

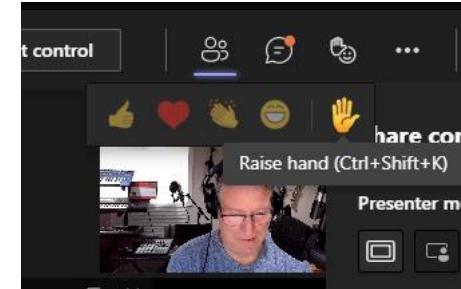


KickStart DP 500 certification





Ctrl+Shift+K is a quick way to raise or lower your hand



Questions during the session

In a regular teams meeting, please use the “Raise Your Hand” feature if you want to ask a question versus posting it in the chat window



Microsoft Confidential

All content is NDA unless otherwise stated.



Exam overview and objective domain DP-500



Become Microsoft Certified



Fundamental Certifications

Recommended start
Ideal for individuals just starting in technology or thinking about a career change.

Associate Certifications

Two years of comprehensive working experience.
It is helpful to have related Fundamental certifications but is not required.

Expert Certifications

Two to five years of deep technical experience.
Many Expert certifications require a specific Associate level certification.

Azure Enterprise Data Analyst Associate Role



Data Analyst Role

Audience profile:

Responsibilities for this role include performing **advanced data analytics at scale**, such as cleaning and transforming data, designing and building enterprise data models, incorporating advanced analytics capabilities, integrating with IT infrastructure, and applying development lifecycle practices

Candidates should have advanced Power BI skills, **including managing data repositories** and data processing in the cloud and on-premises, along with using **Power Query** and Data Analysis Expressions (**DAX**). They should also be proficient in consuming data from **Azure Synapse Analytics** and should have experience querying relational databases, analyzing data by using Transact-SQL (**T-SQL**), and **visualizing data**.

Azure Enterprise Data Analyst Associate workload





Microsoft Certified: Azure Enterprise Data Analyst Associate

Take one exam



CERTIFICATION EXAM

[Designing and Implementing Enterprise-Scale Analytics Solutions Using Microsoft Azure and Microsoft Power BI](#)

Earn the certification

Earn the certification



ASSOCIATE CERTIFICATION

Microsoft Certified: Azure Enterprise Data Analyst Associate

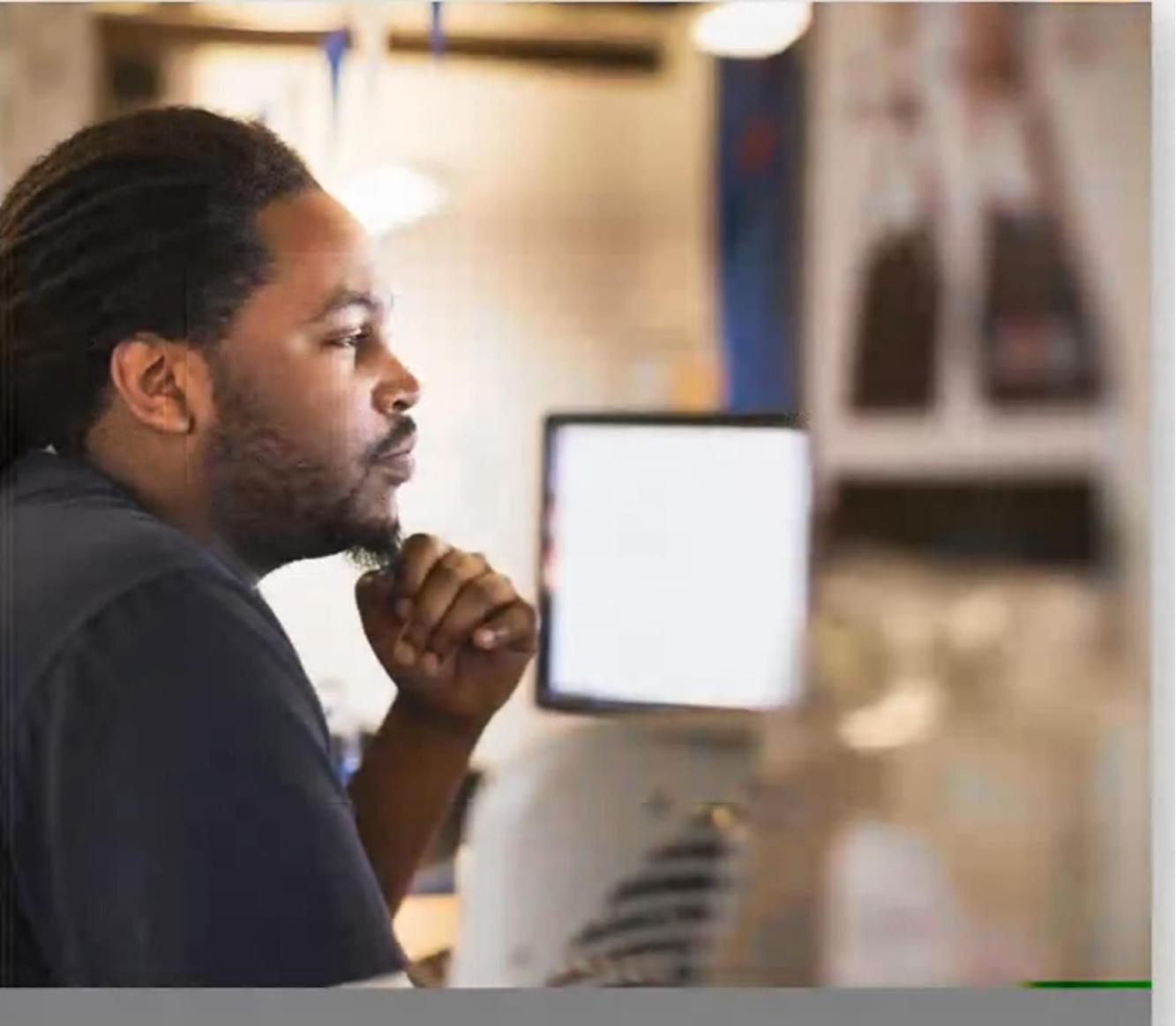
Exam objectives

Skills Measured	Weights
Implement and manage a data analytics environment	25–30%
Query and transform data	20–25%
Implement and manage data models	25–30%
Explore and visualize data	20–25%

- Percentages indicate the relative weight of each area on the exam
- The higher the percentage, the more questions you are likely to see in that area

Passing score is 700

How to prepare



Training and certification

Microlearning

- Step-by-step, bite-sized tutorials by product, skill level, and job role
- Hands-on learning with interactive browser-based scripting environments
- Immediate assessment via knowledge quizzes

 Microsoft Learn

Online Courses

- Self-paced, structured courses with the ability to track progress and maintain a transcript
- Hands-on learning with hands-on labs
- Knowledge assessments

 Microsoft Learning Partners

 LinkedIn Learning

 Pluralsight

Instructor-led Training

- Blended learning, in-person, and online to suit learning needs
- Delivered by Microsoft Certified Trainers
- Deep technical training to give you the technical expertise

 Microsoft Learning Partners

Certification

- New role-based certifications, including varying levels (Fundamentals, Associate, Expert)
- Industry-recognized technical certifications
- Share “credentials” with your professional network with an online badge

 aka.ms/MSCert

 Microsoft Learning Partners

https://learn.microsoft.com/en-us/certifications/browse/?roles=data-analyst

A ⌂ ⌂ ⌂ ⌂ Not syncing

Data Analyst

Data Engineer

Data Scientist

Database Administrator

Developer

DevOps Engineer

Functional Consultant

Levels

Advanced

Beginner

Intermediate

Certification Types

Fundamentals

MCSA

Role-based

Specialty

Types

Certification

Exam

SQL Server • Data Analyst • Intermediate

+ Save

SQL Server • Data Analyst • Advanced

+ Save

SQL Server • Data Analyst • Intermediate

+ Save

EXAM

Exam 70-768: Developing SQL Data Models

Microsoft EXAM

SQL Server • Data Analyst • Intermediate

+ Save

CERTIFICATION

MCSA: BI Reporting

Exam 70-778, 70-779

Microsoft Power Platform • Data Analyst • Intermediate

Microsoft MCSA

+ Save

CERTIFICATION

Microsoft Certified: Azure Enterprise Data Analyst Associate

Exam DP-500

Azure • Data Analyst • Intermediate

Microsoft ASSOCIATE

+ Save

EXAM

Exam PL-300: Microsoft Power BI Data Analyst

Microsoft Power Platform • Data Analyst • Intermediate

Microsoft EXAM

+ Save

EXAM

Exam 70-779: Analyzing and Visualizing Data with Microsoft Excel

Office • Data Analyst • Advanced

Microsoft EXAM

+ Save

EXAM

Exam PL-100: Microsoft Power Platform App Maker

Dynamics 365 • Business Analyst • Intermediate

Microsoft EXAM

+ Save

EXAM

Exam DA-100: Analyzing Data with Microsoft Power BI

Microsoft Power Platform • Data Analyst • Intermediate

Microsoft EXAM

+ Save

CERTIFICATION

Microsoft Certified: Azure Data Fundamentals

Exam DP-900

Azure • Database Administrator • Beginner

Microsoft FUNDAMENTALS

+ Save

EXAM

Exam DP-500: Designing and Implementing Enterprise-Scale Analytics Solutions Using Microsoft Azure and Microsoft Power BI

Azure • Data Analyst • Intermediate

Microsoft EXAM

+ Save

EXAM

Exam DP-900: Microsoft Azure Data Fundamentals

Azure • Database Administrator • Beginner

Microsoft EXAM

+ Save

CERTIFICATION

MCSA: SQL 2016 BI Development

Exam 70-767, 70-768

SQL Server • Data Analyst • Intermediate

Microsoft MCSA

+ Save

CERTIFICATION

Microsoft Certified: Power BI Data Analyst Associate

Exam PL-300

Microsoft Power Platform • Data Analyst • Intermediate

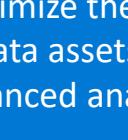
Microsoft ASSOCIATE

+ Save

Build enterprise-scale data analytics solutions



Maximize the value of data assets with advanced analytics



<https://learn.microsoft.com/en-us/certifications/azure-enterprise-data-analyst-associate/>

Two ways to prepare

Online - Free

Instructor-led - Paid

LEARNING PATH

Introduction to data analytics on Azure

0 of 3 modules completed

Intermediate • Data Analyst • SQL Database



Start

+ Save

LEARNING PATH

Govern data across an enterprise

0 of 5 modules completed

Intermediate • Data Analyst • Power BI



+ Save

LEARNING PATH

Model, query, and explore data in Azure Synapse

0 of 4 modules completed

Intermediate • Data Analyst • Synapse Analytics



+ Save

LEARNING PATH

Prepare data for tabular models in Power BI

0 of 3 modules completed

Intermediate • Data Analyst • Power BI



+ Save

LEARNING PATH

Design and build tabular models

0 of 5 modules completed

Intermediate • Data Analyst • Power BI



+ Save

LEARNING PATH

Implement advanced data visualization techniques by using Power BI

0 of 3 modules completed

Intermediate • Data Analyst • Power BI



+ Save

LEARNING PATH

Implement and manage an analytics environment

0 of 8 modules completed

Intermediate • Data Analyst • Power BI



+ Save

LEARNING PATH

Manage the analytics development lifecycle

0 of 3 modules completed

Intermediate • Data Analyst • Power BI



+ Save



Exam AA-001_Sandbox

AA-001_Sandbox

Welcome !

Maximum time for this session, including instructions, survey, and exam: 480 minutes

Number of exam questions: 10

Maximum time for exam: 480 minutes

Minimum score required to pass this exam: 700



Demo the exam experience by visiting our [Exam Sandbox](#)





Implement and manage a data analytics environment (25–30%)

Query and transform data (20–25%)

Implement and manage data models (25–30%)

Explore and visualize data (20–25%)

Govern and administer a data analytics environment

- Manage Power BI assets by using Azure Purview
- Identify data sources in Azure by using Azure Purview
- Recommend settings in the Power BI admin portal
- Recommend a monitoring and auditing solution for a data analytics environment, including Power BI REST API and PowerShell cmdlets

Integrate an analytics platform into an existing IT infrastructure

- Identify requirements for a solution, including features, performance, and licensing strategy
- Configure and manage Power BI capacity
- Recommend and configure an on-premises gateway in Power BI
- Recommend and configure a Power BI tenant or workspace to integrate with Azure Data Lake Storage Gen2
- Integrate an existing Power BI workspace into Azure Synapse Analytics

Manage the analytics development lifecycle

- Commit code and artifacts to a source control repository in Azure Synapse Analytics
- Recommend a deployment strategy for Power BI assets
- Recommend a source control strategy for Power BI assets
- Implement and manage deployment pipelines in Power BI
- Perform impact analysis of downstream dependencies from dataflows and datasets
- Recommend automation solutions for the analytics development lifecycle, including Power BI REST API and PowerShell cmdlets
- Deploy and manage datasets by using the XMLA endpoint
- Create reusable assets, including Power BI templates, Power BI data source (.pbids) files, and shared datasets



Implement and manage a data analytics environment
(25–30%)

Query and transform data
(20–25%)

Implement and manage data models (25–30%)

Explore and visualize data (20–25%)

Query data by using Azure Synapse Analytics

- Identify an appropriate Azure Synapse pool when analyzing data
- Recommend appropriate file types for querying serverless SQL pools
- Query relational data sources in dedicated or serverless SQL pools, including querying partitioned data sources
- Use a machine learning PREDICT function in a query

Ingest and transform data by using Power BI

- Identify data loading performance bottlenecks in Power Query or data sources
- Implement performance improvements in Power Query and data sources
- Create and manage scalable Power BI dataflows
- Identify and manage privacy settings on data sources
- Create queries, functions, and parameters by using the Power Query Advanced Editor
- Query advanced data sources, including JSON, Parquet, APIs, and Azure Machine Learning models



Implement and manage a data analytics environment (25–30%)

Query and transform data (20–25%)

Implement and manage data models (25–30%)

Explore and visualize data (20–25%)

Design and build tabular models

- Choose when to use DirectQuery for Power BI datasets
- Choose when to use external tools, including DAX Studio and Tabular Editor 2
- Create calculation groups
- Write calculations that use DAX variables and functions, for example handling blanks or errors, creating virtual relationships, and working with iterators
- Design and build a large format dataset
- Design and build composite models, including aggregations
- Design and implement enterprise-scale row-level security and object-level security

Optimize enterprise-scale data models

- Identify and implement performance improvements in queries and report visuals
- Troubleshoot DAX performance by using DAX Studio
- Optimize a data model by using Tabular Editor 2
- Analyze data model efficiency by using VertiPaq Analyzer
- Implement incremental refresh
- Optimize a data model by using denormalization



Implement and manage a data analytics environment (25–30%)

Query and transform data (20–25%)

Implement and manage data models (25–30%)

Explore and visualize data (20–25%)

Explore data by using Azure Synapse Analytics

- Explore data by using native visuals in Spark notebooks
- Explore and visualize data by using the Azure Synapse SQL results pane

Visualize data by using Power BI

- Create and import a custom report theme
- Create R or Python visuals in Power BI
- Connect to and query datasets by using the XMLA endpoint
- Design and configure Power BI reports for accessibility
- Enable personalized visuals in a report
- Configure automatic page refresh
- Create and distribute paginated reports in Power BI Report Builder

Implement and manage a data analytics environment



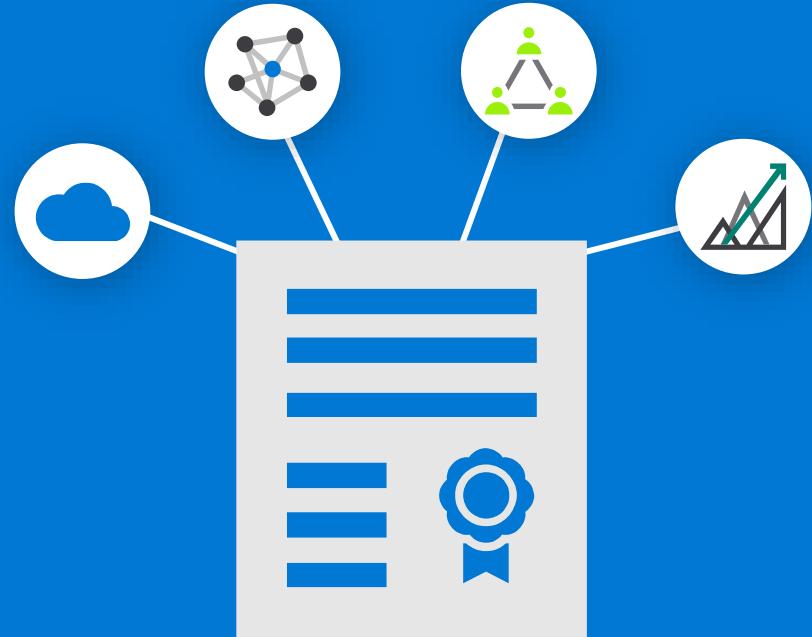
Query and transform data



Implement and manage data models



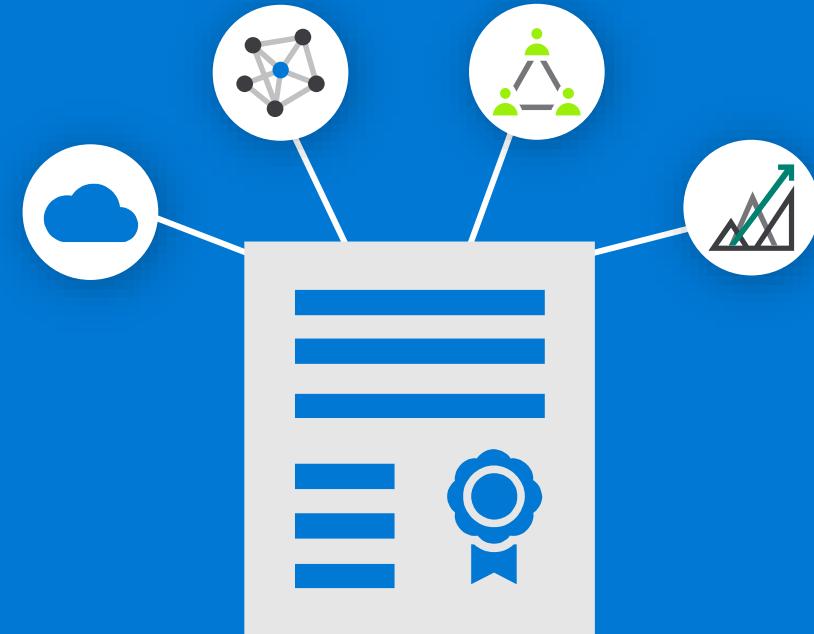
Explore and visualize data



Implement and manage a data analytics environment

Govern and administer a data analytics environment

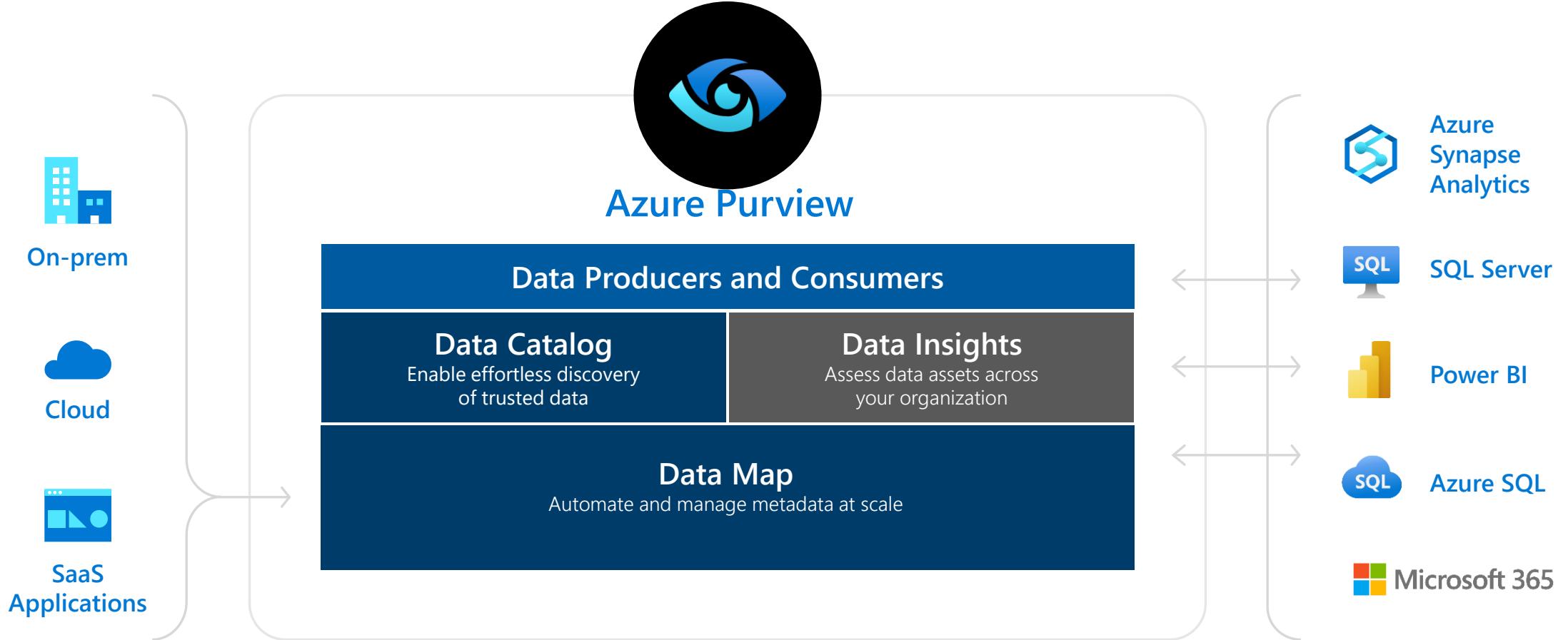
- Manage Power BI assets by using Azure Purview
- Identify data sources in Azure by using Azure Purview
- Recommend settings in the Power BI admin portal
- Recommend a monitoring and auditing solution for a data analytics environment, including Power BI REST API and PowerShell cmdlets



Generally Available

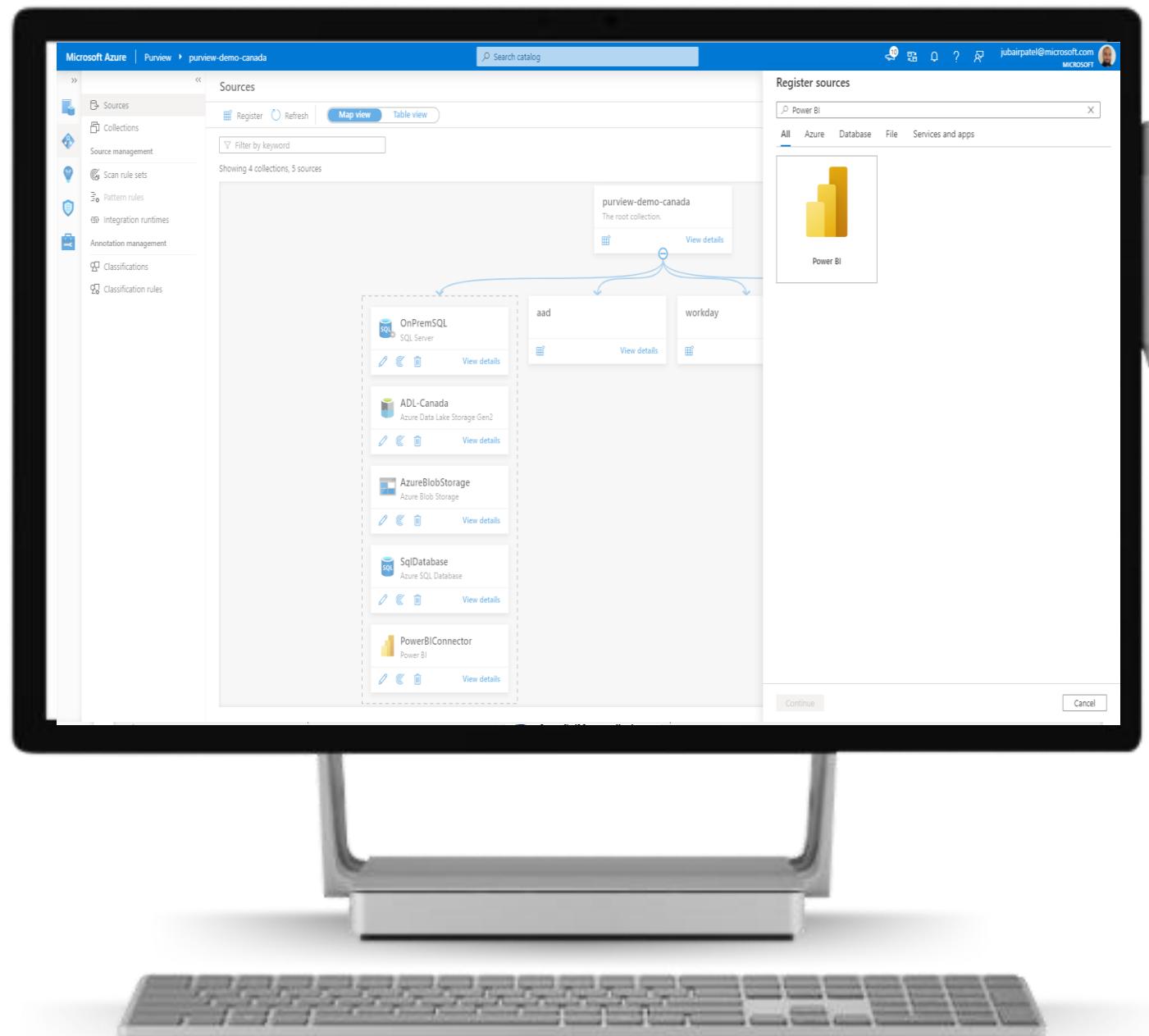
Preview

Unified Data Governance with Azure Purview



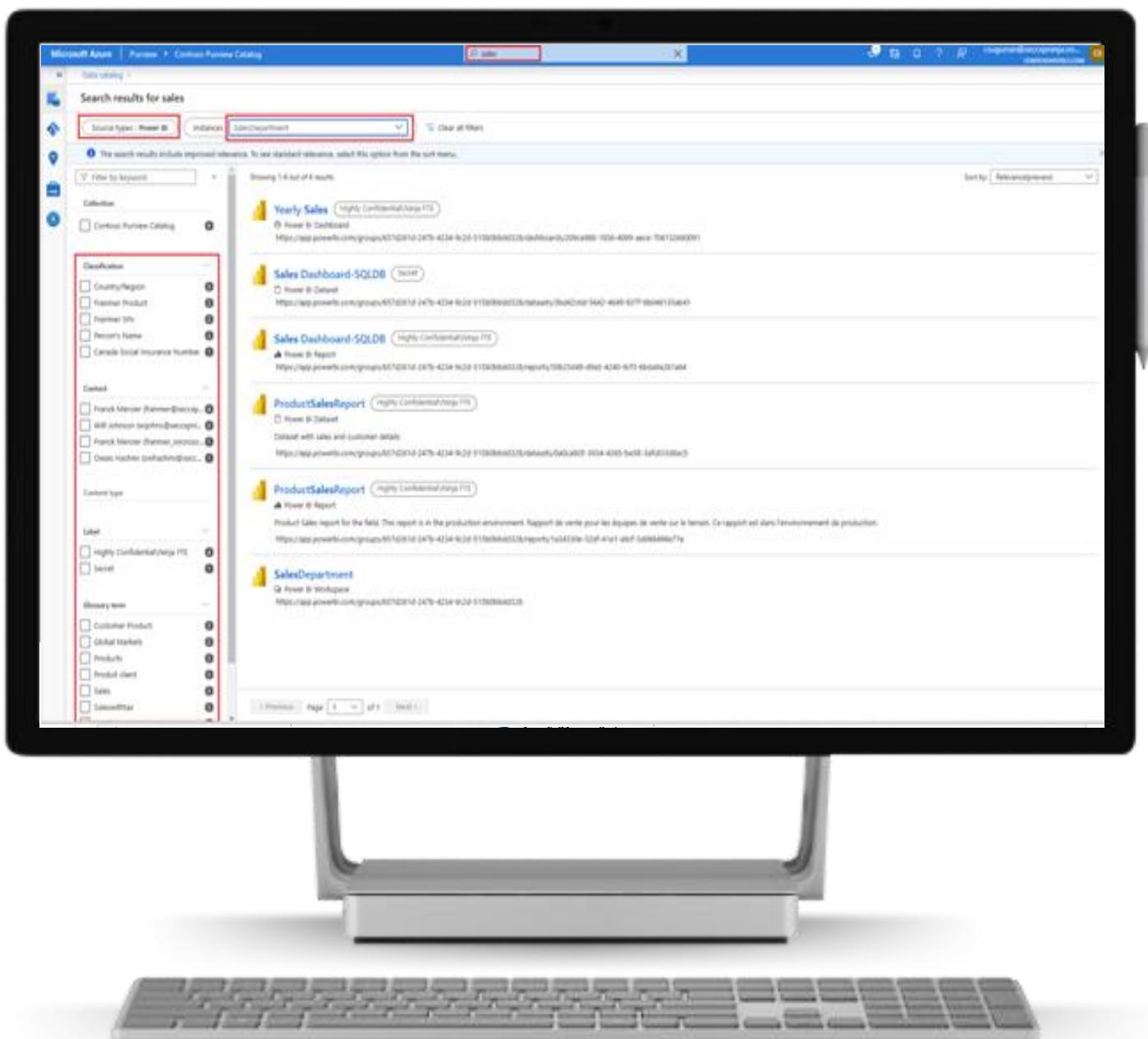
Manage Power BI assets by using Azure Purview

- Native connection with just a few clicks - no infrastructure or code setup
- Scan huge numbers of Power BI assets quickly with auto scaled performance
- Easily set up scheduled and incremental scans to keep information up-to-date



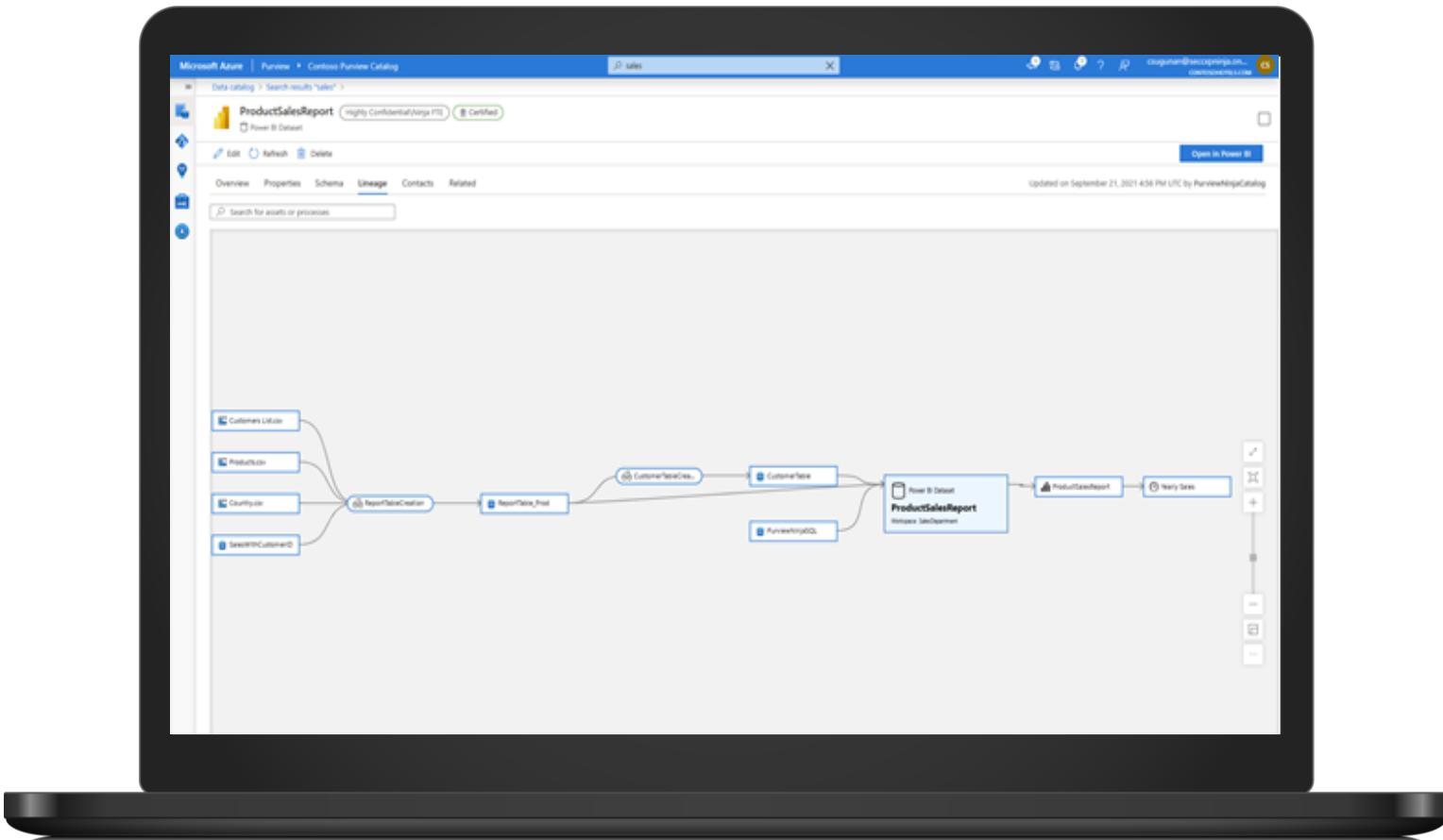
Search and browse Power BI assets in Azure

- Search across all Power BI assets using business glossary and keyword search
- Govern column, dashboard, dataset and workspace level details in Purview
- Set ownership for governance of Power BI assets



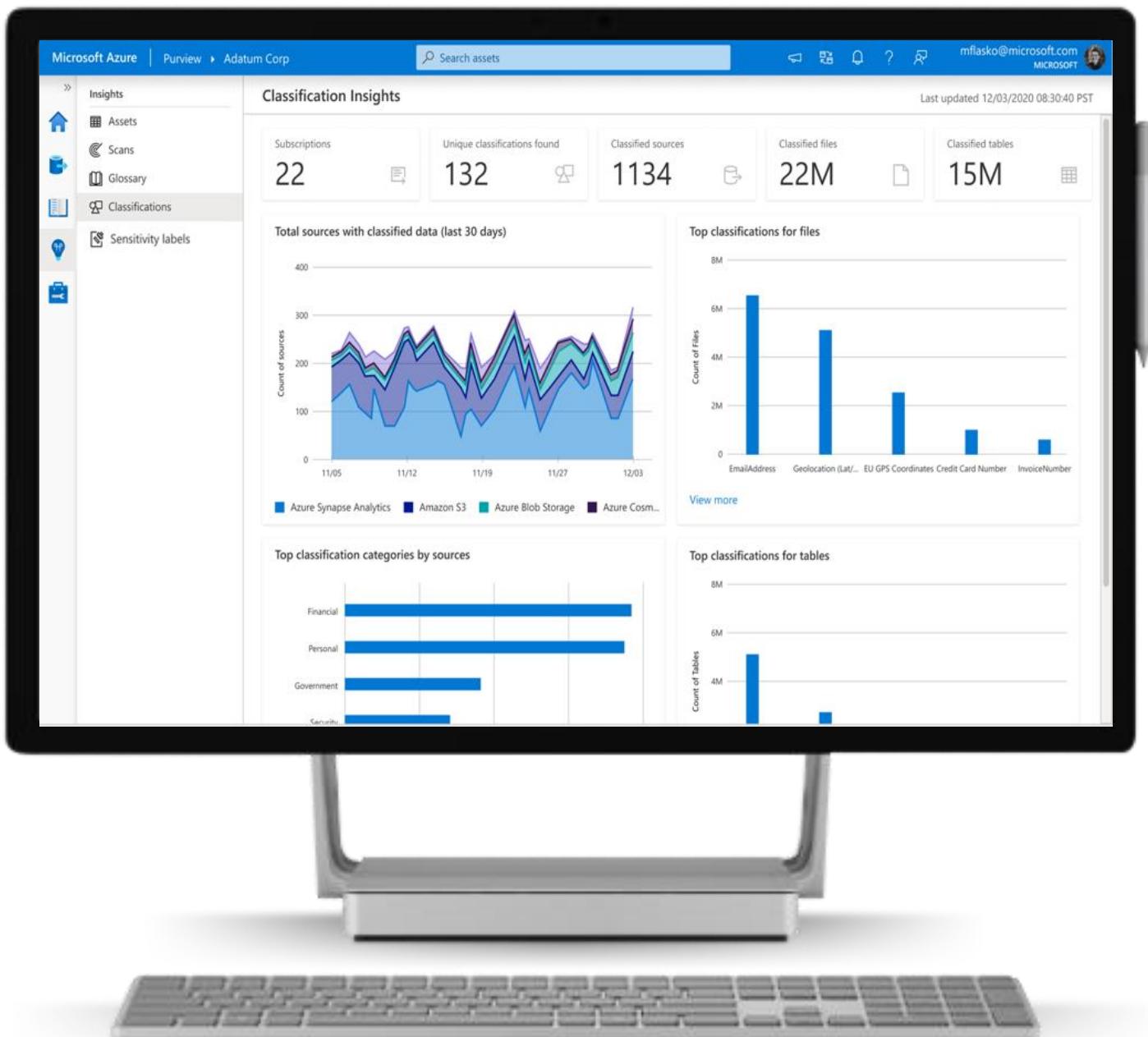
Trace End to End lineage of Power BI Reports

- Automatically scan lineage of Power BI assets and connect with Azure data lake lineage
- Perform forward and backward-looking scenarios for impact analysis
- Lineage visualization at column level



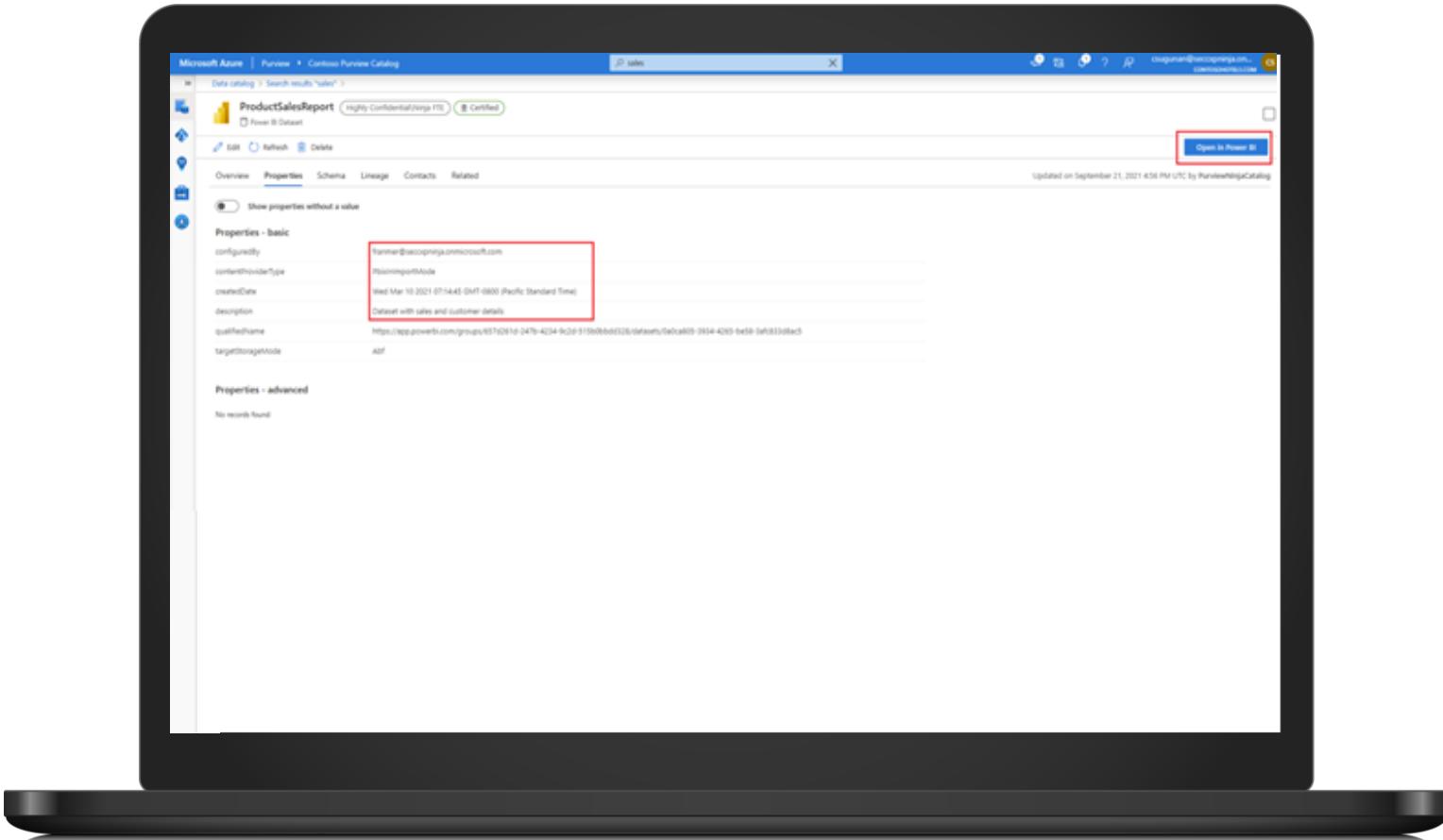
Govern sensitive data being used for insights and visualization

- Automatically scan MIP sensitivity labels from Power BI into Purview
- Apply classifications to Power BI assets in Purview
- Apply ownership to Power BI assets
- Pre-built reports to show how sensitive data is being used

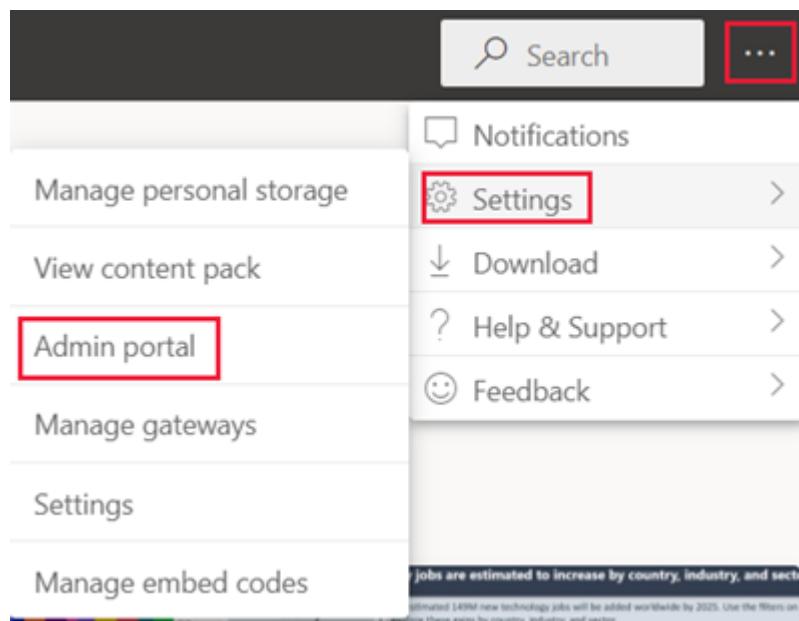


Open Data sets from Purview directly in Power BI

- One-click to open data sets from the Purview Data Catalog in Power BI
- Open Azure SQL and Azure Synapse datasets in Power BI securely



Recommend settings in the Power BI admin portal



Feature	Description
Tenant settings	Enable, disable, and configure Power BI.
Usage metrics	View metrics about Power BI usage in your organization.
Users	Manage users in the Microsoft 365 admin center.
Premium Per User	Configure auto refresh and dataset workload settings.
Audit logs	Audit Power BI activities in the Microsoft Purview compliance portal.
Capacity settings	Manage any Power BI Premium capacities (EM or P SKU) that have been purchased for your organization.
Refresh summary	Schedule refresh on a capacity and also view the details of refreshes that have occurred.
Embed codes	View and manage the embed codes that have been generated for your organization to share reports publicly.
Organizational visuals	View, add, and manage which type of Power BI visuals users can access across the organization.
Azure connections	Configure and manage connections to Azure resources.
Workspaces	View and manage the workspaces that exist in your organization.
Custom branding	Change the look and feel of the Power BI service to match your organization's own branding.
Protection metrics	View a metric to monitor and track sensitivity label usage and adoption in your organization.
Featured content	Manage all the reports, dashboards, and apps that have been promoted to the Featured section on Power BI Home across your organization.

Recommend a monitoring and auditing solution for a data analytics environment including Power BI REST API and PowerShell cmdlets

Admin portal

The screenshot shows the 'Audit logs' section of the Microsoft 365 Admin Center. It includes a sidebar with links like Usage metrics, Users, Premium Per User, Audit logs (which is selected), Tenant settings, Capacity settings, Refresh summary, Embed Codes, Organizational visuals, Azure connections, Workspaces, Custom branding, Protection metrics, and Featured content. The main area displays 'R and Python visuals settings' and 'Audit and usage settings'. In the 'Audit and usage settings' section, there's a red box around the 'Usage metrics for content creators' subsection. This subsection contains a toggle switch labeled 'Enabled' (which is checked) and a dropdown menu for 'Apply to' with options: 'The entire organization' (selected), 'Specific security groups', and 'Except specific security groups'. There are also input fields for 'Enter security groups' and 'Delete all existing usage metrics content'. A yellow 'Apply' button is at the bottom.

Usage metrics for content creators

Admin portal

The screenshot shows the 'Audit logs' section of the Microsoft 365 Admin Center. It includes a sidebar with links like Usage metrics, Users, Premium Per User (preview), Audit logs (selected), and Tenant settings. The main area displays a message: 'Audit logs are managed in the Microsoft 365 Admin Center. Go there to view tenant activity and export logs.' Below this, it says 'Auditing is only available in certain regions while the feature is in preview. Learn more about audit logs' and has a 'Go to Microsoft 365 Admin Center' button.

Access your audit logs

[https://api.powerbi.com/v1.0/myorg/admin/activityevents
?startDateTime='2019-08-31T00:00:00'&endDateTime='2019-08-31T23:59:59'](https://api.powerbi.com/v1.0/myorg/admin/activityevents?startDateTime='2019-08-31T00:00:00'&endDateTime='2019-08-31T23:59:59')

[https://api.powerbi.com/v1.0/myorg/admin/activityevents
?continuationToken=%2BRID%3ARthsAlwfWGcVAAAAAA
AAAA%3D%3D%23RT%3A4%23TRC%3A20%23FPC%3AA
RUAAAAAAAAAFwAAAAAAAA%3D](https://api.powerbi.com/v1.0/myorg/admin/activityevents?continuationToken=%2BRID%3ARthsAlwfWGcVAAAAAA
AAAA%3D%3D%23RT%3A4%23TRC%3A20%23FPC%3AA
RUAAAAAAAAAFwAAAAAAAA%3D)

```
while(response.ContinuationToken != null)  
{  
    // Store the activity event results in a list for example  
  
    completeListOfActivityEvents.AddRange(response.ActivityEventEntities);  
  
    // Make another call to the API with continuation token  
    response =  
    GetPowerBIActivityEvents(response.ContinuationToken)  
}  
completeListOfActivityEvents.AddRange(response.ActivityEventEntities);
```

ActivityEvents REST API

Login-PowerBI

```
$activities = Get-PowerBIActivityEvent -StartTime  
'2019-08-31T00:00:00' -EndTime '2019-08-  
31T23:59:59' | ConvertFrom-Json
```

```
$activities.Count  
$activities[0]
```

--Filter activity data
Login-PowerBI

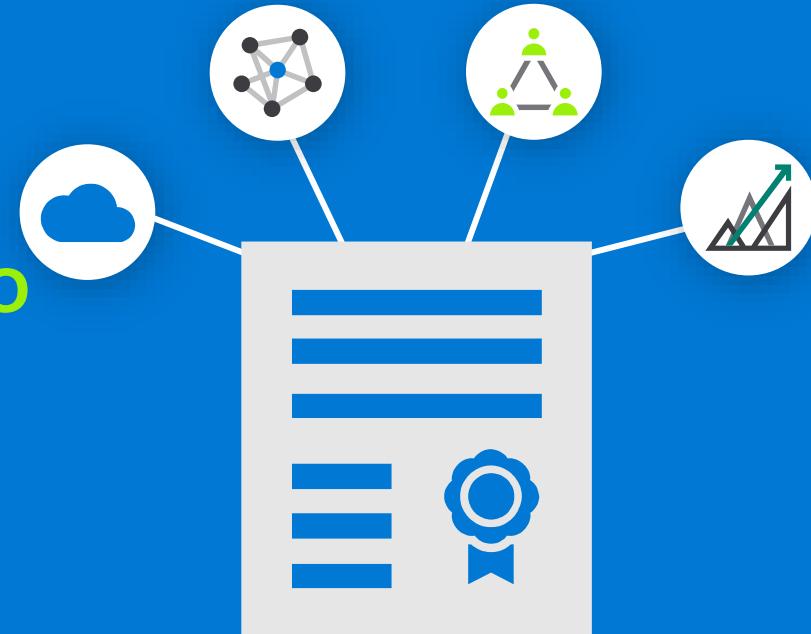
```
$activities = Get-PowerBIActivityEvent -StartTime  
'2019-08-31T00:00:00' -EndTime '2019-08-  
31T23:59:59' -ActivityType 'ViewDashboard' |  
ConvertFrom-Json
```

```
$activities.Count  
$activities[0]
```

Get-PowerBIActivityEvent cmdlet

Implement and manage a data analytics environment

Integrate an analytics platform into an existing IT infrastructure



- Identify requirements for a solution, including features, performance, and licensing strategy
- Configure and manage Power BI capacity
- Recommend and configure an on-premises gateway in Power BI
- Recommend and configure a Power BI tenant or workspace to integrate with Azure Data Lake Storage Gen2
- Integrate an existing Power BI workspace into Azure Synapse Analytics

Recommend and configure an on-premises gateway in Power BI

Personal gateways versus enterprise gateways

	On-premises Data Gateway (Personal Mode)	On-premises Data Gateway (aka Enterprise Gateway)
Target Persona	Business analyst sets up and uses the gateway for her data sources only	<ul style="list-style-type: none">BI Admins set up the gateway for their department/ companyMultiple users use the gateway setup by the admins
Features	<p>Import with scheduled refresh</p> <p>No additional data source configuration to set up a refresh</p> <p>Data source connections managed per user</p>	<p>Direct query and scheduled refresh</p> <p>Gateway admin needs to configure each data source explicitly</p> <p>Central data source management and access control</p>
Services supported	Power BI	Power BI, PowerApps, Microsoft Flow, Azure Logic Apps, Azure Analysis Services, Power Query Online with Dataverse

#Data connectivity modes available in Power BI

	Import	DirectQuery	Live/Exploration	Composite
Overview	ETL Data download	Select specific tables No data download Queries triggered from Report visuals	Explore source objects from Report surface No data download Queries triggered from Report visuals	Combine multiple data connections including direct query connections or import or a combination thereof
Supported Data Sources	All sources (>80 sources)	SQL Server Azure SQL Database Azure SQL Data Warehouse SAP HANA Oracle Teradata	SQL Server Analysis Services (Tabular & Multidimensional)	Following multidimensional data sources are NOT supported SAP HANA and BW SAP Business Warehouse SQL Server Analysis Services Power BI Datasets
Max # of data sources per report	Unlimited	One	One	Unlimited
Data Transformations	All transformations (100's)	Partial support (varies by data source)	None	All transformations for import queries Partial support for DirectQuery (by data source)
Mashup Capabilities	Merge (Joins) Append (Union) Parameterized queries	Merge (Joins) Append (Union)	None	Import capabilities for imported queries Direct Query capabilities for Direct queries
Modeling Capabilities	Relationships Calculated Columns & Tables Measures Hierarchies	Calculated Columns Measures Change Column Types	None	Relationship between queries from different data sources

How data is refreshed

	Import (cached mode)	Direct Query or Live connection
Refresh frequency	Scheduled – hourly (8x a day) Premium allows for 48x a day	Real-time
Refresh Performance	No noticeable delay since data is already cached	Depends on how fast the data source is as queries are executed in real-time
Data storage in Power BI	Since it is cached mode, data is stored in the cloud	No data is stored in Power BI. Data is always on-premises*
Data size	Current limit of 1GB (compressed) per model; Up to 10GB with Premium	The on-premises database is the limit; no Power BI limitation
Security	Can create row-level security on the Power BI dataset	Re-use on-premises row level security (for Analysis Services, additional SSO based on Kerberos)
Data sources	All on-premises data sources supported can be used in import mode	Select data sources

Recommend and configure a Power BI tenant or workspace to integrate with Azure Data Lake Storage Gen2



Power BI

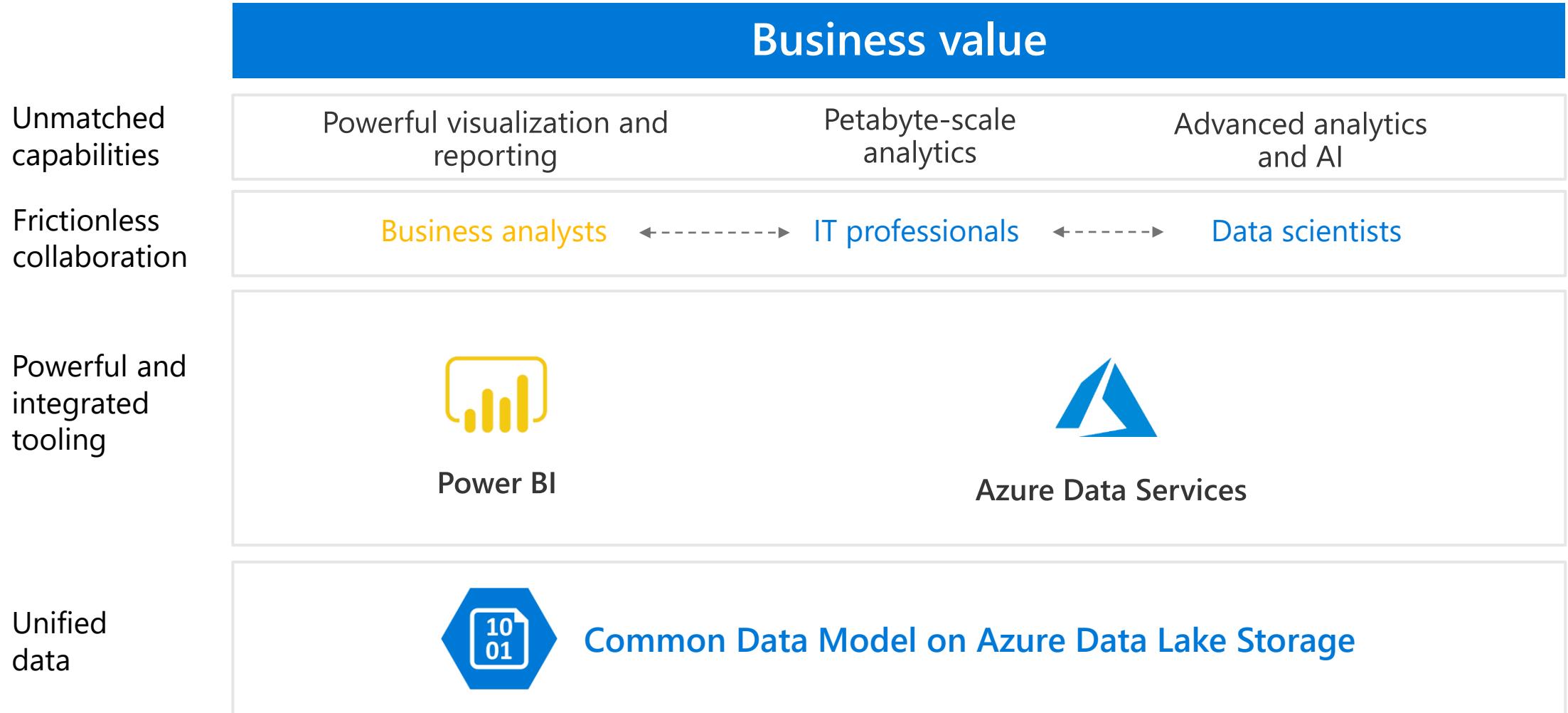


Azure Data Services



Common Data Model on Azure Data Lake Storage

Unleash data value with Power BI and Azure Data Services



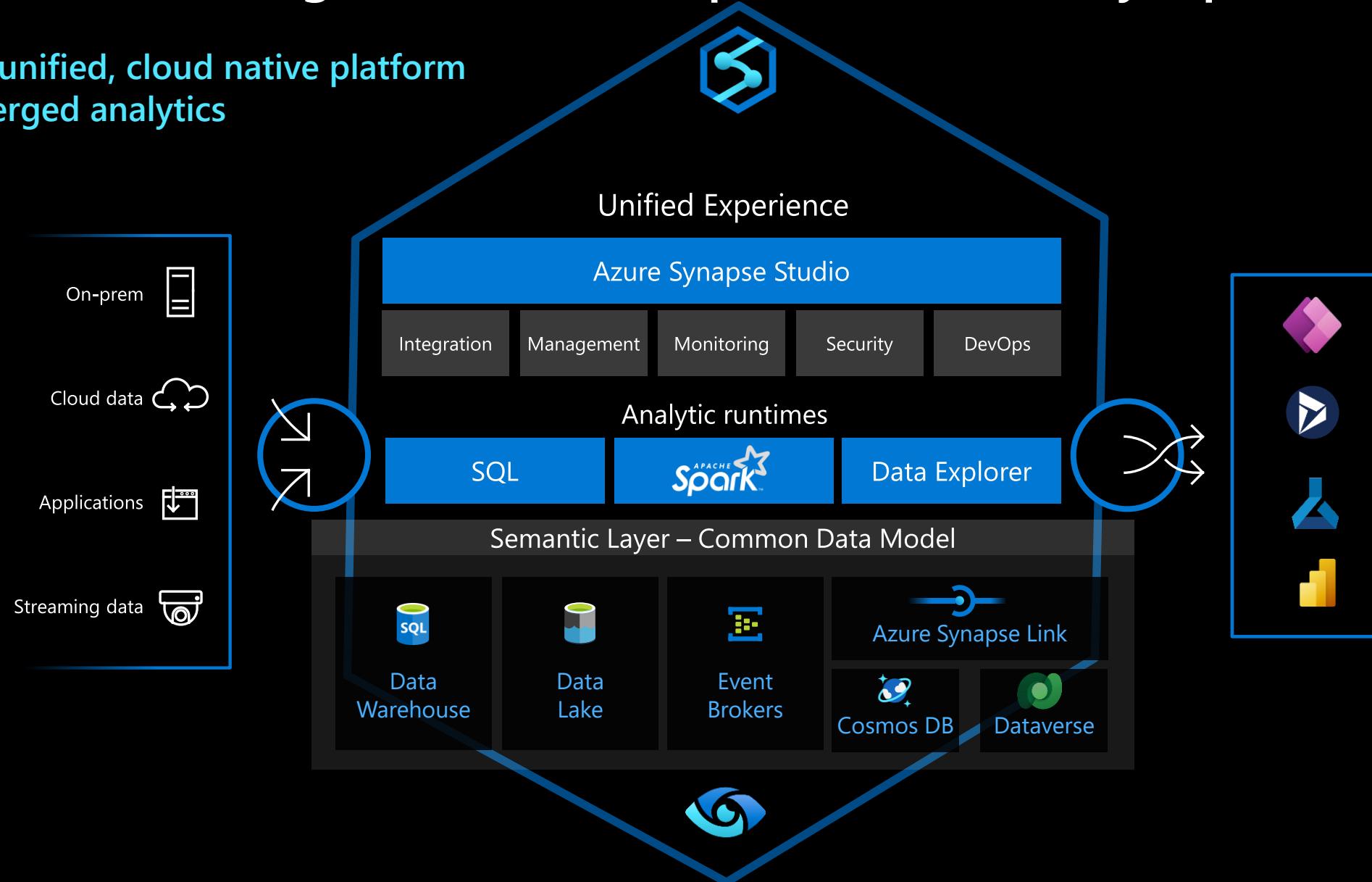
Modern BI = Serverless + Spark + ML

Modern BI <> Data visualization



Integrate an existing Power BI workspace into Azure Synapse Analytics

The first unified, cloud native platform
for converged analytics



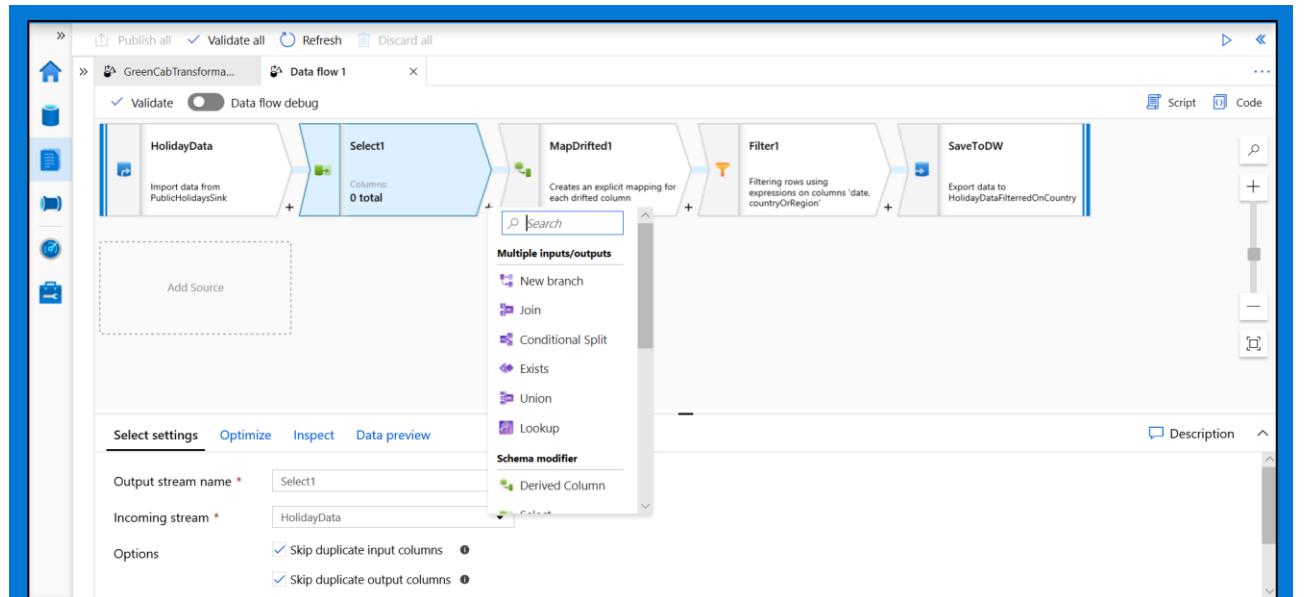
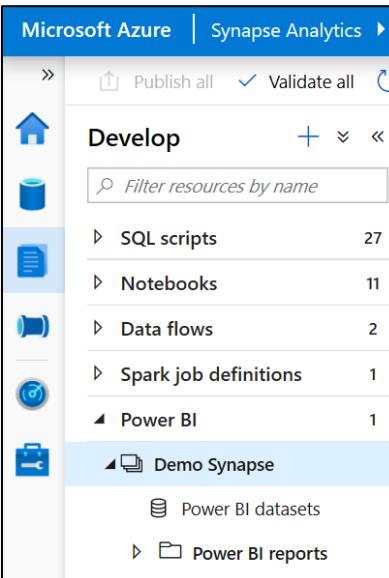
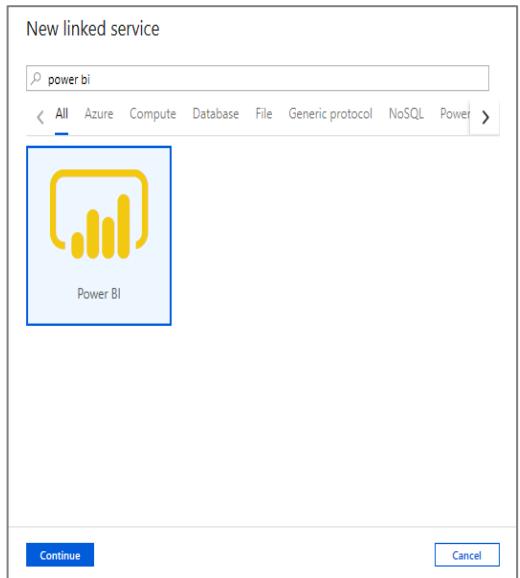
Integrate an existing Power BI workspace into Azure Synapse Analytics

Create Power BI reports in the workspace

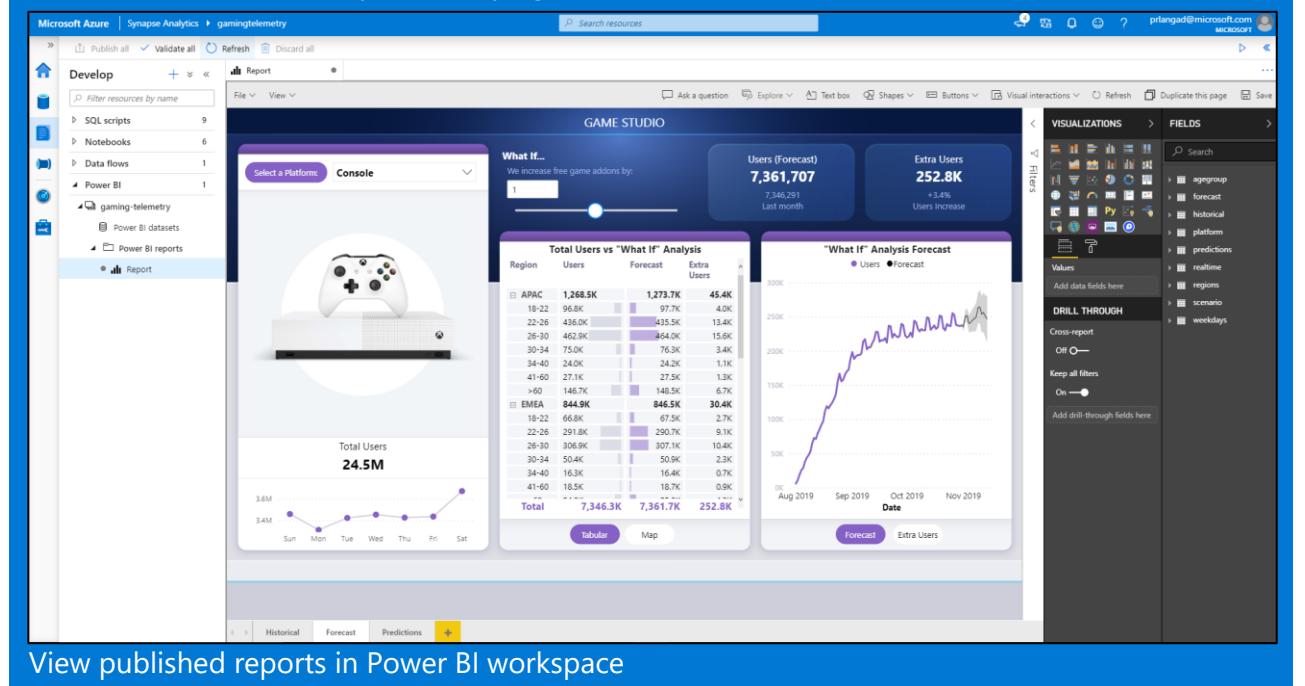
Provides access to published reports in the workspace

Update reports real time from Synapse workspace to get it reflected on Power BI service

Visually explore and analyze data



Data flows are a visual way of specifying how to transform data - Provides a code-free experience.



View published reports in Power BI workspace

Full Stack Analytics

Pro Dev

n/a

Azure Analysis Services

SQL Server Reporting Services

Azure Machine Learning

Azure Data Factory

Data Exploration

BI Semantic Models

Enterprise Reporting

AI/ML

Data prep

No-Code

Power BI Reports

Power BI Datasets

Power BI Paginated Reports

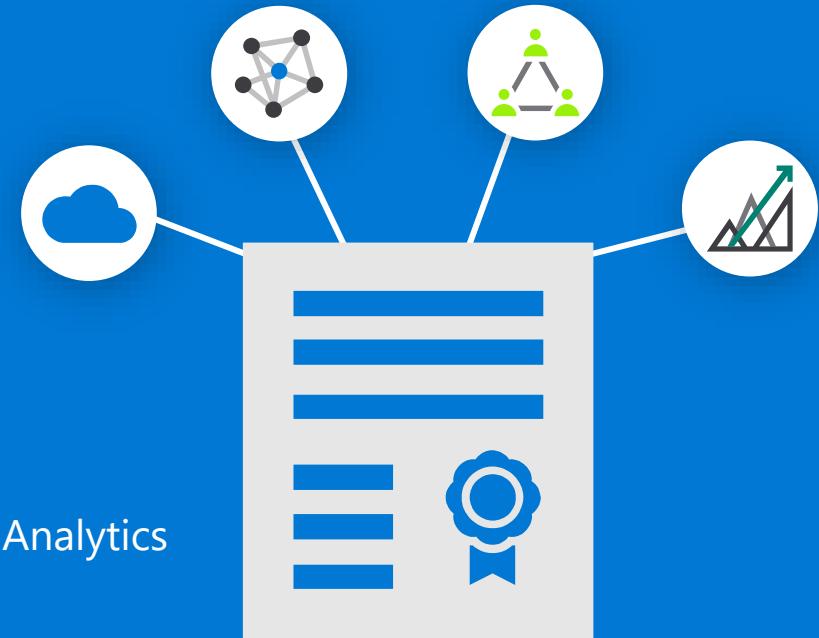
Power BI Augmented Analytics

Power BI Dataflows

Implement and manage a data analytics environment

Manage the analytics development lifecycle

- Commit code and artifacts to a source control repository in Azure Synapse Analytics
- Recommend a deployment strategy for Power BI assets
- Recommend a source control strategy for Power BI assets
- Implement and manage deployment pipelines in Power BI
- Perform impact analysis of downstream dependencies from dataflows and datasets
- Recommend automation solutions for the analytics development lifecycle, including Power BI REST API and PowerShell cmdlets
- Deploy and manage datasets by using the XMLA endpoint
- Create reusable assets, including Power BI templates, Power BI data source (.pbids) files, and shared datasets



Commit code and artifacts to a source control repository in Azure Synapse Analytics

Associate Synapse workspace with a Git repository, Azure DevOps, or GitHub

Microsoft Azure | Synapse Analytics > wsazuresynapseanalytics

Configure a repository

SynapseTestDemo

Specify the settings that you want to use when connecting to your repository.

Enter manually Use repository link

Git repository name * synapsetestdemo-ws-01

Collaboration branch * dev

Publish branch * main

Root folder * /

Import existing resource Import existing resources to repository

Import resource into this branch

Apply Back Cancel

Source control
Git configuration

Microsoft Azure | Synapse Analytics > wsazuresynapseanalytics

Configure a repository

/ main branch

Filter...

dev branch

main branch

workspace_publish branch

Create pull request [Alt+P]

New branch [Alt+N]

Switch to live mode

Validate all Commit all Publish

Setting Disconnect

Repository type GitHub

GitHub account SynapseTestDemo

Git repository name synapsetestdemo-ws-01

Collaboration branch dev

Publish branch main

Root folder /

Integration runtimes

Security

Access control

Credentials

Managed private endpoints

Source control

Git configuration

Recommend a deployment strategy for Power BI assets

The screenshot shows the Microsoft DXT Power BI Deployment pipelines interface. At the top, there's a navigation bar with icons for Microsoft DXT, Power BI, Deployment pipelines, Search, and user profile.

The main area displays a "MyFoods- Full production pipeline" with a subtitle "Demo pipeline with rules applied". On the left, a sidebar contains various icons for navigation.

The pipeline consists of three stages:

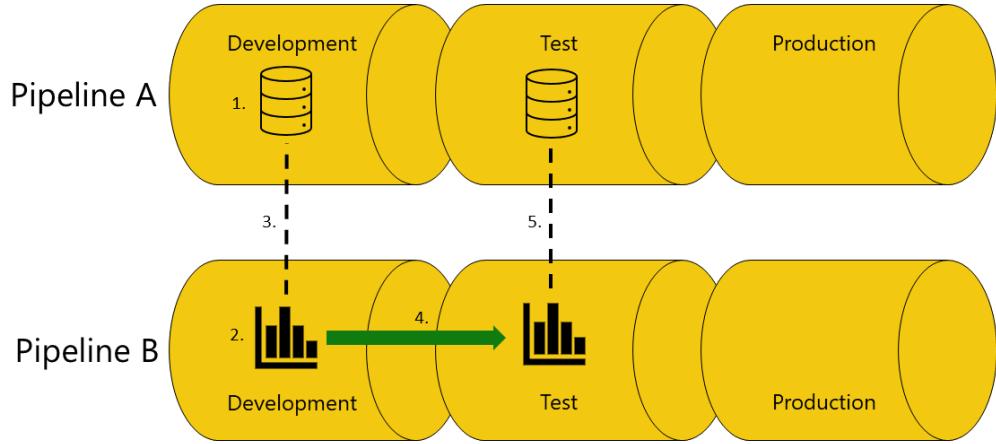
- Development**: Design, review, and revise your content in a development workspace. When it's ready to test and preview, deploy the content to the test stage.
- Test**: Test and verify your content in a preproduction workspace. When it's ready to distribute, deploy the content to the production stage.
- Production**: Your content has been tested and is ready to distribute to your consumers as an app or by access to the production workspace.

Below the stages, there are three cards for the "Sales of MyFoods.com" app across different environments:

- Dev Environment**: Shows 1 Dataflows (preview), 5 Datasets, 5 Reports, and 4 Dashboards. Includes "Show more" and "Publish app" buttons, and a yellow "Deploy to test" button.
- Test Environment**: Shows 1 Dataflows (preview), 5 Datasets, 5 Reports, and 4 Dashboards. Includes "Show more" and "Publish app" buttons, and a yellow "Deploy to production" button.
- Production Environment**: Shows 0 Dataflows (preview), 4 Datasets, 6 Reports, and 3 Dashboards. Includes "Show more" and "View app" buttons, and a yellow "Update app" button.

A large mouse cursor icon is visible in the bottom right corner of the interface.

Deploy content to an existing workspace



Deployed items

When you deploy content from one pipeline stage to another, the copied content contains the following Power BI items:

- Datasets
- Reports
- Dataflows
- Datamarts
- Dashboards
- Paginated reports

The screenshot shows the Power BI service interface with a list of deployed items. At the top, there's a header for 'Test' with a 'Learn more' link and a refresh button. Below the header, a message says 'Test and verify your content in a preproduction workspace. When it's ready to distribute, deploy the content to the production stage.' On the right, a context menu is open for a 'Sales and marketing doc...' item, which was deployed on 2/14/2022 at 12:53:5. The menu options are: Refresh now, Create report, Delete, Manage permissions, Security, Rename, Settings, View lineage (which is highlighted with a red box), and Configure rules. At the bottom of the list, there are buttons for 'Show less ▾', 'Publish app ↗', and a prominent yellow 'Deploy to production' button.

Dataflows	Datasets	Reports	Dashboards
1	2	4	2

Sales and marketing doc... Deployed: Deployed: 2/14/2022, 12:53:5

CDM
Marketing analysis
US_Sales_Analysis
Global sales and marketing
Marketing analysis
US_Sales_Analysis
RegionalSales

Show less ▾ Publish app ↗ Deploy to production

Automate your deployment pipeline using APIs and Azure DevOps

To achieve continuous integration and continuous delivery (CI/CD) of content, many organizations use various automation tools, including [Azure DevOps](#). Organizations that use Azure DevOps, can use the [Power BI automation tool](#) extension, which supports many of the deployment pipelines API operations.

Deployment pipelines API functions

The [deployment pipelines Power BI REST APIs](#), allow you to perform the following functions:

- **Get pipeline information** - Retrieve information about your pipelines and their content. Getting the pipeline information will enable you to dynamically build the deployment API calls. You can also check the [status of a deployment](#), or the [deployment history](#).
- **Deploy** - The REST calls enables developers to use any type of deployment available in the Power BI service.
- **Create and delete** pipelines - Use the [Create Pipeline](#) and [Delete Pipeline](#) to perform these operations.
- **Manage workspaces** - With [Assign Workspace](#) and [Unassign Workspace](#) you can assign and unassign workspaces to specific pipeline stages.
- **Manage pipeline users** - [Delete Pipeline User](#) lets you remove a user from a pipeline. [Update Pipeline User](#) allows you to add a user to your pipeline.

- **Deploy all** - A single API call that deploys all the content in the workspace to the next stage in the pipeline. For this operation, use the [Deploy all](#) API.
- **Selective deploy** - Deploy only specific Power BI items (such as reports or dashboards) in the pipeline. For this operation, use the [Selective deploy](#) API.
- **Backward deploy** - Use to deploy new Power BI items to the previous stage. Backward deployment only works if the Power BI items that are deployed, don't already exist in the target stage. For this operation use either the [Deploy all](#) or the [Selective deploy](#) APIs, with isBackwardDeployment set to True.
- **Update App** - As part of the deployment API call, you can update the content of the app that's related to that stage. Updated Power BI items will automatically be available to your end users, once a deployment has been completed. For this operation use either the [Deploy all](#) or the [Selective deploy](#) APIs, with [PipelineUpdateAppSettings](#).

Query and transform data



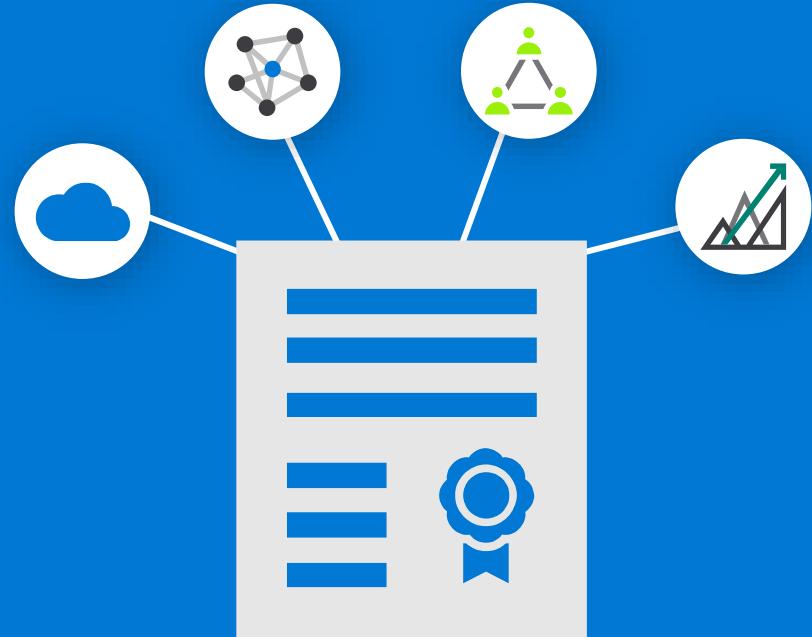
Query and transform data



Implement and manage data models

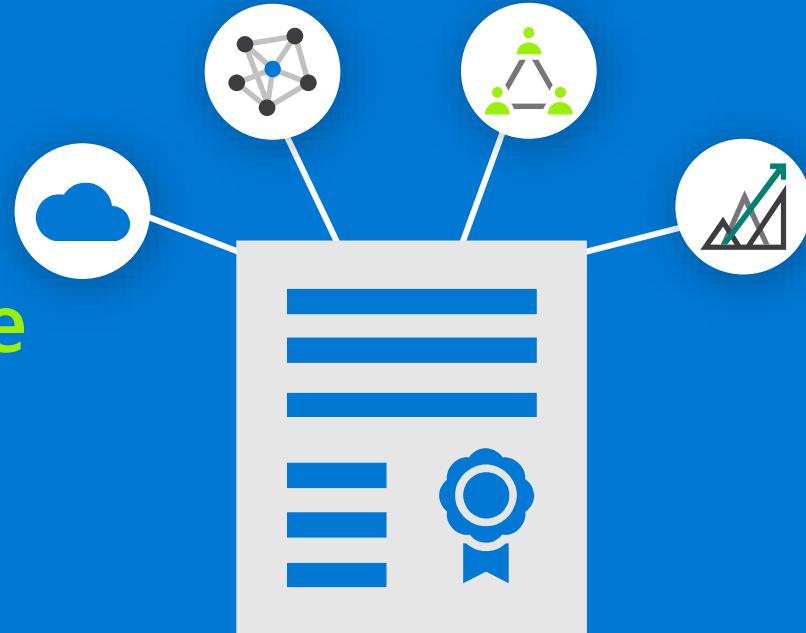


Explore and visualize data



Query and transform data

Query data by using Azure Synapse Analytics



- Identify an appropriate Azure Synapse pool when analyzing data
- Recommend appropriate file types for querying serverless SQL pools
- Query relational data sources in dedicated or serverless SQL pools, including querying partitioned data sources
- Use a machine learning PREDICT function in a query

Analyzing data

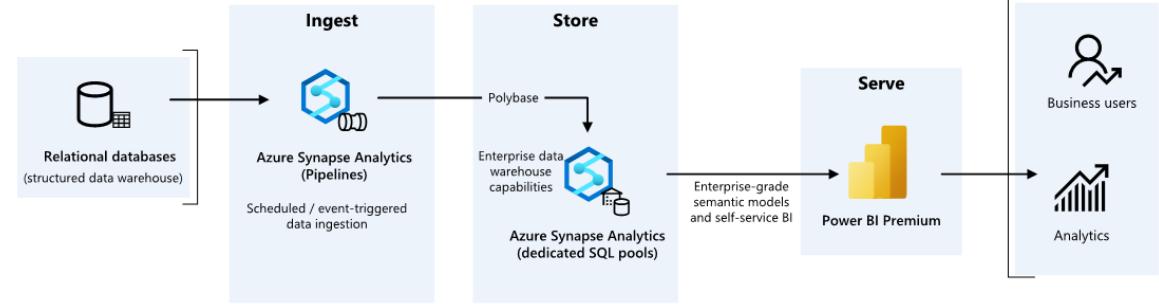
Using

Azure Synapse SQL pool

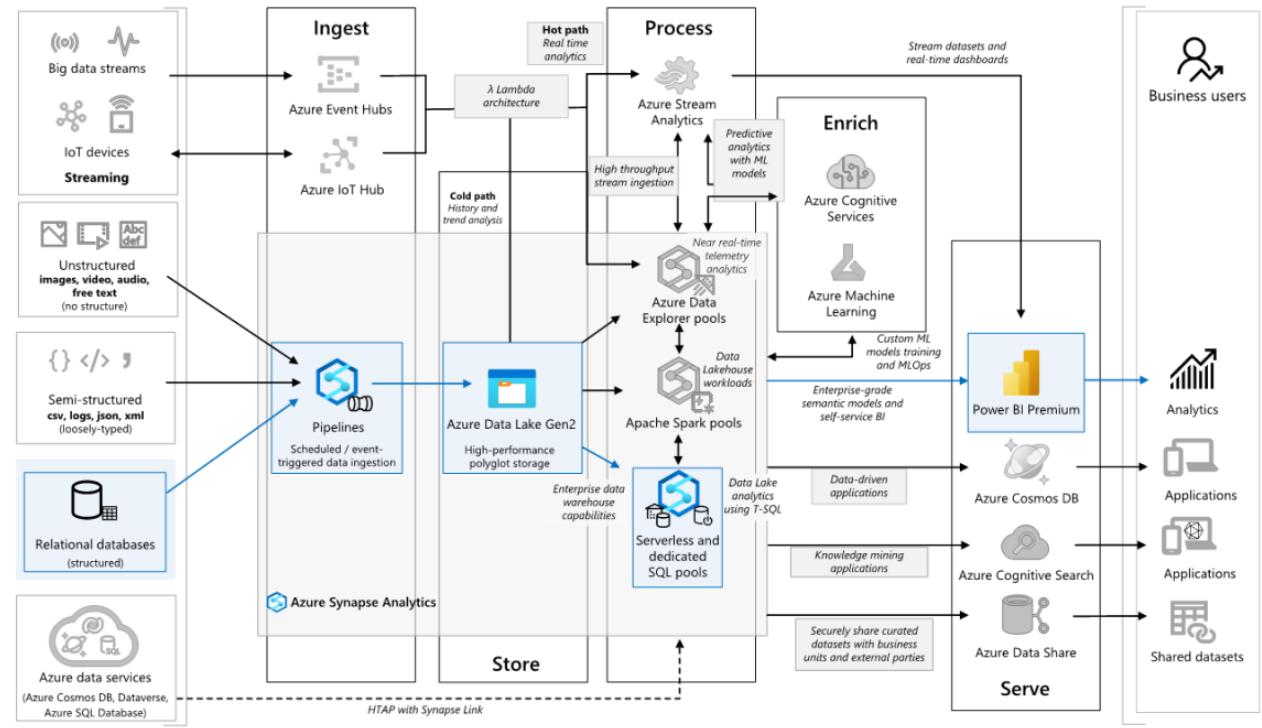
Azure Synapse SQL Serverless Pool

- Moteur SQL dédié**
 - Puissance dédiée scalable horizontalement
 - Moteur MPP (Massivement parallèle)
 - 60 nœuds de stockages et n nœuds de calcul
 - Réponse linéaire des requêtes
 - Exposition d'objets
 - Logiques sous la forme de tables externes
 - Logiques sous la forme de vues (pouvant être matérialisées, donc objets physiques)
 - Physiques sous la forme de tables ou vues avec données embarquées
- Moteur SQL Serverless**
 - Puissance "on demand"
 - Moteur non MPP
 - Scalabilité automatique et non linéaire (mais prédictible)
 - Exposition d'objets **logiques**

Simplified architecture



Architecture



Platform

Azure Active Directory

Cost Management

Azure Key Vault

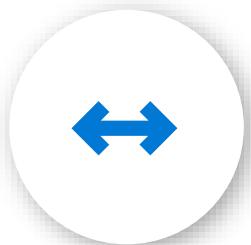
Azure Monitor

Microsoft Defender for Cloud

Azure DevOps & GitHub

Azure Policy

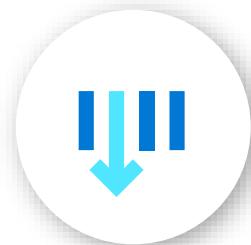
SQL Pool Performance



Elastic Architecture



Columnar Storage



Columnar Ordering

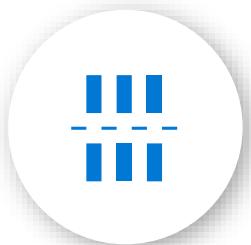


Table Partitioning



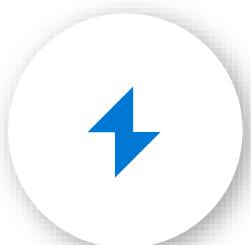
Secondary Indexes



Hash Distribution



Materialized Views



Resultset Cache

Serverless SQL pools

Recommend appropriate file types for querying

The image shows two side-by-side screenshots of the Microsoft Azure Synapse Analytics workspace.

Left Screenshot: A Power BI report titled "Sales Summary". It features a map of the world showing sales by territory, with a large green square highlighting a specific region. Two key figures are displayed: "\$7" (Average of UnitPrice) and "\$479". Below the map is a bar chart titled "Sales Amount by territory" showing sales for different territories, with one bar reaching \$479.36.

Right Screenshot: An SSMS-like interface for querying a "SQL DW Pool (SQL pool)". The query window contains the following T-SQL code:

```
1 SELECT TOP (100) [DateID]
2 ,[Date]
3 ,[DateKey]
4 ,[DayOfMonth]
5 ,[DaySuffix]
6 ,[DayName]
7 ,[DayOfWeek]
8 ,[DayOfWeekInMonth]
9 ,[DayOfWeekInYear]
10 ,[DayOfQuarter]
11 ,[DayOfYear]
12 ,[WeekOfMonth]
13 ,[WeekOfQuarter]
14 ,[WeekOfYear]
15 ,[Month]
16 ,[MonthName]
17 ,[MonthOfQuarter]
18 ,[Quarter]
19 ,[QuarterName]
20 ,[Year]
21 ,[YearName]
22 ,[MonthYear]
23 ,[MYYYY]
24 ,[FirstDayOfMonth]
```

The results pane displays a table with 24 columns corresponding to the selected date-related metrics. The first few rows of the result set are:

DATEID	DATE	DATEKEY	DAYOFMONTH	DAY_SUFFIX	DAYNAME	DAYOFWEEK	DAYOFWEEKIN...	DAYOFWEEKIN...	DAYOFQUARTER	DAYOFYEAR	WEEKOFMONTH	WEEKOFQU...
20100202	2010-02-02T00...	02/02/2010	2	2nd	Tuesday	3	1	5	5	33	1	5
20070612	2007-06-12T00...	06/12/2007	12	12th	Tuesday	3	2	24	11	163	3	11
20100207	2010-02-07T00...	02/07/2010	7	7th	Sunday	1	1	6	6	38	2	6
20010507	2001-05-07T00...	05/07/2001	7	7th	Monday	2	1	19	6	127	2	6
20130422	2013-04-22T00...	04/22/2013	22	22nd	Monday	2	4	16	4	112	4	4
20130417	2013-04-17T00...	04/17/2013	17	17th	Wednesday	4	3	16	3	107	3	3
20030425	2003-04-25T00...	04/25/2003	25	25th	Friday	6	4	17	4	115	4	4
20130420	2013-04-20T00...	04/20/2013	20	20th	Saturday	7	3	16	3	110	3	3

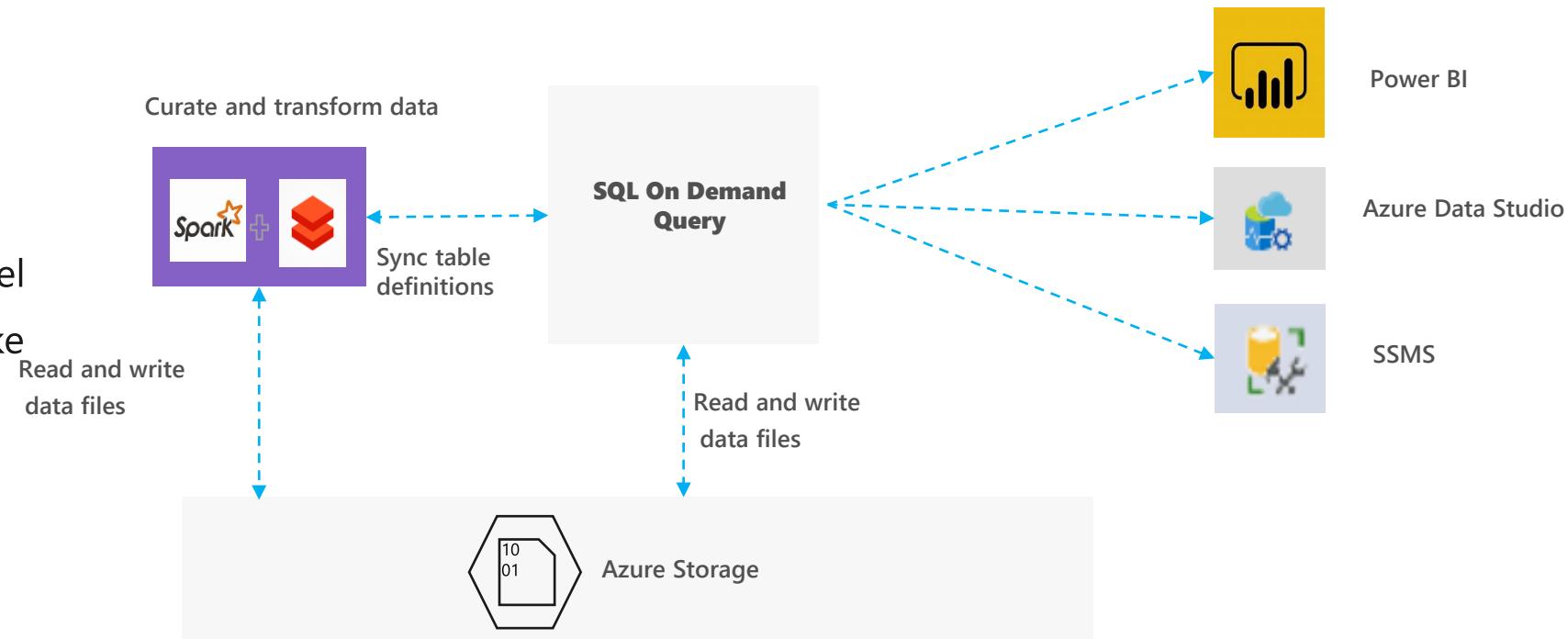
Transform with Synapse SQL Serverless / Pricing

Overview

An interactive query service that provides T-SQL queries over high scale data in Azure Storage.

Benefits

- Pay-per-query with serverless model
- Query data in-place on the data lake with T-SQL (no ETL)
- Supports various file formats (Parquet, CSV, JSON)
- Integrates with Databricks, HDInsight, PowerBI, and the shared Synapse metastore



Synapse SQL Serverless – Querying on storage

The screenshot displays two side-by-side views of the Microsoft Azure Synapse Analytics portal.

Left View: Shows the storage exploration interface under the 'Data' tab. A context menu is open over a file named 'part-00133-19836543-aea5b543-5e83-4a7d-8d31-69f72c50b05d-15253-1.c000.snappy.parquet'. The menu options include 'New SQL script' (which is highlighted), 'New notebook', 'Copy ABFS path', 'Manage Access...', 'Rename...', 'Download', 'Delete', and 'Properties...'. The left sidebar shows 'Storage accounts' (including 'nyctlc' which is selected), 'Databases' (including 'prlangaddemosa'), and 'Datasets'.

Right View: Shows the query editor interface. The 'SQL Analytics on-demand' dropdown is highlighted with a red box. The query window contains the following T-SQL code:

```
1 SELECT
2     TOP 100 *
3 FROM
4     OPENROWSET(
5         BULK 'https://prlangaddemosa.dfs.core.windows.net/nyctlc/yellow/puYear=2015/puMonth=3/part-00133-tid-210938564719836543-aea5b543-5e83-4a7d-8d31-69f72c50b05d-15253-1.c000.snappy.parquet'
6         FORMAT='PARQUET'
7     ) AS nyc;
```

The results pane shows a table with columns: VENDORID, TPEPICKUPDATETIME, TEPDROPPOFFDATETIME, PASSENGERCOUNT, TRIPDISTANCE, PULOCATIONID, DOLOCATIONID, STARTLON, STARTLAT, ENDLON, and ENDLAT. The results are as follows:

VENDORID	TPEPICKUPDATETIME	TEPDROPPOFFDATETIME	PASSENGERCOUNT	TRIPDISTANCE	PULOCATIONID	DOLOCATIONID	STARTLON	STARTLAT	ENDLON	ENDLAT
2	2015-02-28T23:5...	2015-03-01T00:0...	6	1.63	NULL	NULL	-74.000846862793	40.7306938171387	-73.	
1	2015-03-28T19:2...	2015-03-28T19:2...	1	2.2	NULL	NULL	-73.977653503418	40.7631607055664	-73.	
2	2015-02-28T23:5...	2015-03-01T00:1...	5	3.23	NULL	NULL	-73.96012878417...	40.7621574401855	-73.	
1	2015-03-28T19:2...	2015-03-28T19:3...	1	2.1	NULL	NULL	-73.98143005371...	40.7815055847168	-74.	
2	2015-02-28T23:5...	2015-03-01T00:1...	1	3.52	NULL	NULL	-73.98373413085...	40.7497062683105	-74.	

At the bottom of the results pane, a message states: '00:01:00 Query executed successfully.'

Synapse SQL Serverless – Querying CSV File

Overview

Uses OPENROWSET function to access data

Benefits

Ability to read CSV File with

- no header row, Windows style new line
- no header row, Unix-style new line
- header row, Unix-style new line
- header row, Unix-style new line, quoted
- header row, Unix-style new line, escape
- header row, Unix-style new line, tab-delimited
- without specifying all columns

```
SELECT *
FROM OPENROWSET(
    BULK 'https://XXX.blob.core.windows.net/csv/population/population.csv',
    FORMAT = 'CSV',
    FIELDTERMINATOR = ',',
    ROWTERMINATOR = '\n'
)
WITH (
    [country_code] VARCHAR (5) COLLATE Latin1_General_BIN2,
    [country_name] VARCHAR (100) COLLATE Latin1_General_BIN2,
    [year] smallint,
    [population] bigint
) AS [r]
WHERE
    country_name = 'Luxembourg'
    AND year = 2017
```

	country_code	country_name	year	population
1	LU	Luxembourg	2017	594130

Synapse SQL Serverless – Querying specific files

Overview

filename – Provides file name that originates row result

filepath – Provides full path when no parameter is passed or part of path when parameter is passed that originates result

Benefits

Provides source name/path of file/folder for row result set

```
SELECT
    r.filepath() AS filepath
    ,r.filepath(1) AS [year]
    ,r.filepath(2) AS [month]
    ,COUNT_BIG(*) AS [rows]
FROM OPENROWSET(
    BULK 'https://XXX.blob.core.windows.net/csv/taxi/yellow_tripdata_*.csv',
    FORMAT='CSV',
    FIRSTROW= 2 )
WITH (
    vendor_id INT,
    pickup_datetime DATETIME2,
    dropoff_datetime DATETIME2,
    passenger_count SMALLINT,
    trip_distance FLOAT,
    <...columns>
) AS [r]

WHERE r.filepath(1) IN ('2017')
    AND r.filepath(2) IN ('10', '11', '12')

GROUP BY r.filepath(),r.filepath(1),r.filepath(2)
ORDER BY filepath
```

Example of filename function

```
SELECT
    r.filename() AS [filename]
    ,COUNT_BIG(*) AS [rows]
FROM OPENROWSET(
    BULK 'https://XXX.blob.core.windows.net/csv/taxi/yellow_tripdata_2017-1*.csv',
    FORMAT = 'CSV',
    FIRSTROW = 2
)
WITH (
    vendor_id INT,
    pickup_datetime DATETIME2,
    dropoff_datetime DATETIME2,
    passenger_count SMALLINT,
    trip_distance FLOAT,
    <...columns>
) AS [r]
```

GROUP BY r.filename()

ORDER BY [filename]

	filename	rows
1	yellow_tripdata_2017-10.csv	9768815
2	yellow_tripdata_2017-11.csv	9284803
3	yellow_tripdata_2017-12.csv	9508276

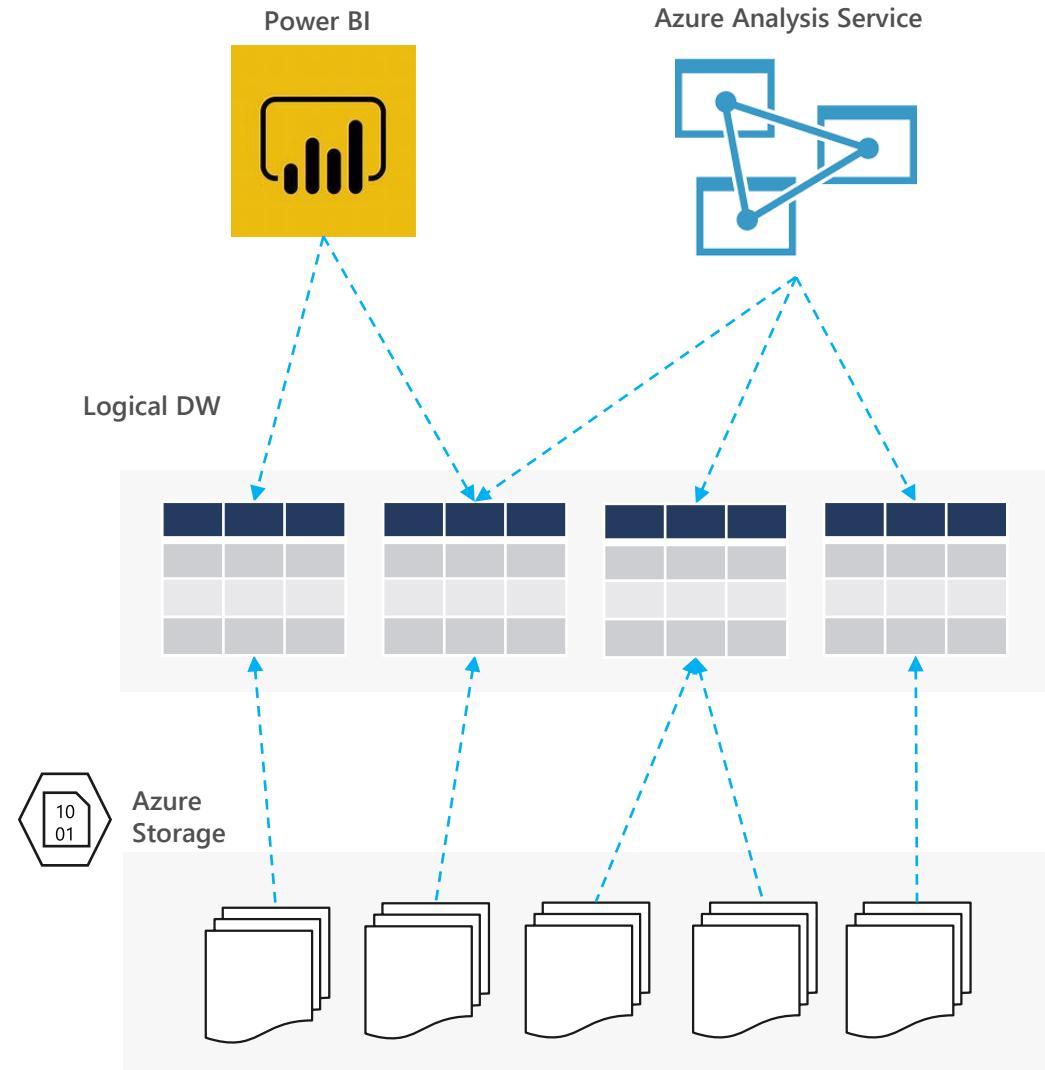
Synapse serverless SQL pool as a logical data warehouse

Overview

Logical relational layer on top of physical files in Azure Storage.

Benefits

- Abstract physical storage and file formats using well understandable relational concepts such as tables and views.
- Direct connector to Azure storage for large ecosystem of BI tools
- BI tools that use SQL can work with files on storage
 - Analytic tools use external tables that represent proxy to actual files.
 - No need for custom connectors in BI tools.
- Provides complex data processing (joining and aggregation) on top of raw files.
- Apply enterprise-ready security model and access control using battle-tested SQL Server permission model on top of Azure storage files



Logical Data Warehouse views

Overview

serverless SQL pool logical data warehouse views are created on external files placed in customer Azure storage

Benefits

Create SQL views on externally stored data

Access files using the view from various tools and language

Leverage rich T-SQL language to process and analyze data in external files exposed via views

Create PowerBI reports on the views created on external data

```
USE [mydbname]
GO

DROP VIEW IF EXISTS populationView
GO

CREATE VIEW populationView AS
SELECT *
FROM OPENROWSET(
    BULK 'https://XYZ.blob.core.windows.net/csv/population/*.csv',
    FORMAT = 'CSV',
    FIELDTERMINATOR = ',',
    ROWTERMINATOR = '\n'
)
WITH (
    [country_code] VARCHAR (5),
    [country_name] VARCHAR (100),
    [year] smallint,
    [population] bigint
) AS [r]
```

```
SELECT
    country_name, population
FROM populationView
WHERE
    [year] = 2019
ORDER BY
    [population] DESC
```

	country_name	population
1	China	1389618778
2	India	1311559204
3	United States	331883986
4	Indonesia	264935824
5	Pakistan	210797836
6	Brazil	210301591
7	Nigeria	208679114
8	Bangladesh	161062905
9	Russia	141944641
10	Mexico	127318112

Creating views

Microsoft Azure | Synapse Analytics > internalsandbox...

Data + <<

Storage accounts 1

Databases 3

Datasets 5

opendataset SQL script 1 SQL script 2 ...

Run Publish Query plan Connect to SQL on-demand Use database DefSQLOnDemand

```
1 CREATE VIEW yellow_2017 AS
2 Select *
3 FROM
4 OPENROWSET(
5     BULK 'https://internalsandboxwe.dfs.core.windows.net/opendataset/nyctlc/yellow/puYear=2017/\*/\*',
6     FORMAT='PARQUET'
7 ) AS [r];
```

Results Messages

00:00:03 Query e

Microsoft Azure | Synapse Analytics > internalsandbox...

Develop + <<

opendataset SQL script 1 SQL script 2 SQL script 3 ...

Run Publish Query plan Connect to SQL on-demand Use database

```
-- type your sql script here, we now have intellisense
SELECT
    YEAR(tpepPickupDateTime),
    passengerCount,
    COUNT(*) AS cnt
FROM
    yellow_2017
GROUP BY
    passengerCount,
    YEAR(tpepPickupDateTime)
ORDER BY
    YEAR(tpepPickupDateTime),
    passengerCount
```

Results Messages

00:01:00 Query executed successfully.

Microsoft Azure | Synapse Analytics > internalsandbox...

Develop + <<

opendataset SQL script 1 SQL script 2 SQL script 3 ...

Run Publish Query plan Connect to SQL on-demand Use database

```
-- type your sql script here, we now have intellisense
SELECT
    YEAR(tpepPickupDateTime),
    passengerCount,
    COUNT(*) AS cnt
FROM
    yellow_2017
GROUP BY
    passengerCount,
    YEAR(tpepPickupDateTime)
ORDER BY
    YEAR(tpepPickupDateTime),
    passengerCount
```

Results Messages

View Table Chart Save as image

00:01:00 Query executed successfully.

Chart type: Line

Category column: (none)

Legend (series) columns: Column 0, passengerCount, cnt

Legend position: center - bottom

Legend (series) label:

(NO COLUMN NAME)	PASSENGERCOUNT	CNT
2017	0	166086
2017	1	81034075
2017	2	16545571
2017	3	4748869
2017	4	2257813
2017	5	5407319

Logical Data Warehouse - tables

Overview

Create external tables that reference external files in your serverless SQL pool logical data warehouse

Benefits

Create external tables that reference set of files on Azure storage.

Join and transform multiple tables in the same query.

Enables you to analyze external files with the same experience that you have in classic databases.

Manage column statistics in external tables.

Manage access rights per table.

Create PowerBI reports on the views created on external data

```
USE [mydbname]
```

```
GO
```

```
DROP TABLE IF EXISTS dbo.Population
```

```
GO
```

```
CREATE EXTERNAL TABLE dbo.Population (
```

```
country_code VARCHAR (5) COLLATE Latin1_General_BIN2,  
country_name VARCHAR (100) COLLATE Latin1_General_BIN2,  
year smallint,  
population bigint
```

```
)
```

```
WITH(
```

```
LOCATION = '/csv/population/population-* .csv',  
DATA_SOURCE = MyAzureStorage,  
FILE_FORMAT = MyAzureCSVFormat
```

```
)
```

```
CREATE STATISTICS stat_country_name  
ON dbo.Population(country_name);
```

```
SELECT
```

```
country_name, population
```

```
FROM population
```

```
WHERE year = 2019
```

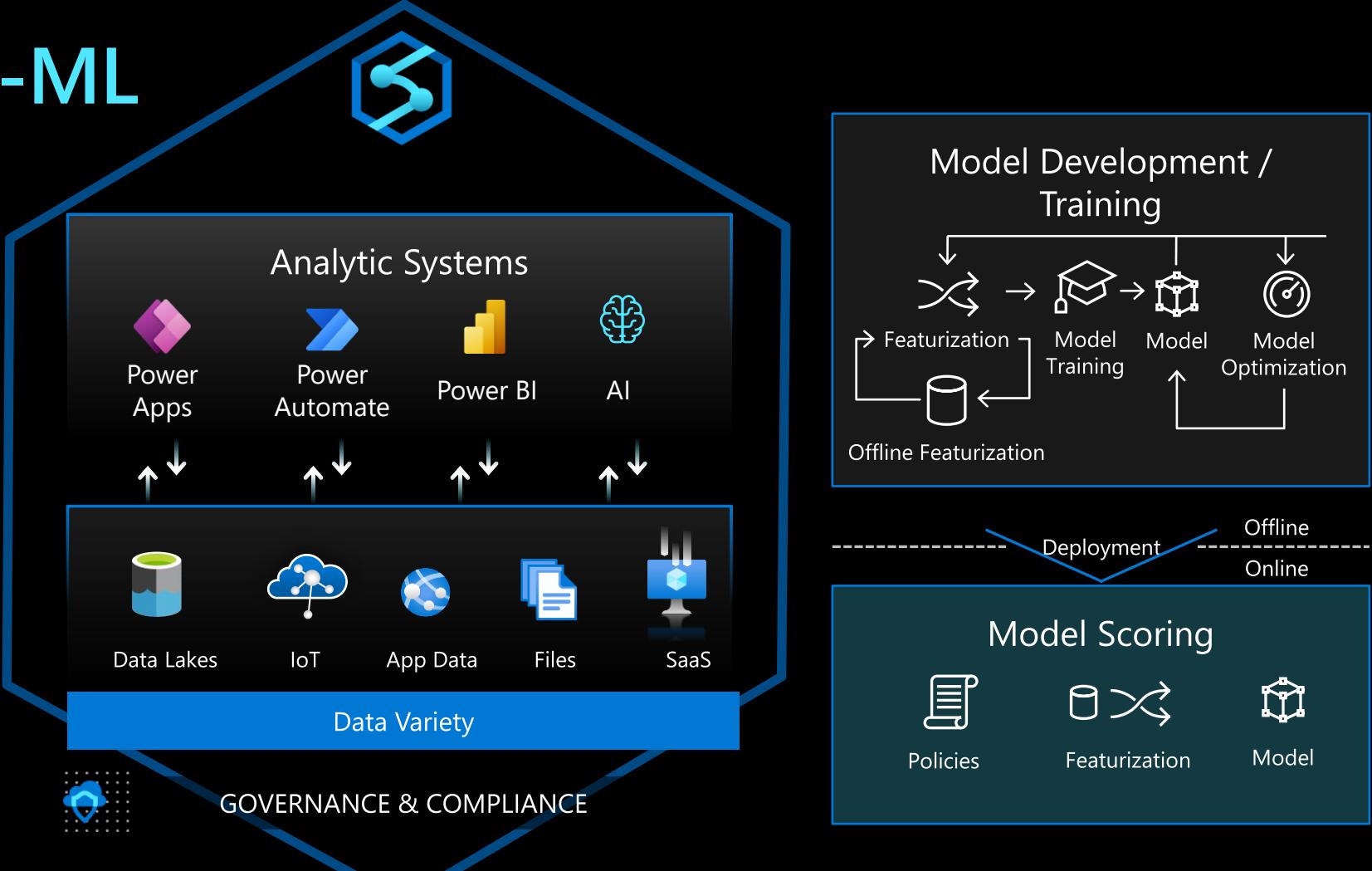
```
ORDER BY population DESC
```

	country_name	population
1	China	1389618778
2	India	1311559204
3	United States	331883986
4	Indonesia	264935824
5	Pakistan	210797836
6	Brazil	210301591
7	Nigeria	208679114
8	Bangladesh	161062905
9	Russia	141944641
10	Mexico	127318112

Why Synapse Enterprise Grade-ML

Productizing AI

- Train in the cloud within the Hub
- Scoring with operational systems
- Governance everywhere (models, lineage)
- Ethical AI
- Control over deployment
- Deployment across Apps, BI, Processes
- Exchange of models (ONNX)
- Enabling Reinforcement learning



```
DECLARE @model VARBINARY(max) = (SELECT model FROM scoring_model WHERE model_name = 'ScoringModelV1');
```

```
INSERT INTO loan_applications (c1, c2, c3, c4, score)
SELECT d.c1, d.c2, d.c3, d.c4, p.score
FROM PREDICT(MODEL = @model, DATA = dbo.mytable AS d) WITH(score FLOAT) AS p;
```

Query and transform data

Ingest and transform data by using Power BI



- Identify data loading performance bottlenecks in Power Query or data sources
- Implement performance improvements in Power Query and data sources
- Create and manage scalable Power BI dataflows
- Identify and manage privacy settings on data sources
- Create queries, functions, and parameters by using the Power Query Advanced Editor
- Query advanced data sources, including JSON, Parquet, APIs, and Azure Machine Learning models

Create queries, functions, and parameters by using the Power Query Advanced Editor

The screenshot illustrates the creation of M query parameters in Power BI Desktop. It shows two main windows: the Power Query Advanced Editor and the Power BI Data View.

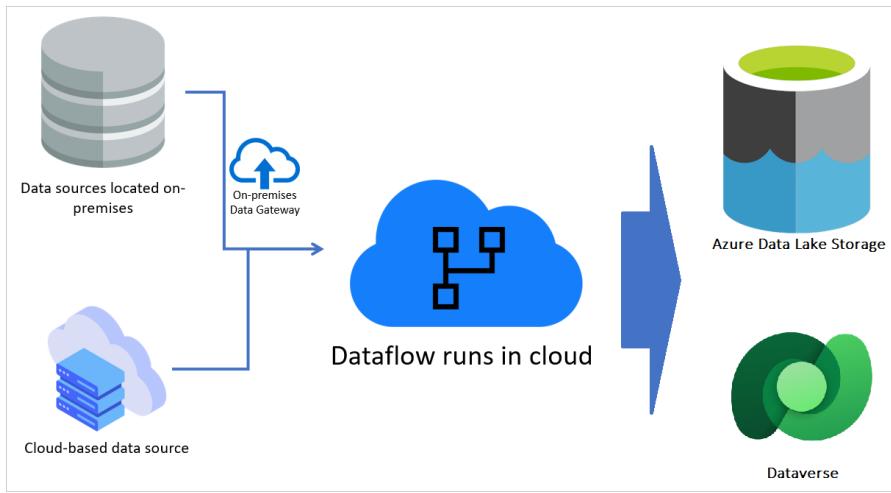
Power Query Advanced Editor: The top window shows the ribbon with "Manage parameters" highlighted. A red arrow points down to the "Manage parameters" dialog, which is displayed below. This dialog allows creating a new parameter named "My Parameter" with a description, type (Text), suggested values (Option 1, Option 2), and default value.

Power BI Data View: The bottom window shows a table with columns: OrderID, Total Units, and % Margin. A context menu is open over the % Margin column, with the "Greater than..." option highlighted with a red box. This leads to the "Filter rows" dialog, where a condition "Margin is greater than 1.2 Minimum Margin" is set using the "Basic" filter type.

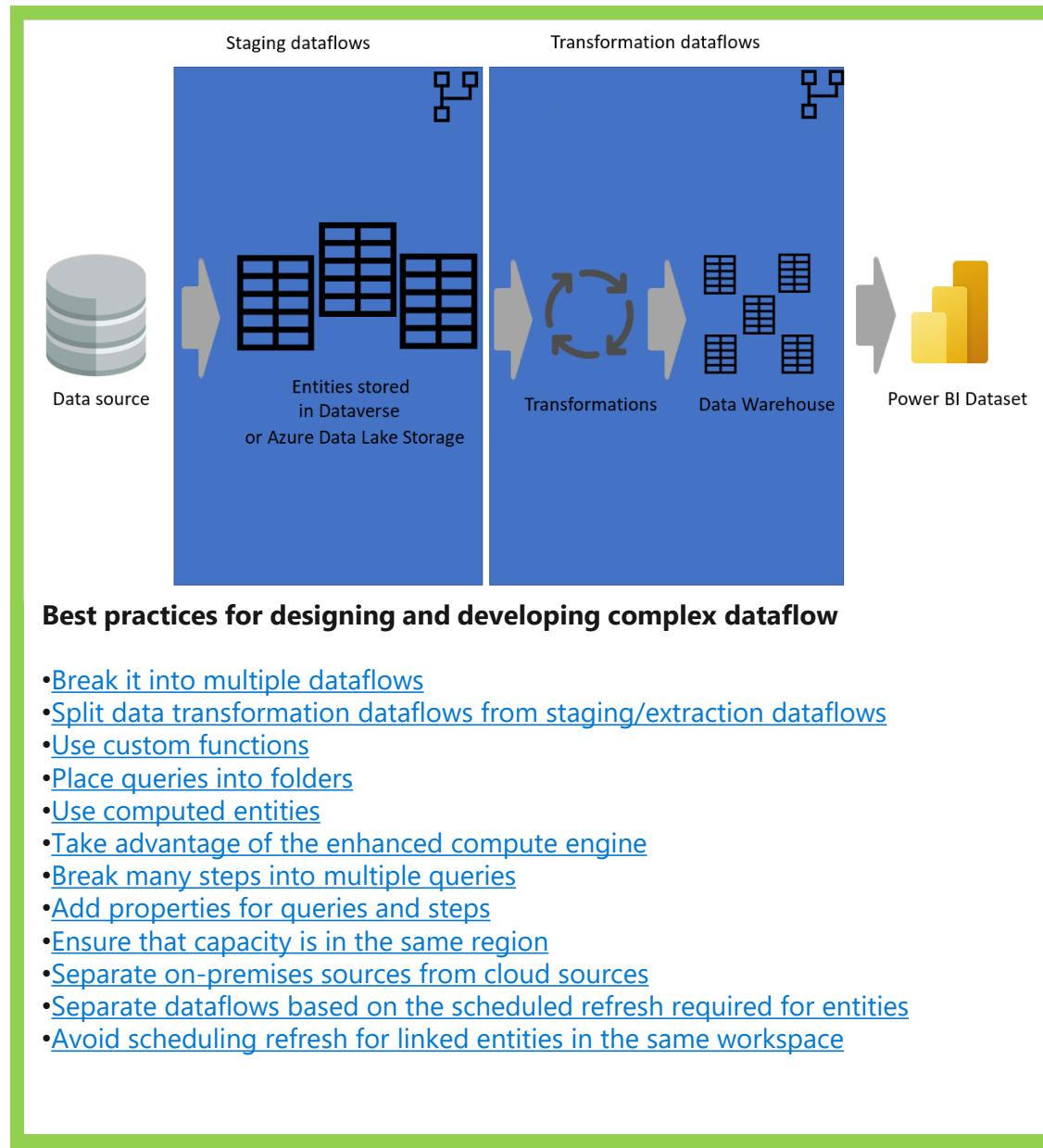
M query parameters in Power BI Desktop applied on #functions and #queries

Dynamic M query parameters in Power BI Desktop

Create and manage scalable Power BI dataflows



Dataflows run in the cloud



Best practices for designing and developing complex dataflow

- [Break it into multiple dataflows](#)
- [Split data transformation dataflows from staging/extraction dataflows](#)
- [Use custom functions](#)
- [Place queries into folders](#)
- [Use computed entities](#)
- [Take advantage of the enhanced compute engine](#)
- [Break many steps into multiple queries](#)
- [Add properties for queries and steps](#)
- [Ensure that capacity is in the same region](#)
- [Separate on-premises sources from cloud sources](#)
- [Separate dataflows based on the scheduled refresh required for entities](#)
- [Avoid scheduling refresh for linked entities in the same workspace](#)

Dataflows use a powerful transformation engine

Implement and manage data models



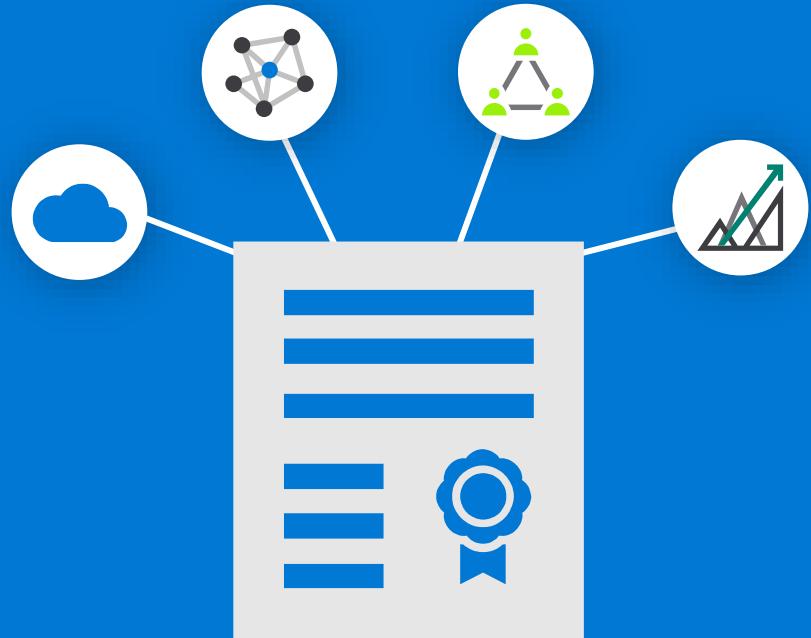
Query and transform data

A light gray rounded rectangle containing a white icon of two computer monitors connected to a cloud.

Implement and manage data models

A light gray rounded rectangle containing a white icon of two computer monitors connected to a cloud.

Explore and visualize data

A light gray rounded rectangle containing a white icon of two computer monitors connected to a cloud.

Implement and manage data models

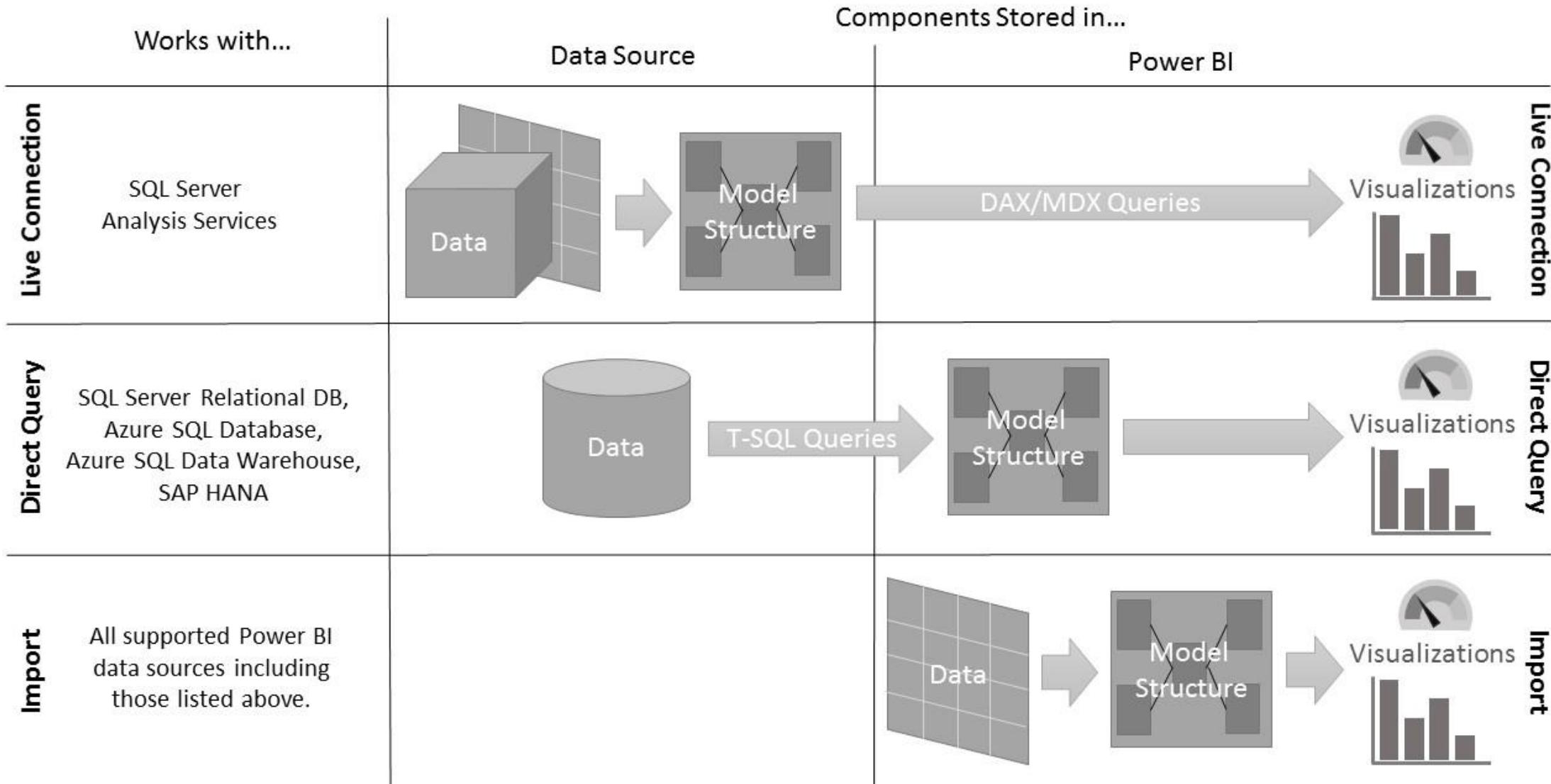
Design and build tabular models



- Choose when to use DirectQuery for Power BI datasets
- Choose when to use external tools, including DAX Studio and Tabular Editor 2
- Create calculation groups
- Write calculations that use DAX variables and functions, for example handling blanks or errors, creating virtual relationships, and working with iterators
- Design and build a large format dataset
- Design and build composite models, including aggregations
- Design and implement enterprise-scale row-level security and object-level security

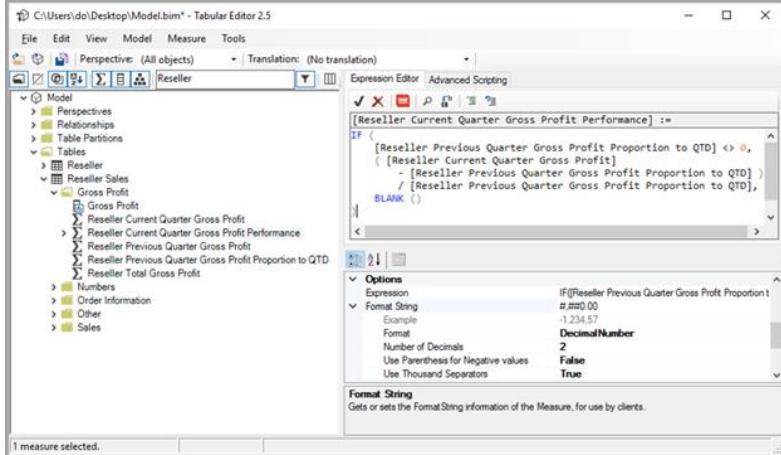
Choose when to use DirectQuery for Power BI datasets

Power BI Connection Types



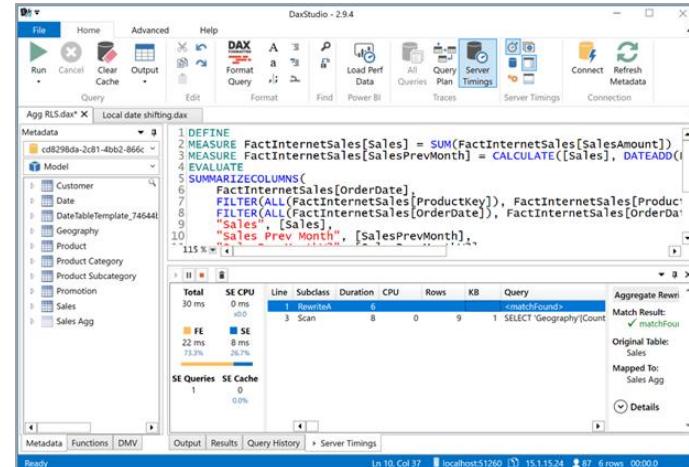
Choose when to use external tools including DAX Studio and Tabular Editor 2

Tabular Editor



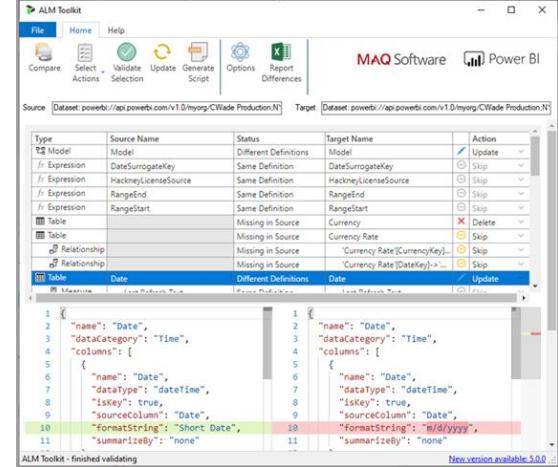
Tabular Editor is an open-source tool for creating, maintaining, and managing tabular models using an intuitive, lightweight editor. You can use this as an alternative to SQL Server Data Tools (SSDT) for all tabular models for analysis services, and it can be used without the workspace server. It provides a hierarchical view shows all objects in your tabular model. Objects are organized by display folders with support for multi-select property editing and DAX syntax highlighting. XMLA read-only is required for query operations. Read-write is required for metadata operations. To learn more, view tabulareditor.github.io.

DAX Studio



DAX Studio is an open-source tool for DAX authoring, diagnosis, performance tuning, and analysis. Features include object browsing, integrated tracing, query execution breakdowns with detailed statistics, DAX syntax highlighting and formatting. XMLA read-only is required for query operations. To learn more, view daxstudio.org.

ALM Toolkit



ALM Toolkit is an open-source schema compare tool for Power BI datasets, most often used for application lifecycle management (ALM) scenarios. Perform deployment across environments and retain incremental refresh historical data. Diff and merge metadata files, branches, and repos. Reuse common definitions between datasets. Read-only is required for query operations. Read-write is required for metadata operations. To learn more, see alm-toolkit.com.

Create calculation groups

Measures



Year	Previous Year			Year over Year		
	Orders	Sales	Profit	Orders	Sales	Profit
■ FY2019	24,349	\$23,860,891	\$3,035,934	58,990	\$10,209,217	\$672,067
+ FY2019 Q1	6,236	\$6,005,207	\$960,439	21,886	\$4,186,891	(\$87,233)
+ FY2019 Q2	5,800	\$5,923,348	\$920,415	14,739	\$2,469,110	(\$35,041)
+ FY2019 Q3	5,215	\$5,965,011	\$912,421	7,943	\$804,477	(\$87,632)
+ FY2019 Q4	7,098	\$5,967,325	\$242,659	14,422	\$2,748,740	\$881,973
■ FY2020	83,339	\$34,070,109	\$3,708,000	83,749	\$17,808,166	\$2,099,432
+ FY2020 Q1	28,122	\$10,192,098	\$873,205	23,512	\$3,672,219	(\$360,712)
+ FY2020 Q2	20,539	\$8,392,458	\$885,374	22,105	\$5,152,782	\$928,118
+ FY2020 Q3	13,158	\$6,769,488	\$824,789	19,074	\$4,965,578	\$1,111,852
+ FY2020 Q4	21,520	\$8,716,064	\$1,124,632	19,058	\$4,017,587	\$420,175
Total	107,688	\$57,931,000	\$6,743,934	142,739	\$28,017,383	\$2,771,499

Calculation Items

Benefits of using a calculation group

The main benefit of using calculation groups is a reduction in the overall number of measures you need to create and maintain. Calculation groups also enable the creation of creative report features, such as switching measures using a slicer, dynamic formatting, and even turning display labels on and off.

The recommended tool for creating calculation groups in Power BI is the free, open-source [Tabular Editor 2.x](#) tool. Tabular Editor 2.x lets you manipulate and manage measures, calculated columns, display folders, perspectives, and translations in Analysis Services Tabular and Power BI XMLA Models (from Compatibility Level 1200 and onwards). Power BI Desktop doesn't have the user interface to create calculation groups.

Write calculations that use DAX variables and functions

for example handling blanks or errors, creating virtual relationships, and working with iterators

PERCENTILE()
**A
LCULATETABLE()**
 CONTAINS()
 EVALUATE
 ROW()
 EOMONTH()
 DIVIDE()
 FIRSTNONBLANK()
 SAMEPERIODLASTYEAR()

IF()
SWITCH()
MEASURES
CALCULATED COLUMNS
CALCULATE()
FILTER()
SUM()
SUMX()
RANKX()
ALL()
ALLSELECTED()
 ISBLANK()
DATE()
 NOW()
 PERTVIOUSMONTH()
SUMMARIZE()
DATESMTD()
EARLIER()
DATEDIFF()
TIME()
 DATESINPERIOD()
HASONEVALUE()
TODAY()
TOTALMTD()
CONFIDENCE.NORM

Learning DAX



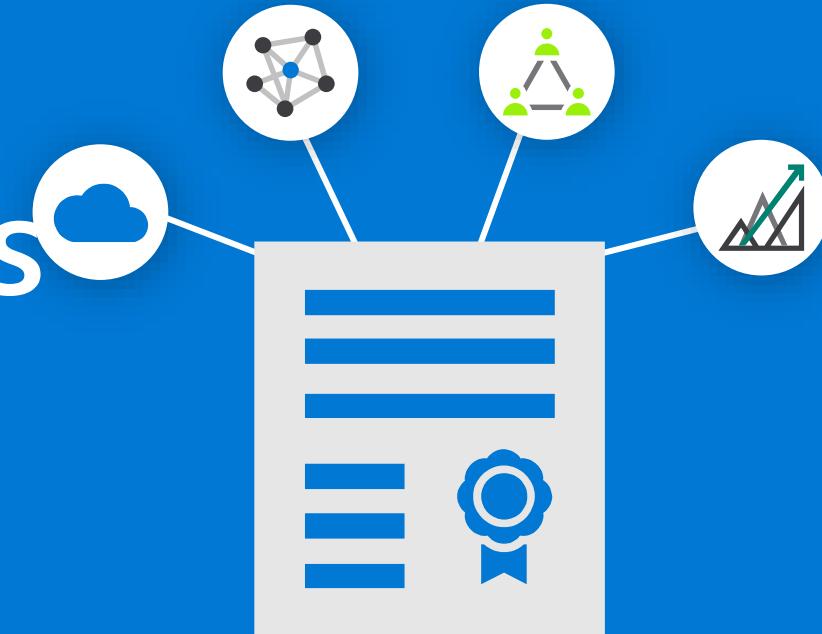
- Understanding essential concepts is more important than memorizing functions

- You can always lookup function syntax
- Keeping a library of working examples may be more valuable than a web search

ACOS	CONFIDENCE.T	EOMONTH	ISNOTTEXT
ACOSH	CONTAINS	EVEN	ISNUMBER
ACOT	COS	EXACT	ISO.CEILING
ACOTH	COSH	EXCEPT	ISODD
ADDCOLUMNS	COT	EXP	ISONORAFTER
ADDMISSINGITEMS	COTH	EXPON.DIST	ISSUBTOTAL
ALL	COUNT	FACT	ISTEXT
ALLEXCEPT	COUNTA	FILTER	KEEPFILTERS
ALLNONBLANKROW	COUNTAX	FILTERS	LASTDATE
ALLSELECTED	COUNTBLANK	FIND	LASTNONBLANK
AND	COUNTROWS	FIRSTDATE	LCM
ASIN	COUNTX	FIRSTNONBLANK	LEFT
ASINH	CROSSFILTER	FIXED	LEN
ATAN	CROSSJOIN	FLOOR	LN
ATANH	CURRENCY	FORMAT	LOG
AVERAGE	CURRENTGROUP	GCD	LOG10
AVERAGEA	CUSTOMDATA	GENERATE	LOOKUPVALUE
AVERAGEG	DATABLE	GENERATEALL	LOWER
BETA.DIST	DATE	GEOMEAN	MAX
BETA.INV	DATEADD	GEOMEANX	MAXA
BLANK	DATEDIFF	GROUPBY	MAXX
CALCULATE	DATESBETWEEN	HASONEFILTER	MEDIAN
CALCULATETABLE	DATESINPERIOD	HASONEVALUE	MEDIANX
CALENDAR	DATESMTD	HOUR	MID
CALENDARAUTO	DATESQTD	IF	MIN
CEILING	DATESYTD	IFERROR	MINA
CHISQ.DIST	DATEVALUE	IGNORE	MINUTE
CHISQ.DIST.RT	DAY	INT	MINX
CHISQ.INV	DEGREES	INTERSECT	MOD
CHISQ.INV.RT	DISTINCT	ISBLANK	MONTH
CLOSINGBALANCEMONTH	DISTINCTCOUNT	ISCROSSFILTERED	MROUND
CLOSINGBALANCEQUARTER	DIVIDE	ISEMPTY	NATURALINNERJOIN
CLOSINGBALANCEYEAR	EARLIER	ISERROR	NATURALLEFTOUTERJOIN
COMBIN	EARLIEST	ISEVEN	NEXTDAY
COMBINA	EDATE	ISFILTERED	NEXTMONTH
CONCATENATE	ENDOFMONTH	ISLOGICAL	NEXTQUARTER
CONCATENATEX	ENDOFQUARTER		NEXTYEAR
CONFIDENCE.NORM	ENDOFYEAR		NOT
			NOW
			...

Implement and manage data models

Optimize enterprise-scale data models



- Identify and implement performance improvements in queries and report visuals
- Troubleshoot DAX performance by using DAX Studio
- Optimize a data model by using Tabular Editor 2
- Analyze data model efficiency by using VertiPaq Analyzer
- Implement incremental refresh
- Optimize a data model by using denormalization

Explore and visualize data



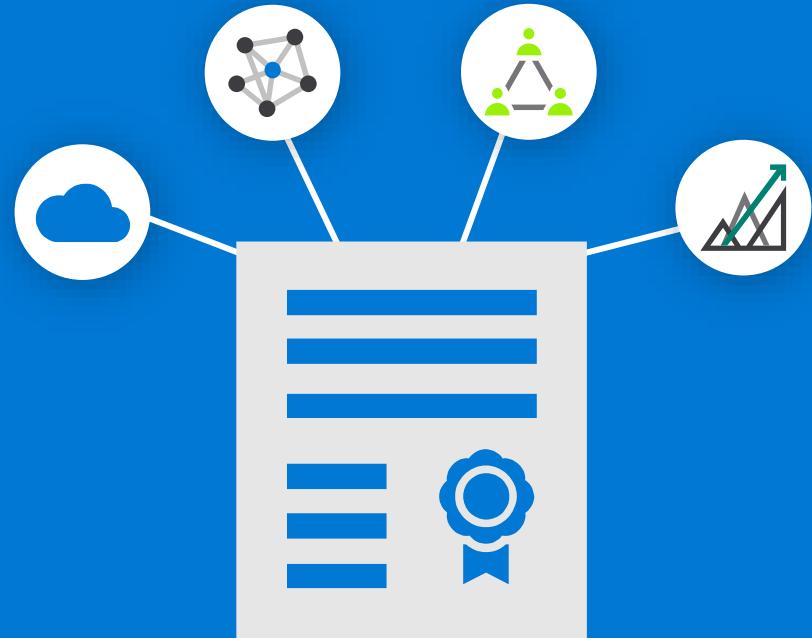
Query and transform data



Implement and manage data models



Explore and visualize data



Explore and visualize data

Explore data by using Azure Synapse Analytics

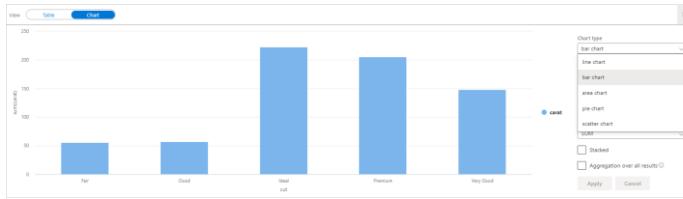


- Explore data by using native visuals in Spark notebooks
- Explore and visualize data by using the Azure Synapse SQL results pane

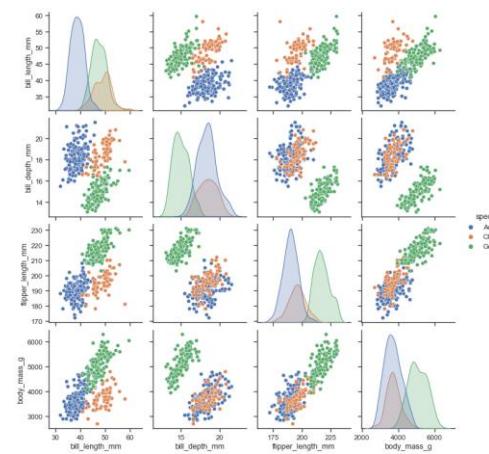
Develop Hub

Explore data by using native visuals in Spark notebooks

Fonction display(df)



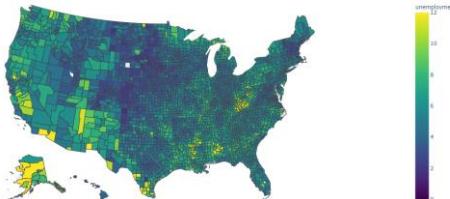
Seaborn



```
import seaborn as sns
sns.set_theme(style="ticks")
```

```
df = sns.load_dataset("penguins")
sns.pairplot(df, hue="species")
```

Plotly



```
import json
with
urlopen('https://raw.githubusercontent.com/plotly/datasets/master/geojson-counties-fips.json') as response:
    counties = json.load(response)
```

```
import pandas as pd
df =
pd.read_csv("https://raw.githubusercontent.com/plotly/datasets/master/fips-unemp-16.csv",
            dtype={"fips": str})
```

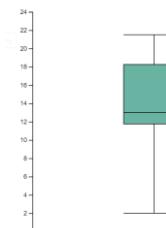
```
import plotly
import plotly.express as px
```

```
fig = px.choropleth(df, geojson=counties, locations='fips',
color='unemp',
        color_continuous_scale="Viridis",
        range_color=(0, 12),
        scope="usa",
        labels={'unemp':'unemployment rate'}
    )
fig.update_layout(margin={"r":0,"t":0,"l":0,"b":0})
```

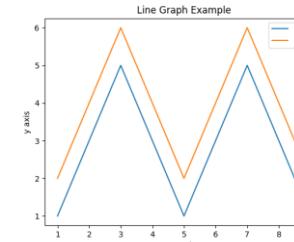
```
# create an html document that embeds the Plotly plot
h = plotly.offline.plot(fig, output_type='div')
```

```
# display this html
displayHTML(h)
```

displayHTML() option



Matplotlib



```
# Bar chart
```

```
import matplotlib.pyplot as plt
```

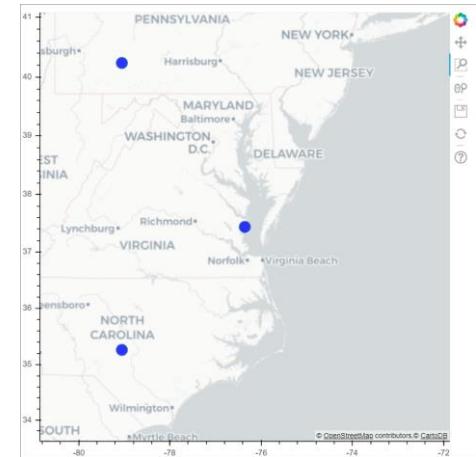
```
x1 = [1, 3, 4, 5, 6, 7, 9]
y1 = [4, 7, 2, 4, 7, 8, 3]
```

```
x2 = [2, 4, 6, 8, 10]
y2 = [5, 6, 2, 6, 2]
```

```
plt.bar(x1, y1, label="Blue Bar", color='b')
plt.bar(x2, y2, label="Green Bar", color='g')
plt.plot()
```

```
plt.xlabel("bar number")
plt.ylabel("bar height")
plt.title("Bar Chart Example")
plt.legend()
plt.show()
```

Bokeh



```
from bokeh.plotting import figure, output_file
from bokeh.tile_providers import get_provider, Vendors
from bokeh.embed import file_html
from bokeh.resources import CDN
from bokeh.models import ColumnDataSource
```

```
tile_provider = get_provider(Vendors.CARTODBPOSITRON)

# range bounds supplied in web mercator coordinates
p = figure(x_range=(-9000000,-8000000),
y_range=(4000000,5000000),
x_axis_type="mercator", y_axis_type="mercator")
p.add_tile(tile_provider)
```

```
# plot datapoints on the map
source = ColumnDataSource(
    data=dict(x=[ -8800000, -8500000 , -8800000],
              y=[4200000, 4500000, 4900000])
)
```

```
p.circle(x="x", y="y", size=15, fill_color="blue", fill_alpha=0.8,
source=source)
```

```
# create an html document that embeds the Bokeh plot
html = file_html(p, CDN, "my plot1")
```

```
# display this html
displayHTML(html)
```

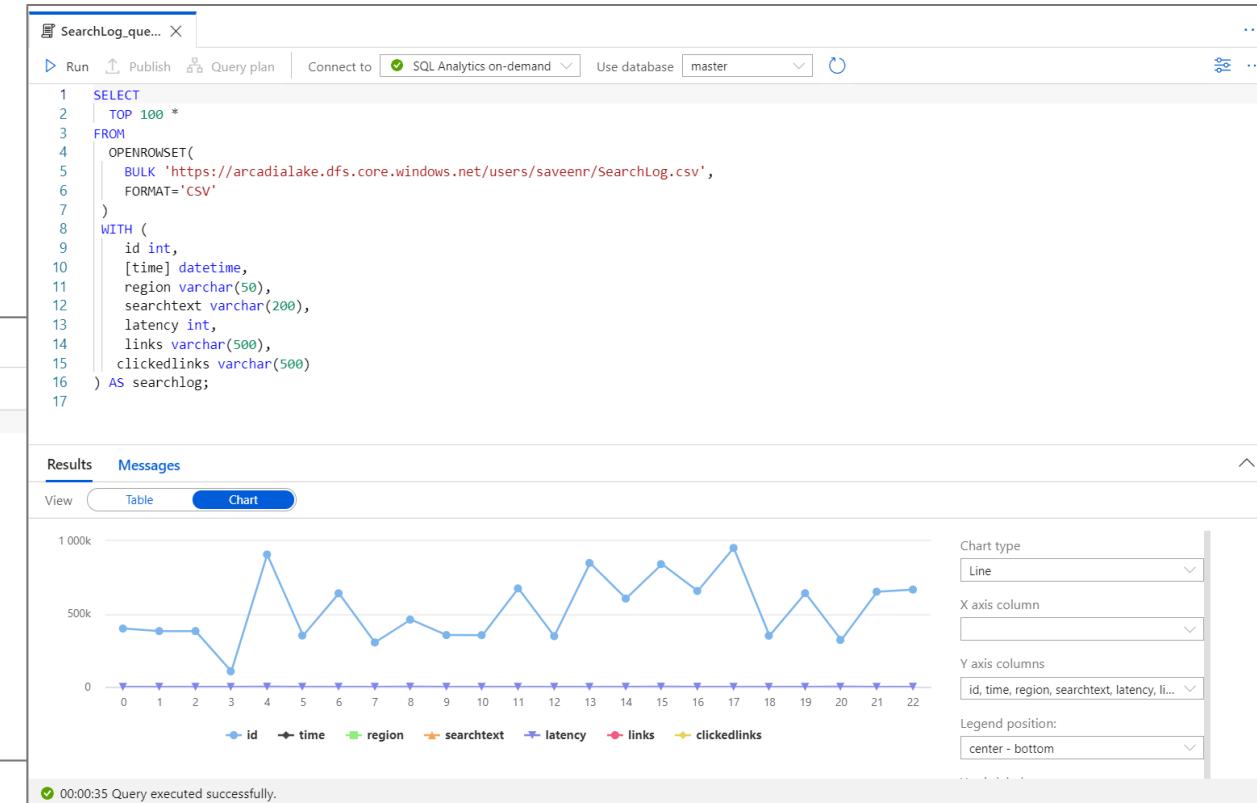
Develop Hub

Explore and visualize data by using the Azure Synapse SQL results pane

The screenshot shows the Azure Synapse Analytics Develop Hub interface. On the left, a code editor window displays a T-SQL query for reading data from a CSV file stored in Azure Data Lake. The query uses OPENROWSET and BULK options to read data from a URL. Below the code editor is a results pane with tabs for 'Results' (selected) and 'Messages'. The 'Results' tab shows a table view with columns ID, TIME, and REGION. The 'Messages' tab is also visible. A red box highlights the 'Table' button in the results pane, and a red arrow points from it to a larger view of the results table below. The table contains several rows of log data. At the bottom of the results pane, a message indicates the query was executed successfully.

ID	TIME	REGION
399266	2019-10-15T11:53:04.0000000	en-us
382045	2019-10-15T11:53:25.0000000	en-gb
382045	2019-10-16T11:53:42.0000000	en-gb
106479	2019-10-16T11:53:10.0000000	en-ca
906441	2019-10-16T11:54:18.0000000	en-us

00:00:35 Query executed successfully.



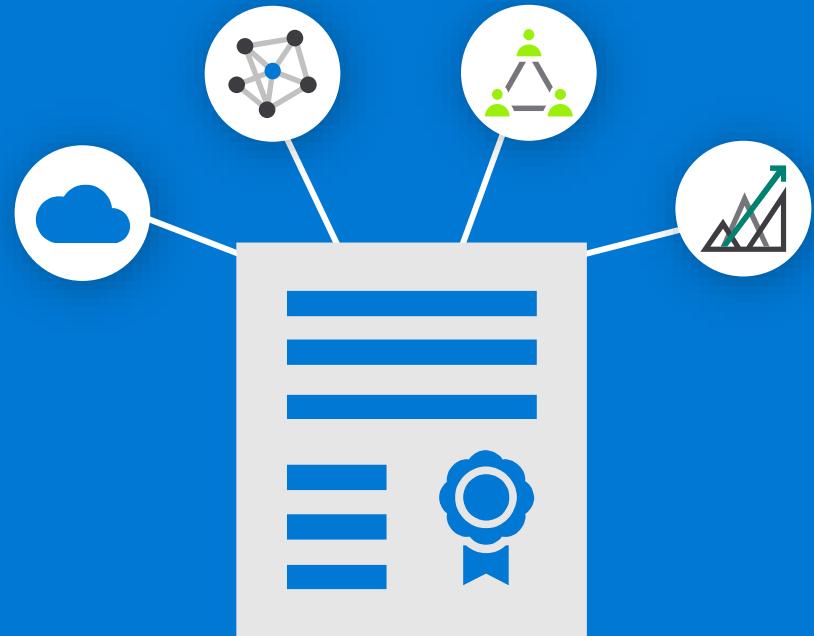
The screenshot shows the Azure Synapse Analytics Develop Hub interface with a dropdown menu for exporting results. The menu is titled 'Export results' and includes options for CSV, Excel, JSON, and XML. A red box highlights this dropdown menu. Below the menu, there is a table with columns ID, TIME, and REGION. The table contains three rows of data, matching the results shown in the previous screenshots. The table is partially cut off on the right side.

ID	TIME	REGION
399266	2019-10-15T11:53:04.0000000	en-u
382045	2019-10-15T11:53:25.0000000	en-g
382045	2019-10-16T11:53:42.0000000	en-g

Explore and visualize data

Visualize data by using Power BI

- Create and import a custom report theme
- Create R or Python visuals in Power BI
- Connect to and query datasets by using the XMLA endpoint
- Design and configure Power BI reports for accessibility
- Enable personalized visuals in a report
- Configure automatic page refresh
- Create and distribute paginated reports in Power BI Report Builder





End