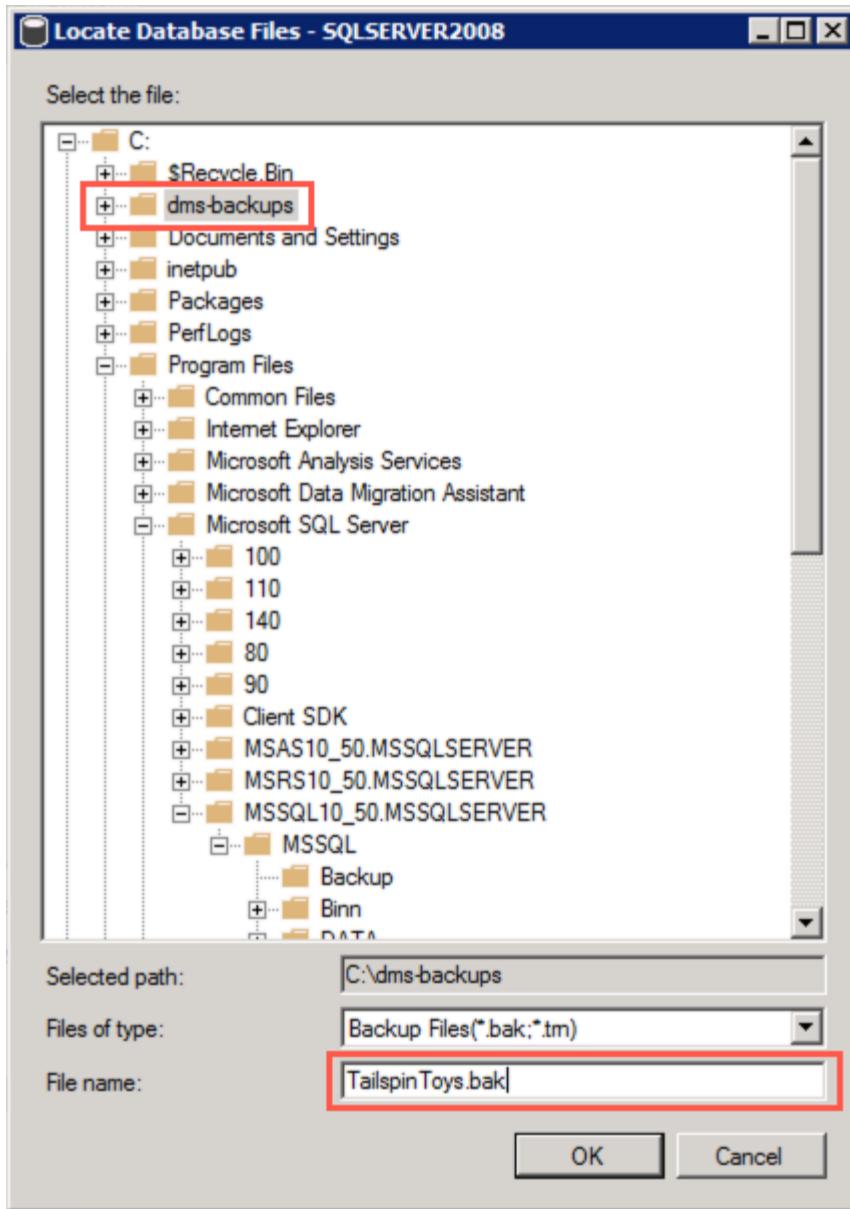
5. Next, select **Add** to add the SMB network share as a backup destination:



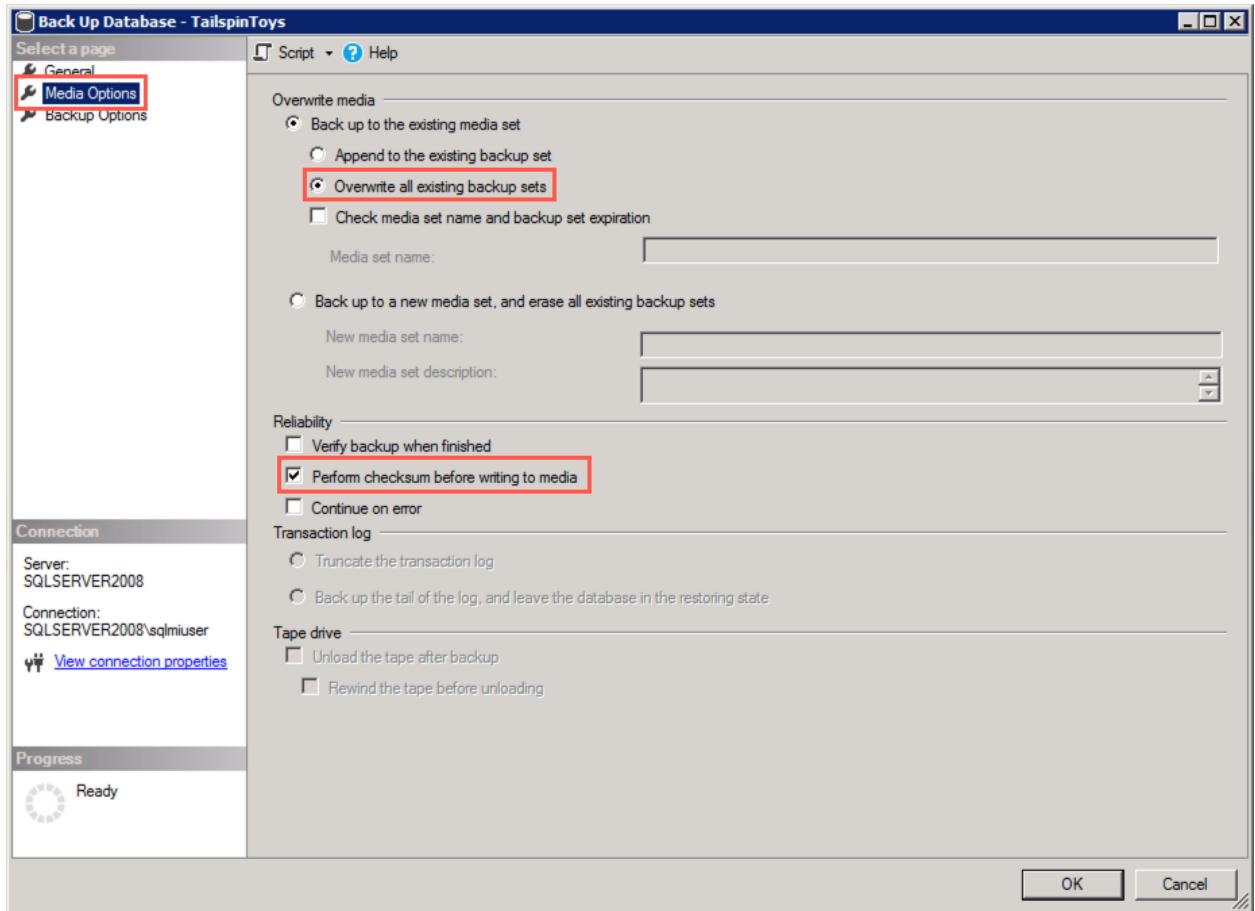
6. In the *Select Backup Destination* dialog, select the **Browse (...)** button:



7. In the *Locate Database Files* dialog, select the c:\dms-backups folder, enter **TailspinToys.bak** into the *File name* field, and then select **OK**:

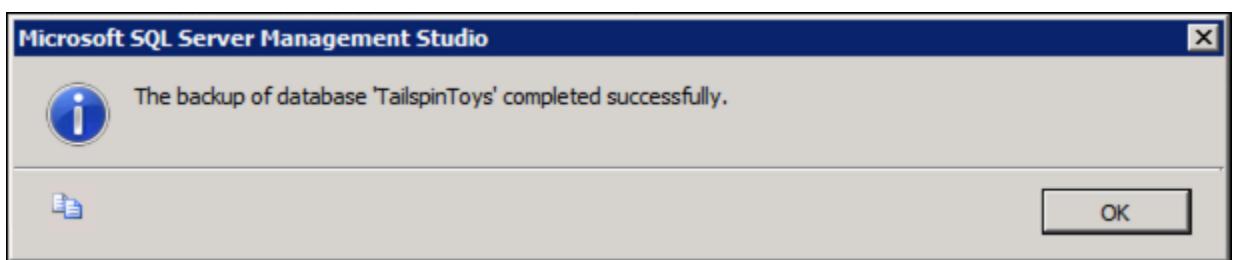


8. Select **OK** to close the Select Backup Destination dialog
9. In the *Back Up Database* dialog, select the **Media Options** in the *Select a page* pane, and then set the following:
 - o Select **Back up to the existing media set** and then select **Overwrite all existing backup sets**
 - o Under *Reliability*, check the box for **Perform checksum before writing to media**. This is required by DMS when using the backup to restore the database to SQL MI:



10. Select **OK** to perform the backup

11. You will receive a message when the backup is complete. Select **OK**:



 **Activity 4: Retrieve SQL MI, SQL Server 2008 VM, and service principal connection information**

Note:

If you're doing this lab as part of a workshop and were provided an environment to use, this step has already been completed. You can review, but **there is nothing you need to do**. Please refer to instructor guidance.

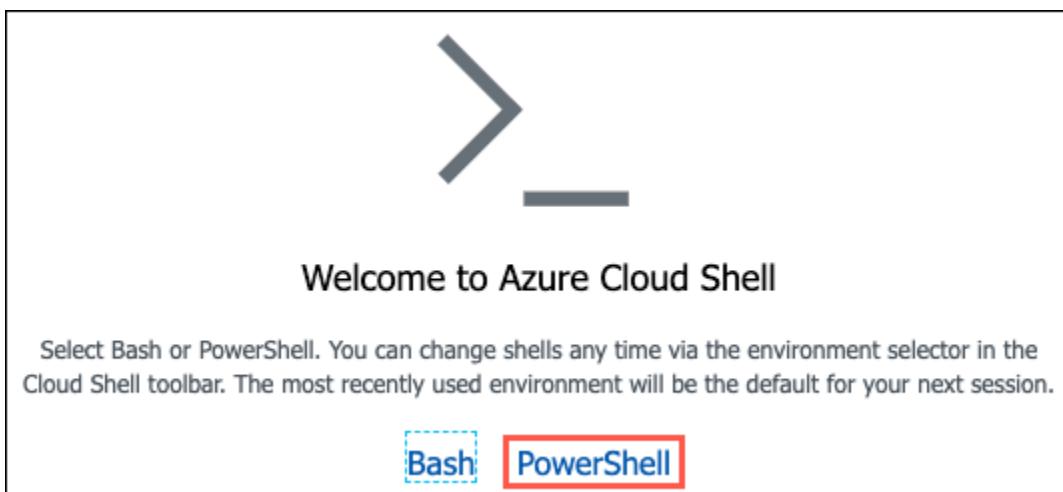
In this task, you will use the Azure Cloud shell to retrieve the information necessary to connect to your SQL MI and SqlServer2008 VM from DMS.

✓ Steps

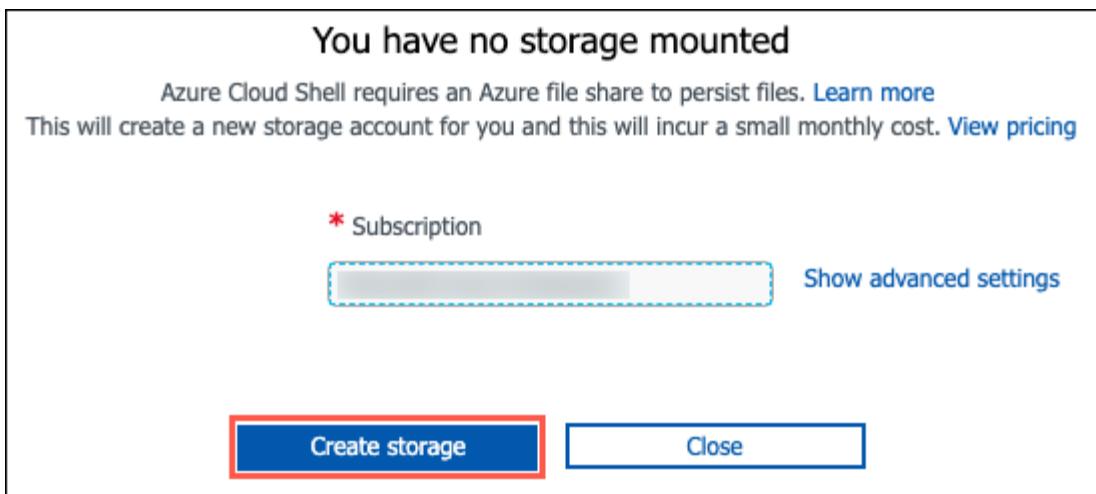
1. In the [Azure portal](#), select the **Azure Cloud Shell** icon from the top menu:



2. In the *Cloud Shell* window that opens at the bottom of your browser window, select **PowerShell**:



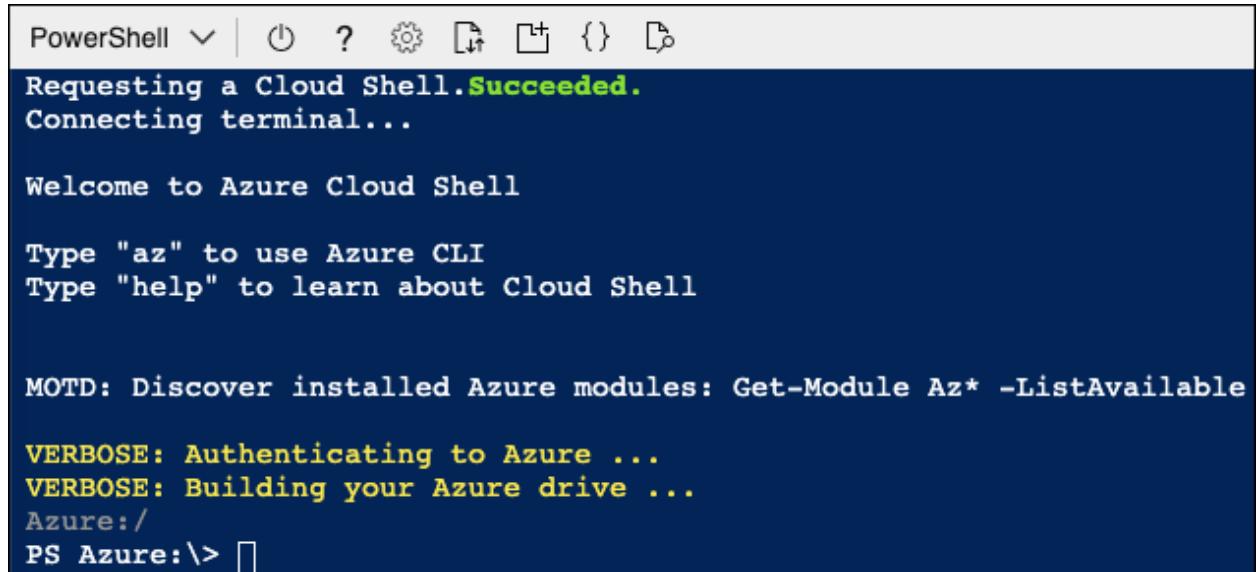
3. If prompted that you have no storage mounted, select the subscription you are using for this hands-on lab and select **Create storage**:



Note:

If creation fails, you may need to select **Advanced settings** and specify the subscription, region and resource group for the new storage account.

4. After a moment, you will receive a message that you have successfully requested a Cloud Shell, and be presented with a PS Azure prompt:



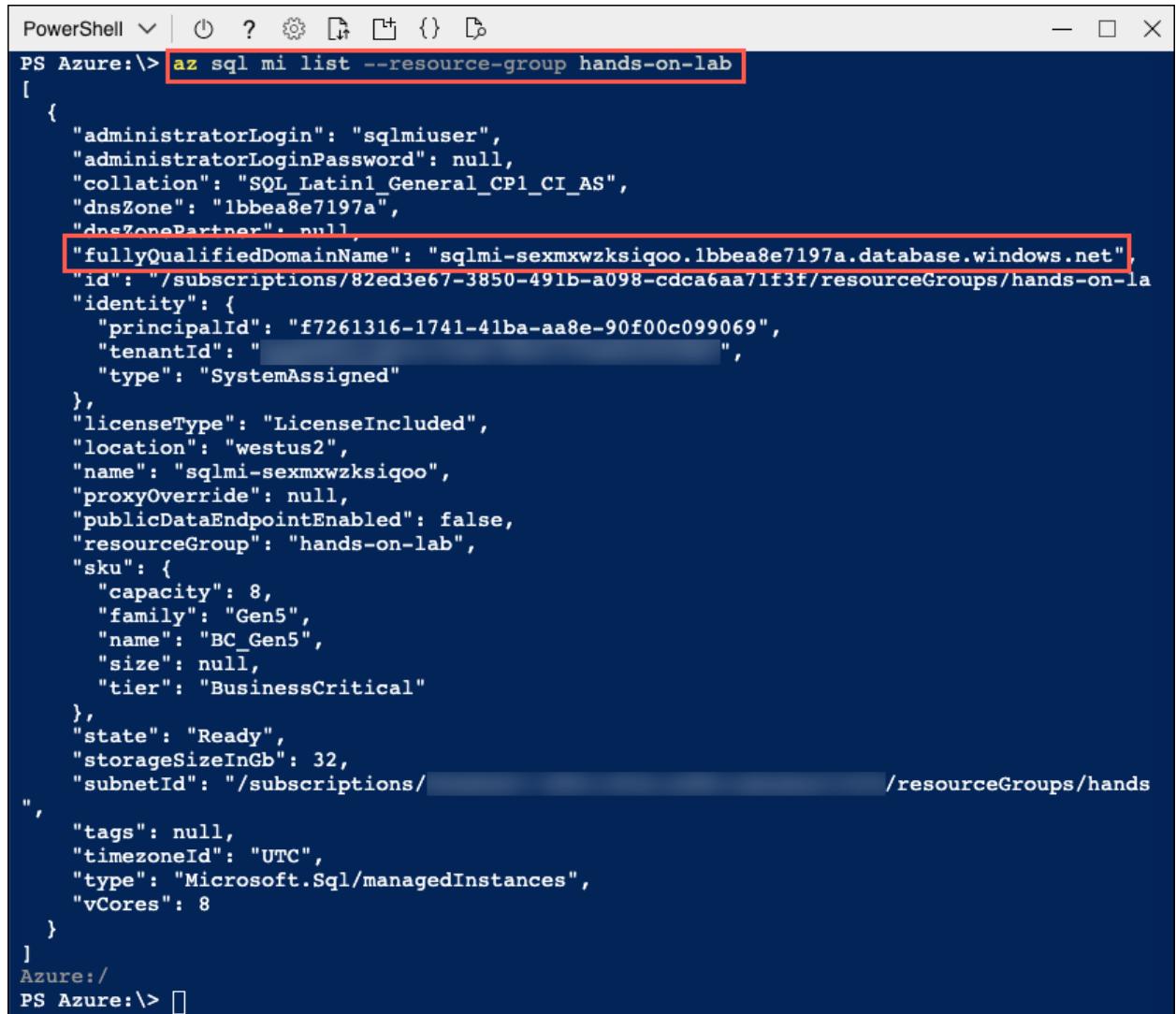
The screenshot shows a PowerShell terminal window titled "PowerShell". The output shows the following sequence of events:

- "Requesting a Cloud Shell. **Succeeded.**"
- "Connecting terminal..."
- "Welcome to Azure Cloud Shell"
- "Type "az" to use Azure CLI"
- "Type "help" to learn about Cloud Shell"
- "MOTD: Discover installed Azure modules: Get-Module Az* -ListAvailable"
- "VERBOSE: Authenticating to Azure ..."
- "VERBOSE: Building your Azure drive ..."
- "Azure:/"
- "PS Azure:\> "

5. At the prompt, you will retrieve information about SQL MI in the hands-on-lab-SUFFIX resource group by entering the following PowerShell command, **replacing SUFFIX** with your unique identifier or specific "SUFFIX" assigned to you for the labs:

```
az sql mi list --resource-group hands-on-lab-SUFFIX
```

6. Within the output of the above command, locate and copy the value of the `fullyQualifiedDomainName` property. Paste the value into a text editor, such as *Notepad.exe*, for later reference:



```
PowerShell | ⌂ ? ⌂ { } ⌂
PS Azure:\> az sql mi list --resource-group hands-on-lab
[
  {
    "administratorLogin": "sqlmiuser",
    "administratorLoginPassword": null,
    "collation": "SQL_Latin1_General_CI_AS",
    "dnsZone": "1bbea8e7197a",
    "dnsZonePartner": null,
    "fullyQualifiedDomainName": "sqlmi-sexmxwzksiqoo.1bbea8e7197a.database.windows.net",
    "id": "/subscriptions/82ed3e67-3850-491b-a098-cdca6aa71f3f/resourceGroups/hands-on-la",
    "identity": {
      "principalId": "f7261316-1741-41ba-aa8e-90f00c099069",
      "tenantId": "REDACTED",
      "type": "SystemAssigned"
    },
    "licenseType": "LicenseIncluded",
    "location": "westus2",
    "name": "sqlmi-sexmxwzksiqoo",
    "proxyOverride": null,
    "publicDataEndpointEnabled": false,
    "resourceGroup": "hands-on-lab",
    "sku": {
      "capacity": 8,
      "family": "Gen5",
      "name": "BC_Gen5",
      "size": null,
      "tier": "BusinessCritical"
    },
    "state": "Ready",
    "storageSizeInGb": 32,
    "subnetId": "/subscriptions/REDACTED/resourceGroups/hands-on-lab",
    "tags": null,
    "timezoneId": "UTC",
    "type": "Microsoft.Sql/managedInstances",
    "vCores": 8
  }
]
Azure:/
```

7. Next, you will enter a second command to retrieve the public IP address of the SqlServer2008 VM, which you will use to connect to the database on that server. Enter the following PowerShell command, **replacing SUFFIX** with your unique identifier or specific "SUFFIX" assigned to you for the labs:

```
az vm list-ip-addresses -g hands-on-lab-SUFFIX -n SqlServer2008
```

8. Within the output of the command above, locate and copy the value of the ipAddress property within the publicIpAddresses object. Paste the value into a text editor, such as *Notepad.exe*, for later reference:

```
PowerShell PS Azure:\> az vm list-ip-addresses -g hands-on-lab -n SqlServer2008
[{"virtualMachine": {"name": "SqlServer2008", "network": {"privateIpAddresses": ["10.0.1.5"], "publicIpAddresses": [{"id": "/subscriptions//resourceGroups/hands-on-lab/providers/Microsoft.Network/publicIPAddresses/SqlServer2008-ip", "ipAddress": "13.66.228.107", "ipAllocationMethod": "Dynamic", "name": "SqlServer2008-ip", "resourceGroup": "hands-on-lab"}]}, "resourceGroup": "hands-on-lab"}]
Azure:/ PS Azure:\>
```

9. (Leave the Azure Cloud Shell open for the next set of tasks)



Activity 5: Create a service principal

Note:

If you're doing this lab as part of a workshop and were provided an environment to use, this step has already been completed. You can review, but **there is nothing you need to do**. Please refer to instructor guidance.

In this task, you will use the Azure Cloud Shell to create an Azure Active Directory (Azure AD) application and service principal (SP) that will provide DMS access to Azure SQL MI. You will grant the SP permissions to the hands-on-lab-SUFFIX resource group.

Note:

You must have rights within your Azure AD tenant to create applications and assign roles to complete this task. If you are blocked by this, but still want to do a migration with Azure Database Migration Services, you can perform an offline migration. In Activity 6, select offline instead of online migration in Step 3, and instead of Step 7, you can refer to [this section of a migrating to Azure SQL Database Managed Instance offline tutorial](#).



Steps

Note:

If you're doing this lab as part of a workshop and were provided an environment to use, this step has already been completed. You can review, but **there is nothing you need to do**. Please refer to instructor guidance.

1. Next, you will issue a command to create a service principal named **tailspin-toys** and assign it contributor permissions to your *hands-on-lab-SUFFIX* resource group.
2. First, you need to retrieve your subscription ID. Enter the following at the Cloud Shell prompt:

```
az account list --output table
```

3. In the *output* table, locate the subscription you are using for this hands-on lab, and copy the *SubscriptionId* value into a text editor for use later.
4. Next, enter the following command at the Cloud Shell prompt, replacing {SubscriptionID} with the value you copied above and {ResourceGroupName} with the name of your *hands-on-lab-SUFFIX* resource group, and then press Enter to run the command:

```
az ad sp create-for-rbac -n "tailspin-toys" --role owner --scopes subscriptions/{SubscriptionID}/resourceGroups/{ResourceGroupName}
```

```
PS Azure:\> az ad sp create-for-rbac -n "tailspin-toys" --role contributor --scopes
Changing "tailspin-toys" to a valid URI of "http://tailspin-toys", which is the req
Found an existing application instance of "aeab3b83-9080-426c-94a3-4828db8532e9". W
{
  "appId": "aeab3b83-9080-426c-94a3-4828db8532e9",
  "displayName": "tailspin-toys",
  "name": "http://tailspin-toys",
  "password": "76ff5bae-8d25-469a-a74b-4a33ad868585",
  "tenant": "d280491c-[REDACTED]"
}
Azure:\>
PS Azure:\>
```

5. Copy the output from the command into a text editor, as you will need the appId and password in the next task. The output should be similar to:

```
6. {
7.   "appId": "aeab3b83-9080-426c-94a3-4828db8532e9",
8.   "displayName": "tailspin-toys",
9.   "name": "http://tailspin-toys",
10.  "password": "76ff5bae-8d25-469a-a74b-4a33ad868585",
11.  "tenant": "d280491c-b27a-XXXX-XXXX-XXXXXXXXXXXX"
```

12. To verify the role assignment, select **Access control (IAM)** from the left-hand menu of the *hands-on-lab-SUFFIX* resource group blade, and then select the **Role assignments** tab and locate *tailspin-toys* under the **OWNER** role.

NAME	TYPE	ROLE	SCOPE
KB	User	Owner	Subscription (Inherited)
tailspin-toys-sp	App	Owner	This resource

13. Next, you will issue another command to grant the **CONTRIBUTOR** role at the subscription level to the newly created service principal. At the *Cloud Shell* prompt, run the following command:

```
az role assignment create --assignee http://tailspin-toys --role contributor
```



Activity 6: Create and run an online data migration project

In this task, you will create a new online data migration project in DMS for the TailspinToys database.



Steps

1. In the [Azure portal](#), navigate to the *Azure Database Migration Service* by selecting **Resource groups** from the left-hand navigation menu, selecting the **hands-on-lab-SUFFIX** resource group, and then selecting the **tailspin-dms** Azure Database Migration Service in the list of resources:

NAME ↑↓	TYPE ↑↓	LOCATION ↑↓
SqlServer2008-ip	Public IP address	West US 2
SqlServer2008-nic	Network interface	West US 2
SqlServer2008-nsg	Network security group	West US 2
tailspin-dms	Azure Database Migration Service	West US 2

Note:

If you were provided with an environment for these labs, the "SUFFIX" part of the **hands-on-lab-SUFFIX** will be used in several portions of the lab. You should note this value from the credentials provided. Any time "SUFFIX" is referenced, use this number. **Do not use other "SUFFIX" values or you will disrupt another attendees' environment.**

2. On the *Azure Database Migration Service* blade, select **+New Migration Project**:

Note:

If you were provided an environment for this lab, your Database Migration Service may be paused due to inactivity. You can select **Start Service** to start it, if you get the message "The service is currently unavailable for migration."

The screenshot shows the Azure Database Migration Service blade for the 'tailspin-dms' service. The left sidebar has tabs for Overview, Activity log, Access control (IAM), and Tags. The Overview tab is selected. At the top, there's a search bar, a '+ New Migration Project' button (which is highlighted with a red box), and service management buttons (Delete service, Refresh, Start Service, Stop Service). A green success message box displays: 'Great job! Your database migration service was successfully created. You can create your first migration project now.' Below the message, under the 'Essentials' section, it shows the Resource group (change) as 'hands-on-lab', Status as 'Online', and Location as 'West US 2'. It also lists the Virtual network & IP Address as 'hands-on-lab-vnet/subnets/Management 10.0.1.6'.

3. On the New migration project blade, enter the following:

- *Project name:* Enter **OnPremToSqlMi**
- *Source server type:* Select **SQL Server**
- *Target server type:* Select **Azure SQL Database Managed Instance**
- *Choose type of activity:* Select **Online data migration** and select **Save**

New migration project		Type of activity
Project name <input type="text" value="OnPremToSqlMi"/> ✓	Choose type of activity <input type="text" value="Online data migration"/> ▼	
* Source server type <input type="text" value="SQL Server"/> ▼	Use this option to migrate databases that must be accessible and continuously updated during migration.	
* Target server type <input type="text" value="Azure SQL Database Managed Instance"/> ▼		
* Choose type of activity <input type="text" value="Online data migration"/> >		
Create and run activity	Save	

4. Select **Create and run activity**

5. On the Migration Wizard **Select source** blade, enter the following:

- *Source SQL Server instance name:* Enter the Public IP address of your SqlServer2008 VM that you copied into a text editor in the previous task.
For example, **13.66.228.107**

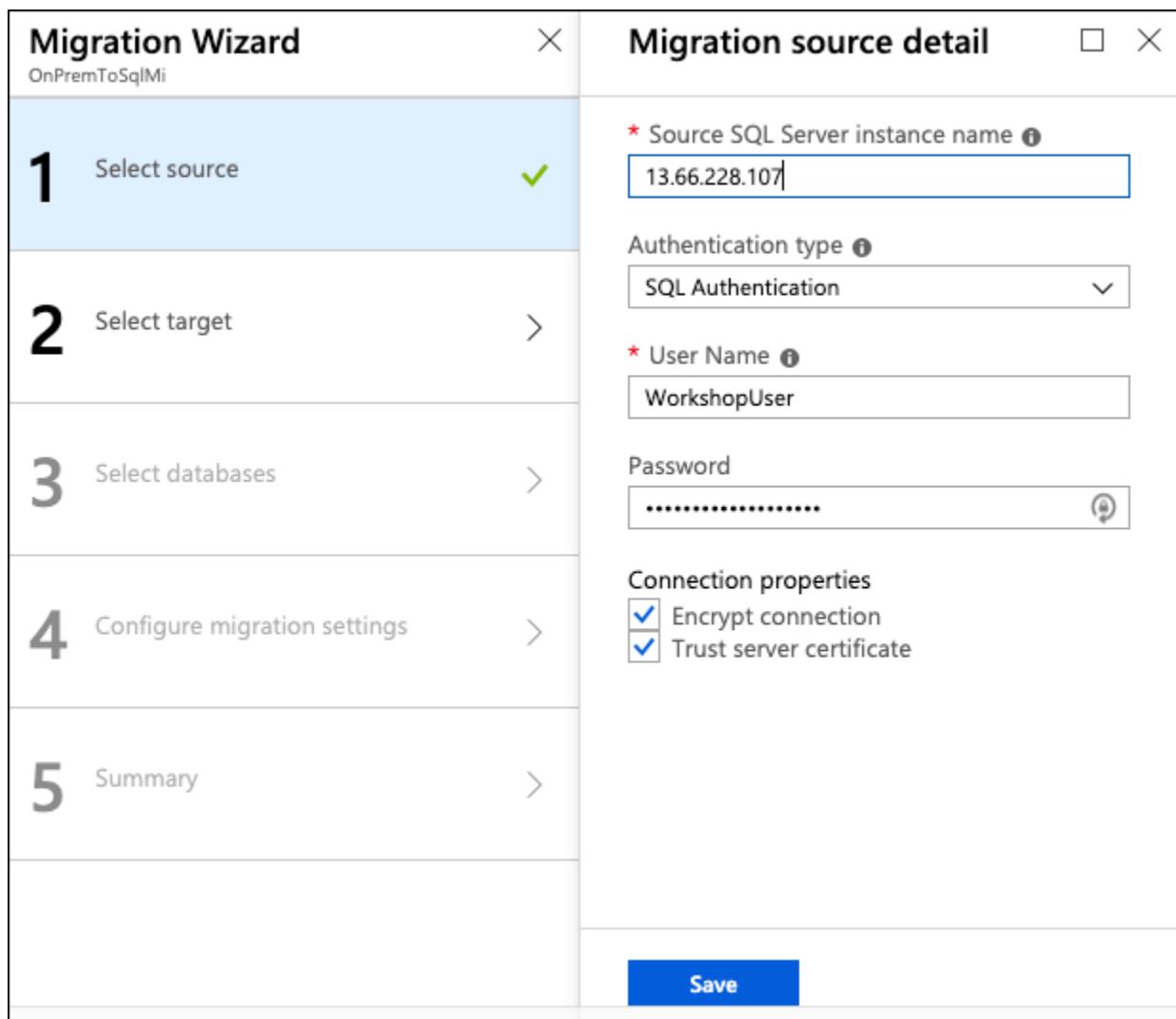
Note:

If you're doing this lab as part of a workshop and were provided an environment to use, please refer to instructor guidance to obtain your SQL Server VM's Public IP address.

- *User Name:* Enter **WorkshopUser**
- *Password:* Enter your password
- *Connection properties:* Check both **Encrypt connection** and **Trust server certificate**

Note:

The Password should be consistent among all labs. Your instructor will provide the password if you are taking this Lab in person.



6. Select **Save**

7. On the *Migration Wizard | Select target* blade, enter the following:

- *Application ID:* Enter the appId value from the output of the `az ad sp create-for-rbac` command you executed in the last task
- *Key:* Enter the password value from the output of the `az ad sp create-for-rbac` command you executed in the last task

Note:

If you're doing this lab as part of a workshop and were provided an environment to use, please refer to instructor guidance to obtain the Application ID and Application secret key.

- *Subscription:* Select the subscription you are using for this hand-on lab

- *Target Azure SQL Managed Instance:* Select the **mi-SUFFIX** instance

Note:

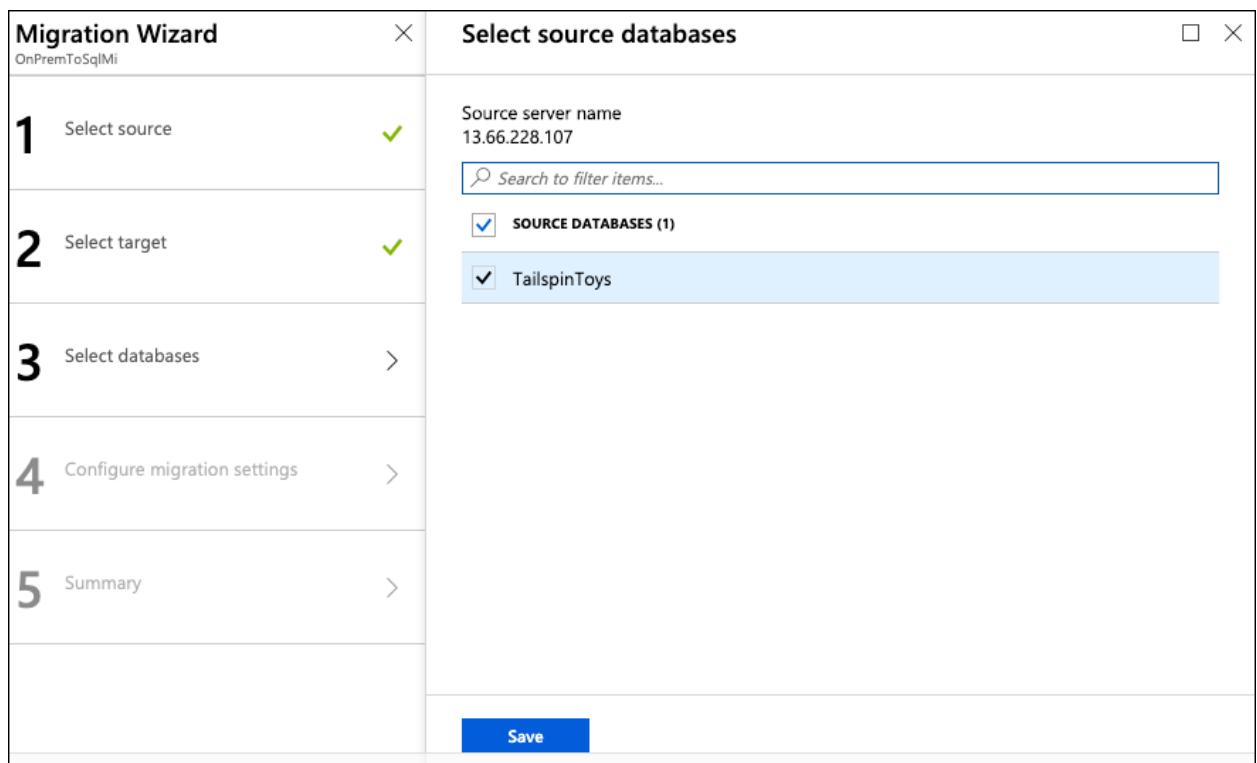
If you're doing this lab as part of a workshop and were provided an environment to use, the "SUFFIX" should be the same as the "SUFFIX" for your Resource Group. **Do not use other "SUFFIX" values or you will disrupt another attendees' environment.**

- *SQL Username:* Enter **sqlmiuser**
- *Password:* Enter your password

Migration Wizard		Migration target details
OnPremToSqlMi		
1	Select source	✓
2	Select target	>
3	Select databases	>
4	Configure migration settings	>
5	Summary	>
<div style="border: 1px solid #ccc; padding: 5px;"> <p>* Application ID that Azure Database Migration Service will use to call restore service ⓘ aeab3b83-9080-426c-94a3-4828db8532e9</p> <p>* Key ⚡ ✓</p> <p>Learn how to create</p> <p>* Select subscription containing the target Azure SQL Database Managed Instance server</p> <p>* Select target Azure SQL Database Managed Instance ⓘ sqlmi</p> <p>Learn how to create</p> <p>* SQL User Name ⓘ sqlmiuser ✓</p> <p>Password ⚡ ✓</p> <p>Save</p> </div>		

8. Select **Save**

9. On the Migration Wizard *Select databases* blade, select **TailspinToys**:



10. Select **Save**

11. On the *Migration Wizard | Configure migration settings* blade, enter the following configuration:

- *Network share location:* Enter **\SQLSERVER2008\dms-backups**. This is the path of the SMB network share you created during the before the hands-on lab exercises

Note:

If you're doing this lab as part of a workshop and were provided an environment to use, if *SQLSERVER2008* errors, use the **SQL VM Private IP Address** which was provided to you (e.g. *\10.0.0.5\dms-backups*)

- *Windows User Azure Database Migration Service impersonates to upload files to Azure Storage:* Enter **SQLSERVER2008\sqlmiuser**
- *Password:* Enter your password
- *Subscription containing storage account:* Select the subscription you are using for this hands-on lab
- *Storage account:* Select the **sqlmistoreUNIQUEID** storage account from the drop-down

Migration Wizard		X
OnPremToSqlMi		
1	Select source	✓
2	Select target	✓
3	Select databases	✓
4	Configure migration settings	>
5	Summary	>

Configure migration settings

Backup settings

! Ensure that the service account running the source SQL Server instance and the service account running the target SQL Server instance have read privileges on the network share that you provide.

* Network share location that Azure Database Migration Service will read backups from
\\SQLSERVER2008\dms-backups

! Make sure the Windows user has read access on the network share that you created above. The Azure Database Migration Service will impersonate the user credential to upload the backup files to Azure storage container for restore operation.

* Windows User Azure Database Migration Service impersonates to upload files to Azure Storage
SQLSERVER2008\sqlmiuser

Password
***** 

Storage account settings

* Select the subscription containing the desired storage account
Microsoft Azure Enterprise

i Select a Storage account created in location 'West US 2' and configured for standard performance tier that allows Azure Database Migration Service to upload database backup files to and use for migrating databases to a Azure SQL Database Managed Instance. Use this link to learn more about creating a Storage account

* Storage account that Azure Database Migration Service will upload the files to
sqlmistore

Advanced settings

Save

12. Select **Save** on the *Configure migration setting* blade

13. On the Migration Wizard *Summary* blade, enter the following:

- *Activity name:* Enter **TailspinToysMigration**

Migration Wizard		Migration summary
OnPremToSqlMi		
1	Select source	Activity name TailspinToysMigration
2	Select target	Target server name Target server version Azure SQL Database Managed Instance
3	Select databases	Source server name 13.66.228.107
4	Configure migration settings	Source server version SQL Server 2008 R2 10.50.6560.0
5	Summary >	Database(s) to migrate 1 of 1 Type of activity Online data migration
		Run migration

14. Select **Run migration**

15. Monitor the migration on the status screen that appears. Select the refresh icon in the toolbar to retrieve the latest status:

TailspinToysMigration			
<input type="button" value="Delete migration"/>	<input type="button" value="Stop migration"/>		
<input type="button" value="Refresh"/>	<input type="button" value="Retry"/>		
<input type="button" value="Download report"/>			
Source server : 13.77.159.115	Target server : sqlmi-semxwzksiqoo.15b8611394c5.database.windows.net		
Source version : 10.50.6560.0	Target version : Azure SQL Database Managed Instance		
SQL Server 2008 R2			
Databases : 1	Type of activity : Online		
Application ID : 26e033d5-cc55-4642-843b-3e8dcaaebe1	Activity status : Running		
<input type="text" value="Search"/> x			
1 item(s)			
DATABASE NAME	STATUS	DURATION	FINISH DATE
TailspinToys	Full backup uploading	00:00:01	---

16. Continue selecting **Refresh** every 5-10 seconds, until you see the status change to **Log files uploading** or **Log Shipping in progress** (when the log files are done uploading, it changes to **Log Shipping in progress**, but you don't need to wait for that to happen). When either status appears, move on to the next task:

DATABASE NAME	STATUS	DURATION	FINISH DATE
TailspinToys	Log files uploading	00:05:56	---



Activity 7: Perform migration cutover

Since you performed the migration as an "online data migration," the migration wizard will continue to monitor the SMB network share for newly added log files. This allows for any updates that happen on the source database to be captured until you cut over to the SQL MI database. In this task, you will add a record to one of the database tables, backup the logs, and complete the migration of the TailspinToys database by cutting over to the SQL MI database.

✓ Steps

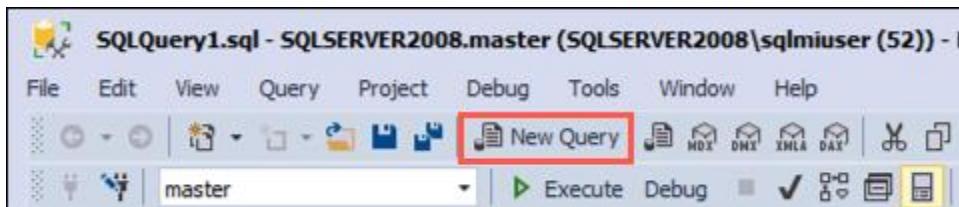
1. In the migration status window in the Azure portal and select **TailspinToys** under *database name* to view further details about the database migration:

DATABASE NAME	STATUS	DURATION	FINISH DATE
TailspinToys	Log files uploading	00:05:56	---

2. On the *TailspinToys* screen you will see a status of *Restored* for the *TailspinToys.bak* file:

TailspinToys						
Source server		Target server				
13.77.159.115		sqlmi-sewmwzksiqoo.15b8611394c5.database.windows.net		Database status		
Source version		Target version		Log files uploading		
10.50.6560.0		Azure SQL Database Managed Instance		Last applied LSN		
SQL Server 2008 R2		Full backup file(s)		177000000003200000		
		TailspinToys.bak		Last applied backup file(s)		
		TailspinToys.bak		Last applied backup file(s) taken on		
				5/13/2019, 12:19:04 PM		
<input type="button" value="Search"/> X						
1 item(s)						
ACTIVE BACKUP FILE(S)	TYPE	STATUS	BACKUP START TIME	BACKUP FINISH TIME		
TailspinToys.bak	Database	Restored	5/13/2019, 12:18:53 PM	5/13/2019, 12:19:04 PM		

3. To demonstrate log shipping and how transactions made on the source database during the migration process will be added to the target SQL MI database, you will add a record to one of the database tables.
4. Return to SSMS on your SqlServer2008 VM and select **New Query** from the toolbar:



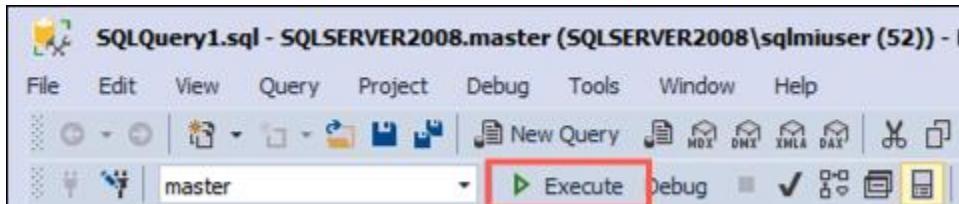
5. Paste the following SQL script, which inserts a record into the Game table, into the new query window:
6. USE TailspinToys;
7. GO
- 8.
9. INSERT [dbo].[Game] (Title, Description, Rating, IsOnlineMultiplayer)

VALUES ('Space Adventure', 'Explore the universe with are newest online'

multiplayer gaming experience. Build your own rocket ships, and take off for

the stars in an infinite open world adventure.', 'T', 1)

10. Execute the query by selecting **Execute** in the SSMS toolbar:



11. With the new record added to the Games table, you will now backup the transaction logs, which will be shipped to DMS. Select **New Query** again in the toolbar, and paste the following script into the new query window:
12. USE master;

```

13. GO
14.
15. BACKUP LOG TailspinToys
16. TO DISK = 'c:\dms-backups\TailspinToysLog.trn'
17. WITH CHECKSUM
    GO

```

18. Execute the query by selecting **Execute** in the SSMS toolbar:

19. Return to the migration status page in the Azure portal. On the TailspinToys screen, select **Refresh** you should see the **TailspinToysLog.trn** file appear, with a status of **Uploaded**:

TailspinToys						
Refresh		Start Cutover				
Source server	Target server	Database status	Last applied LSN			
13.77.159.15	sqlmi-sexmwxzksiqoo.15b8611394c5.database.windows.net	Log files uploading	179000000001600000			
Source version	Target version	Full backup file(s)	Last applied backup file(s)			
10.50.6560.0	Azure SQL Database Managed Instance	TailspinToys.bak	TailspinToys.bak			
SQL Server 2008 R2			Last applied backup file(s) taken on			
			5/13/2019, 4:20:03 PM			
<input type="text" value="Search"/>						
2 item(s)						
ACTIVE BACKUP FILE(S)	TYPE	STATUS	BACKUP START TIME	BACKUP FINISH TIME		
TailspinToysLog.trn	Transaction log	Uploaded	5/13/2019, 4:22:15 PM	5/13/2019, 4:22:15 PM		
TailspinToys.bak	Database	Restored	5/13/2019, 4:19:52 PM	5/13/2019, 4:20:03 PM		

Note: If you don't see it the transaction logs entry, continue selecting Refresh every few seconds until it appears.

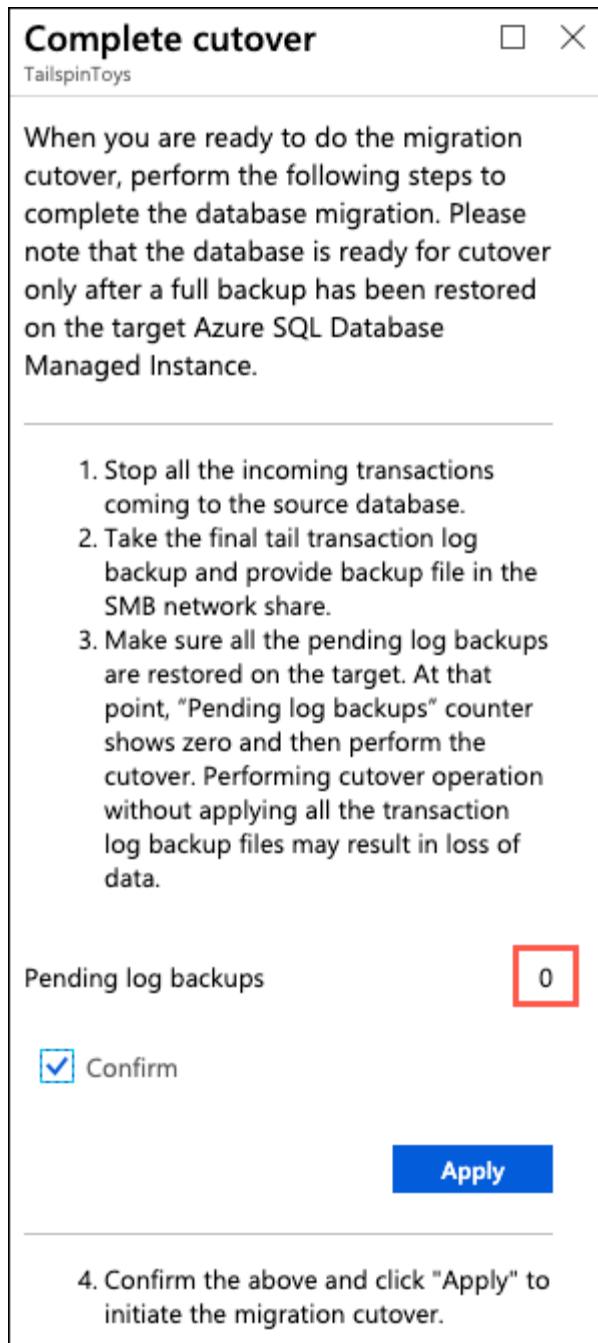
20. Once the transaction logs are uploaded, they need to be restored to the database. Select **Refresh** every 10-15 seconds until you see the status change to *Restored*, which can take a minute or two:

TailspinToys						
Refresh		Start Cutover				
Source server	Target server	Database status	Last applied LSN			
13.77.159.15	sqlmi-sexmwxzksiqoo.15b8611394c5.database.windows.net	Log files uploading	179000000034400000			
Source version	Target version	Full backup file(s)	Last applied backup file(s)			
10.50.6560.0	Azure SQL Database Managed Instance	TailspinToys.bak	TailspinToysLog.trn			
SQL Server 2008 R2			Last applied backup file(s) taken on			
			5/13/2019, 4:43:55 PM			
<input type="text" value="Search"/>						
2 item(s)						
ACTIVE BACKUP FILE(S)	TYPE	STATUS	BACKUP START TIME	BACKUP FINISH TIME		
TailspinToysLog.trn	Transaction log	Restored	5/13/2019, 4:43:55 PM	5/13/2019, 4:43:55 PM		
TailspinToys.bak	Database	Restored	5/13/2019, 4:19:52 PM	5/13/2019, 4:20:03 PM		

21. After verifying the transaction log status of *Restored*, select **Start Cutover**:



22. On the *Complete cutover* dialog, verify *pending log backups* is 0, check **Confirm**, and select **Apply**:



23. You will be given a progress bar below the *Apply* button in the *Complete cutover* dialog. When the migration is complete, you will see the status as *Completed*:

Complete cutover □ X

TailspinToys

When you are ready to do the migration cutover, perform the following steps to complete the database migration. Please note that the database is ready for cutover only after a full backup has been restored on the target Azure SQL Database Managed Instance.

1. Stop all the incoming transactions coming to the source database.

2. Take the final tail transaction log backup and provide backup file in the SMB network share.

3. Make sure all the pending log backups are restored on the target. At that point, "Pending log backups" counter shows zero and then perform the cutover. Performing cutover operation without applying all the transaction log backup files may result in loss of data.

Pending log backups 0

Confirm

Apply

4. Confirm the above and click "Apply" to initiate the migration cutover.

Completed

Note:

This will take between 10-30 minutes, so it might be a good time to take a break, or to review what you've done so far. Sometimes the progress bar is delayed, select **Refresh** from the panel to the left (don't worry, it won't close your cutover) to update manually, every ~5 minutes.

24. Close the *Complete cutover* dialog by selecting the **X** in the upper right corner of the dialog, and do the same thing for the *TailspinToys* blade. This will return you to the *TailspinToysMigration* blade. Select **Refresh**, and you should see a status of *Completed* from the *TailspinToys* database.

The screenshot shows the 'TailspinToysMigration' blade. At the top, there are buttons for 'Delete migration', 'Stop migration', 'Refresh', 'Retry', and 'Download report'. Below this, it displays migration details: Source server (13.77.159.115), Source version (10.50.6560.0, SQL Server 2008 R2), Target server (sqlmi-sexmwzksiqoo.15b8611394c5.database.windows.net), Target version (Azure SQL Database Managed Instance), Databases (1), Application ID (26e033d5-cc55-4642-843b-3e8dcaaeb1). It also shows Type of activity (Online) and Activity status (Succeeded). A search bar and a table below show one item: TailspinToys with Status 'Completed', Duration 00:39:50, and Finish Date 5/13/2019, 10:12:07 AM. Navigation links for 'prev', 'next', and 'Page 1 of 1' are at the bottom.

25. You have now successfully migrated the *TailspinToys* database to Azure SQL Managed Instance.



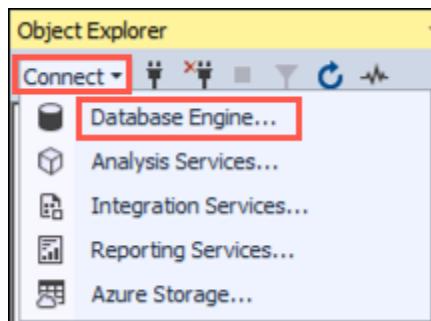
Activity 8: Verify database and transaction log migration

In this task, you will connect to the SQL MI database using SSMS, and quickly verify the migration.



Steps

1. Return to SSMS on your SqlServer2008 VM, and then select **Connect** and **Database Engine** from the *Object Explorer* menu:



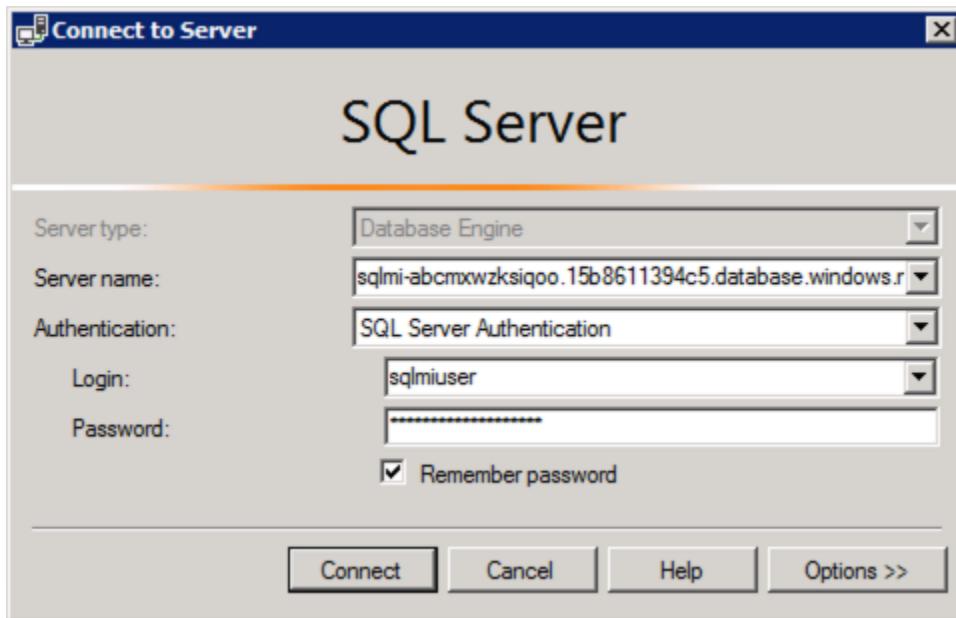
2. In the *Connect to Server* dialog, enter the following:

- o *Server name*: Enter the fully qualified domain name of your SQL managed instance, which you copied from the Azure Cloud Shell in a previous task

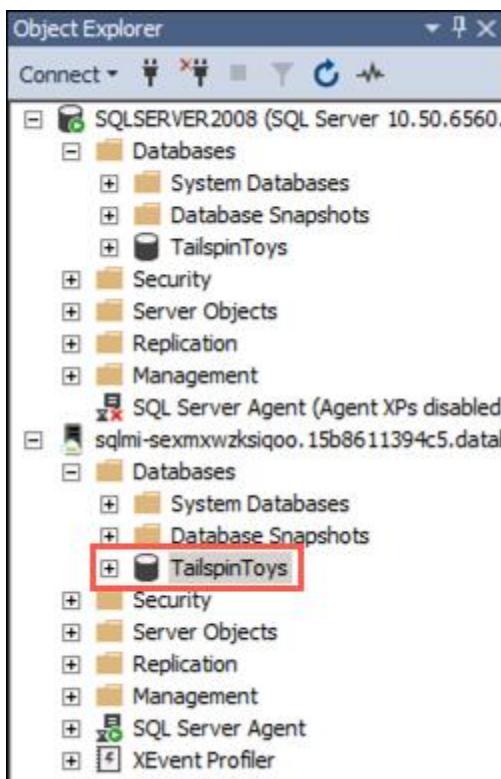
Note:

If you were provided an environment for this lab, the fully qualified domain name (also referenced as "MiFQDN" or "FQDN") has been provided to you in your environment details.

- *Authentication:* Select **SQL Server Authentication**
- *Login:* Enter **sqlmiuser**
- *Password:* Enter your password
- Check the **Remember password** box



3. Select **Connect**
4. You will see your SQL MI connection appear below the SQLSERVER2008 connection. Expand *Databases* in the SQL MI connection and select the TailspinToys database:



5. With the TailspinToys database selected, select **New Query** on the SSMS toolbar to open a new query window.
6. In the new query window, enter the following SQL script:

```
SELECT * FROM Game
```

7. Select **Execute** on the SSMS toolbar to run the query. You will see the records contained in the Game table displayed, including the new Space Adventure you added after initiating the migration process:

The screenshot shows the SSMS interface with three tabs at the top: 'SQLQuery18.sql - sql...oys (sqlmiuser (149))' (selected), 'SQLQuery17.sql - sql...oys (sqlmiuser (184))', and 'SQLQuery3.sql - SQL...'. Below the tabs is a toolbar with a magnifying glass icon. The main area has a 'Results' tab selected. A query 'SELECT * FROM Game' is entered in the text input field. The results grid displays four rows of data:

	Id	Title	Description	Rating	IsOnlineMultiplayer
1	1	Combat Fighter Pilot	Combat Figher Pilot is our number one selling gam...	T	1
2	2	Block Builder	Build your own interactive worlds and make your o...	E	1
3	3	Superheros vs. Supervillians	Battle one-on-one against other plavers in this ulti...	E-10	0
4	10	Space Adventure	Explore the universe with are newest online multipl...	T	1

8. You are now done using the SqlServer2008 VM. Close any open windows and log off of the VM. You will use the "JumpBox" VM for the remaining tasks of this hands-on-lab.



Activity 9: Update the application

With the `TailspinToys` database now running on SQL MI in Azure, the next step is to make the required modifications to the TailspinToys gamer information web application.

Note: SQL Managed Instance has a private IP address in its own VNet, so to connect an application you need to configure access to the VNet where Managed Instance is deployed. To learn more, read [Connect your application to Azure SQL Database Managed Instance](#).

Note: Due to time constraints, the lab will deal with switching the app running on a Jumpbox VM locally from leveraging data in SQL Server 2008 to SQL MI (but not the deployment to Azure or integrating the App Service with the Virtual Network). In the [extended version of these labs](#), or if you have time at the end of the lab, you can do that.

In this activity, you will create an RDP connection to the JumpBox VM, and then using Visual Studio on the JumpBox, run the `TailspinToysWeb` application on the VM.



Steps

Note:

If you were provided an environment for this lab, you may already be in the JumpBox VM. If you are, you can **skip to step 9**.

1. In the [Azure portal](#), select **Resource groups** in the *Azure navigation pane*, and select the **hands-on-lab-SUFFIX** resource group from the list:

The screenshot shows the Azure portal's 'Resource groups' blade. On the left, a sidebar menu includes 'Create a resource', 'Home', 'Dashboard', 'All services', 'FAVORITES', 'All resources', and 'Resource groups' (which is highlighted with a red box). Below these are 'App Services' and 'Virtual machines'. The main area is titled 'Resource groups' and shows a table with the following data:

	NAME	TYPE	LOCATION
<input type="checkbox"/>	hands-on-lab	Resource group	West US 2

2. In the list of resources for your resource group, select the JumpBox VM:

The screenshot shows the list of resources within the 'hands-on-lab' resource group. The table has columns for NAME, TYPE, and LOCATION. The 'JumpBox' VM is highlighted with a red box.

	NAME	TYPE	LOCATION
<input type="checkbox"/>	hands-on-lab-route-table	Route table	West US 2
<input type="checkbox"/>	hands-on-lab-vnet	Virtual network	West US 2
<input type="checkbox"/>	JumpBox	Virtual machine	West US 2
<input type="checkbox"/>	JumpBox_OsDisk_1_ebf408acaf0b499db112d2...	Disk	West US 2
<input type="checkbox"/>	jumpbox944	Network interface	West US 2
<input type="checkbox"/>	JumpBox-ip	Public IP address	West US 2
<input type="checkbox"/>	JumpBox-nsg	Network security group	West US 2
<input type="checkbox"/>	sqlmi-nsg	Network security group	West US 2

3. On your JumpBox VM blade, select **Connect** from the top menu:



4. On the *Connect to virtual machine* blade, select **Download RDP File**, then open the downloaded RDP file:

RDP SSH

To connect to your virtual machine via RDP, select an IP address, optionally change the port number, and download the RDP file.

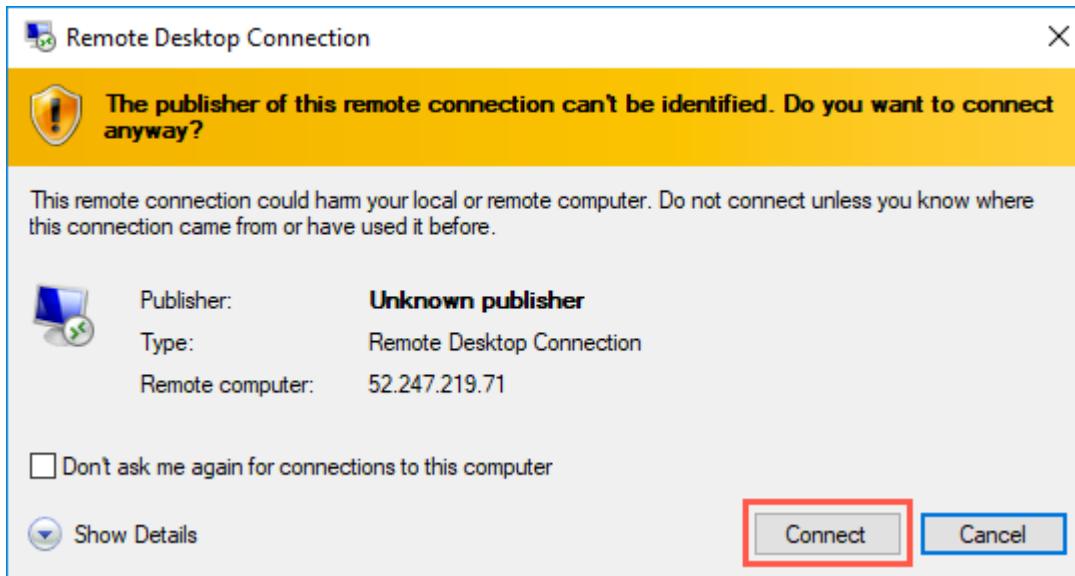
o

* IP address
Public IP address (52.247.219.71)

* Port number
3389

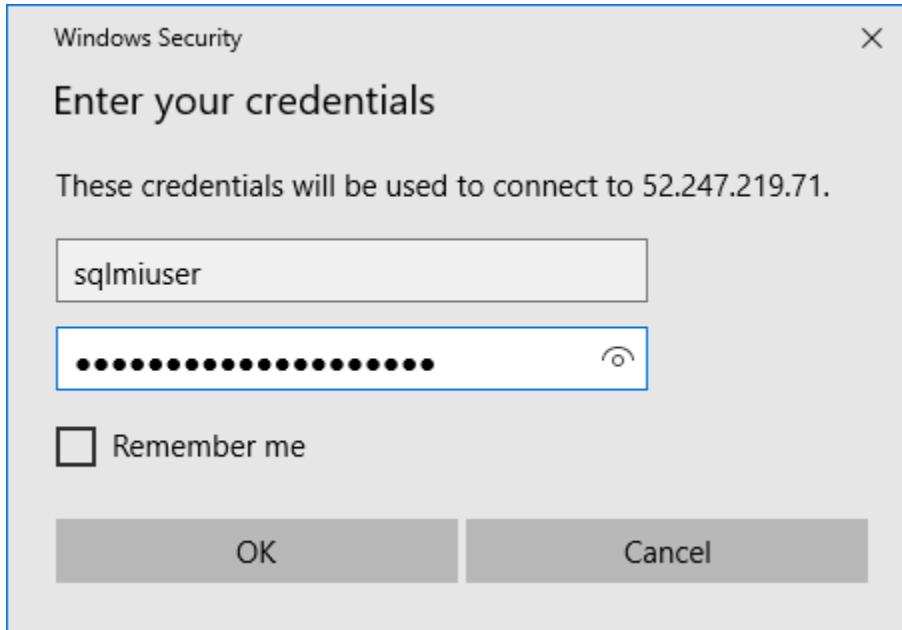
Download RDP File

5. Select **Connect** on the *Remote Desktop Connection* dialog:

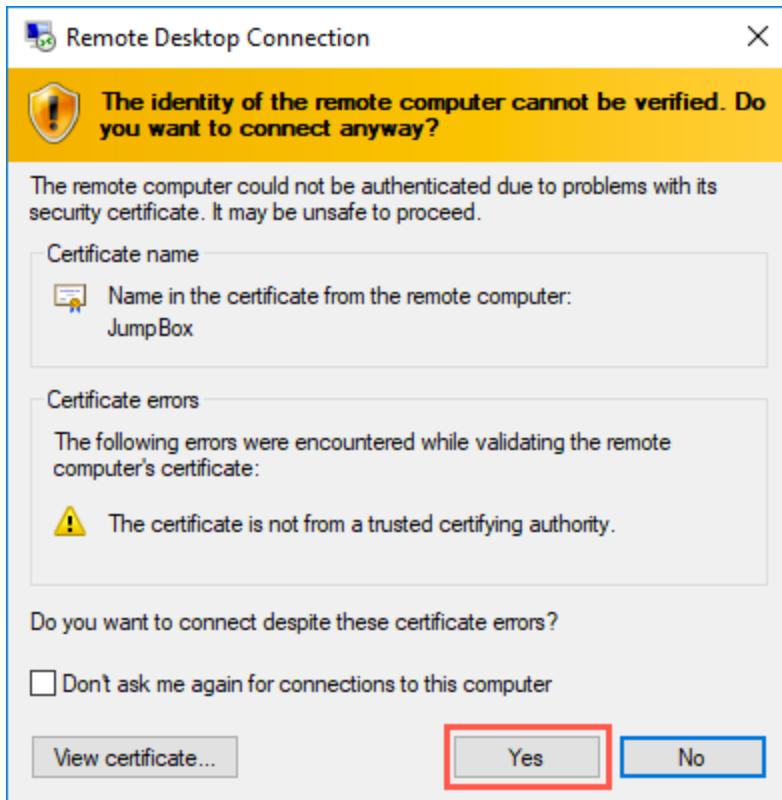


6. Enter the following credentials when prompted, and then select **OK**:

- o *Username: sqlmiuser*



7. Select **Yes** to connect, if prompted that the identity of the remote computer cannot be verified:



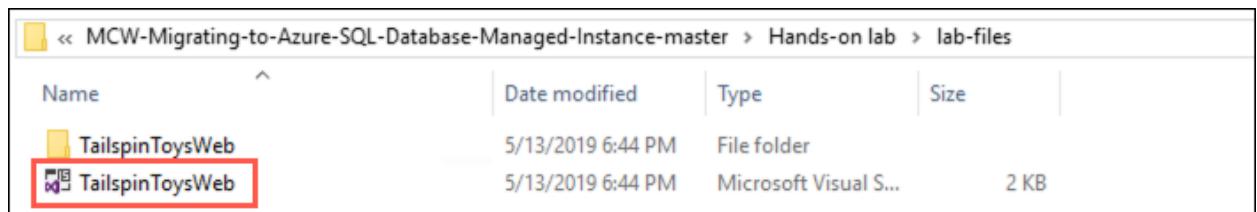
- Once logged in, the repository containing all of these files can be cloned to C:/users/[username]/sqlworkshops/SQLGroundtoCloud by opening the command prompt and running the following command:

```
git clone https://github.com/microsoft/sqlworkshops.git
```

Note:

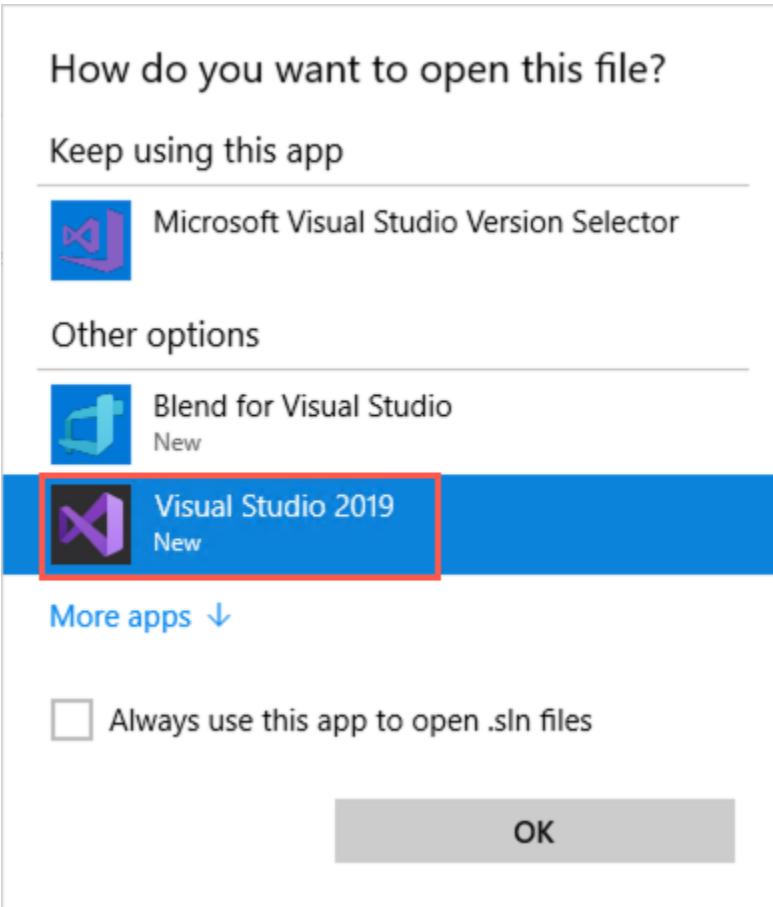
If you're doing this lab as part of a workshop and were provided an environment to use, or you have already cloned the repository earlier in the workshop, you do not need to clone the workshop again.

- Open the c:/users/[username]/sqlworkshops/SQLGroundtoCloud folder. In the lab-files folder, double-click TailspinToysWeb.sln to open the solution in Visual Studio:



Name	Date modified	Type	Size
TailspinToysWeb	5/13/2019 6:44 PM	File folder	
TailspinToysWeb	5/13/2019 6:44 PM	Microsoft Visual S...	2 KB

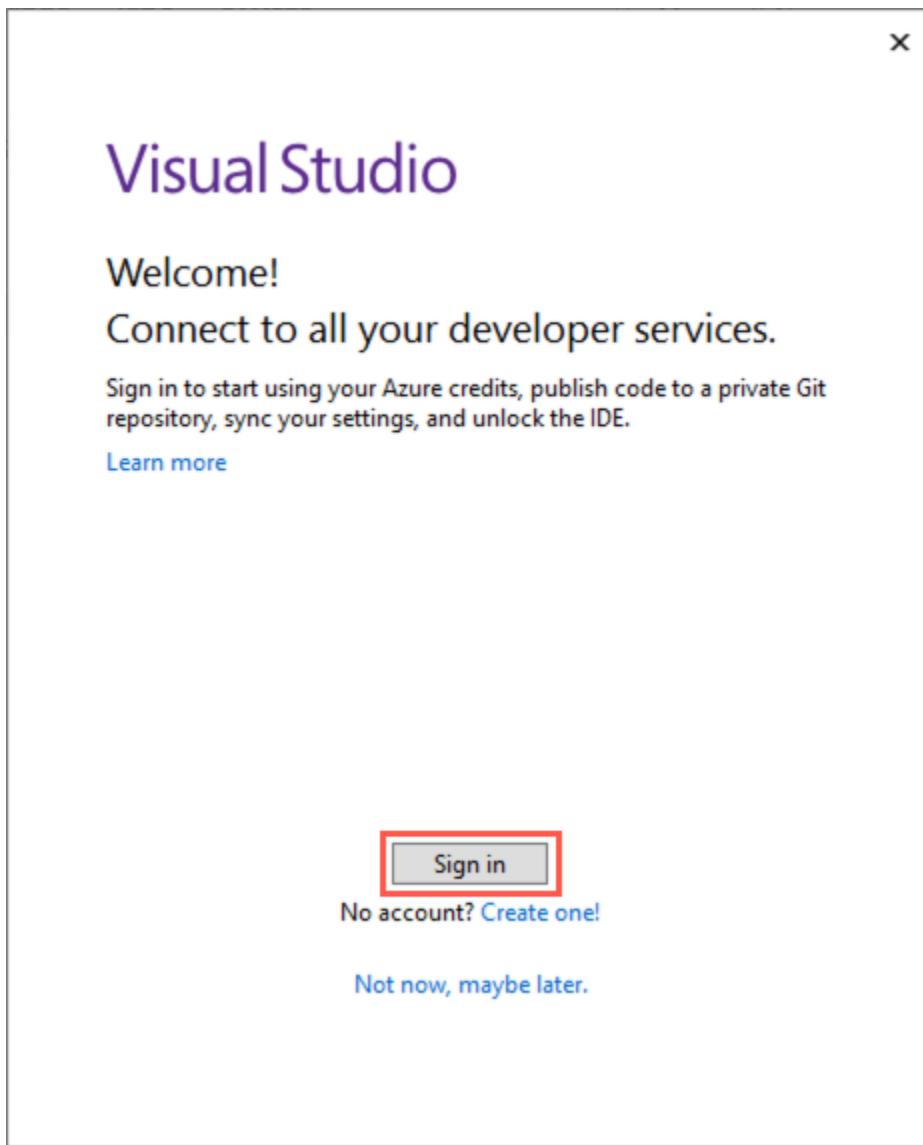
- If prompted about how you want to open the file, select **Visual Studio 2019** and then select **OK**:



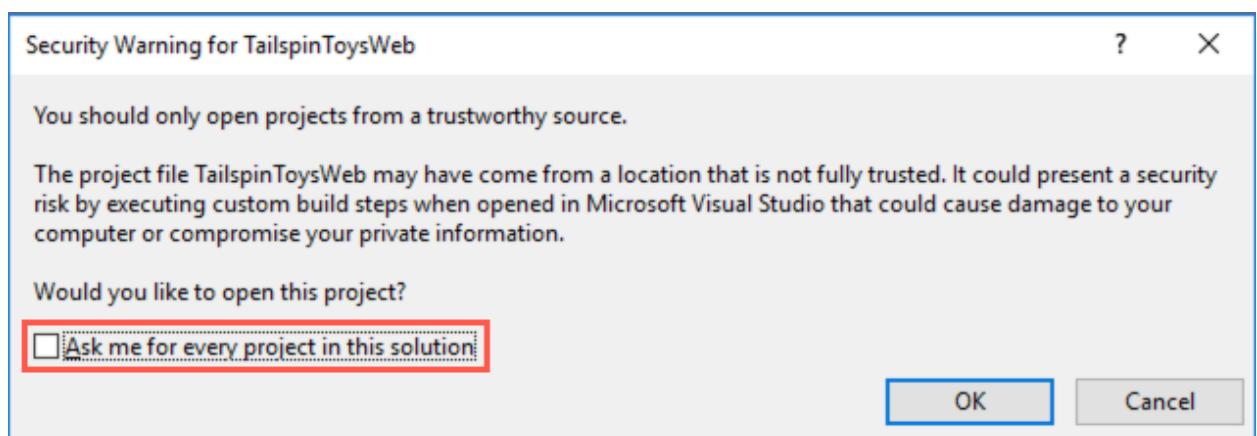
11. Select **Sign in** and enter your Azure account credentials when prompted:

Note:

If you're doing this lab as part of a workshop and were provided an environment to use, please use the Azure account credentials provided to you. If you are not prompted, you can skip to the next step for now.



12. At the security warning prompt, uncheck Ask me for every project in this solution, and then select **OK**:



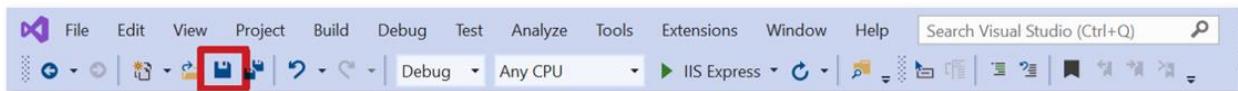
13. Open appsettings.json and enter your SQL 2008 VM information and password in the Connection strings section:

```
"ConnectionStrings": {  
    "TailspinToysContext": "Server=tcp:<your-sql-2008-vm-public-ip>,1433;Database=TailspinToys;User ID=WorkshopUser;Password=<your-password>;Trusted_Connection=False;Encrypt=True;TrustServerCertificate=True;" ,  
    "TailspinToysReadOnlyContext": "Server=tcp:<your-sql-2008-vm-public-ip>,1433;Database=TailspinToys;User ID=WorkshopUser;Password=<your-password>;Trusted_Connection=False;Encrypt=True;TrustServerCertificate=True;"  
}
```

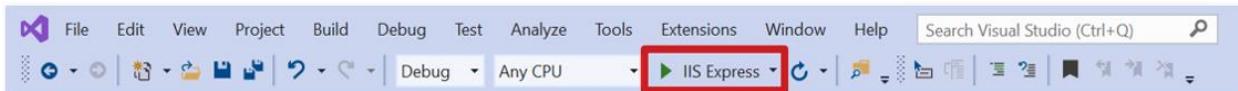
Note:

Use the same login "WorkshopUser" and password from previous activities.

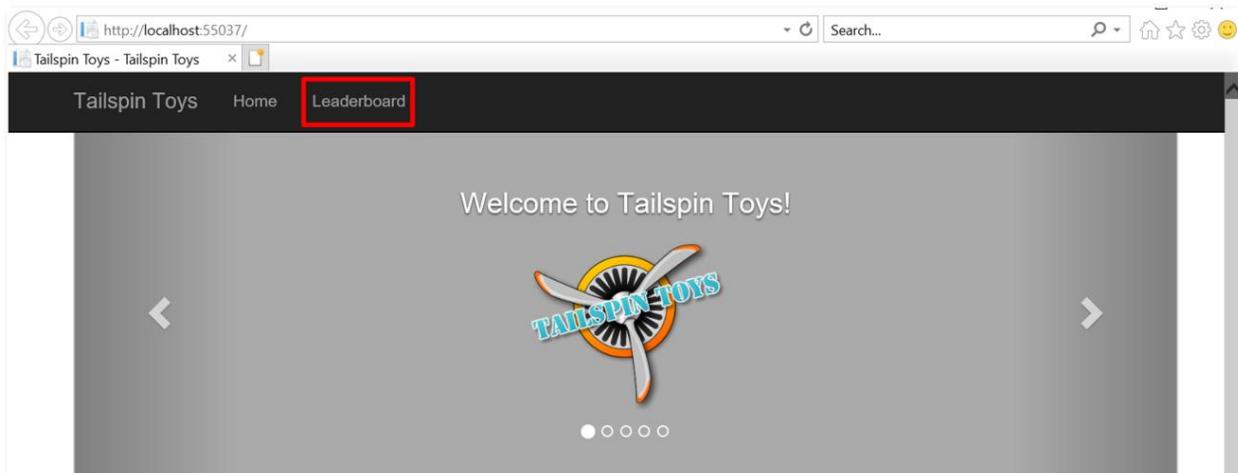
14. Save the file:



15. Run the application (IIS Express button):



16. You should now see the app running locally and view the site and it's data which is accessing the on-prem data (Select Leaderboard):



Tailspin Toys: The best in online gaming!

Tailspin Toys is the developer of several popular online video games. Founded in 2010, we continue to innovate in online multiplayer gameplay, bringing you the best games on the market!

[About Us](#)

[Investor Relations](#)

[Careers](#)

[Contact Us](#)

Position	Gamer Tag	Name	Score	Diff	Games Played	Online
1	abel978	Andrea Bell	232723	0	2621	
2	ddiaz2679	Destiny Diaz	232705	-18	3306	✓
3	acooper11383	Alex Cooper	232704	-19	3606	
4	kpal15499	Kelli Pal	232702	-21	2204	✓
5	cturner9405	Charles Turner	232686	-37	3246	✓
6	ahenderson3918	Anna Henderson	232664	-59	3593	✓
7	kbrooks1613	Kimberly Brooks	232654	-69	3610	

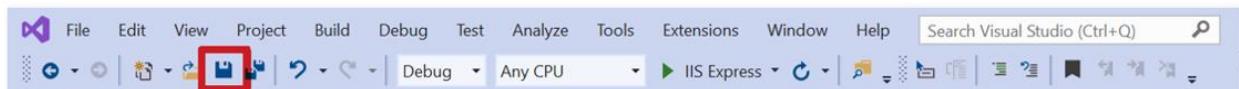
17. Stop the application by closing the browser.

18. Now, in order to have the app run with the data in SQL MI, update `appsettings.json` by replacing `Server=tcp:<your-sql-2008-vm-public-ip>,1433;` with the fully qualified domain name for your MI (something like `Server=mi-SUFFIX.fdsor39943234j3oj4.database.windows.net;`). Then, replace `WorkshopUser` with `sqlmiuser`, as that's how you log into your SQL MI.

Note:

If your passwords for `WorkshopUser` and `sqlmiuser` are different, don't forget to update to the correct password as well.

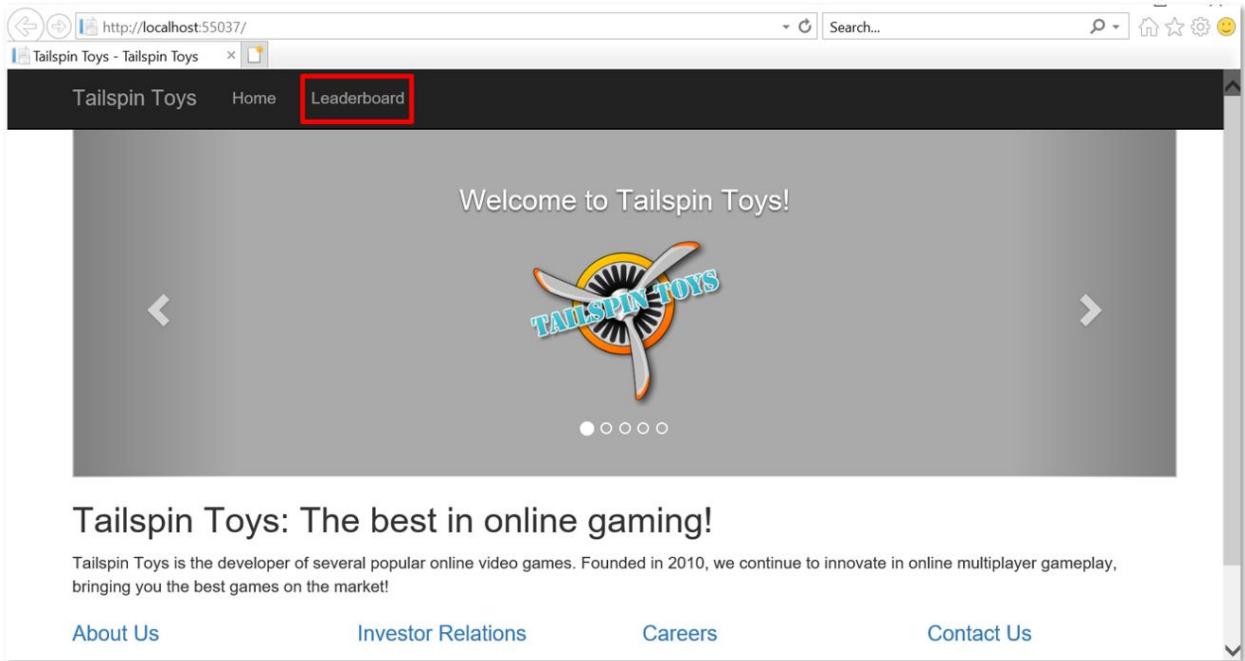
19. Save the file:



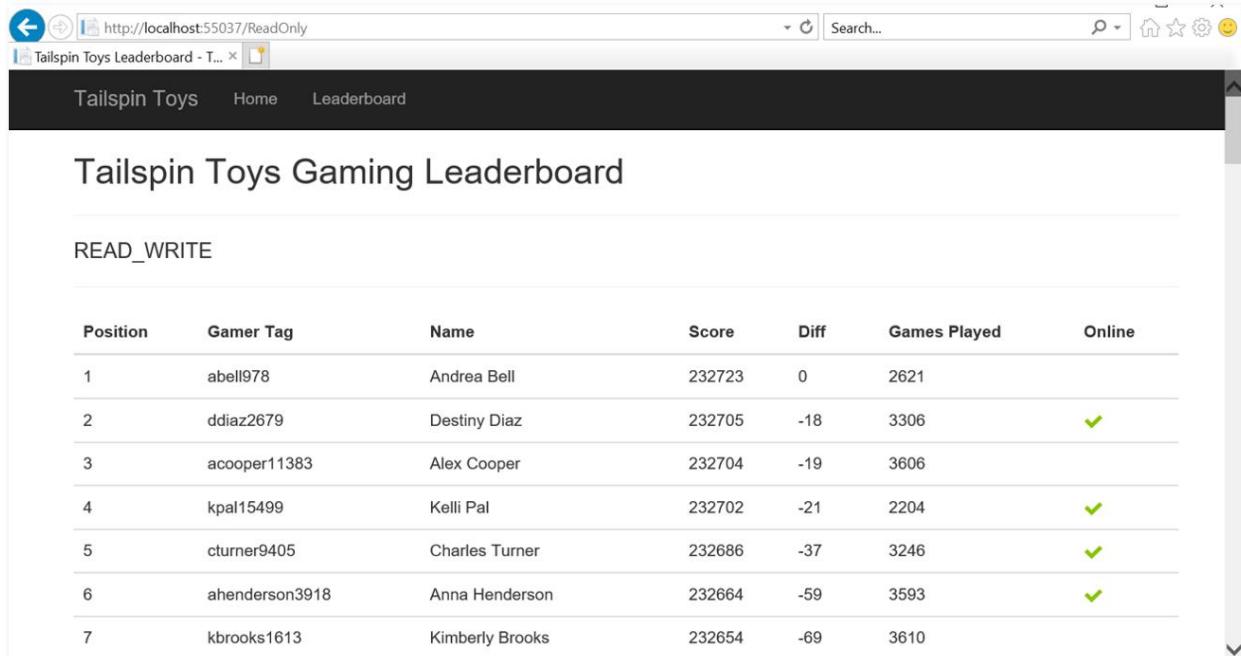
20. Run the application (IIS Express button):



21. You should see the same results as before, but this time, the data is coming from your SQL MI in Azure:



The screenshot shows a web browser window for 'Tailspin Toys - Tailspin Toys' at 'http://localhost:55037/'. The 'Leaderboard' tab is highlighted with a red box. The main content area displays the 'Welcome to Tailspin Toys!' message, the Tailspin Toys logo, and navigation arrows. Below this, there's a section titled 'Tailspin Toys: The best in online gaming!' with a brief description and links to 'About Us', 'Investor Relations', 'Careers', and 'Contact Us'.



The screenshot shows a web browser window for 'Tailspin Toys Leaderboard - T...' at 'http://localhost:55037/ReadOnly'. The 'Leaderboard' tab is selected. The main content area displays the 'Tailspin Toys Gaming Leaderboard' title and a table titled 'READ_WRITE' with the following data:

Position	Gamer Tag	Name	Score	Diff	Games Played	Online
1	abel978	Andrea Bell	232723	0	2621	
2	ddiaz2679	Destiny Diaz	232705	-18	3306	✓
3	acooper11383	Alex Cooper	232704	-19	3606	
4	kpal15499	Kelli Pal	232702	-21	2204	✓
5	cturner9405	Charles Turner	232686	-37	3246	✓
6	ahenderson3918	Anna Henderson	232664	-59	3593	✓
7	kbrooks1613	Kimberly Brooks	232654	-69	3610	

Note: If you want to complete an extension of this lab where you deploy the web app to Azure and integrate the App Service within the virtual network using point-to-site and VNet integration, see exercises 3 and 4 in the non-abbreviated lab [here](#).



5.2 Improve database security with Advanced Data Security (*Bonus*)

[Advanced Data Security](#) is a unified package for advanced SQL security capabilities. It includes functionality for discovering and classifying sensitive data, surfacing and mitigating potential database vulnerabilities, and detecting anomalous activities that could indicate a threat to your database. It provides a single go-to location for enabling and managing these capabilities.

In this exercise, you'll enable Advanced Data Security, configure Data Discovery and Classification, and review the Vulnerability Assessment. At the end, you'll also receive a pointer to a Dynamic Data Masking lab extension.



[Activities](#)

Complete activities 1-3 by following the links below. Once you're in Activity 1, you can continue to the next activity through Activity 3 (the links below all point to different parts of the same overall lab). When are you done with activities 1-3, come back here for the next section (5.3).

[Activity 1](#): Enable Advanced Data Security

[Activity 2](#): Configure SQL Data Discovery and Classification

[Activity 3](#): Review Advanced Data Security Vulnerability Assessment



Lab Exercise 2: Improve database security with Advanced Data Security

[Advanced Data Security](#) is a unified package for advanced SQL security capabilities. It includes functionality for discovering and classifying sensitive data, surfacing and mitigating potential database vulnerabilities, and detecting anomalous activities that could indicate a threat to your database. It provides a single go-to location for enabling and managing these capabilities.

In this exercise, you'll enable Advanced Data Security, configure Data Discovery and Classification, and review the Vulnerability Assessment. At the end, you'll also receive a pointer to a Dynamic Data Masking lab extension.



Activity 1: Enable Advanced Data Security

In this task, you will enable Advanced Data Security (ADS) for all databases on the Managed Instance.



Steps

1. In the [Azure portal](#), select **Resource groups** from the left-hand menu, select the **hands-on-lab-SUFFIX** resource group, and then select the **TailspinToys** Managed database resource from the list:

	NAME ↑↓	TYPE ↑↓	LOCATION ↑↓
<input type="checkbox"/>	NIC-dsnargfxg5ny4cdyvxu4pp7e	Network interface	West US 2
<input type="checkbox"/>	sqlmi-nsg	Network security group	West US 2
<input type="checkbox"/>	sqlmi	SQL managed instance	West US 2
<input type="checkbox"/>	TailspinToys (sqlmi/TailspinToys)	Managed database	West US 2
<input type="checkbox"/>	sqlmistore	Storage account	West US 2
<input type="checkbox"/>	SqlServer2008	Virtual machine	West US 2

2. On the *TailspinToys Managed database* blade, select **Advanced Data Security** from the left-hand menu, under Security, and then select **Enable Advanced Data Security on the managed instance**:

The screenshot shows the 'TailspinToys - Advanced Data Security' blade in the Azure portal. On the left, there's a sidebar with links like 'Overview', 'Activity log', 'Diagnose and solve problems', 'Settings' (with 'Locks' and 'Export template'), 'Security', and 'Advanced Data Security' (which is highlighted with a red box). The main area has a large 'i' icon with text about enabling ADS for all databases. Below it is a button 'Enable Advanced Data Security on the managed instance' (also highlighted with a red box). There are three tiles: 'Data Discovery & Classification (preview)' (0 TOTAL), 'Vulnerability Assessment' (27 TOTAL, with a pie chart showing failure distribution), and 'Advanced Threat Protection' (0 TOTAL).

- Within a few minutes, ADS will be enabled for all databases on the Managed Instance. You will see the three tiles on the *Advanced Data Security* blade become enabled when it has been enabled:

The screenshot shows three tiles from the 'Advanced Data Security' blade:

- Data Discovery & Classification (preview)**: Shows 0 TOTAL. Below it is a table of recommended columns to classify:

COLUMN	SENSITIVITY LABEL
AddressLine1	Confidential
AddressLine2	Confidential
City	Confidential

- Vulnerability Assessment**: Shows 27 TOTAL. It includes a pie chart and a table of failed checks:

SECURITY CHECK	RISK
The dbo information should be consisten...	High
Database communication using TDS shou...	High
Execute permissions to access the registr...	High

- Advanced Threat Protection**: Shows 0 TOTAL. It includes a table of security alerts:

DESCRIPTION	DATE
There are no alerts or recommendations to display.	



Activity 2: Configure SQL Data Discovery and Classification

In this task, you will look at the [SQL Data Discovery and Classification](#) feature of Advanced Data Security. Data Discovery and Classification introduces a new tool for discovering, classifying, labeling & reporting the sensitive data in your databases. It introduces a set of advanced services, forming a new SQL Information Protection paradigm aimed at protecting the data in your database, not just the database.

Discovering and classifying your most sensitive data (business, financial, healthcare, etc.) can play a pivotal role in your organizational information protection stature.

Note: This functionality is currently available *in Preview* for SQL MI through the Azure portal.

✓ Steps

1. On the *Advanced Data Security* blade, select the **Data Discovery & Classification** tile:

The screenshot shows the 'Data Discovery & Classification (preview)' blade. At the top, there is a circular progress bar with the number '0' and the word 'TOTAL' below it. Below the progress bar, the heading 'Recommended columns to classify' is displayed. A table follows, listing three columns: 'AddressLine1', 'AddressLine2', and 'City', each assigned the 'Confidential' sensitivity label. An ellipsis '...' is shown at the bottom right of the table.

COLUMN	SENSITIVITY LABEL
AddressLine1	Confidential
AddressLine2	Confidential
City	Confidential
...	

2. In the *Data Discovery & Classification* blade, select the **info link** with the message *We have found 40 columns with classification recommendations*:

The screenshot shows the 'Data Discovery & Classification (preview)' blade. At the top, there are three buttons: 'Export', 'Configure', and 'Feedback'. Below them, an info link is highlighted with a red box, containing the text 'We have found 40 columns with classification recommendations →'. The rest of the blade is visible but not highlighted.

- Look over the list of recommendations to get a better understanding of the types of data and classifications assigned, based on the built-in classification settings. In the list of *classification recommendations*, select the recommendation for the **Sales - CreditCard - CardNumber** field:

The screenshot shows the 'Classification' blade with the following interface elements:

- Overview** and **Classification** tabs.
- A message bar at the top stating "40 columns with classification recommendations (Click to minimize)".
- Buttons: "Accept selected recommendations", "Dismiss selected recommendations", and "Show dismissed recommendations".
- A table listing recommendations. The row for "Sales - CreditCard - CardNumber" is highlighted with a red border and has a checked checkbox in the first column.
- Column headers: Sales, CountryRegionCurrency, CurrencyCode, Financial, Confidential.
- Row details: Sales, CreditCard, CreditCardID, Credit Card, Confidential; Sales, CreditCard, CardType, Credit Card, Confidential; Sales, CreditCard, CardNumber, Credit Card, Confidential; Sales, CreditCard, ExpYear, Credit Card, Confidential.

- Due to the risk of exposing credit card information, Tailspin Toys would like a way to classify it as *highly confidential*, not just *Confidential*, as the recommendation suggests. To correct this, select **+ Add classification** at the top of the *Data Discovery and Classification* blade:



- Expand the **Sensitivity label** field, and review the various built-in labels you can choose from. You can also add your own labels, should you desire:

The screenshot shows a dropdown menu titled "Sensitivity label" with the following options:

- [n/a]
- Public
- General
- Confidential
- Confidential - GDPR
- Highly Confidential
- Highly Confidential - GDPR

- In the *Add classification* dialog, enter the following:
 - Schema name*: Select **Sales**
 - Table name*: Select **CreditCard**
 - Column name*: Select **CardNumber (nvarchar)**
 - Information type*: Select **Credit Card**

- Sensitivity level: Select **Highly Confidential**

Add classification

* Schema name
Sales

* Table name
CreditCard

* Column name
CardNumber (nvarchar)

Information type
Credit Card

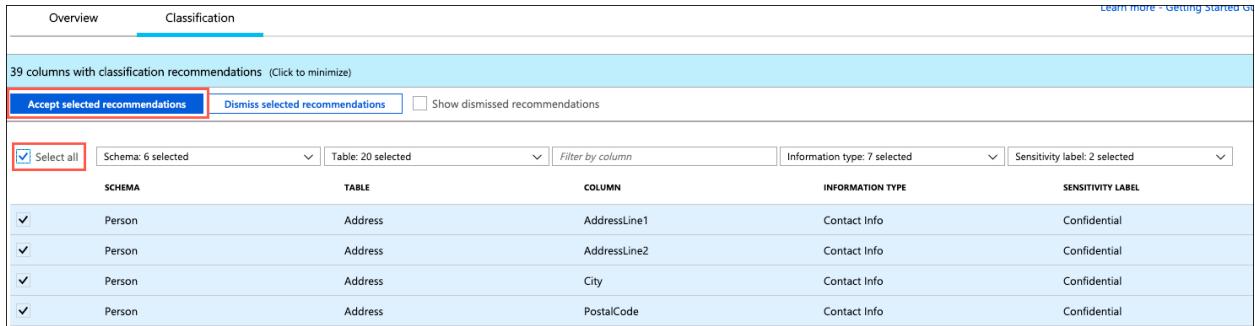
Sensitivity label
Highly Confidential

Add classification **Cancel**

- Select **Add classification**
- You will see the *Sales - CreditCard - CardNumber* field disappear from the recommendations list, and the number of recommendations drop by 1
- Other recommendations you can review are the *HumanResources - Employee* fields for *NationalIDNumber* and *BirthDate*. Note that these have been flagged by the recommendation service as *Confidential - GDPR*. As Tailspin Toys maintains data about gamers from around the world, including Europe, having a tool which helps them discover data which may be relevant to GDPR compliance will be very helpful:

Person	EmailAddress	EmailAddress	Contact Info	Confidential
HumanResources	Employee	NationalIDNumber	National ID	Confidential - GDPR
HumanResources	Employee	BirthDate	Date Of Birth	Confidential - GDPR
Sales	SalesOrderHeader	AccountNumber	Financial	Confidential

- Check the **Select all** check box at the top of the list to select all the remaining recommended classifications, and then select **Accept selected recommendations**:



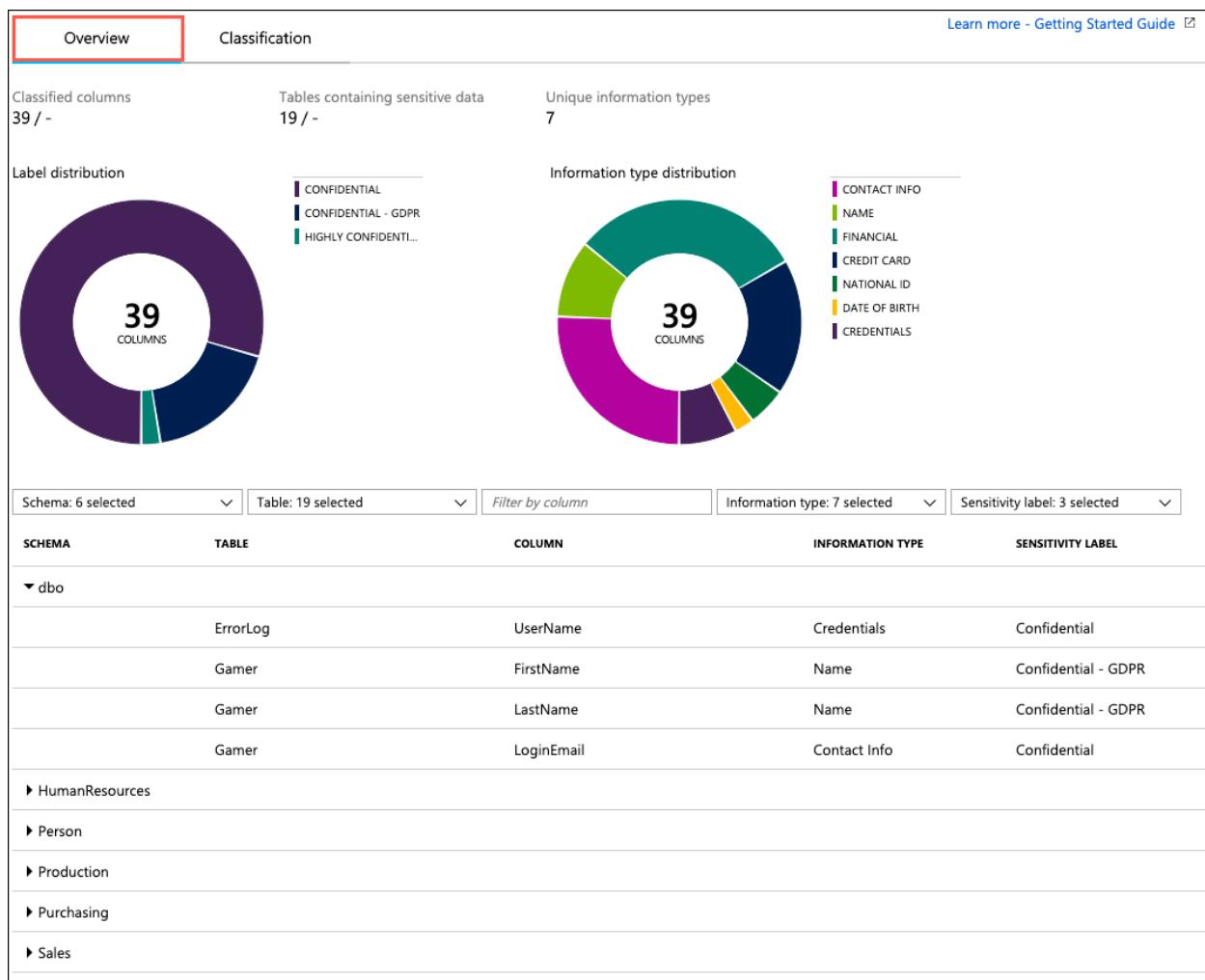
Classification						Learn more - Getting Started
39 columns with classification recommendations (Click to minimize)						
<input type="button" value="Accept selected recommendations"/>		<input type="button" value="Dismiss selected recommendations"/>		<input type="checkbox"/> Show dismissed recommendations		
<input checked="" type="checkbox"/> Select all	Schema: 6 selected	Table: 20 selected	Filter by column	Information type: 7 selected	Sensitivity label: 2 selected	
SCHEMA	TABLE	COLUMN	INFORMATION TYPE	SENSITIVITY LABEL		
✓ Person	Address	AddressLine1	Contact Info	Confidential		
✓ Person	Address	AddressLine2	Contact Info	Confidential		
✓ Person	Address	City	Contact Info	Confidential		
✓ Person	Address	PostalCode	Contact Info	Confidential		

11. Select **Save** on the toolbar of the Data Classification window. It may take several minutes for the save to complete:



Note: This feature is still in preview. If you receive an error when saving, try returning to the Advanced Data Security blade, and selecting the Data Discovery & Classification tile again to see the results.

12. When the save completes, select the **Overview** tab on the *Data Discovery and Classification* blade to view a report with a full summary of the database classification state:



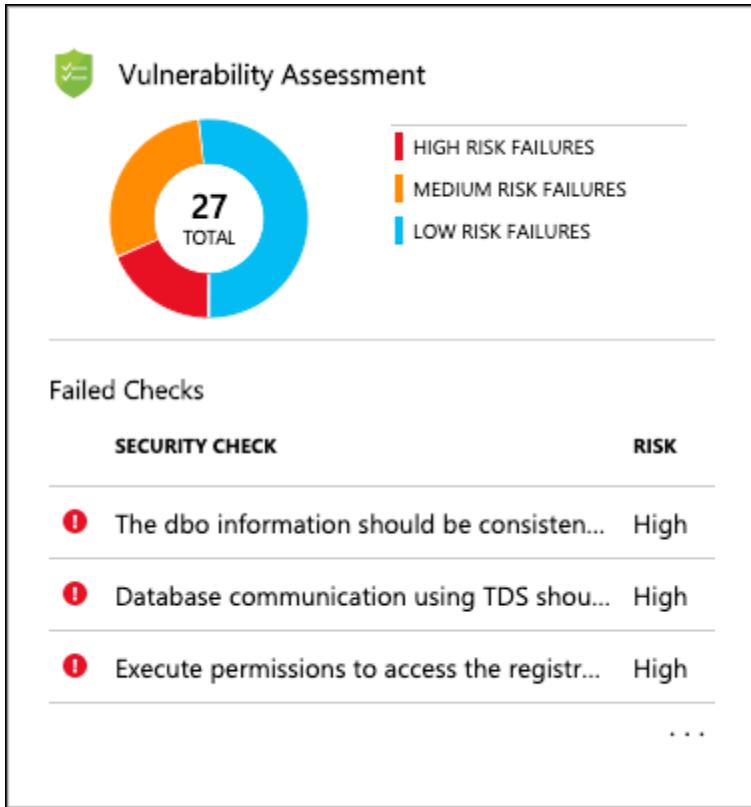
Activity 3: Review Advanced Data Security Vulnerability Assessment

In this task, you will review an assessment report generated by ADS for the TailspinToys database and take action to remediate one of the findings in the TailspinToys database. The [SQL Vulnerability Assessment service](#) is a service that provides visibility into your security state, and includes actionable steps to resolve security issues, and enhance your database security.

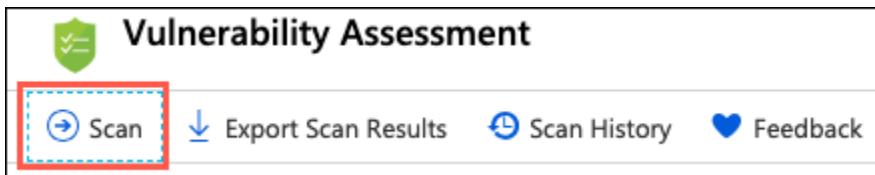


Steps

1. Return to the **Advanced Data Security** blade for the TailspinToys Managed database and then select the **Vulnerability Assessment** tile:



2. On the *Vulnerability Assessment* blade, select **Scan** on the toolbar:



3. When the scan completes, you will see a dashboard, displaying the number of failing checks, passing checks, and a breakdown of the risk summary by severity level:

Vulnerability Assessment

Scan Export Scan Results Scan History Feedback

Total failing checks: 18 Failed (18) Total passing checks: 102 Passed (102)

Risk summary:

- High Risk: 3
- Medium Risk: 6
- Low Risk: 9

Last scan time: Mon, 11 Mar 2019 16:11:25 UTC

ID	SECURITY CHECK	APPLIES TO	CATEGORY	RISK
VA1220	Database communication using TDS should be protected through TLS	master	Data protection	High
VA2110	Execute permissions to access the registry should be revoked	master	Authentication & Authorization	High
VA2114	Minimal set of principals should be members of high impact fixed server roles	master	Authentication & Authorization	High
VA1143	'dbo' user should not be used for normal service operation	ContosoAutoDb	Surface area reduction	Medium
VA1219	Transparent data encryption should be enabled	ContosoAutoDb	Data protection	Medium

Note: Scans are run on a schedule, so if you see a message that no vulnerabilities are found your database may not have been scanned yet. You will need to run a scan manually. To do this, select the **Scan** button on the toolbar, and follow any prompts to start a scan. This will take a minute or so to complete.

- In the scan results, take a few minutes to browse both the *Failed* and *Passed* checks, and review the types of checks that are performed. In the *Failed* list, locate the security check for *Transparent data encryption*. This check has an ID of VA1219:

Failed (18) Passed (102)

Filter by ID or security check Category: All selected Risk: All selected

ID	SECURITY CHECK	APPLIES TO	CATEGORY	RISK
VA1220	Database communication using TDS should be protected through TLS	master	Data protection	High
VA2110	Execute permissions to access the registry should be revoked	master	Authentication & Authoriza...	High
VA2114	Minimal set of principals should be members of high impact fixed server roles	master	Authentication & Authoriza...	High
VA1143	'dbo' user should not be used for normal service operation	ContosoAutoDb	Surface area reduction	Medium
VA1219	Transparent data encryption should be enabled	ContosoAutoDb	Data protection	Medium
VA1276	Agent XPs feature should be disabled	master	Auditing & Logging	Medium

- Select the **VA1219** finding to view the detailed description:

VA1219 - Transparent data encryption should be enabled	
<input checked="" type="checkbox"/> Approve as Baseline	<input type="checkbox"/> Clear Baseline
name	VA1219 - Transparent data encryption should be enabled
risk	Medium
status	✖ FAIL
Applies To	ContosoAutoDb
description	Transparent data encryption (TDE) helps protect against the threat of malicious activity by performing real-time encryption and decryption of the database, associated backups, and transaction log files 'at rest', without requiring changes to the application. This rule checks that TDE is enabled on the database.
impact	Transparent Data Encryption (TDE) protects data 'at rest', meaning the data and log files are encrypted when stored on disk.
BENCHMARK REFERENCES	<ul style="list-style-type: none"> FedRAMP
RULE QUERY	<pre>SELECT CASE WHEN EXISTS (SELECT * FROM sys.databases</pre>
MICROSOFT RECOMMENDATION	True
BASELINE ✖	Not set
ACTUAL RESULT	False
REMEDIATION	Enable TDE on the affected database. Please follow the instructions on https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/transparent-data-encryption

The details for each finding provide more insight into the reason. Note the fields describing the finding, the impact of the recommended settings, and details on remediation for the finding.

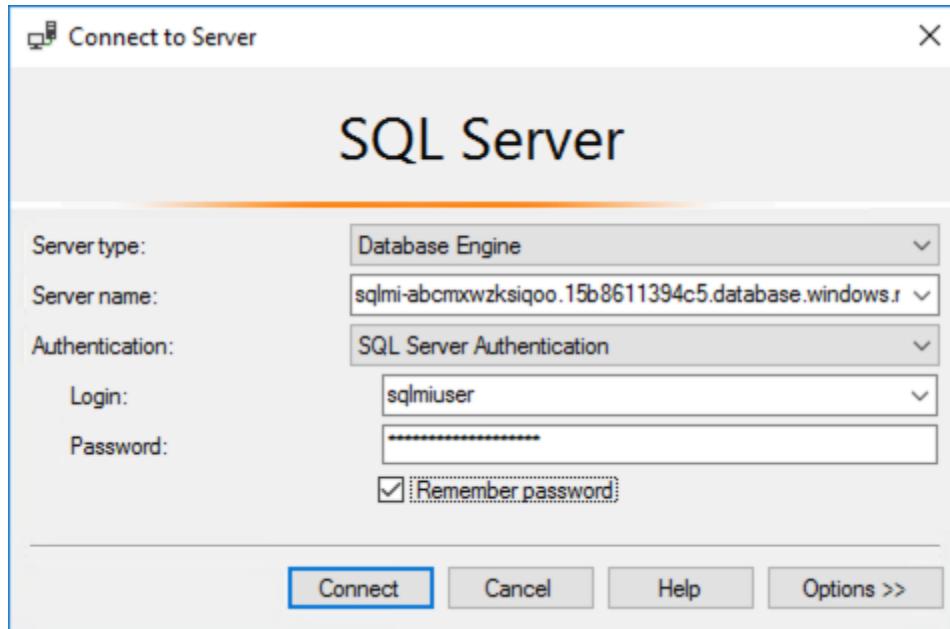
6. You will now act on the recommendation remediation steps for the finding, and enable [Transparent Data Encryption](#) for the TailspinToys database. To accomplish this, you will switch over to using SSMS on your JumpBox VM for the next few steps.

Note:

Transparent data encryption (TDE) needs to be manually enabled for Azure SQL Managed Instance. TDE helps protect Azure SQL Database, Azure SQL Managed Instance, and Azure Data Warehouse against the threat of malicious activity. It performs real-time encryption and decryption of the database, associated backups, and transaction log files at rest without requiring changes to the application.

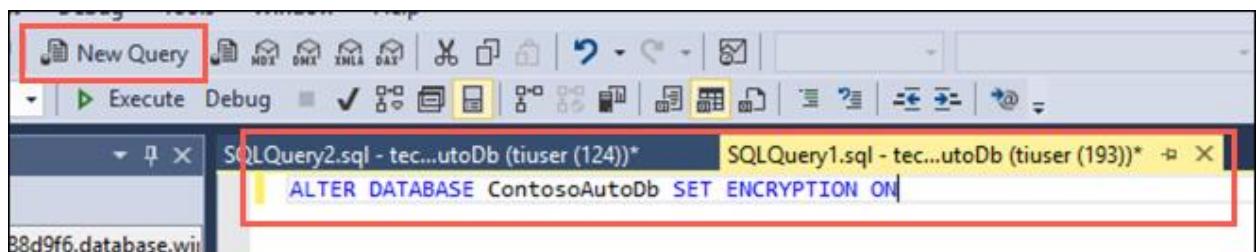
7. On your JumpBox VM, open Microsoft SQL Server Management Studio 18 from the Start menu, and enter the following information in the *Connect to Server* dialog.

- *Server name*: Enter the fully qualified domain name of your SQL managed instance, which you copied from the Azure Cloud Shell in a previous task
- *Authentication*: Select **SQL Server Authentication**
- *Login*: Enter **sqlmiuser**
- *Password*: Enter your password
- Check the **Remember password** box



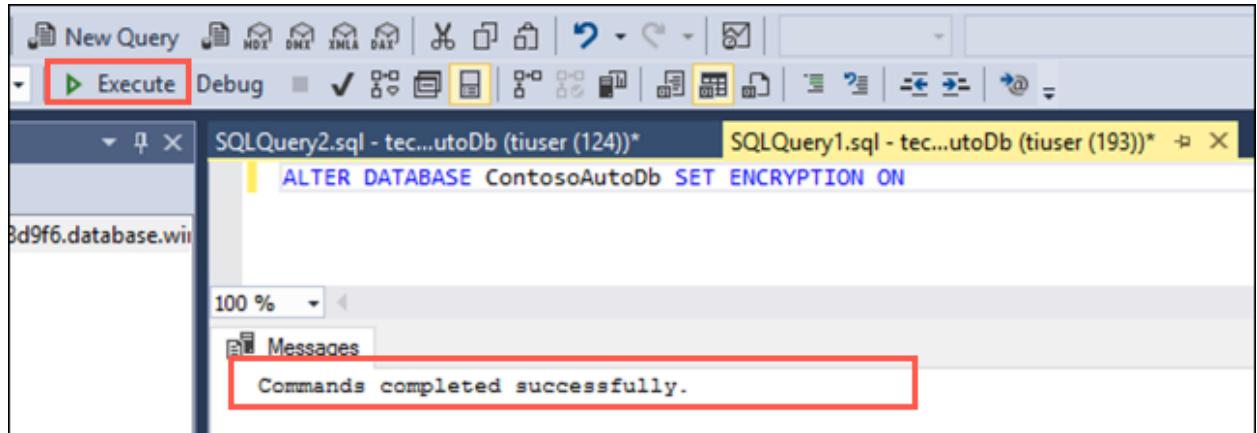
8. In SSMS, select **New Query** from the toolbar, paste the following SQL script into the new query window:

```
ALTER DATABASE [TailspinToys] SET ENCRYPTION ON
```



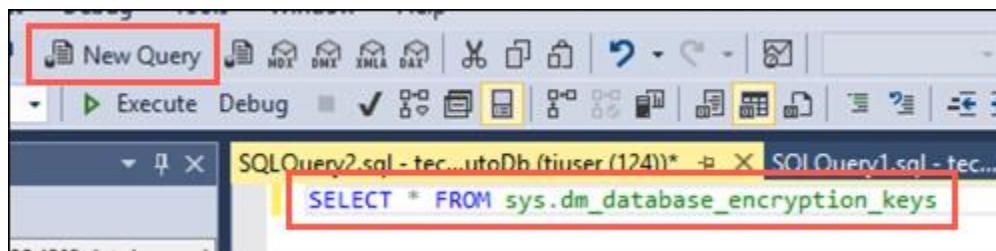
You turn transparent data encryption on and off on the database level. To enable transparent data encryption on a database in Azure SQL Managed Instance use must use T-SQL.

9. Select **Execute** from the SSMS toolbar. After a few seconds, you will see a message that the "Commands completed successfully":



10. You can verify the encryption state and view information the associated encryption keys by using the [sys.dm_database_encryption_keys view](#). Select **New Query** on the SSMS toolbar again, and paste the following query into the new query window:

```
SELECT * FROM sys.dm_database_encryption_keys
```



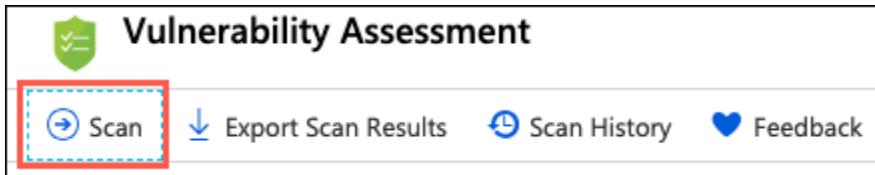
11. Select **Execute** from the SSMS toolbar. You will see two records in the Results window, which provide information about the encryption state and keys used for encryption:

The screenshot shows the SSMS interface with the 'Results' tab selected. The toolbar at the top has a red box around the 'Execute' button. Below the toolbar, there are two query windows: 'SQLQuery2.sql - tec...utoDb (tiuser (124))' and 'SQLQuery1.sql - tec...utoDb (tiuser (193))'. The second window contains the query: 'SELECT * FROM sys.dm_database_encryption_keys'. The results grid shows two rows of data:

database_id	encryption_state	create_date	regenerate_date	modify_date	set_date	opened_date	key_algorithm	key_length	
1	2	3	2019-03-11 18:37:18.973	2019-03-11 18:37:18.973	2019-03-11 18:37:18.973	1900-01-01 00:00:00.000	2019-03-11 18:37:18.973	AES	256
2	5	3	2019-03-11 18:37:18.950	2019-03-11 18:37:18.950	2019-03-11 18:37:18.950	2019-03-11 18:37:18.950	2019-03-11 18:37:18.970	AES	256

By default, service-managed transparent data encryption is used. A transparent data encryption certificate is automatically generated for the server that contains the database.

12. Return to the *Azure portal* and the *Advanced Data Security - Vulnerability Assessment* blade of the TailspinToys managed database. On the toolbar, select **Scan** to start a new assessment of the database:



13. When the scan completes, select the **Failed** tab, enter **VA1219** into the search filter box, and observe that the previous failure is no longer in the Failed list:

Failed (17)	Passed (103)
VA1219	
ID	
SECURITY CHECK	
No results	

14. Now, select the **Passed** tab, and observe the **VA1219** check is listed with a status of **PASS**:

Failed (17)	Passed (103)
VA1219	Category: All selected
SECURITY CHECK	
VA1219 Transparent data encryption should be enabled	
APPLIES TO	
CATEGORY	
STATUS	
ContosoAutoDb Data protection PASS	

Using the SQL Vulnerability Assessment, it is simple to identify and remediate potential database vulnerabilities, allowing you to proactively improve your database security.

Note: If you want to complete an extension of this lab where you also explore the capabilities of [Dynamic Data Masking](#), see exercise 6 and 4 in the non-abbreviated lab [here](#).



5.3 Use online secondary for read-only queries (Bonus)

In this exercise, you will look at how you can use the automatically created online secondary for reporting, without feeling the impacts of a heavy transactional load on the primary database. Each database in the SQL MI Business Critical tier is automatically provisioned with several AlwaysON replicas to support the availability SLA. Using **Read Scale-Out** allows you to load balance Azure SQL Database read-only workloads using the capacity of one read-only replica.



Activities

Complete activities 1-3 by following the links below. Once you're in Activity 1, you can continue to the next activity through Activity 3 (the links below all point to different parts of the same overall lab). When are you done with activities 1-3, come back here for the next section (5.4).

[Activity 1](#): View Leaderboard report in Tailspin Toys web application

[Activity 2](#): Update read only connection string

[Activity 3](#): Reload leaderboard report in the Tailspin Toys web application



Lab Exercise 3: Use an online secondary for read-only queries

In this exercise, you will look at how you can use the automatically created online secondary for reporting, without feeling the impacts of a heavy transactional load on the primary database. Each database in the SQL MI Business Critical tier is automatically provisioned with several AlwaysON replicas to support the availability SLA. Using **Read Scale-Out** allows you to load balance Azure SQL Database read-only workloads using the capacity of one read-only replica.



Activity 1: View Leaderboard report in Tailspin Toys web application

In this task, you will open a web report using the web application you deployed to your App Service.

✓ Steps

1. Return to your JumpBox VM, and run the web application:



2. In the *TailspinToys* web app, select **Leaderboard** from the menu:

A screenshot of a web browser displaying the 'Tailspin Toys Gaming Leaderboard'. The 'Leaderboard' tab is active. In the table header, the word 'READ_WRITE' is highlighted with a red box. The table data is as follows:

Position	Gamer Tag	Name	Score	Diff	Games Played	Online
1	abell978	Andrea Bell	232723	0	2621	
2	ddiaz2679	Destiny Diaz	232705	-18	3306	✓
3	acooperr11383	Alex Cooper	232704	-19	3606	

Note the `READ_WRITE` string on the page. This is the output from reading the `Updateability` property associated with the `ApplicationIntent` option on the target database. This can be retrieved using the SQL query `SELECT DATABASEPROPERTYEX(DB_NAME(), "Updateability")`.



Activity 2: Update read only connection string

In this task, you will enable Read Scale-Out for the `TailspinToys` database, using the `ApplicationIntent` option in the connection string. This option dictates whether the connection is routed to the write replica or to a read-only replica. Specifically, if the `ApplicationIntent` value is `ReadWrite` (the default value), the connection will be directed to the database's read-write replica. If the `ApplicationIntent` value is `ReadOnly`, the connection is routed to a read-only replica.

✓ Steps

1. Stop the application by closing the browser
2. In order to have the app connect to the read-only replica, open `appsettings.json`. In the `TailspinToysReadOnlyContext` line, paste the following parameter to end the line:
`ApplicationIntent=ReadOnly;`

3. The TailspinToysReadOnlyContext connection string should now look something like the following:

```
Server=mi-SUFFIX.15b8611394c.database.windows.net;Database=TailspinToys;User  
ID=sqlmiuser;Password=<your-  
password>;Trusted_Connection=False;Encrypt=True;TrustServerCertificate=True;Ap  
plicationIntent=ReadOnly;
```

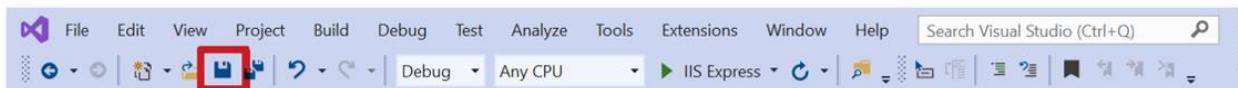


Activity 3: Reload leaderboard report in the Tailspin Toys web application

In this task, you will refresh the Leaderboard report in the Tailspin Toys web app, and observe the results.

✓ Steps

3. Save the appsettings.json file in Visual Studio:



4. Run the application (IIS Express button):



5. Return to the TailspinToys gamer information website you opened previously, on the **Leaderboard** page. The page should now look similar to the following:

Tailspin Toys						
Tailspin Toys Gaming Leaderboard						
READ_ONLY						
Position	Gamer Tag	Name	Score	Diff	Games Played	Online
1	abel1978	Andrea Bell	232723	0	2621	
2	ddiaz2679	Destiny Diaz	232705	-18	3306	✓
3	acooper11383	Alex Cooper	232704	-19	3606	

Notice the updateability option is now displaying as READ_ONLY. With a simple addition to your database connection string, you are able to send read-only queries to the online secondary of your SQL MI Business critical database, allowing you to load-balance read-only workloads using the capacity of one

read-only replica. The SQL MI Business Critical cluster has built-in Read Scale-Out capability that provides free-of charge built-in read-only node that can be used to run read-only queries that should not affect performance of your primary workload.

Note:

If you are attending this lab as part of a day-long workshop, this may still read as READ-WRITE, if the lab was configured using [Instance Pools](#), since Instance Pools currently only support General Purpose (not Business Critical which is needed here). You have finished the activities for Module 5. [Return to it here](#), review, and refer to instructor guidance.



5.4 After the Migration

In this module, you used the [Azure Database Migration Service](#) (DMS) to migrate the TailspinToys database from the on-premises SQL 2008 R2 database to SQL MI. You then updated the web application to use the SQL MI created, and enabled [advanced security features](#). Finally, you set up your application to leverage the [online secondary replica](#) to handle heavy read workloads.

Now that Tailspin Toys has completed a migration for their gaming database. They'll want to leverage the [post-migration validation and optimization guide](#) to ensure data completeness and uncover and resolve performance issues.

If and when Tailspin Toys chooses to scale their migration to other instances and databases, they can leverage the same process you've seen in Modules 4 and 5, but should also refer to the guidance Microsoft provides on [scaling a migration to Azure](#).



Lab Exercise 4: After the Migration

In this Lab, you used the [Azure Database Migration Service](#) (DMS) to migrate the TailspinToys database from the on-premises SQL 2008 R2 database to SQL MI. You then updated the web application to use the SQL MI created, and enabled [advanced security features](#). Finally, you set up your application to leverage the [online secondary replica](#) to handle heavy read workloads.

Now that Tailspin Toys has completed a migration for their gaming database. They'll want to leverage the [post-migration validation and optimization guide](#) to ensure data completeness and uncover and resolve performance issues.

If and when Tailspin Toys chooses to scale their migration to other instances and databases, they can leverage the same process you've seen in Labs 4 and 5, but should also refer to the guidance Microsoft provides on [scaling a migration to Azure](#).



For Further Study

- [Microsoft Hands On Labs](#) offers free self-paced lab environments as well as a request form for instructor led lab environments. As of last update, there are about eight labs available around Azure SQL (assessment, migration, app innovation, row level security, managed instance, and more).
- [Microsoft Hands On Demos](#) is similar to Hands On Labs, but offers easy to set up demos that are free for certain internals and partners. As of last update, there are about seven demos available around Azure SQL.
- [Azure Database Migration Guide](#) contains lots of resources that will help in guiding and supporting database migrations to Azure.
- [The Azure Architecture Documentation](#) contains many migration patterns as well as full code samples, scenarios, and guidance about how to migrate on-prem estates. There are useful, detailed scenarios about rehosting to SQL MI and SQL VMs, as well as guidance of how to scale a migration, after you've done a PoC.
- [MCW: Migrating SQL Databases to Azure](#) contains extended labs from what you've seen in these exercises. There is an opportunity to see how the networking was configured, and deeper dives around the network and setup.
- [How to Deploy an Azure Infrastructure](#) and Best Practices for setting up networking are also two very useful resources when moving to Azure.
- [Best practices for costing and sizing workloads migrated to Azure](#)
- [Best practices for securing and managing workloads migrated to Azure](#)

Next, Continue to [06 - What to Use When](#).

Workshop: SQL Ground-to-Cloud

A Microsoft workshop from the SQL Server team



06 - What to Use When: Designing a Solution

In this workshop you have covered using SQL Server both on-premises and in-cloud configurations, as well as hybrid applications as a solution for data processing. The end of this Module contains several helpful references you can use in these exercises and in production.

This module can be used stand-alone, and does not require any [prerequisite](#) other than a laptop and some sort of design software (such as Microsoft Visio) - although you may also just use a whiteboard or paper for your design.

There are many elements in a single solution, and in this module you'll learn how to take the business scenario and determine the best resources and processes to use to satisfy the requirements while considering the constraints within the scenario.

In production, there are normally 6 phases to create a solution. These can be done in-person, or through recorded documents:

- 01 **Discovery**: The original statement of the problem from the customer
- 02 **Envisioning**: A "blue-sky" description of what success in the project would look like. Often phrased as "*I can...*" statements
- 03 **Architecture Design Session**: An initial layout of the technology options and choices for a preliminary solution
- 04 **Proof-Of-Concept** (POC): After the optimal solution technologies and processes are selected, a POC is set up with a small representative example of what a solution might look like, as much as possible. If available, a currently-running solution in a parallel example can be used
- 05 **Implementation**: Implementing a phased-in rollout of the completed solution based on findings from the previous phases
- 06 **Handoff**: A post-mortem on the project with a discussion of future enhancements

Throughout this module, you can use various templates, icons, stencils and other assets to assist you with each phase and also use these with your exercises. These assets can also be used in your production workloads: <https://github.com/microsoft/sqlworkshops/tree/master/ProjectResources>

For this module, you'll focus on the *Discovery* and the *Architecture Design Session* phases only. If you wish to develop your solution further after the course, you can use the assets above to complete all phases.



6.1 Understand the Problem Space

The first step in any project is to fully understand the problem the company needs to solve, and any requirements and constraints they have on those goals. This is often in the form of a "Problem Statement", which is a formal set of paragraphs clearly defining the circumstances, present condition, and desired outcomes for a solution. At this point you want to avoid exploring how to solve the problem, and focus on what you want to solve.

Begin with as complete an examination of the company and organization as you can. Gather information from as many sources as possible, and simplify the descriptions to have specific measurements and depictions of the environment.

From there, lay out the problem, and then review that with all stakeholders.

After everyone agrees on the problem statement, pull out as many requirements (*goals*) for the project as you can find, and then lay in any constraints the solution has. At this point, it's acceptable to have unrealistic constraints - later you can pull those back after showing a cost/benefit ratio on each requirement and constraint.



Activity: Review Business Scenarios

In this activity you will review three business scenarios, and pick one to focus on for the rest of this module. The company descriptions, project goals, and constraints have already been laid out for you.

After you make your choice, copy the problem statement into your working documents (see the [Resources](#) for examples) and make any changes or additions you want to make to the scenario. Feel free to adapt it to have more information where you want clarity - you can make assumptions about any part of the scenario. Are there sub-goals that have been left out? Any other constraints you can think of?



AdventureWorks

[Adventure Works Cycles](#) is a large, multinational manufacturing company. The company manufactures and sells metal and composite bicycles to North American, European and Asian commercial markets. While its base operation is located in Bothell, Washington with 290 employees, several regional sales teams are located throughout their market base.

Starting in the year 2000, Adventure Works Cycles bought a small manufacturing plant, Importadores Neptuno, located in Mexico. Importadores Neptuno manufactures several critical subcomponents for the Adventure Works Cycles product line. These subcomponents are shipped to the Bothell location for final product assembly. In 2001, Importadores Neptuno, became the sole manufacturer and distributor of the touring bicycle product group.

Coming off a successful fiscal year, Adventure Works Cycles is looking to broaden its market share by targeting their sales to their best customers, extending their product availability through an external Web site, and reducing their cost of sales through lower production costs. They are also looking to modernize their data estate.

Project Goals

- Modernize to a newer SQL version
- Move to Cloud wherever possible
- Cloud Integration
- Increase Performance
- Publish Product Catalog to the Web
- Enable Business-To-Business (B2B) systems

Project Constraints

- Some systems must stay on-premises
- In some cases, no code change is possible
- The B2B system should be a "Pull" from partners



Contoso

The Contoso company is a multi-national business with headquarters in Paris, France. It is a conglomerate manufacturing, sales, and support organization with over 100,000 products. They are embarking on a multi-year process of migrating from company-owned datacenters to a cloud provider. They have narrowed the list of potential vendors to three, including Microsoft. They have [high security](#) and [interoperability with mobile device](#) concerns. There is also an Open-Source (OSS) investigation at the company.

Project Goals

- Move everything to Cloud
- Multi-Cloud strategy desired - standards-based
- All client apps should be available worldwide
- Server-side should be API's by default
- Interest in parity for platforms (OSS Support)

Project Constraints

- High Security and Auditing capabilities required
- International Compliance required
- Access Tracking required
- Must be user-friendly on mobile devices



Wide World Importers

[Wide World Importers](#) (WWI) is a wholesale novelty goods importer and distributor operating from the San Francisco bay area in the United States.

As a wholesaler, WWI's customers are mostly companies who resell to individuals. WWI sells to retail customers across the United States including specialty stores, supermarkets, computing stores, tourist attraction shops, and some individuals. WWI also sells to other wholesalers via a network of agents who promote the products on WWI's behalf. While all of WWI's customers are currently based in the United States, the company is intending to push for expansion into other countries.

WWI buys goods from suppliers including novelty and toy manufacturers, and other novelty wholesalers. They stock the goods in their WWI warehouse and reorder from

suppliers as needed to fulfil customer orders. They also purchase large volumes of packaging materials, and sell these in smaller quantities as a convenience for the customers.

Recently WWI started to sell a variety of edible novelties such as "chili chocolates". The company previously did not have to handle chilled items. Now, to meet food handling requirements, they must monitor the temperature in their chiller room and any of their trucks that have chiller sections.

Project Goals

- Enable Big Data processing
- Enable Machine Learning and Artificial Intelligence prediction capabilities
- Cloud platform desired, but may need to consider on-premises options

Project Constraints

- Short timeframe
- Unknown Data Sources
- Use only API calls for the predictions

6.2 Understand the Technologies and Processes for Solving the Problem

With a firm understanding of what the customer needs, you can now consider the technologies and processes at your disposal for the solution. Each technology will have benefits and considerations, so at this point you just want to list out all of your options. Everything is on the table at this phase, and ensure that you check the list you create with other professionals to ensure you have considered everything that could solve the problem.



Activity: List the Technologies and Processes for the Problem Space

In this activity you will list out all of the options you have for your problem space.

Steps

- Open your Architecture Design Session (ADS) document and detail all technologies you have studied in this workshop, and list them out. (Order is not important during this step)
- Next, write the problem element next to each technology that it could solve
- Document any processes you should follow when using each technology



6.3 The Decision Matrix

Following the process, you now know the problems you want to solve, the desired outcomes for the solution, and several tools and technique options that you can use to achieve your goals. In most situations, there are several ways to solve a given problem. Sometimes the "best" solution is too costly, inconvenient or unworkable due to the requirements or constraints the customer puts on the solution.

Because most solutions are fairly complex, and there are multiple technology and process choices, considerations and requirements, a *Decision Matrix* that lists these elements is useful. It contains columns for the technology and process options you have, and the requirements and constraints as rows. Each column gets a score you assign from a low number (does not meet this requirement) to a higher one (does meet the requirement). These numbers are summed at the end of each row, per requirement. The highest number is usually the best technology for that aspect of the solution.

As an example, assume you have an application that is written using T-SQL statements, and you want to store data that has high security requirements and is available online:

Requirement/Constraint	SQL Server in Azure VM	Azure SQL DB	Postgres as a Service
Low Cost	2	3	3
Easy to Manage	2	3	3
Highly Securable	3	3	2
Fully Supports T-SQL	3	3	0

Score:	10	12	8
--------	----	----	---

In this simple example, Azure SQL DB is a high candidate for your solution. (In production, there would be far more requirements and constraints, and you may need to use a 1-5 scale rather than 1-3)



Activity: Create a Decision Matrix

In this activity you will use the scenario you selected from above and create a Decision Matrix using a spreadsheet.



Steps

- [Open this reference and read through it.](#)
- Create a spreadsheet for your Decision Matrix, or download one of the samples from the reference.
- Fill out the Decision Matrix based on the problem requirements and constraints using the technologies and processes you developed in the previous steps



6.4 Explain the Solution

The architecture design session above is most often conducted by the technical staff with representation from the business. After the design session is complete, the findings should be condensed into an instrument (slides, notes or other graphics tool) that allow the team to explain the proposal along with a set of options. At the end of the presentation, your team should also include a description of the project, timelines, and responsibilities.



Activity: Create a Solution Presentation

In this activity your team will create a solution briefing with options and timelines.



Steps

- Open this reference: <https://github.com/microsoft/sqlworkshops/tree/master/ProjectResources>

- You may use the PowerPoint template provided (6 - Handoff.pptx) or create your own briefing tool, using anything you like.
- Make sure you include at least one alternative option and explain why you chose your original one.
- If time permits, review the schedule documents in the Resources area and assign reasonable timelines to your solution.



References

- [How to Write a Problem Statement](#) - Article on writing effective problem statements
- [Decision Matrix Analysis](#) - Article on creating a Decision Matrix
- [Azure Pricing Calculator](#) - Create a cost analysis of your solution
- [Azure Data Architecture Guide](#) - This guide presents a structured approach for designing data-centric solutions on Microsoft Azure. It is based on proven practices derived from customer engagements
- [Azure Reference Architectures](#) - Recommended practices, along with considerations for scalability, availability, manageability, and security
- [Microsoft Cloud Adoption Framework for Azure](#) - Article on writing effective problem statements
- [Microsoft Azure Trust Center](#) - Full reference site for Azure security, privacy and compliance

Congratulations! You have completed this workshop on "SQL Ground-to-Cloud". You now have the tools, assets, and processes you need to extrapolate this information into other applications.