

Data Factory documentation in Microsoft Fabric

Data factory in Microsoft Fabric provides cloud-scale data movement and data transformation services that allow you to solve the most complex data factory and ETL scenarios. It's intended to make your data factory experience easy to use, powerful, and truly enterprise-grade.

About Data Factory

OVERVIEW

- What is Data Factory?
- Connector overview
- What's new in Data Factory?

Using Data Factory

GET STARTED

- Create your first dataflow
- Create your first data pipeline
- Move and transform data
- Migrate to Data Factory
- Pricing in Data Factory

TUTORIAL

- Create an end-to-end data integration scenario

TRAINING

- Data Factory training

Working with data pipelines

HOW-TO GUIDE

- Data pipeline runs
- Copy data - copy activity
- Monitor data pipeline runs
- Activity Overview
- Move data with copy assistant

Working with dataflows

HOW-TO GUIDE

- Data source management
- Monitor your dataflows
- Save a draft of your dataflow

FAQs

GET STARTED

- Frequently asked questions

What is Data Factory in Microsoft Fabric?

Article • 11/15/2023

Data Factory empowers you with a modern data integration experience to ingest, prepare and transform data from a rich set of data sources (for example, databases, data warehouse, Lakehouse, real-time data, and more). Whether you are a citizen or professional developer, you will be able to transform the data with intelligent transformations and leverage a rich set of activities. With Data Factory in Microsoft Fabric, we are bringing fast copy (data movement) capabilities to both dataflows and data pipelines. With Fast Copy, you can move data between your favorite data stores blazing fast. Most importantly, Fast Copy enables you to bring data to your Lakehouse and Data Warehouse in Microsoft Fabric for analytics.

There are two primary high-level features Data Factory implements: dataflows and pipelines.

- Dataflows enable you to leverage more than 300 transformations in the dataflows designer, letting you transform data easier and with more flexibility than any other tool - including smart AI-based data transformations.
- Data pipelines enable you to leverage the out-of-the-box rich data orchestration capabilities to compose flexible data workflows that meet your enterprise needs.

Dataflows

Dataflows provide a low-code interface for ingesting data from hundreds of data sources, transforming your data using 300+ data transformations. You can then load the resulting data into multiple destinations, such as Azure SQL databases and more. Dataflows can be run repeatedly using manual or scheduled refresh, or as part of a data pipeline orchestration.

Dataflows are built using the familiar Power Query experience that's available today across several Microsoft products and services such as Excel, Power BI, Power Platform, Dynamics 365 Insights applications, and more. Power Query empowers all users, from citizen to professional data integrators, to perform data ingestion and data transformations across their data estate. Perform joins, aggregations, data cleansing, custom transformations, and much more all from an easy-to-use, highly visual, low-code UI.



Data pipelines

Data pipelines enable powerful workflow capabilities at cloud-scale. With data pipelines, you can build complex workflows that can refresh your dataflow, move PB-size data, and define sophisticated control flow pipelines.

Use data pipelines to build complex ETL and data factory workflows that can perform many different tasks at scale. Control flow capabilities are built into data pipelines that allow you to build workflow logic, which provides loops and conditionals.

Add a configuration-driven copy activity together with your low-code dataflow refresh in a single pipeline for an end-to-end ETL data pipeline. You can even add code-first activities for Spark Notebooks, SQL scripts, stored procs, and more.



Next steps

To get started with Microsoft Fabric, go to Quickstart: Create your first Dataflow Gen2 to get and transform data.

Feedback

Was this page helpful?

Yes

No

Provide product feedback [↗](#) | Ask the community [↗](#)

What's new in Microsoft Fabric?

Article • 12/21/2023

This page is continuously updated with a recent review of what's new in Microsoft Fabric. To follow the latest in Fabric news and features, see the Microsoft Fabric Blog [↗](#). Also follow the latest in Power BI at What's new in Power BI?

For older updates, review previous updates in Microsoft Fabric.

ⓘ Important

Microsoft Fabric has been announced! [↗](#)

New to Microsoft Fabric?

This section includes articles and announcements for users new to Microsoft Fabric.

- Learning Paths for Fabric
- Get started with Microsoft Fabric
- End-to-end tutorials in Microsoft Fabric
- Definitions of terms used in Microsoft Fabric

[] Expand table

Month	Feature	Learn more
December 2023	Fabric platform Security Fundamentals	Learn more about the big-picture perspective of the Microsoft Fabric security architecture by describing how the main security flows in the system work.
November 2023	Microsoft Fabric, explained for existing Synapse users	A focus on what customers using the current Platform-as-a-Service (PaaS) version of Synapse can expect ↗ . We'll explain what the general availability of Fabric means for your current investments (spoiler: we fully support them), but also how to think about the future.
November 2023	Microsoft Fabric is now generally available	Microsoft Fabric is now generally available for purchase ↗ . Microsoft Fabric can reshape how your teams work with data by bringing everyone together on a single, AI-powered platform built for the era of AI. This includes the experiences of Fabric: Power BI, Data Factory, Data Engineering, Data Science, Real-Time Analytics, Data Warehouse, and the overall Fabric platform.

Month	Feature	Learn more
November 2023	Fabric workloads are now generally available!	Microsoft Fabric is now generally available! Microsoft Fabric Synapse Data Warehouse, Data Engineering & Data Science, Real-Time Analytics, Data Factory, OneLake, and the overall Fabric platform are now generally available.
November 2023	Implement medallion lakehouse architecture in Microsoft Fabric	An introduction to medallion lake architecture and how you can implement a lakehouse in Microsoft Fabric.
October 2023	Announcing the Fabric roadmap	Announcing the Fabric Roadmap [↗] . One place you can see what we are working on and when you can expect it to be available.
October 2023	Get started with semantic link	<p>Explore how semantic link seamlessly connects Power BI semantic models with Synapse Data Science within Microsoft Fabric. Learn more at Semantic link in Microsoft Fabric: Bridging BI and Data Science [↗].</p> <p>You can also check out the semantic link sample notebooks that are now available in the fabric-samples GitHub repository. These notebooks showcase the use of semantic link's Python library, SemPy, in Microsoft Fabric.</p>
September 2023	Fabric Capacities – Everything you need to know about what's new and what's coming	Read more about the improvements we're making to the Fabric capacity management platform for Fabric and Power BI users [↗] .
August 2023	Accessing Microsoft Fabric for developers, startups and enterprises!	Learn how to enable Microsoft Fabric as a developer, as a startup or as an enterprise has different steps. Learn more at Enabling Microsoft Fabric for developers, startups and enterprises [↗] .
August 2023	Strong, useful, beautiful: Designing a new way of getting data	From the Data Integration Design Team, learn about the strong, creative, and function design [↗] of Microsoft Fabric, as Microsoft designs for the future of data integration.
August 2023	Learn Live: Get started with Microsoft Fabric	Calling all professionals, enthusiasts, and learners! On August 29, we'll be kicking off the "Learn Live: Get started with Microsoft Fabric" series in partnership with Microsoft's Data Advocacy teams and Microsoft WorldWide Learning teams to deliver 9x live-streamed lessons covering topics related to Microsoft Fabric! [↗]
July 2023	Step-by-Step Tutorial: Building ETLs with	In this comprehensive guide, we walk you through the process of creating Extract, Transform, Load (ETL)

Month	Feature	Learn more
	Microsoft Fabric	pipelines using Microsoft Fabric ↗ .
July 2023	Free preview usage of Microsoft Fabric experiences extended to October 1, 2023	We're extending the free preview usage of Fabric experiences (other than Power BI). These experiences won't count against purchased capacity until October 1, 2023 ↗ .
June 2023	Get skilled on Microsoft Fabric - the AI-powered analytics platform	Who is Fabric for? How can I get skilled? ↗ This blog post answers these questions about Microsoft Fabric, a comprehensive data analytics solution by unifying many experiences on a single platform.
June 2023	Introducing the end-to-end scenarios in Microsoft Fabric	In this blog, we explore four end-to-end scenarios that are typical paths our customers take to extract value and insights from their data using Microsoft Fabric ↗ .
May 2023	Get Started with Microsoft Fabric - All in-one place for all your Analytical needs	A technical overview and introduction to everything from data movement to data science, real-time analytics, and business intelligence in Microsoft Fabric ↗ .
May 2023	Microsoft OneLake in Fabric, the OneDrive for data	Microsoft OneLake brings the first multicloud SaaS data lake for the entire organization ↗ .

Features currently in preview

The following table lists the features of Microsoft Fabric that are currently in preview. Preview features are sorted alphabetically.

Note

Features currently in preview are available under [supplemental terms of use](#) [↗](#), review for legal terms that apply to Azure features that are in beta, preview, or otherwise not yet released into general availability. Microsoft Fabric provides previews to give you a chance to evaluate and [share feedback with the product group](#) [↗](#) on features before they become generally available (GA).

 [Expand table](#)

Feature	Learn more
Copilot in notebooks preview	The Copilot in Fabric Data Science and Data Engineering notebooks ↗ is designed to accelerate productivity, provide helpful answers and

Feature	Learn more
	guidance, and generate code for common tasks like data exploration, data preparation and machine learning with. You can interact and engage with the AI from either the chat panel or even from within notebooks cells using magic commands to get insights from data faster. For more information, see Copilot in notebooks ↗ .
Data Activator preview	We are thrilled to announce that Data Activator is now in preview ↗ and is enabled for all existing Microsoft Fabric users.
Data Engineering: Environment preview	We are thrilled to announce preview of the Environment in Fabric. The Environment is a centralized item ↗ that allows you to configure all the required settings for running a Spark job in one place.
Data Wrangler for Spark DataFrames preview	Data Wrangler now supports Spark DataFrames in preview. Until now, users have been able to explore and transform pandas DataFrames using common operations that can be converted to Python code in real time. The new release allows users to edit Spark DataFrames in addition to pandas DataFrames with Data Wrangler ↗ .
Lakehouse support for git integration and deployment pipelines (preview)	The Lakehouse now integrates with the lifecycle management capabilities in Microsoft Fabric ↗ , providing a standardized collaboration between all development team members throughout the product's life. Lifecycle management facilitates an effective product versioning and release process by continuously delivering features and bug fixes into multiple environments.
Microsoft 365 connector now supports ingesting data into Lakehouse (preview)	The Microsoft 365 connector now supports ingesting data into Lakehouse tables ↗ .
Microsoft Fabric Admin APIs	Fabric Admin APIs ↗ are designed to streamline administrative tasks. The initial set of Fabric Admin APIs is tailored to simplify the discovery of workspaces, Fabric items, and user access details.
Microsoft Fabric User APIs	We're happy to announce the preview of Microsoft Fabric User APIs. The Fabric user APIs ↗ are a major enabler for both enterprises and partners to use Microsoft Fabric as they enable end-to-end fully automated interaction with the service, enable integration of Microsoft Fabric into external web applications, and generally enable customers and partners to scale their solutions more easily.
Notebook Git integration preview	Fabric notebooks now offer Git integration for source control using Azure DevOps ↗ . It allows users to easily control the notebook code versions and manage the git branches by leveraging the Fabric Git functions and Azure DevOps.

Feature	Learn more
Notebook in Deployment Pipeline Preview	Now you can also use notebooks to deploy your code across different environments ↗ , such as development, test, and production. You can also use deployment rules to customize the behavior of your notebooks when they are deployed, such as changing the default Lakehouse of a Notebook. Get started with deployment pipelines to set up your deployment pipeline, Notebook will show up in the deployment content automatically.
Prebuilt Azure AI services in Fabric preview	The preview of prebuilt AI services in Fabric ↗ is an integration with Azure AI services ↗ , formerly known as Azure Cognitive Services. Prebuilt Azure AI services allow for easy enhancement of data with prebuilt AI models without any prerequisites. Currently, prebuilt AI services are in public preview and include support for Azure Open AI service ↗ , Azure AI Language ↗ , and Azure AI Translator ↗ .
Splunk add-on preview	Microsoft Fabric add-on for Splunk ↗ allows users to ingest logs from Splunk platform into a Fabric KQL DB using the Kusto python SDK.
Warehouse restore points and restore in place	You can now create restore points and perform an in-place restore of a warehouse to a past point in time. Restore in-place is an essential part of data warehouse recovery ↗ which allows to restore the data warehouse to a prior known reliable state by replacing or over-writing the existing data warehouse from which the restore point was created.
VNet Gateways in Dataflow Gen2 preview	VNet Data Gateway support for Dataflows Gen2 in Fabric is now in preview. The VNet data gateway ↗ helps to connect from Fabric Dataflows Gen2 to Azure data services within a VNet, without the need of an on-premises data gateway.

Generally available features

The following table lists the features of Microsoft Fabric that have transitioned from preview to general availability (GA) within the last 12 months.

 Expand table

Month	Feature	Learn more
November 2023	Microsoft Fabric is now generally available	Microsoft Fabric is now generally available for purchase ↗ . Microsoft Fabric can reshape how your teams work with data by bringing everyone together on a single, AI-powered platform built for the era of AI. This includes the experiences of Fabric: Power BI, Data Factory, Data Engineering, Data Science, Real-Time Analytics, Data Warehouse, and the overall Fabric platform ↗ .

Community

This section summarizes new Microsoft Fabric community opportunities for prospective and current influencers and MVPs.

- Join a local Fabric User Group [↗](#) or join a local event [↗](#).
- To learn about the Microsoft MVP Award and to find MVPs, see [mvp.microsoft.com ↗](#).
- Are you a student? Learn more about the Microsoft Learn Student Ambassadors program [↗](#).

[] [Expand table](#)

Month	Feature	Learn more
December 2023	Microsoft Fabric Community Conference	Join us in Las Vegas March 26-28, 2024 for the first annual Microsoft Fabric Community Conference. See firsthand how Microsoft Fabric and the rest of the data and AI products at Microsoft can help your organization prepare for the era of AI. Register today ↗ to immerse yourself in the future of data and AI and connect with thousands of data innovators like yourself eager to share their insights.
December 2023	Fabric Cloud skills challenge	Join the thousands of professionals that have completed the Microsoft Fabric Challenge as part of the Microsoft Ignite Cloud Skills Challenge ↗ . In less than 8 hours, you will learn how to connect, ingest, store, analyze, and report on data with Power BI and Microsoft Fabric. The Microsoft Fabric challenge prepares you for the Microsoft Certified: Fabric Analytics Engineer Associate certification, available soon. The challenge is on until January 15, 2024.
November 2023	Microsoft Fabric MVP Corner – Special Edition (Ignite)	A special edition of the "Microsoft Fabric MVP Corner" blog series highlights selected content related to Fabric and created by MVPs around the Microsoft Ignite 2023 conference ↗ , when we announced Microsoft Fabric generally available.
November 2023	Skill up on Fabric with the Microsoft Learn Cloud Skills Challenge	We are excited to announce the Microsoft Ignite: Microsoft Fabric Challenge ↗ as part of the Microsoft Learn Cloud Skills Challenge. Skill up for in-demand tech scenarios ↗ , and enter to win a VIP pass to the next Microsoft Ignite. The challenge is on until January 15, 2024. The challenge helps you prepare for the Microsoft Certified: Fabric Analytics Engineer Associate certification and new Microsoft Applied Skills credentials covering the lakehouse and data warehouse scenarios, which are coming in the next months.

Month	Feature	Learn more
October 2023	Microsoft Fabric MVP Corner – October 2023	Highlights of selected content related to Fabric and created by MVPs from October 2023 ↗ .
September 2023	Microsoft Fabric MVP Corner – September 2023	Highlights of selected content related to Fabric and created by MVPs from September 2023 ↗ .
August 2023	Microsoft Fabric MVP Corner – August 2023	Highlights of selected content related to Fabric and created by MVPs from August 2023 ↗ .
July 2023	Microsoft Fabric MVP Corner – July 2023	Highlights of selected content related to Fabric and created by MVPs in July 2023 ↗ .
June 2023	Microsoft Fabric MVP Corner – June 2023	The Fabric MVP Corner blog series to highlight selected content related to Fabric and created by MVPs in June 2023 ↗ .
May 2023	Fabric User Groups	Power BI User Groups are now Fabric User Groups ↗ !

Power BI

Updates to Power BI Desktop and the Power BI service are summarized at [What's new in Power BI?](#)

Fabric samples and guidance

This section summarizes new guidance and sample project resources for Microsoft Fabric.

[Expand table](#)

Month	Feature	Learn more
December 2023	Working with OneLake using Azure Storage Explorer	If you want to use an application that directly integrates with Windows File Explorer, check out OneLake file explorer ↗ . However, if you are accustomed to using Azure Storage Explorer for your data management tasks ↗ , you can continue to harness its functionalities with OneLake and some of its key benefits.

Month	Feature	Learn more
November 2023	Semantic Link: OneLake integrated Semantic Models	Semantic Link adds support for the recently released OneLake integrated semantic models! You can now directly access data using your semantic model's name via OneLake ↗ using the <code>read_table</code> function and the new <code>mode</code> parameter set to <code>onelake</code> .
November 2023	Integrate your SAP data into Microsoft Fabric	Using the built-in connectivity of Microsoft Fabric is, of course, the easiest and least-effort way of adding SAP data to your Fabric data estate ↗ .
November 2023	Fabric Changing the game: Validate dependencies with Semantic Link – Data Quality	Follow this step-by-step example of how to explore the functional dependencies between columns in a table using the semantic link ↗ . The semantic link is a feature that allows you to establish a connection between Power BI datasets and Synapse Data Science in Microsoft Fabric.
November 2023	Implement medallion lakehouse architecture in Microsoft Fabric	An introduction to medallion lake architecture and how you can implement a lakehouse in Microsoft Fabric.
October 2023	Fabric Change the Game: Exploring the data	Follow this realistic example of reading data from Azure Data Lake Storage using shortcuts, organizing raw data into structured tables, and basic data exploration. Our data exploration uses as a source the diverse and captivating city of London with information extracted from data.london.gov.uk/↗ .
September 2023	Announcing an end-to-end workshop: Analyzing Wildlife Data with Microsoft Fabric	A new workshop guides you in building a hands-on, end-to-end data analytics solution ↗ for the Snapshot Serengeti dataset using Microsoft Fabric. The dataset consists of approximately 1.68M wildlife images and image annotations provided in json files.
September 2023	New learning path: Implement a Lakehouse with Microsoft Fabric	The new Implement a Lakehouse with Microsoft Fabric learning path introduces the foundational components of implementing a data lakehouse with Microsoft Fabric with seven in-depth modules.
September 2023	Fabric Readiness repository	The Fabric Readiness repository ↗ is a treasure trove of resources for anyone interested in exploring the exciting world of Microsoft Fabric.
July 2023	Connecting to OneLake	How do I connect to OneLake? This blog covers how to connect and interact with OneLake, including how OneLake achieves its compatibility with any tool used over ADLS Gen2! ↗

Month	Feature	Learn more
June 2023	Using Azure Databricks with Microsoft Fabric and OneLake	How does Azure Databricks work with Microsoft Fabric? ↗ This blog post answers that question and more details on how the two systems can work together.

Data Factory in Microsoft Fabric

This section summarizes recent new features and capabilities of Data Factory in Microsoft Fabric. Follow issues and feedback through the Data Factory Community Forum [↗](#).

[] [Expand table](#)

Month	Feature	Learn more
December 2023	Read and Write to the Fabric Lakehouse using Azure Data Factory (ADF)	You can now read and write data in the Microsoft Fabric Lakehouse from ADF (Azure Data Factory). Using either Copy Activity or Mapping Data Flows, you can read, write, transform, and process data using ADF or Synapse Analytics, currently in public preview.
December 2023	Set activity state for easy pipeline debugging	In Fabric Data Factory data pipelines, you can now set an activity's state to inactive so that you can save your pipeline even with incomplete, invalid configurations. Think of it as "commenting out" part of your pipeline code.
December 2023	Connection editing in pipeline editor	You can now edit your existing data connections while you are designing your pipeline without leaving the pipeline editor! When setting your connection, simply select Edit and a pop-up will appear.
December 2023	Azure Databricks Notebook executions in Fabric Data Factory	You can now create powerful data pipeline workflows that include Notebook executions from your Azure Databricks clusters using Fabric Data Factory ↗ . Add a Databricks activity to your pipeline, point to your existing cluster, or request a new cluster, and Data Factory will execute your Notebook code for you.
November 2023	Implement medallion lakehouse architecture in Microsoft Fabric	An introduction to medallion lake architecture and how you can implement a lakehouse in Microsoft Fabric.
November 2023	Dataflow Gen2 General availability of Fabric connectors	The connectors for Lakehouse, Warehouse, and KQL Database are now generally available ↗ . We encourage

Month	Feature	<p>Learn more these connectors when trying to connect to data from any of these Fabric experiences.</p>
November 2023	Dataflow Gen2 Automatic refresh cancellation	<p>To prevent unnecessary resources from being consumed, we've implemented a new mechanism that stops the refresh of a Dataflow as soon as the results of the refresh are known to have no impact ↗. This is to reduce consumption more proactively.</p>
November 2023	Dataflow Gen2 Error message propagation through gateway	<p>We made diagnostics improvement to provide meaningful error messages when Dataflow refresh fails ↗ for those Dataflows running through the Enterprise Data Gateway.</p>
November 2023	Dataflow Gen2 Support for column binding for SAP HANA connector	<p>Column binding support is enabled for SAP HANA. This optional parameter results in significantly improved performance. For more information, see Support for column binding for SAP HANA connector ↗.</p>
November 2023	Dataflow Gen2 staging artifacts hidden	<p>When using a Dataflow Gen2 in Fabric, the system will automatically create a set of staging artifacts. Now, these staging artifacts will be abstracted from the Dataflow Gen2 experience ↗ and will be hidden from the workspace list. No action is required by the user and this change has no impact on existing Dataflows.</p>
November 2023	Dataflow Gen2 Support for VNet Gateways preview	<p>VNet Data Gateway support for Dataflows Gen2 in Fabric is now in preview. The VNet data gateway ↗ helps to connect from Fabric Dataflows Gen2 to Azure data services within a VNet, without the need of an on-premises data gateway.</p>
November 2023	Cross workspace "Save as"	<p>You can now clone your data pipelines across workspaces by using the "Save as" button ↗.</p>
November 2023	Dynamic content flyout integration with Email and Teams activity	<p>In the Email and Teams activities, you can now add dynamic content with ease. With this new pipeline expression integration, you will now see a flyout menu to help you select and build your message content ↗ quickly without needing to learn the pipeline expression language.</p>
November 2023	Copy activity now supports fault tolerance for Fabric Data Warehouse connector	<p>The Copy activity in data pipelines now supports fault tolerance for Fabric Warehouse ↗. Fault tolerance allows you to handle certain errors without interrupting data movement. By enabling fault tolerance, you can continue to copy data while skipping incompatible data like duplicated rows.</p>
November 2023	MongoDB and MongoDB Atlas connectors	<p>MongoDB and MongoDB Atlas connectors are now available ↗ to use in your Data Factory data pipelines as sources and destinations.</p>

Month	Feature	Learn more
November 2023	Microsoft 365 connector now supports ingesting data into Lakehouse (preview)	The Microsoft 365 connector now supports ingesting data into Lakehouse tables ↗ .
November 2023	Multi-task support for editing pipelines in the designer	You can now open and edit data pipelines from different workspaces ↗ and navigate between them using the multi-tasking capabilities in Fabric.
November 2023	String interpolation added to pipeline return value	You can now edit your data connections within your data pipelines ↗ . Previously, a new tab would open when connections needed editing. Now, you can remain within your pipeline and seamlessly update your connections.
October 2023	Category redesign of activities	We've redesigned the way activities are categorized to make it easier for you to find the activities you're looking for with new categories like Control flow, Notifications, and more.
October 2023	Copy runtime performance improvement	We've made improvements to the Copy runtime performance. According to our tests results, with the improvements users can expect to see the duration of copying from parquet/csv files into Lakehouse table to improve by ~25%-35%.
October 2023	Integer data type available for variables	We now support variables as integers! When creating a new variable, you can now choose to set the variable type to Integer, making it easier to use arithmetic functions with your variables.
October 2023	Pipeline name now supported in System variables.	We've added a new system variable called Pipeline Name so that you can inspect and pass the name of your pipeline inside of the pipeline expression editor, enabling a more powerful workflow in Fabric Data Factory.
October 2023	Support for Type editing in Copy activity Mappings	You can now edit column types when you land data into your Lakehouse table(s). This makes it easier to customize the schema of your data in your destination. Simply navigate to the Mapping tab, import your schemas, if you don't see any mappings, and use the drop-down list to make changes.
October 2023	New certified connector: Emplifi Metrics	Announcing the release of the new Emplifi Metrics connector. The Power BI Connector is a layer between Emplifi Public API and Power BI itself. For more information, see Emplifi Public API documentation ↗ .

Month	Feature	Learn more
October 2023	SAP HANA (Connector Update)	The update enhances the SAP HANA connector with the capability to consume HANA Calculation Views deployed in SAP Datasphere by taking into account SAP Datasphere's additional security concepts.
October 2023	Set Activity State to "Comment Out" Part of Pipeline	Activity State is now available in Fabric Data Factory data pipelines ↗ , giving you the ability to comment out part of your pipeline without deleting the definition.
August 2023	Staging labels	The concept of staging data was introduced in Dataflows Gen2 for Microsoft Fabric and now you have the ability to define what queries within your Dataflow should use the staging mechanisms or not.
August 2023	Secure input/output for logs	We've added advanced settings for the Set Variable activity called Secure input and Secure output. When you enable secure input or output, you can hide sensitive information from being captured in logs.
August 2023	Pipeline run status added to Output panel	We've recently added Pipeline status so that developers can easily see the status of the pipeline run. You can now view your Pipeline run status from the Output panel.
August 2023	Data pipelines FTP connector	The FTP connector is now available to use in your Data Factory data pipelines in Microsoft Fabric. Look for it in the New connection menu.
August 2023	Maximum number of entities in a Dataflow	The new maximum number of entities that can be part of a Dataflow has been raised to 50.
August 2023	Manage connections feature	The Manage Connections option now allows you to view the linked connections to your dataflow, unlink a connection, or edit connection credentials and gateway.
August 2023	Power BI Lakehouses connector	An update to the Lakehouses connector in the August version of the Power BI Desktop and Gateway includes significant performance improvements.
July 2023	New modern data connectivity and discovery experience in Dataflows	An improved experience aims to expedite the process of discovering data in Dataflow, Dataflow Gen2, and Datamart ↗ .

Data Factory in Microsoft Fabric samples and guidance

[\[\] Expand table](#)

Month	Feature	Learn more
December 2023	Read data from Delta Lake tables with the DeltaLake.Table M function	The DeltaLake.Table is a new function in Power Query's M language for reading data from Delta Lake tables ↗ . This function is now available in Power Query in Power BI Desktop and in Dataflows Gen1 and Gen2 and replaces the need to use community-developed solution.
October 2023	Microsoft Fabric Data Factory Webinar Series – October 2023	You are invited to join our October webinar series ↗ , where we will show you how to use Data Factory to transform and orchestrate your data in various scenarios.
September 2023	Notify Outlook and Teams channel/group from a Microsoft Fabric pipeline	Learn how to send notifications to both Teams channels/groups and Outlook emails ↗ .
September 2023	Microsoft Fabric Data Factory Webinar Series – September 2023	Join our Data Factory webinar series ↗ where we will show you how to use Data Factory to transform and orchestrate your data in various scenarios.
August 2023	Metadata Driven Pipelines for Microsoft Fabric – Part 2, Data Warehouse Style	An overview of a metadata-driven pipeline in Microsoft Fabric that follows the medallion architecture with Data Warehouse serving as the Gold layer ↗ .
August 2023	Metadata Driven Pipelines for Microsoft Fabric	An overview of a Metadata driven pipeline in Microsoft Fabric ↗ that follows the medallion architecture (Bronze, Silver, Gold).
August 2023	Using Data pipelines for copying data to/from KQL Databases and crafting workflows with the Lookup activity	Real-Time Analytics' KQL DB is supported as both a destination and a source with data pipelines ↗ , allowing you to build and manage various extract, transform, and load (ETL) activities, leveraging the power and capabilities of KQL DBs.
August 2023	Incrementally amass data	With Dataflows Gen2 that comes with support for data destinations, you can setup your own pattern to load new data incrementally ↗ , replace some old data and keep your reports up to date with your source data.
August 2023	Data Pipeline Performance Improvement Part 3: Gaining more than 50% improvement for Historical Loads	Learn how to account for pagination given the current state of Fabric Data Pipelines in preview. This pipeline is performant when the number of paginated pages isn't too large. Read more at Gaining more than 50% improvement for Historical Loads ↗ .

Month	Feature	Learn more
August 2023	Data Pipeline Performance Improvements Part 2: Creating an Array of JSONs	Examples from this blog series include how to merge two arrays into an array of JSON objects, and how to take a date range and create multiple subranges then store these as an array of JSONs. Read more at Creating an Array of JSONs .
July 2023	Data Pipeline Performance Improvements Part 1: How to convert a time interval (dd.hh:mm:ss) into seconds	Part one of a series of blogs on moving data with multiple Copy Activities moving smaller volumes in parallel: How to convert a time interval (dd.hh:mm:ss) into seconds.
July 2023	Construct a data analytics workflow with a Fabric Data Factory data pipeline	A blog covering data pipelines in Data Factory and the advantages you find by using pipelines to orchestrate your Fabric data analytics projects and activities.
July 2023	Data Pipelines Tutorial: Ingest files into a Lakehouse from a REST API with pagination ft. AVEVA Data Hub	In this blog, we will act in the persona of an AVEVA customer who needs to retrieve operations data from AVEVA Data Hub into a Microsoft Fabric Lakehouse.
July 2023	Data Factory Spotlight: Dataflow Gen2	This blog spotlight covers the two primary high-level features Data Factory implements: dataflows and pipelines.

Synapse Data Engineering in Microsoft Fabric

This section summarizes recent new features and capabilities of the Data Engineering experience in Microsoft Fabric.

[] [Expand table](#)

Month	Feature	Learn more
December 2023	%%configure – personalize your Spark session in Notebook	Now you can personalize your Spark session with the magic command %%configure, in both interactive notebook and pipeline notebook activities.
December 2023	Rich dataframe preview in Notebook	The display() function has been updated on Fabric Notebook, now named the <i>Rich dataframe preview</i> . Now when you use <code>display()</code> to preview your dataframe, you can easily specify the range, view the dataframe summary and column statistics, check invalid values or missing values, and preview the long cell.

Month	Feature	Learn more
December 2023	Working with OneLake using Azure Storage Explorer	If you want to use an application that directly integrates with Windows File Explorer, check out OneLake file explorer ↗ . However, if you are accustomed to using Azure Storage Explorer for your data management tasks ↗ , you can continue to harness its functionalities with OneLake and some of its key benefits.
November 2023	Accessibility support for Lakehouse	To provide a more inclusive and user-friendly interaction, we have implemented improvements so far to support accessibility in the Lakehouse experience ↗ , including screen reader compatibility, responsive design text reflow, keyboard navigation, alternative text for images, and form fields and labels.
November 2023	Enhanced multitasking experience in Lakehouse	We've introduced new capabilities to enhance the multi-tasking experience in Lakehouse ↗ , including multitasking during running operations, nonblocking reloading, and clearer notifications.
November 2023	Upgraded DataGridView capabilities in Lakehouse	An upgraded DataGridView for the Lakehouse table preview experience ↗ now features sorting, filtering, and resizing of columns.
November 2023	SQL analytics endpoint re-provisioning	You can now retry the SQL analytics endpoint provisioning directly within the Lakehouse experience ↗ . This means that if your initial provisioning attempt fails, you have the option to try again without the need to create an entirely new Lakehouse.
November 2023	Microsoft Fabric Runtime 1.2	The Microsoft Fabric Runtime 1.2 ↗ is a significant advancement in our data processing capabilities. Microsoft Fabric Runtime 1.2 includes Apache Spark 3.4.1, Mariner 2.0 as the operating system, Java 11, Scala 2.12.17, Python 3.10, Delta Lake 2.4, and R 4.2.2, ensuring you have the most cutting-edge tools at your disposal. In addition, this release comes bundled with default packages, encompassing a complete Anaconda installation and essential libraries for Java/Scala, Python, and R, simplifying your workflow.
November 2023	Multiple Runtimes Support	With the introduction of Runtime 1.2, Fabric supports multiple runtimes ↗ , offering users the flexibility to seamlessly switch between them, minimizing the risk of incompatibilities or disruptions. When changing runtimes, all system-created items within the workspace, including Lakehouses, SJDs, and Notebooks, will operate using the newly selected workspace-level runtime version starting from the next Spark Session.

Month	Feature	Learn more
November 2023	Delta as the default table format in the new Runtime 1.2	The default Spark session parameter <code>spark.sql.sources.default</code> is now <code>delta</code> . All tables created using Spark SQL, PySpark, Scala Spark, and Spark R, whenever the table type is omitted, will create the table as Delta by default ↗ .
November 2023	Intelligent Cache	By default, the newly revamped and optimized Intelligent Cache feature is enabled in Fabric Spark. The intelligent cache ↗ works seamlessly behind the scenes and caches data to help speed-up the execution of Spark jobs in Microsoft Fabric as it reads from your OneLake or ADLS Gen2 storage via shortcuts.
November 2023	Monitoring Hub for Spark enhancements	The latest enhancements in the monitoring hub are designed to provide a comprehensive and detailed view of Spark and Lakehouse activities ↗ , including executor allocations, runtime version for a Spark application, a related items link in the detail page.
November 2023	Monitoring for Lakehouse operations	Users can now view the progress and status of Lakehouse maintenance jobs ↗ and table load activities.
November 2023	Spark application resource Usage Analysis	Responding to customers' requests for monitoring Spark resource usage metrics for performance tuning and optimization, we are excited to introduce the Spark resource usage analysis feature ↗ , now available in preview. This newly released feature enables users to monitor allocated executors, running executors, and idle executors, alongside Spark executions.
November 2023	REST API support for Spark Job Definition preview	REST Public APIs for Spark Job Definition are now available, making it easy for users to manage and manipulate SJD items ↗ .
November 2023	REST API support for Lakehouse artifact, Load to tables and table maintenance	As a key requirement for workload integration, REST Public APIs for Lakehouse are now available. The Lakehouse REST Public APIs ↗ makes it easy for users to manage and manipulate Lakehouse artifacts items programmatically.
November 2023	Lakehouse support for git integration and deployment pipelines (preview)	The Lakehouse now integrates with the lifecycle management capabilities in Microsoft Fabric ↗ , providing a standardized collaboration between all development team members throughout the product's life. Lifecycle management facilitates an effective product versioning and release process by continuously delivering features and bug fixes into multiple environments.

Month	Feature	Learn more
November 2023	Embed a Power BI report in Notebook	We are thrilled to announce that the <code>powerbiclient</code> Python package ↗ is now natively supported in Fabric notebooks. This means you can easily embed and interact with Power BI reports in your notebooks ↗ with just a few lines of code. To learn more about how to use the <code>powerbiclient</code> package to embed a Power BI component.
November 2023	Mssparkutils new API – reference run multiple notebooks in parallel	A new <code>runMultiple</code> API in <code>mssparkutils</code> ↗ called <code>mssparkutils.notebook.runMultiple()</code> allows you to run multiple notebooks in parallel, or with a predefined topological structure. For more information, see Notebook utilities .
November 2023	Notebook resources .JAR file support	We now support uploading the .jar files in the Notebook Resources explorer ↗ . You can add your own compiled libs, use drag & drop to generate a code snippet to install them in the session, and load the libraries in code conveniently.
November 2023	Notebook Git integration preview	Fabric notebooks now offer Git integration for source control using Azure DevOps ↗ . It allows users to easily control the notebook code versions and manage the git branches by leveraging the Fabric Git functions and Azure DevOps.
November 2023	Notebook in Deployment Pipeline Preview	Now you can also use notebooks to deploy your code across different environments ↗ , such as development, test, and production. You can also use deployment rules to customize the behavior of your notebooks when they are deployed, such as changing the default Lakehouse of a Notebook. Get started with deployment pipelines to set up your deployment pipeline, Notebook will show up in the deployment content automatically.
November 2023	Notebook REST APIs Preview	With REST Public APIs for the Notebook items, data engineers/data scientists can automate their pipelines and establish CI/CD conveniently and efficiently. The <code>notebook</code> Restful Public API ↗ can make it easy for users to manage and manipulate Fabric notebook items and integrate notebook with other tools and systems.
November 2023	Environment preview	We are thrilled to announce preview of the Environment in Fabric. The Environment is a centralized item ↗ that allows you to configure all the required settings for running a Spark job in one place.
November 2023	Synapse VS Code extension in vscode.dev preview	With support for the Synapse VS Code extension on <code>vscode.dev</code> , users can now seamlessly edit and execute Fabric Notebooks without ever leaving their browser window ↗ .

Month	Feature	Learn more
		Additionally, all the native pro-developer features of VS Code are now accessible to end-users in this environment.
October 2023	Create multiple OneLake shortcuts at once	Creating multiple OneLake shortcuts just got easier. Rather than creating shortcuts one at a time, you can now browse to your desired location and select multiple targets at once. All your selected targets then get created as new shortcuts in a single operation ♂ .
October 2023	Delta-RS introduces native support for OneLake	The OneLake team worked with the Delta-RS community to help introduce support for recognizing OneLake URLs in both Delta-RS and the Rust Object Store ♂ .
September 2023	Import notebook to your Workspace	The new "Import Notebook" entry on the Workspace -> New menu ♂ lets you easily import new Fabric Notebook items in the target workspace. You can upload one or more files, including <code>.ipynb</code> , <code>.py</code> , <code>.sql</code> , <code>.scala</code> , and <code>.r</code> file formats.
September 2023	Notebook file system support in Synapse VS Code extension	The Synapse VS Code extension now supports notebook File System ♂ for Data Engineering and Data Science in Microsoft Fabric. The Synapse VS Code extension empowers users to develop their notebook artifacts directly within the Visual Studio Code environment.
September 2023	Notebook sharing execute-only mode	We now support checking the "Run" operation separately when sharing a notebook, if you just selected the "Run" operation, the recipient would see a "Execution-only" notebook ♂ .
September 2023	Notebook save conflict resolution	We now support viewing and comparing the differences between two versions of the same notebook ♂ when there are saving conflicts.
September 2023	Mssparkutils new API for fast data copy	We now support a new method in mssparkutils that can enable large volume of data move/copy much faster ♂ , <code>Mssparkutils.fs.fastcp()</code> . You can use <code>mssparkutils.fs.help("fastcp")</code> to check the detailed usage.
September 2023	Notebook resources .whl file support	We now support uploading .whl files in the Notebook Resources explorer ♂ .
August 2023	Introducing High Concurrency Mode in Notebooks for Data Engineering and Data Science	High concurrency mode allows you to run notebooks simultaneously on the same cluster without compromising performance or security when paying for a single session. High concurrency mode offers several benefits ♂ for Fabric Spark users.

Month	Feature	Learn more
	workloads in Microsoft Fabric	
August 2023	Service principal support to connect to data in Dataflow, Datamart, Dataset and Dataflow Gen 2	Azure service principal has been added as an authentication type ↗ for a set of data sources that can be used in Dataset, Dataflow, Dataflow Gen2 and Datamart.
August 2023	Announcing XMLA Write support for Direct Lake datasets	Direct Lake datasets now support XMLA-Write operations. Now you can use your favorite BI Pro tools and scripts to create and manage Direct Lake datasets using XMLA endpoints ↗ .

Synapse Data Engineering samples and guidance

[Expand table](#)

Month	Feature	Learn more
November 2023	Fabric Changing the game: Using your own library with Microsoft Fabric	A step-by-step guide to use your own Python library in the Lakehouse ↗ . It is quite simple to create your own library with Python and even simpler to reuse it on Fabric.
August 2023	Fabric changing the game: Logging your workload using Notebooks	Learn more about logging your workload into OneLake using notebooks ↗ , using the OneLake API Path inside the notebook.
July 2023	Lakehouse Sharing and Access Permission Management	Share a lakehouse and manage permissions ↗ so that users can access lakehouse data through the Data Hub, the SQL analytics endpoint, and the default semantic model.
June 2023	Virtualize your existing data into OneLake with shortcuts	Connect data silos without moving or copying data with OneLake, which allows you to create special folders called shortcuts that point to other storage locations ↗ .

Synapse Data Science in Microsoft Fabric

This section summarizes recent improvements and features for the Data Science experience in Microsoft Fabric.

Month	Feature	Learn more
December 2023	Semantic Link update	We're excited to announce the latest update of Semantic Link ↗ ! Apart from many improvements we also added many new features for our Power BI engineering community that you can use from Fabric notebooks to satisfy all your automation needs.
December 2023	Prebuilt Azure AI services in Fabric preview	The preview of prebuilt AI services in Fabric ↗ is an integration with Azure AI services ↗ , formerly known as Azure Cognitive Services. Prebuilt Azure AI services allow for easy enhancement of data with prebuilt AI models without any prerequisites. Currently, prebuilt AI services are in public preview and include support for Azure Open AI service ↗ , Azure AI Language ↗ , and Azure AI Translator ↗ .
November 2023	Copilot in notebooks preview	The Copilot in Fabric Data Science and Data Engineering notebooks ↗ is designed to accelerate productivity, provide helpful answers and guidance, and generate code for common tasks like data exploration, data preparation and machine learning with. You can interact and engage with the AI from either the chat panel or even from within notebooks cells using magic commands to get insights from data faster. For more information, see Copilot in notebooks ↗ .
November 2023	Custom Python Operations in Data Wrangler	Data Wrangler, a notebook-based tool for exploratory data analysis, has always allowed users to browse and apply common data-cleaning operations, generating the corresponding code in real time. Now, in addition to generating code from the UI, users can also write their own code with custom operations in Data Wrangler ↗ .
November 2023	Data Wrangler for Spark DataFrames preview	Data Wrangler now supports Spark DataFrames in preview. Until now, users have been able to explore and transform pandas DataFrames using common operations that can be converted to Python code in real time. The new release allows users to edit Spark DataFrames in addition to pandas DataFrames with Data Wrangler ↗ .
November 2023	MLflow Notebook Widget	The MLflow inline authoring widget enables users to effortlessly track their experiment runs along with metrics and parameters, all directly from within their notebook ↗ .
November 2023	New Model & Experiment Item	New enhancements to our model and experiment tracking features ↗ are based on valuable user feedback. The new tree-control in the run details view makes tracking easier

Month	Feature	Learn more
	Usability Improvements	by showing which run is selected. We've enhanced the comparison feature, allowing you to easily adjust the comparison pane for a more user-friendly experience. Now you can select the run name to see the Run Details view.
November 2023	Recent Experiment Runs	It is now simpler for users to check out recent runs for an experiment directly from the workspace list view ↗ . This update makes it easier to keep track of recent activity, quickly jump to the related Spark application, and apply filters based on the run status.
November 2023	Models renamed to ML Models	Microsoft has renamed "Models" to "ML Models" ↗ to ensure clarity and avoid any confusion with other Fabric elements. For more information, see Machine learning experiments in Microsoft Fabric.
November 2023	SynapseML v1.0	SynapseML v1.0 is now released. SynapseML v1.0 makes it easy to build production ready machine learning systems on Fabric ↗ and has been in use at Microsoft for over six years.
November 2023	Train Interpretable Explainable Boosting Machines with SynapseML	We've introduced a scalable implementation of Explainable Boosting Machines (EBM) powered by Apache Spark in SynapseML ↗ . EBMs are a powerful machine learning technique that combines the accuracy of gradient boosting with a strong focus on model interpretability.
November 2023	Prebuilt AI models in Microsoft Fabric preview	We are excited to announce the preview for prebuilt AI models in Fabric ↗ . Azure OpenAI Service ↗ , Text Analytics ↗ , and Azure AI Translator ↗ are prebuilt models available in Fabric, with support for both RESTful API and SynapseML. You can also use the OpenAI Python Library ↗ to access Azure OpenAI service in Fabric.
November 2023	Reusing existing Spark Session in sparklyr	We have added support for a new connection method called "synapse" in sparklyr ↗ , which enables users to connect to an existing Spark session. Additionally, we have contributed this connection method to the OSS sparklyr project. Users can now use both sparklyr and SparkR in the same session and easily share data between them.
November 2023	REST API Support for ML Experiments and ML Models	REST APIs for ML Experiment and ML Model ↗ are now available. These REST APIs for ML Experiments and ML Models begin to empower users to create and manage machine-learning artifacts programmatically, a key requirement for pipeline automation and workload integration.

Month	Feature	Learn more
October 2023	Semantic link (preview)	Semantic Link is an innovative feature that seamlessly connects Power BI semantic models with Synapse Data Science within Microsoft Fabric. As the gold layer in a medallion architecture, Power BI semantic models contain the most refined and valuable data in your organization.
October 2023	Semantic link in Microsoft Fabric: Bridging BI and Data Science	We are pleased to introduce the preview of semantic link ↗ , an innovative feature that seamlessly connects Power BI semantic models with Synapse Data Science within Microsoft Fabric.
October 2023	Get started with semantic link (preview)	Explore how semantic link seamlessly connects Power BI semantic models with Synapse Data Science within Microsoft Fabric. Learn more at Semantic link in Microsoft Fabric: Bridging BI and Data Science ↗ .
		You can also check out the semantic link sample notebooks that are now available in the fabric-samples ↗ GitHub repository. These notebooks showcase the use of semantic link's Python library, SemPy, in Microsoft Fabric.
August 2023	Harness the Power of LangChain in Microsoft Fabric for Advanced Document Summarization	Harness the potential of Microsoft Fabric and SynapseML LLM capabilities ↗ to effectively summarize and organize your own documents.
July 2023	Unleashing the Power of SynapseML and Microsoft Fabric: A Guide to Q&A on PDF Documents	In this blog post, we delve into the exciting functionalities and features of Microsoft Fabric and SynapseML ↗ to demonstrate how to leverage Generative AI models or Large Language Models (LLMs) to perform question and answer (Q&A) tasks on any PDF document ↗ .

Synapse Data Science samples and guidance

[\[\] Expand table](#)

Month	Feature	Learn more
November 2023	New data science happy path tutorial in Microsoft Fabric	We've updated the Data Science Happy Path tutorial for Microsoft Fabric ↗ . This new comprehensive tutorial demonstrates the entire data science workflow ↗ , using a bank customer churn problem as the context.
November 2023	New data science samples	We've expanded our collection of data science samples ↗ to include new end-to-end R samples and new quick tutorial

Month	Feature	Learn more
		samples for "Explaining Model Outputs" and "Visualizing Model Behavior." ↗ .
November 2023	New data science forecasting sample	The new Data Science sample on sales forecasting ↗ was developed in collaboration with Sonata Software ↗ . This new sample encompasses the entire data science workflow, spanning from data cleaning to Power BI visualization. The notebook covers the steps to develop, evaluate, and score a forecasting model for superstore sales, harnessing the power of the SARIMAX algorithm.
August 2023	New Machine failure and Customer churn samples	More samples have been added to the Microsoft Fabric Synapse Data Science Use a sample menu. To check these Data Science samples, select Synapse Data Science , then Use a sample .
August 2023	Use Semantic Kernel with Lakehouse in Microsoft Fabric	Learn how Fabric allows data scientists to use Semantic Kernel with Lakehouse in Microsoft Fabric ↗ .

Synapse Data Warehouse in Microsoft Fabric

This section summarizes recent improvements and features for Synapse Data Warehouse in Microsoft Fabric.

[\[\]](#) [Expand table](#)

Month	Feature	Learn more
December 2023	Automatic Log Checkpointing for Fabric Warehouse	Automatic Log Checkpointing ↗ is one of the ways that we help your Data Warehouse to provide you with great performance and best of all, it involves no additional work from you!
December 2023	Restore points and restore in place	You can now create restore points and perform an in-place restore of a warehouse to a past point in time. The restore points and restore in place features are currently in preview. Restore in-place is an essential part of data warehouse recovery ↗ which allows to restore the data warehouse to a prior known reliable state by replacing or over-writing the existing data warehouse from which the restore point was created.
November 2023	Mirroring in Microsoft Fabric	Any database can be accessed and managed centrally from within Fabric without having to switch database

Month	Feature	Learn more
		clients. By just providing connection details, your database is instantly available in Fabric as a Mirrored database ↗ . Azure Cosmos DB, Azure SQL Database, and Snowflake customers will be able to use Mirroring. SQL Server, Azure PostgreSQL, Azure MySQL, MongoDB, and other databases and data warehouses will be coming in CY24.
November 2023	TRIM T-SQL support	You can now use the <code>TRIM</code> command to remove spaces or specific characters from strings by using the keywords LEADING, TRAILING or BOTH in TRIM (Transact-SQL).
November 2023	GENERATE_SERIES T-SQL support	Generates a series of numbers within a given interval with <code>GENERATE_SERIES</code> (Transact-SQL). The interval and the step between series values are defined by the user.
November 2023	SSD metadata caching	File and rowgroup metadata are now also cached with in-memory and SSD cache, further improving performance.
November 2023	PARSER 2.0 improvements for CSV ingestion	CSV file parser version 2.0 for <code>COPY INTO</code> ↗ builds an innovation from Microsoft Research's Data Platform and Analytics group to make CSV file ingestion blazing fast on Fabric Warehouse. For more information, see <code>COPY INTO</code> (Transact-SQL).
November 2023	Fast compute resource assignment enabled	All query executions in Fabric Warehouse are now powered by the new technology recently deployed as part of the Global Resource Governance component that assigns compute resources in milliseconds.
November 2023	REST API support for Warehouse	With the Warehouse public APIs, SQL developers can now automate their pipelines and establish CI/CD conveniently and efficiently. The Warehouse REST Public APIs ↗ makes it easy for users to manage and manipulate Fabric Warehouse items.
November 2023	SQLPackage support for Fabric Warehouse	SQLPackage now supports Fabric Warehouse ↗ . SqlPackage is a command-line utility that automates the following database development tasks by exposing some of the public Data-Tier Application Framework (DacFx) APIs. The SqlPackage command line tool allows you to specify these actions along with action-specific parameters and properties.
November 2023	Power BI semantic models	Microsoft has renamed the Power BI <i>dataset</i> content type to <i>semantic model</i> . This applies to Microsoft Fabric

Month	Feature	Learn more
		semantic models as well. For more information, see New name for Power BI datasets .
November 2023	SQL analytics endpoint	Microsoft has renamed the SQL endpoint of a Lakehouse to the SQL analytics endpoint of a Lakehouse.
November 2023	Dynamic data masking	Dynamic Data Masking (DDM) ↗ for Fabric Warehouse and the SQL analytics endpoint in the Lakehouse. For more information and samples, see Dynamic data masking in Fabric data warehousing and How to implement dynamic data masking in Synapse Data Warehouse .
November 2023	Clone tables with time travel	You can now use table clones to create a clone of a table based on data up to seven calendar days in the past ↗ .
November 2023	User experience updates	Several user experiences in Warehouse have landed. For more information, see Fabric Warehouse user experience updates ↗ .
November 2023	Automatic data compaction	Automatic data compaction ↗ will rewrite many smaller parquet files into a few larger parquet files, which will improve the performance of reading the table. Data Compaction is one of the ways that we help your Warehouse to provide you with great performance and no effort on your part.
October 2023	Support for sp_rename	Support for the T-SQL <code>sp_rename</code> syntax is now available for both Warehouse and SQL analytics endpoint. For more information, see Fabric Warehouse support for sp_rename ↗ .
October 2023	Query insights	The query insights feature is a scalable, sustainable, and extendable solution to enhance the SQL analytics experience. With historic query data, aggregated insights, and access to actual query text, you can analyze and tune your query performance.
October 2023	Full DML to Delta Lake Logs	Fabric Warehouse now publishes all Inserts, Updates and Deletes ↗ for each table to their Delta Lake Log in OneLake.
October 2023	V-Order write optimization	V-Order optimizes parquet files to enable lightning-fast reads under the Microsoft Fabric compute engines such as Power BI, SQL, Spark and others. Warehouse queries in general benefit from faster read times with this optimization, still ensuring the parquet files are 100%

Month	Feature	Learn more
		compliant to its open-source specification. Starting this month, all data ingested into Fabric Warehouses use V-Order optimization.
October 2023	Burstable capacity	Burstable capacity allows workloads to use more resources to achieve better performance. Burstable capacity is finite, with a limit applied to the backend compute resources to greatly reduce the risk of throttling. For more information, see Warehouse SKU Guardrails for Burstable Capacity .
October 2023	Throttling and smoothing in Synapse Data Warehouse	A new article details the throttling and smoothing behavior in Synapse Data Warehouse, where almost all activity is classified as <i>background</i> to take advantage of the 24-hr smoothing window before throttling takes effect. Learn more about how to observe utilization in Synapse Data Warehouse.
September 2023	Default semantic model improvements	The default semantic model no longer automatically adds new objects. This can be enabled in the Warehouse item settings.
September 2023	Deployment pipelines now support warehouses	Deployment pipelines enable creators to develop and test content in the service before it reaches the users. Supported content types include reports, paginated reports, dashboards, semantic models, dataflows, and now warehouses. Learn how to deploy content programmatically using REST APIs and DevOps.
September 2023	SQL Projects support for Warehouse in Microsoft Fabric	Microsoft Fabric Data Warehouse is now supported in the SQL Database Projects extension available inside of Azure Data Studio and Visual Studio Code.
September 2023	Announcing: Column-level & Row-level security for Fabric Warehouse & SQL analytics endpoint	Column-level and row-level security in Fabric Warehouse and SQL analytics endpoint are now in preview, behaving similarly to the same features in SQL Server.
September 2023	Usage reporting	Utilization and billing reporting is available for Fabric data warehousing in the Microsoft Fabric Capacity Metrics app. For more information, read about Utilization and billing reporting Fabric data warehousing.
August 2023	SSD Caching enabled	Local SSD caching stores frequently accessed data on local disks in highly optimized format, significantly reducing I/O latency. This benefits you immediately, with no action required or configuration necessary.

Month	Feature	Learn more
July 2023	Sharing	Any Admin or Member within a workspace can share a Warehouse with another recipient ↗ within your organization. You can also grant these permissions using the "Manage permissions" experience.
July 2023	Table clone	A zero-copy clone ↗ creates a replica of the table by copying the metadata, while referencing the same data files in OneLake. This avoids the need to store multiple copies of data, thereby saving on storage costs when you clone a table in Microsoft Fabric. For more information, see tutorials to Clone a table with T-SQL or Clone tables in the Fabric portal.
May 2023	Introducing Synapse Data Warehouse in Microsoft Fabric	Synapse Data Warehouse is the next generation of data warehousing in Microsoft Fabric ↗ that is the first transactional data warehouse to natively support an open data format, Delta-Parquet.

Synapse Data Warehouse samples and guidance

[\[+\] Expand table](#)

Month	Feature	Learn more
November 2023	Migrate from Azure Synapse dedicated SQL pools	A detailed guide with a migration runbook is available for migrations from Azure Synapse Data Warehouse dedicated SQL pools into Microsoft Fabric.
August 2023	Efficient Data Partitioning with Microsoft Fabric: Best Practices and Implementation Guide	A proposed method for data partitioning using Fabric notebooks ↗ . Data partitioning is a data management technique used to divide a large dataset into smaller, more manageable subsets called partitions or shards.
May 2023	Microsoft Fabric - How can a SQL user or DBA connect	This blog reviews how to connect to a SQL analytics endpoint of the Lakehouse or the Warehouse through the Tabular Data Stream, or TDS endpoint ↗ , familiar to all modern web applications that interact with a SQL Server endpoint.

Synapse Real-Time Analytics in Microsoft Fabric

This section summarizes recent improvements and features for real-time analytics in Microsoft Fabric.

[+] Expand table

Month	Feature	Learn more
December 2023	Calculating distinct counts in Power BI running reports on KQL Databases	New Fabric KQL database dcount and dcountif functions use a special algorithm to return an estimate of distinct counts ↗ , even in extremely large datasets. The new functions count_distinct and count_distinctif calculate exact distinct counts.
December 2023	Create a Notebook with pre-configured connection to your KQL DB	You can now just create a new Notebook from KQL DB editor with a preconfigured connection to your KQL DB ↗ and explore the data using PySpark. This option creates a PySpark Notebook with a ready-to execute code cell to read data from the selected KQL DB.
December 2023	KQL Database schema validation	The new Kusto command <code>.show database schema violations</code> was designed to validate the current state of your database schema and find inconsistencies. You can use <code>.show database schema violations</code> for a spot check on your database or in CI/CD automation ↗ .
December 2023	Enabling Data Availability of KQL Database in OneLake	Data availability of KQL Database in OneLake ↗ means you can enjoy the best of both worlds. You can query the data with high performance and low latency in their KQL database, and you can query the same data in Delta Parquet via Power BI Direct Lake mode, Warehouse, Lakehouse, Notebooks and more.
December 2023	Fabric Change the Game: Real-time Analytics	Real-Time Analytics is a formidable tool ↗ , diminishing complexity and streamlining data integration processes. Microsoft Fabric allows you to build Real-Time streaming analytics with eventstream or Spark Stream.
November 2023	Announcing Delta Lake support in Real-Time Analytics KQL Database	You can now enable availability of KQL Database in Delta Lake format ↗ . Delta Lake is the unified data lake table format chosen to achieve seamless data access across all compute engines in Microsoft Fabric.
November 2023	Real-Time Analytics in Microsoft Fabric general availability (GA)	Announcing the general availability of Real-Time Analytics in Microsoft Fabric ↗ ! Real-Time Analytics offers countless features all aimed at making your data analysis more efficient and effective.
November 2023	Delta Parquet support in KQL Database	As part of the one logical copy promise, we are excited to announce that data in KQL Database can now be made available in OneLake in delta parquet format ↗ . You can

Month	Feature	Learn more
		now access this Delta table by creating a OneLake shortcut from Lakehouse, Warehouse, or directly via Power BI Direct Lake mode.
November 2023	Open Source Connectors for KQL Database	Several open-source connectors for real-time analytics are now supported ↗ to enable users to ingest data from various sources and process it using KQL DB.
November 2023	REST API Support for KQL Database	We're excited to announce the launch of REST Public APIs for KQL DB. The Public REST APIs of KQL DB ↗ enables users to manage and automate their flows programmatically.
November 2023	Eventstream now Generally Available	Eventstream is now generally available, adding enhancements aimed at taking your data processing experience to the next level.
November 2023	Eventstream Data Transformation for KQL Database	Now, you can transform your data streams into real time within Eventstream before they're sent to your KQL Database ↗ . When you create a KQL Database destination in the eventstream, you can set the ingestion mode to "Event processing before ingestion" and add event processing logics such as filtering and aggregation to transform your data streams.
November 2023	Splunk add-on preview	Microsoft Fabric add-on for Splunk ↗ allows users to ingest logs from Splunk platform into a Fabric KQL DB using the Kusto python SDK.
November 2023	Get Data from Eventstream anywhere in Fabric	If you're working on other Fabric items and are looking to ingest data from Eventstream, our new "Get Data from Eventstream" feature ↗ simplifies the process, you can Get data from Eventstream while you're working with a KQL database and Lakehouse.
November 2023	Two ingestion modes for Lakehouse Destination	We've introduced two distinct ingestion modes for your Lakehouse Destination: Rows per file and Duration ↗ .
November 2023	Optimize Tables Before Ingesting Data to Lakehouse	The table optimization shortcut is now available inside Eventstream Lakehouse destination ↗ to compact numerous small streaming files generated on a Lakehouse table. Table optimization shortcut works by opening a Notebook with Spark job, which would compact small streaming files in the destination Lakehouse table.
November 2023	Create a Cloud Connection within	We've simplified the process of establishing a cloud connection to your Azure services within Eventstream ↗ .

Month	Feature	Learn more
	Eventstream	When adding an Azure resource, such as Azure IoT Hub and Azure Event Hubs, to your Eventstream, you can now create the cloud connection and enter your Azure resource credentials right within Eventstream. This enhancement significantly improves the process of adding new data sources to your Eventstream, saving you time and effort.
November 2023	Get Data in Real-Time Analytics: A New and Improved Experience	A new Get Data experience ↗ simplifies the data ingestion process in your KQL database.
October 2023	Expanded Custom App Connections	New custom app connections ↗ provide more flexibility when it comes to bringing your data streams into Eventstream.
October 2023	Enhanced UX on Event Processor	New UX improvements on the no-code Event Processor ↗ provide an intuitive experience, allowing you to effortlessly add or delete operations on the canvas.
October 2023	Eventstream Kafka Endpoints and Sample Code	The Custom App feature has new endpoints in sources and destinations ↗ , including sample Java code for your convenience. Simply add it to your application, and you'll be all set to stream your real-time event to Eventstream.
October 2023	Event processing editor UX improvements	Recent UX improvements ↗ introduce a full-screen mode, providing a more spacious workspace for designing your data processing workflows. The insertion and deletion of data stream operations have been made more intuitive, making it easier to drag and drop and connect your data transformations.
October 2023	KQL Database Auto scale algorithm improvements	Users do not need to worry about how many resources are needed to support their workloads in a KQL database. KQL Database has a sophisticated in-built, multi-dimensional, auto scaling algorithm. We recently implemented some optimizations that will make some time series analysis more efficient ↗ .
October 2023	Understanding Fabric KQL DB Capacity	Read more about how a KQL database is billed ↗ in the SaaS world of Microsoft Fabric.
September 2023	OneLake shortcut to delta tables from KQL DB	Now you can create a shortcut from KQL DB to delta tables in OneLake, allowing in-place data queries. Now you query delta tables in your Lakehouse or Warehouse directly from KQL DB.

Month	Feature	Learn more
September 2023	Model and Query data as graphs using KQL	Kusto Query Language (KQL) now allows you to model and query data as graphs. This feature is currently in preview. Learn more at Introduction to graph semantics in KQL and Graph operators and functions .
September 2023	Easily connect to KQL Database from Power BI desktop	Power BI desktop has released two new ways to easily connect to a KQL database, in the Get Data dialogue and in the OneLake data hub menus.
September 2023	Eventstream now supports AMQP format connection string for data ingestion	AMQP stands for Advanced Message Queuing Protocol, a protocol that supports a wide range of messaging patterns. In Eventstream, you can now create a Custom App source or destination and select AMQP format connection string for ingesting data into Fabric or consuming data from Fabric.
September 2023	Eventstream supports data ingestion from Azure IoT Hub	Azure IoT Hub is a cloud-hosted solution that provides secure communication channels for sending and receiving data from IoT devices. In Eventstream, you can now stream your Azure IoT Hub data into Fabric and perform real-time processing before storing it in a Kusto Database or Lakehouse.
September 2023	Real-Time Data Sharing in Microsoft Fabric	A database shortcut in Real-Time Analytics is an embedded reference within a KQL database to a source database in Azure Data Explorer (ADX) allowing in-place data sharing. The behavior exhibited by the database shortcut is similar to that of an Azure Data Explorer follower database.
August 2023	Provisioning optimization	The KQL Database provisioning process has been optimized. Now you can provision a KQL Database within a few seconds.
August 2023	KQL Database support for inline Python	Fabric KQL Database supports running Python code embedded in Kusto Query Language (KQL) using the <code>python()</code> plugin. The plugin is disabled by default. Before you start, enable the Python plugin in your KQL database.
July 2023	Microsoft Fabric eventstreams: Generating Real-time Insights with Python, KQL, and Power BI	Microsoft Fabric eventstreams are a high-throughput, low-latency data ingestion and transformation service.
July 2023	Stream Real-time Events to Microsoft Fabric with	Eventstreams under Real-Time Analytics are a centralized platform within Fabric, allowing you to capture, transform,

Month	Feature	Learn more
	eventstreams from a custom application	and route real-time events to multiple destinations effortlessly, all through a user-friendly, no-code experience.
June 2023	Unveiling the Epic Opportunity: A Fun Game to Explore the Synapse Real-Time Analytics	As part of the Kusto Detective Agency Season 2 ↗ , we're excited to introduce an epic opportunity for all investigators and data enthusiasts to learn about the new portfolio in a fun and engaging way. Recruiting now at ↗!

Synapse Real-Time Analytics samples and guidance

[\[\] Expand table](#)

Month	Feature	Learn more
November 2023	Semantic Link: Data validation using Great Expectations	Great Expectations Open Source (GX OSS) is a popular Python library that provides a framework for describing and validating the acceptable state of data. With the recent integration of Microsoft Fabric semantic link, GX can now access semantic models ↗ , further enabling seamless collaboration between data scientists and business analysts.
November 2023	Explore Data Transformation in Eventstream for KQL Database Integration	Dive into a practical scenario using real-world bike-sharing data and learn to compute the number of bikes rented every minute on each street, using Eventstream's powerful event processor, mastering real-time data transformations, and effortlessly directing the processed data to your KQL Database. ↗ .
October 2023	From RabbitMQ to PowerBI reports with Microsoft Fabric Real-Time Analytics	A walkthrough of an end-to-end scenario sending data from RabbitMQ to a KQL Database in Microsoft Fabric ↗ .
October 2023	Stream Azure IoT Hub Data into Fabric Eventstream for Email Alerting	A demo of using Fabric Eventstream to seamlessly ingest and transform real-time data streams ↗ before they reach various Fabric destinations such as Lakehouse, KQL Database, and Reflex. Then, configure email alerts in Reflex with Data Activator triggers.
September 2023	Real-Time Analytics sample gallery	Real-Time Analytics now offers a comprehensive sample gallery with multiple datasets allowing you to explore, learn and get started quickly. Access the samples by selecting Use a sample from the Real-Time Analytics experience home ↗ .

Month	Feature	Learn more
September 2023	Quick start: Sending data to Synapse Real-Time Analytics in Fabric from Apache Kafka Ecosystems using Java	Learn how to send data from Kafka to Synapse Real-time Analytics in Fabric ↗ .
June 2023	From raw data to insights: How to ingest data from Azure Event Hubs into a KQL database	Learn about the integration between Azure Event Hubs and your KQL database ↗ .
June 2023	From raw data to insights: How to ingest data from eventstreams into a KQL database	Learn about the integration between eventstreams and a KQL database ↗ , both of which are a part of the Real-Time Analytics experience.
June 2023	Discovering the best ways to get data into a KQL database	This blog covers different options for bringing data into a KQL database ↗ .
June 2023	Get started with exploring your data with KQL – a purpose-built tool for petabyte scale data analytics	In this blog, we focus on the different ways of querying data in Synapse Real-Time Analytics ↗ .

Microsoft Copilot in Microsoft Fabric

[\[\] Expand table](#)

Month	Feature	Learn more
November 2023	Empower Power BI users with Microsoft Fabric and Copilot	We are thrilled to announce the general availability of Microsoft Fabric and the preview of Copilot in Microsoft Fabric, including the experience for Power BI ↗ .
November 2023	Copilot for Power BI in Microsoft Fabric preview	We are thrilled to announce the preview of Copilot in Microsoft Fabric ↗ , including the experience for Power BI, which helps users quickly get started by helping them create reports in the Power BI web experience. For more information, see Copilot for Power BI ↗ .
October 2023	Chat your data in Microsoft Fabric	Learn how to construct Copilot tools based on business data in Microsoft Fabric ↗ .

Month	Feature	Learn more
	with Semantic Kernel	

Microsoft Fabric core features

News and feature announcements core to the Microsoft Fabric experience.

[Expand table](#)

Month	Feature	Learn more
December 2023	Microsoft Fabric Admin APIs preview	Fabric Admin APIs ↗ are designed to streamline administrative tasks. The initial set of Fabric Admin APIs is tailored to simplify the discovery of workspaces, Fabric items, and user access details.
December 2023	Workspace retention changes in Fabric and Power BI	Starting November 15, 2023, the retention period for collaborative workspaces will be configurable from 7 to 90 days ↗ . The workspace retention setting will be enabled by default and the default retention period will be 7 days.
November 2023	Fabric workloads are now generally available!	Microsoft Fabric is now generally available! ↗ Microsoft Fabric Synapse Data Warehouse, Data Engineering & Data Science, Real-Time Analytics, Data Factory, OneLake, and the overall Fabric platform are now generally available.
November 2023	Microsoft Fabric User APIs preview	We're happy to announce the preview of Microsoft Fabric User APIs. The Fabric user APIs ↗ are a major enabler for both enterprises and partners to use Microsoft Fabric as they enable end-to-end fully automated interaction with the service, enable integration of Microsoft Fabric into external web applications, and generally enable customers and partners to scale their solutions more easily.
October 2023	Item type icons	Our design team has completed a rework of the item type icons across the platform ↗ to improve visual parsing.
October 2023	Keyword-Based Filtering of Tenant Settings	Microsoft Fabric has recently introduced keyword-based filtering for the tenant settings page in the admin portal ↗ .
September 2023	Monitoring hub – column options	Column options inside the monitoring hub give ↗ users a better customization experience and more room to operate.
September 2023	OneLake File Explorer v1.0.10	The OneLake file explorer ↗ automatically syncs all Microsoft OneLake items that you have access to in

Month	Feature	Learn more
		<p>Windows File Explorer. With the latest version, you can seamlessly transition between using the OneLake file explorer app and the Fabric web portal. You can also right-click on the OneLake icon in the Windows notification area, and select Diagnostic Operations to view client-site logs.</p> <p>Learn more about easy access to open workspaces and items online ↗.</p>
August 2023	Multitasking navigation improvement	<p>Now, all Fabric items are opened in a single browser tab on the left navigation bar, even in the event of a page refresh. This ensures you can refresh the page without the concern of losing context.</p>
August 2023	Monitoring Hub support for personalized column options	<p>We have updated Monitoring Hub to allow users to personalize activity-specific columns. You now have the flexibility to display columns that are relevant to the activities you're focused on.</p>
July 2023	New OneLake file explorer update with support for switching organizational accounts	<p>With OneLake file explorer ↗ v1.0.9.0, it's simple to choose and switch between different Microsoft Entra ID (formerly Azure Active Directory) accounts ↗.</p>
July 2023	Help pane	<p>The Help pane is feature-aware and displays articles about the actions and features available on the current Fabric screen. For more information, see Help pane ↗ in the monthly Fabric update.</p>

Continuous Integration/Continuous Delivery (CI/CD) in Microsoft Fabric

This section includes guidance and documentation updates on development process, tools, and versioning in the Microsoft Fabric workspace.

[\[+\] Expand table](#)

Month	Feature	Learn more
November 2023	Microsoft Fabric User APIs	<p>Microsoft Fabric User APIs are now available for Fabric experiences. The Fabric user APIs ↗ are a major enabler for both enterprises and partners to use Microsoft Fabric as they enable end-to-end fully automated interaction with the service, enable integration of Microsoft Fabric into</p>

Month	Feature	Learn more
		external web applications, and generally enable customers and partners to scale their solutions more easily.
November 2023	Notebook in Deployment Pipeline Preview	Now you can also use notebooks to deploy your code across different environments, such as development, test, and production. You can also use deployment rules to customize the behavior of your notebooks when they're deployed, such as changing the default Lakehouse of a Notebook. Get started with deployment pipelines to set up your deployment pipeline, Notebook will show up in the deployment content automatically.
November 2023	Notebook Git integration preview	Fabric notebooks now offer Git integration for source control using Azure DevOps. It allows users to easily control the notebook code versions and manage the Git branches by leveraging the Fabric Git functions and Azure DevOps.
November 2023	Notebook REST APIs Preview	With REST Public APIs for the Notebook items, data engineers/data scientists can automate their pipelines and establish CI/CD conveniently and efficiently. The notebook Restful Public API ↗ can make it easy for users to manage and manipulate Fabric notebook items and integrate notebook with other tools and systems.
November 2023	Lakehouse support for git integration and deployment pipelines (preview)	The Lakehouse artifact now integrates with the lifecycle management capabilities in Microsoft Fabric, providing a standardized collaboration between all development team members throughout the product's life. Lifecycle management facilitates an effective product versioning and release process by continuously delivering features and bug fixes into multiple environments.
November 2023	SQLPackage support for Fabric Warehouse	SQLPackage now supports Fabric Warehouse. SqlPackage is a command-line utility that automates the following database development tasks by exposing some of the public Data-Tier Application Framework (DacFx) APIs. The SqlPackage command line tool allows you to specify these actions along with action-specific parameters and properties.
September 2023	SQL Projects support for Warehouse in Microsoft Fabric	Microsoft Fabric Data Warehouse is now supported in the SQL Database Projects extension available inside of Azure Data Studio and Visual Studio Code ↗ .
September 2023	Notebook file system support in Synapse VS Code extension	The Synapse VS Code extension now supports notebook File System ↗ for Data Engineering and Data Science in Microsoft Fabric. The Synapse VS Code extension empowers

Month	Feature	Learn more
		users to develop their notebook artifacts directly within the Visual Studio Code environment.
September 2023	Deployment pipelines now support warehouses	Deployment pipelines enable creators to develop and test content in the service before it reaches the users. Supported content types include reports, paginated reports, dashboards, semantic models, dataflows, and now warehouses. Learn how to deploy content programmatically using REST APIs and DevOps.
September 2023	Git integration with paginated reports in Power BI	You can now publish a Power BI paginated report and keep it in sync with your git workspace. Developers can apply their development processes, tools, and best practices.
August 2023	Introducing the dbt adapter for Synapse Data Warehouse in Microsoft Fabric	The dbt adapter allows you to connect and transform data into Synapse Data Warehouse ↗ . The Data Build Tool (dbt) is an open-source framework that simplifies data transformation and analytics engineering.
May 2023	Introducing git integration in Microsoft Fabric for seamless source control management	While developing in Fabric, developers can back up and version their work, roll back as needed, collaborate or work in isolation using git branches ↗ . Read more about connecting the workspace to an Azure repo.

Data Activator in Microsoft Fabric

Data Activator is a no-code experience in Microsoft Fabric for automatically taking actions when patterns or conditions are detected in changing data. This section summarizes recent new features and capabilities of Data Activator in Microsoft Fabric.

[\[+\] Expand table](#)

Month	Feature	Learn more
October 2023	Announcing the Data Activator preview	We are thrilled to announce that Data Activator is now in preview ↗ and is enabled for all existing Microsoft Fabric users.
August 2023	Updated preview experience for trigger design	We have been working on a new experience for designing triggers ↗ and it's now available in our preview! You now see three cards in every trigger: Select, Detect, and Act.
May 2023	Driving actions from your data with Data	Data Activator is a new no-code Microsoft Fabric experience ↗ that empowers the business analyst to drive

Month	Feature	Learn more
	Activator	actions automatically from your data. To learn more, sign up for the Data Activator limited preview.

Fabric and Microsoft 365

This section includes articles and announcements about Microsoft Fabric integration with Microsoft Graph and Microsoft 365.

[] [Expand table](#)

Month	Feature	Learn more
November 2023	Fabric + Microsoft 365 Data: Better Together	Microsoft Graph is the gateway to data and intelligence in Microsoft 365. Microsoft 365 Data Integration for Microsoft Fabric [↗] enables you to manage your Microsoft 365 alongside your other data sources in one place with a suite of analytical experiences.
November 2023	Microsoft 365 connector now supports ingesting data into Lakehouse (preview)	The Microsoft 365 connector now supports ingesting data into Lakehouse tables [↗] .
October 2023	Microsoft OneLake adds shortcut support to Power Platform and Dynamics 365	You can now create shortcuts directly to your Dynamics 365 and Power Platform data in Dataverse [↗] and analyze it with Microsoft Fabric alongside the rest of your OneLake data. There is no need to export data, build ETL pipelines or use third-party integration tools.

Migration

This section includes guidance and documentation updates on migration to Microsoft Fabric.

[] [Expand table](#)

Month	Feature	Learn more
November 2023	Migrate from Azure Synapse dedicated SQL pools	A detailed guide with a migration runbook is available for migrations from Azure Synapse Data Warehouse dedicated SQL pools into Microsoft Fabric.

Month	Feature	Learn more
November 2023	Migrating from Azure Synapse Spark to Fabric	A detailed set of articles on migration of Azure Synapse Spark to Microsoft Fabric, including a migration process that can involve multiple scenarios and phases.
July 2023	Fabric Changing the game – OneLake integration	This blog post covers OneLake integrations and multiple scenarios to ingest the data inside of Fabric OneLake ↗ , including ADLS, ADF, OneLake Explorer, Databricks.
June 2023	Microsoft Fabric changing the game: Exporting data and building the Lakehouse	This blog post covers the scenario to export data from Azure SQL Database into OneLake ↗ .
June 2023	Copy data to Azure SQL at scale with Microsoft Fabric	Did you know that you can use Microsoft Fabric to copy data at scale from supported data sources to Azure SQL Database or Azure SQL Managed Instance within minutes? ↗
June 2023	Bring your Mainframe DB2 z/OS data to Microsoft Fabric	In this blog, we review the convenience and ease of opening DB2 for z/OS data in Microsoft Fabric ↗ .

Monitoring

This section includes guidance and documentation updates on monitoring your Microsoft Fabric capacity and utilization, including the Monitoring hub.

[\[\] Expand table](#)

Month	Feature	Learn more
October 2023	Throttling and smoothing in Synapse Data Warehouse	A new article helps you understand Fabric capacity throttling. Throttling occurs when a tenant's capacity consumes more capacity resources than it has purchased over a period of time.
September 2023	Monitoring hub - column options	Users can select and reorder the columns according to their customized needs in the Monitoring hub ↗ .
September 2023	Fabric Capacities – Everything you need to know about what's new and what's coming	Read more about the improvements we're making to the Fabric capacity management platform for Fabric and Power BI users ↗ .

Month	Feature	Learn more
September 2023	Microsoft Fabric Capacity Metrics	The Microsoft Fabric Capacity Metrics app is available in App Source for a variety of billing and utilization reporting.
August 2023	Monitoring Hub support for personalized column options	We have updated Monitoring Hub to allow users to personalize activity-specific columns. You now have the flexibility to display columns that are relevant to the activities you're focused on.
May 2023	Capacity metrics in Microsoft Fabric	Learn more about the universal compute capacities and Fabric's capacity metrics governance features ↗ that admins can use to monitor usage ↗ and make data-driven scale-up decisions.

Microsoft Purview

This section summarizes recent announcements about governance and compliance capabilities with Microsoft Purview in Microsoft Fabric. Learn more about Information protection in Microsoft Fabric.

[Expand table](#)

Month	Feature	Learn more
May 2023	Administration, Security and Governance in Microsoft Fabric	Microsoft Fabric provides built-in enterprise grade governance and compliance capabilities ↗ , powered by Microsoft Purview.

Related content

For older updates, review previous updates in Microsoft Fabric.

- Modernization Best Practices and Reusable Assets Blog [↗](#)
- Azure Data Explorer Blog [↗](#)
- Fabric Known Issues [↗](#)
- Microsoft Fabric Blog [↗](#)
- Microsoft Fabric terminology
- What's new in Power BI?

Next step

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Data Factory pricing in Microsoft Fabric

Article • 11/15/2023

Data Factory in Microsoft Fabric provides serverless and elastic data integration service capabilities built for cloud scale. There's no fixed-size compute power that you need to plan for peak load; rather you need to specify which operations to perform while authoring pipelines and dataflows, which translates into an amount of **Fabric Capacity Units** consumed, that you can further track using the Microsoft Fabric Capacity Metrics app to plan and manage your consumption metrics. This allows you to design the ETL processes in a much more scalable manner. In addition, Data Factory, like other Fabric experiences, is billed on a consumption-based plan, which means you only pay for what you use.

Microsoft Fabric Capacities

Fabric is a unified data platform that offers shared experiences, architecture, governance, compliance, and billing. Capacities provide the computing power that drives all of these experiences. They offer a simple and unified way to scale resources to meet customer demand and can be easily increased with a SKU upgrade.



You can manage your Fabric Data Factory run costs easily with simplified billing. Additional users don't require any cost management on a per-user basis, and you can save money by planning and committing Fabric capacities for your data integration projects ahead. With the pay-as-you-go option, you can easily scale your capacities up and down to adjust their computing power and pause their capacities when not in use to save costs. Learn more about Fabric capacities and usage billing.

Data Factory pricing meters

Whether you're a citizen or professional developer, Data Factory enables you to develop enterprise-scale data integration solutions with next-generation dataflows and data pipelines. These experiences operate on multiple services with different capacity meters. Data pipelines use **Data Orchestration** and **Data Movement** meters, while Dataflow Gen2 uses **Standard Compute** and **High Scale Compute**. Additionally, like other Fabric experiences, the common meter for storage consumption is OneLake Storage.

Data Factory

- Pipelines
- Dataflows Gen2

Data Orchestration

Standard Compute

High-Scale Compute

Data Movement*

OneLake Storage

* Data Movement meter will be leveraged by Dataflows Gen2 as part of the upcoming Fast Copy feature

Pricing examples

Here are some example scenarios for pricing of data pipelines:

- Load 1 TB Parquet to a data warehouse
- Load 1 TB Parquet to a data warehouse via staging

- Load 1 TB CSV files to a Lakehouse table
- Load 1 TB CSV files to a Lakehouse files with binary copy
- Load 1 TB Parquet to a Lakehouse table

Here are some pricing examples for Dataflow Gen2:

- Load on-premises 2 GB CSV file to a Lakehouse table
- Load 2 GB Parquet to a Lakehouse table

Next steps

- Data pipelines pricing for Data Factory in Microsoft Fabric
 - Dataflow Gen2 pricing for Data Factory in Microsoft Fabric
-

Feedback

Was this page helpful?

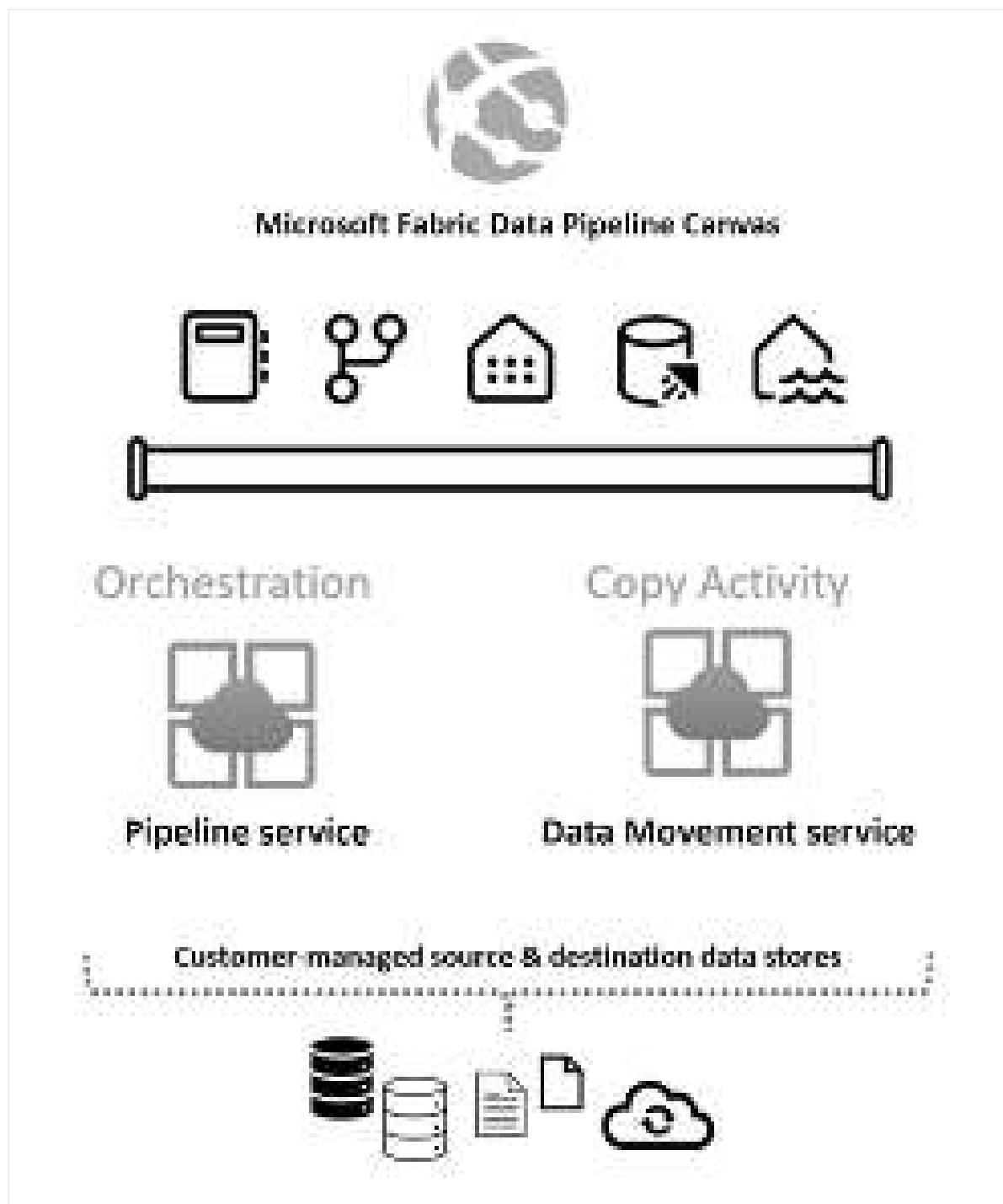


Provide product feedback [↗](#) | Ask the community [↗](#)

Data pipelines pricing for Data Factory in Microsoft Fabric

Article • 11/15/2023

Data pipelines enable you to apply rich out-of-the-box data orchestration capabilities to compose flexible data workflows that meet your enterprise data movement and ingestion needs. These capabilities require different computing service engines that have distinct consumption rates.



When you run a Data Pipeline with Data Factory in Microsoft Fabric, Fabric Capacity Units are consumed for the following services:

- Pipeline services for orchestration of *activity runs*: Your charge is based on the number of activity runs that are orchestrated.
- Data Movement service for Copy activity runs. You are charged based on the Capacity Units consumed during the Copy activity execution duration.

Pricing model

The following table shows a breakdown of the pricing model for data pipelines within Data Factory in Microsoft Fabric:

Data Pipelines Engine Type	Charge Meters and Metric Units	Fabric Capacity Units (CUs) consumption rate
Data orchestration	Based on Copy activity run duration in hours and the used intelligent optimization throughput resources	1.5 CUs / hour
Data movement	Incorporates orchestration activity runs and activity integration runtime charges	0.0056 CUs for each non-copy activity run

It indicates that for each intelligent optimization throughput resource usage in a Data Pipeline execution, 1.5 CU per hour are consumed for data movement Copy activities. Secondly, each orchestration activity run consumes 0.0056 CU. At the end of each pipeline run, the CU consumption for each engine type is summed and is billed as per the translated price of the Fabric Capacity in the region where the capacity is deployed.

ⓘ Note

Whenever a pipeline orchestration activity triggers other Fabric artifacts to run (for example, Notebook or Dataflow Gen2), the consumption for those artifacts needs to be taken into account as well.

Changes to Microsoft Fabric workload consumption rate

Consumption rates are subject to change at any time. Microsoft uses reasonable efforts to provide notice via email and in-product notification. Changes are effective on the date stated in the Release Notes and the Microsoft Fabric Blog ↗. If any change to a Microsoft Fabric Workload Consumption Rate materially increases the Capacity Units (CU) required to use a particular workload, customers can use the cancellation options available for the chosen payment method.

Compute estimated costs using the Fabric Metrics App

The Microsoft Fabric Capacity Metrics app provides visibility into capacity usage for all Fabric workspaces tied to a capacity. It's used by capacity administrators to monitor the performance of workloads and their usage compared to purchased capacity. Using the Fabric Metrics App is the most accurate way to estimate the costs of data pipeline executions.

The following table can be used as a template to compute estimated costs using Fabric Metrics app for a data pipeline run:

Metric	Data movement operation	Activity run operation
Duration in seconds	t in seconds	N/A
CU seconds	x CU seconds	y CU seconds
Effective CU-hour	x CU seconds / (60*60) = X CU-hour	y CU(s) / (60*60) = Y CU-hour

Total cost: (X + Y CU-hour) * (Fabric capacity per unit price)

Next steps

- Pricing example scenarios
- Pricing Dataflow Gen2

Feedback

Was this page helpful?

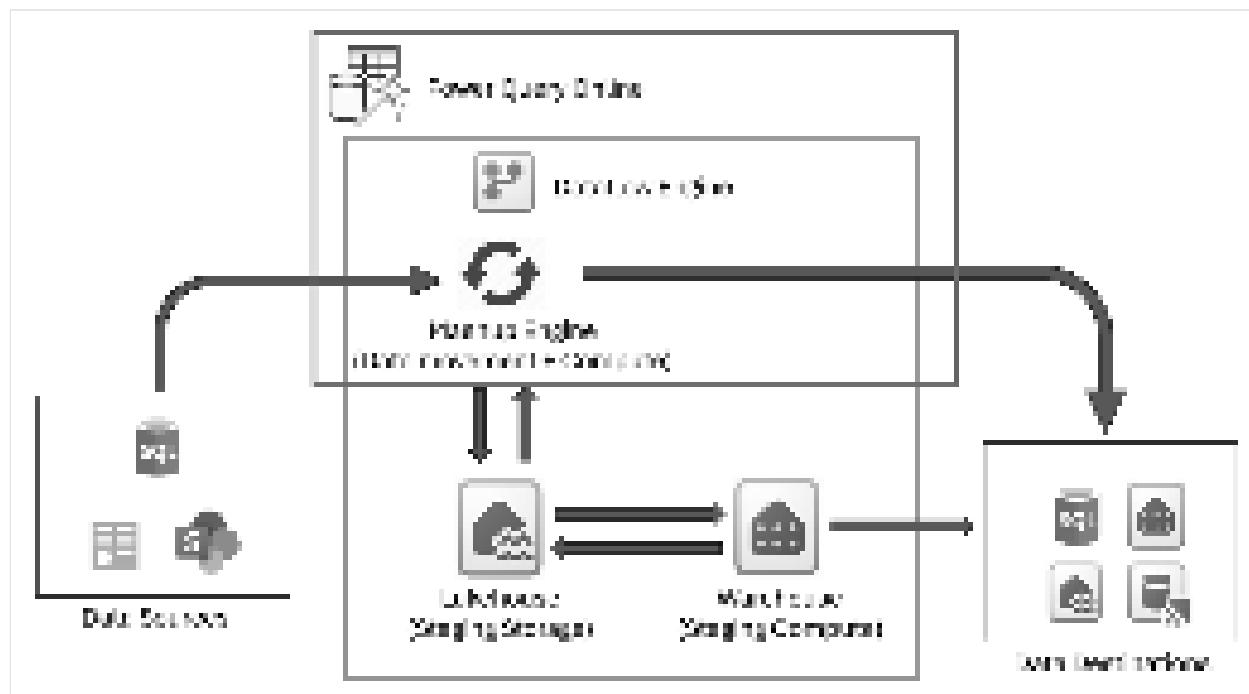


Provide product feedback [↗](#) | Ask the community [↗](#)

Dataflow Gen2 pricing for Data Factory in Microsoft Fabric

Article • 11/15/2023

Dataflow Gen2 enables you to leverage a low-code interface and 300+ data and AI-based transformations, letting you transform data easier and with more flexibility than any other tool. Dataflow Gen2 is authored using the familiar Power Query experience that's available today across several Microsoft products and services such as Excel, Power BI, Power Platform, Dynamics 365 Insights applications, and more. Once you publish a dataflow, the dataflow definition is generated - this is the program that will run once the dataflow is refreshed to produce tables in staging storage and/or output destination. During refresh, the definition of a dataflow is used by the dataflow engine to generate an orchestration plan, manage resources, and orchestrate execution of queries across data sources, gateways, and compute engines, and to create tables in either the staging storage or data destination.



The diagram shown here captures the various components of the Data Factory Dataflow Gen2 architecture, including the Lakehouse used to stage data being ingested, and Warehouse artifact used as a compute engine and means to write back results to staging or supported output destinations faster. When Warehouse compute cannot be used, or when staging is disabled for a query, the Mashup Engine will extract, transform, or load the data to staging or data destinations. You can learn more about how Dataflow Gen2 works in this blog post here: [Data Factory Spotlight: Dataflow Gen2](#).

When you refresh or publish a Dataflow Gen2 item, Fabric Capacity Units are consumed for the following engines.

- Standard Compute: You're charged for it based on the query evaluation time across all your Dataflow queries ran through the Mashup engine.
- High Scale Dataflows Compute: You are charged when staging is enabled based on Lakehouse (Staging storage) and Warehouse (Storage Compute) SQL engine consumption duration.

Dataflow Gen2 pricing model

The following table indicates that to determine Dataflow Gen2 execution costs, each query execution utilizes the mashup engine for standard computing, and that compute execution duration is translated to a consumption rate of 16 CUs per hour. Secondly, for high scale compute scenarios when staging is enabled, Lakehouse/Warehouse SQL engine execution duration should be accounted for as well. Compute execution duration is translated to a consumption rate of 6 CUs per hour. At the end of each Dataflow Gen2 run, the Capacity Unit (CU) consumption for each engine type is summed and is billed according to the translated price for Fabric capacity in the region where it's deployed.

Dataflow Gen2 Engine Type	Consumption Meters	Fabric CU consumption rate	Consumption reporting granularity
Standard Compute	Based on each mashup engine query execution duration in seconds.	16 CUs per hour	Per Dataflow Gen2 item
High Scale Dataflows Compute	Based on Lakehouse/Warehouse SQL engine execution (with staging enabled) duration in seconds.	6 CUs per hour	Per workspace

Note

It isn't currently possible to cancel a Dataflow Gen2 run, but we will add this capability by January, 2024.

Changes to Microsoft Fabric workload consumption rate

Consumption rates are subject to change at any time. Microsoft uses reasonable efforts to provide notice via email and in-product notification. Changes are effective on the

date stated in the Release Notes and the Microsoft Fabric Blog [↗](#). If any change to a Microsoft Fabric Workload Consumption Rate materially increases the Capacity Units (CU) required to use a particular workload, customers can use the cancellation options available for the chosen payment method.

Compute estimated costs using the Fabric Metrics App

The Microsoft Fabric Capacity Metrics app provides visibility into capacity usage for all Fabric workspaces tied to a capacity. It's used by capacity administrators to monitor the performance of workloads and their usage compared to purchased capacity. Using the Metrics app is the most accurate way to estimate the costs of Dataflow Gen2 refresh runs. When you load-test your scenario, create the Dataflow Gen2 item in a new workspace to reduce any reported noise in the Fabric Metrics App.

The following table can be utilized as a template to compute estimated costs using Fabric Metrics app for a Dataflow Gen2 refresh:

Metric	Standard Compute	High Scale Compute
Total CUs	s CU seconds	h CU seconds
Effective CU-hours billed	$s / (60*60) = S$ CU-hour	$h / (60*60) = H$ CU-hour

Total refresh cost = $(S + H)$ CU-hour * (Fabric capacity per unit price)

Next steps

- Pricing example scenarios
- Pricing data pipelines

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback [↗](#) | Ask the community [↗](#)

Pricing scenario using a data pipeline to load 1 TB of Parquet data to a data warehouse

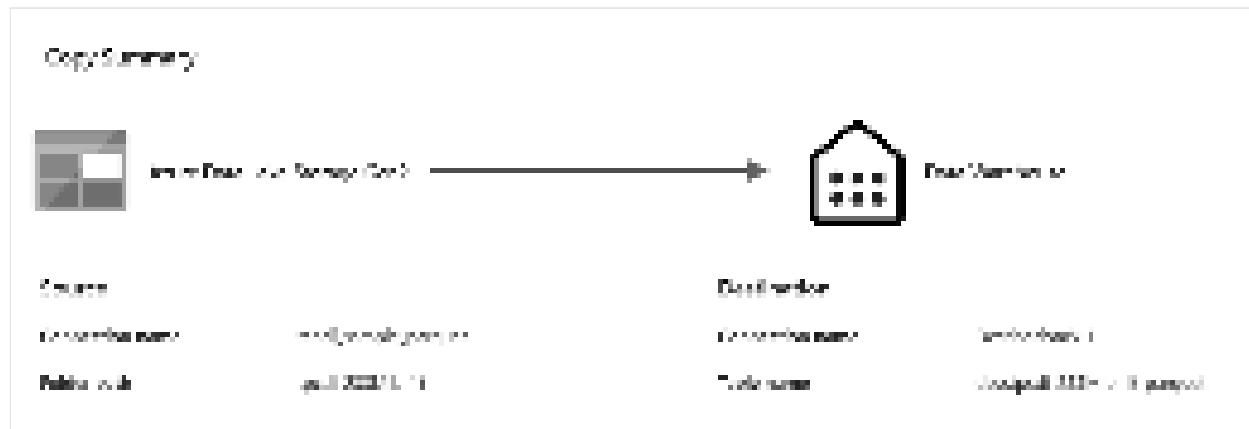
Article • 11/15/2023

In this scenario, a Copy activity was used in a data pipeline to load 1 TB of Parquet table data stored in Azure Data Lake Storage (ADLS) Gen2 to a data warehouse in Microsoft Fabric.

The prices used in the following example are hypothetical and don't intend to imply exact actual pricing. These are just to demonstrate how you can estimate, plan, and manage cost for Data Factory projects in Microsoft Fabric. Also, since Fabric capacities are priced uniquely across regions, we use the pay-as-you-go pricing for a Fabric capacity at US West 2 (a typical Azure region), at \$0.18 per CU per hour. Refer here to Microsoft Fabric - Pricing [↗](#) to explore other Fabric capacity pricing options.

Configuration

To accomplish this scenario, you need to create a pipeline with the following configuration:



Cost estimation using the Fabric Metrics App

The data movement operation utilized 3,960 CU seconds with a 662.64 second duration while activity run operation was null since there weren't any non-copy activities in the

pipeline run.

Note

Although reported as a metric, the actual duration of the run isn't relevant when calculating the effective CU hours with the Fabric Metrics App since the CU seconds metric it also reports already accounts for its duration.

Metric	Data Movement Operation
CU seconds	3,960 CU seconds
Effective CU-hours	$(3,960) / (60*60)$ CU-hours = 1.1 CU-hours

Total run cost at \$0.18/CU hour = $(1.1 \text{ CU-hour}) * (\$0.18/\text{CU hour}) \approx \0.20

Next steps

- Data pipelines pricing for Data Factory in Microsoft Fabric
- Dataflow Gen2 pricing for Data Factory in Microsoft Fabric
- Pricing example scenarios

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Pricing scenario using a data pipeline to load 1 TB of Parquet data to a data warehouse with staging

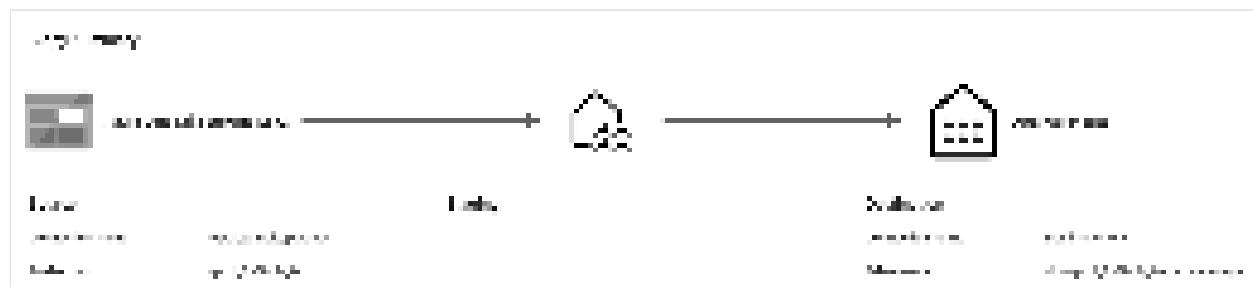
Article • 11/15/2023

In this scenario, a Copy activity was used in a data pipeline to load 1 TB of Parquet table data stored in Azure Data Lake Storage (ADLS) Gen2 to a data warehouse with staging in Microsoft Fabric.

The prices used in the following example are hypothetical and don't intend to imply exact actual pricing. These are just to demonstrate how you can estimate, plan, and manage cost for Data Factory projects in Microsoft Fabric. Also, since Fabric capacities are priced uniquely across regions, we use the pay-as-you-go pricing for a Fabric capacity at US West 2 (a typical Azure region), at \$0.18 per CU per hour. Refer here to Microsoft Fabric - Pricing [↗](#) to explore other Fabric capacity pricing options.

Configuration

To accomplish this scenario, you need to create a pipeline with the following configuration:



Cost estimation using the Fabric Metrics App



The data movement operation utilized 267,480 CU seconds with a 1504.42 second (25.07 minute) duration while activity run operation was null since there weren't any non-copy activities in the pipeline run.

Note

Although reported as a metric, the actual duration of the run isn't relevant when calculating the effective CU hours with the Fabric Metrics App since the CU seconds metric it also reports already accounts for its duration.

Metric	Data Movement Operation
CU seconds	267,480 CU seconds
Effective CU-hours	$(267,480) / (60*60)$ CU-hours = 74.3 CU-hours

Total run cost at \$0.18/CU hour = (74.3 CU-hours) * (\$0.18/CU hour) ~= \$13.37

Next steps

- Data pipelines pricing for Data Factory in Microsoft Fabric
- Dataflow Gen2 pricing for Data Factory in Microsoft Fabric
- Pricing example scenarios

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Pricing scenario using a data pipeline to load 1 TB of CSV data to a Lakehouse table

Article • 11/15/2023

In this scenario, a Copy activity was used in a data pipeline to load 1 TB of CSV data stored in Azure Data Lake Storage (ADLS) Gen2 to a Lakehouse table in Microsoft Fabric.

The prices used in the following example are hypothetical and don't intend to imply exact actual pricing. These are just to demonstrate how you can estimate, plan, and manage cost for Data Factory projects in Microsoft Fabric. Also, since Fabric capacities are priced uniquely across regions, we use the pay-as-you-go pricing for a Fabric capacity at US West 2 (a typical Azure region), at \$0.18 per CU per hour. Refer here to Microsoft Fabric - Pricing [↗](#) to explore other Fabric capacity pricing options.

Configuration

To accomplish this scenario, you need to create a pipeline with the following configuration:



Cost estimation using the Fabric Metrics App



The data movement operation utilized 282,240 CU seconds with a 763.78 second (12.73 minute) duration while activity run operation was null since there weren't any non-copy

activities in the pipeline run.

Note

Although reported as a metric, the actual duration of the run isn't relevant when calculating the effective CU hours with the Fabric Metrics App since the CU seconds metric it also reports already accounts for its duration.

Metric	Data Movement Operation
CU seconds	282,240 CU seconds
Effective CU-hours	$(282,240) / (60*60)$ CU-hours = 78.4 CU-hours

Total run cost at \$0.18/CU hour = $(78.4 \text{ CU-hours}) * (\$0.18/\text{CU hour}) \approx \14.11

Next steps

- Data pipelines pricing for Data Factory in Microsoft Fabric
- Dataflow Gen2 pricing for Data Factory in Microsoft Fabric
- Pricing example scenarios

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Pricing scenario using a data pipeline to load 1 TB of CSV data to Lakehouse files

Article • 11/15/2023

In this scenario, a Copy activity was used in a data pipeline to load 1 TB of CSV data stored in Azure Data Lake Storage (ADLS) Gen2 to Lakehouse files in Microsoft Fabric.

The prices used in the following example are hypothetical and don't intend to imply exact actual pricing. These are just to demonstrate how you can estimate, plan, and manage cost for Data Factory projects in Microsoft Fabric. Also, since Fabric capacities are priced uniquely across regions, we use the pay-as-you-go pricing for a Fabric capacity at US West 2 (a typical Azure region), at \$0.18 per CU per hour. Refer here to Microsoft Fabric - Pricing [↗](#) to explore other Fabric capacity pricing options.

Configuration

To accomplish this scenario, you need to create a pipeline with the following configuration:



Cost estimation using the Fabric Metrics App



The data movement operation utilized 95,760 CU seconds with a 290.27 second (4.84 minute) duration while activity run operation was null since there weren't any non-copy activities in the pipeline run.

Note

Although reported as a metric, the actual duration of the run isn't relevant when calculating the effective CU hours with the Fabric Metrics App since the CU seconds metric it also reports already accounts for its duration.

Metric	Data Movement Operation
CU seconds	95,760 CU seconds
Effective CU-hours	$(95,760) / (60*60)$ CU-hours = 26.6 CU-hours

Total run cost at \$0.18/CU hour = $(26.6 \text{ CU-hours}) * (\$0.18/\text{CU hour}) \approx \4.79

Next steps

- Data pipelines pricing for Data Factory in Microsoft Fabric
- Dataflow Gen2 pricing for Data Factory in Microsoft Fabric
- Pricing example scenarios

Feedback

Was this page helpful?



Yes



No

Provide product feedback  | Ask the community 

Pricing scenario using a data pipeline to load 1 TB of Parquet data to a Lakehouse table

Article • 11/15/2023

In this scenario, a Copy activity was used in a data pipeline to load 1 TB of Parquet data stored in Azure Data Lake Storage (ADLS) Gen2 to a Lakehouse table in Microsoft Fabric.

The prices used in the following example are hypothetical and don't intend to imply exact actual pricing. These are just to demonstrate how you can estimate, plan, and manage cost for Data Factory projects in Microsoft Fabric. Also, since Fabric capacities are priced uniquely across regions, we use the pay-as-you-go pricing for a Fabric capacity at US West 2 (a typical Azure region), at \$0.18 per CU per hour. Refer here to Microsoft Fabric - Pricing [↗](#) to explore other Fabric capacity pricing options.

Configuration

To accomplish this scenario, you need to create a pipeline with the following configuration:



Cost estimation using the Fabric Metrics App

Activity	Duration (s)	CPU (CU)	Memory (GB)	Network (GB)	Total (CU)
Copy	810.16	291,960	1	1	291,960
Activity run	0.00	0	0	0	0

The data movement operation utilized 291,960 CU seconds with a 810.16 second (13.5 minute) duration while activity run operation was null since there weren't any non-copy activities in the pipeline run.

Note

Although reported as a metric, the actual duration of the run isn't relevant when calculating the effective CU hours with the Fabric Metrics App since the CU seconds metric it also reports already accounts for its duration.

Metric	Data Movement Operation
CU seconds	291,960 CU seconds
Effective CU-hours	$(291,960) / (60*60)$ CU-hours = 81.1 CU-hours

Total run cost at \$0.18/CU hour = $(81.1 \text{ CU-hours}) * (\$0.18/\text{CU hour}) \approx \14.60

Next steps

- Data pipelines pricing for Data Factory in Microsoft Fabric
- Dataflow Gen2 pricing for Data Factory in Microsoft Fabric
- Pricing example scenarios

Feedback

Was this page helpful?



Yes



No

Provide product feedback  | Ask the community 

Pricing scenario using Dataflow Gen2 to load 2 GB of on-premises CSV data to a Lakehouse table

Article • 11/15/2023

In this scenario, Dataflow Gen2 was used to load 2 GB of on-premises CSV data to a Lakehouse table in Microsoft Fabric.

The prices used in the following example are hypothetical and don't intend to imply exact actual pricing. These are just to demonstrate how you can estimate, plan, and manage cost for Data Factory projects in Microsoft Fabric. Also, since Fabric capacities are priced uniquely across regions, we use the pay-as-you-go pricing for a Fabric capacity at US West 2 (a typical Azure region), at \$0.18 per CU per hour. Refer here to Microsoft Fabric - Pricing [↗](#) to explore other Fabric capacity pricing options.

Configuration

To accomplish this scenario, you need to create a dataflow with the following steps:

1. Initialize Dataflow: Start by uploading 2 GB CSV files from your on-premises environment into the dataflow.
2. Configure Power Query:
 - a. Navigate to Power Query.
 - b. Disable the option for staging the query.
 - c. Proceed to combine the CSV files.
3. Data Transformation:
 - a. Promote headers for clarity.
 - b. Remove unnecessary columns.
 - c. Adjust column data types as needed.
4. Define Output Data Destination:
 - a. Configure Lakehouse as the data output destination.
 - b. In this example, a Lakehouse within Fabric was created and utilized.

Cost estimation using the Fabric Metrics App

Category	Duration	Cost	Information	Tags
File processing	1 hour	\$0.18	File processing	File processing
File processing	1 hour	\$0.18	File processing	File processing
File processing	1 hour	\$0.18	File processing	File processing
Total	3 hours	\$0.54		

Operation name	CU (s)	Duration (s)	Users	Billing type
Dataflow Gen2 Refresh	4,749.42	309.45	1	Billable
Total	4,749.42	309.45	1	

Operation name	CU (s)	Duration (s)	Users	Billing type
High Scale Dataflow Com...	7.78	0.36	2	Billable
Total	7.78	0.36	2	

Operation name	CU (s)	Duration (s)	Users	Billing type
High Scale Dataflow Com...	7.85	0.37	2	Billable
Total	7.85	0.37	2	

The Dataflow Gen2 Refresh operation consumed 4749.42 CU seconds, and two High Scale Dataflows Compute operations consumed 7.78 CU seconds + 7.85 CU seconds each.

ⓘ Note

Although reported as a metric, the actual duration of the run isn't relevant when calculating the effective CU hours with the Fabric Metrics App since the CU seconds metric it also reports already accounts for its duration.

Metric	Compute consumption
Dataflow Gen2 Refresh CU seconds	4749.42 CU seconds
High Scale Dataflows Compute CU seconds	(7.78 + 7.85) 15.63 CU seconds
Effective CU hours billed	(4749.42 + 15.63) / (60*60) = 1.32 CU hours

Total run cost at \$0.18/CU hour = (1.32 CU-hours) * (\$0.18/CU hour) ~= \$0.24

Next steps

- Data pipelines pricing for Data Factory in Microsoft Fabric
- Dataflow Gen2 pricing for Data Factory in Microsoft Fabric
- Pricing example scenarios

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Pricing scenario using Dataflow Gen2 to load 2 GB of Parquet data to a Lakehouse table

Article • 11/15/2023

In this scenario, Dataflow Gen2 was used to load 2 GB of Parquet data stored in Azure Data Lake Storage (ADLS) Gen2 to a Lakehouse table in Microsoft Fabric. We used the NYC Taxi-green sample data for the Parquet data.

The prices used in the following example are hypothetical and don't intend to imply exact actual pricing. These are just to demonstrate how you can estimate, plan, and manage cost for Data Factory projects in Microsoft Fabric. Also, since Fabric capacities are priced uniquely across regions, we use the pay-as-you-go pricing for a Fabric capacity at US West 2 (a typical Azure region), at \$0.18 per CU per hour. Refer here to Microsoft Fabric - Pricing [↗](#) to explore other Fabric capacity pricing options.

Configuration

To accomplish this scenario, you need to create a dataflow with the following steps:

1. Initialize Dataflow: Get 2 GB Parquet files data from ADLS Gen2 storage account.
2. Configure Power Query:
 - a. Navigate to Power Query.
 - b. Ensure the option for staging the query is enabled.
 - c. Proceed to combine the Parquet files.
3. Data Transformation:
 - a. Promote headers for clarity.
 - b. Remove unnecessary columns.
 - c. Adjust column data types as needed.
4. Define Output Data Destination:
 - a. Configure Lakehouse as the data output destination.
 - b. In this example, a Lakehouse within Fabric was created and utilized.

Cost estimation using the Fabric Metrics App

	CU	Duration	Users	Billing type
Get Public Dataset - Cloud Storage	10.397	00:00:00	1	Billable
Get Public Dataset - Cloud Storage	10	00:00:00	1	Billable
Get Public Dataset - Cloud Storage - 10%	10	00:00:00	1	Billable
Get Public Dataset - Cloud Storage - 90%	10	00:00:00	1	Billable
Get Public Dataset - Cloud Storage - 100%	10	00:00:00	1	Billable
Get Public Dataset - Cloud Storage - 1000%	10	00:00:00	1	Billable
Total	10.397	00:00:00	1	Billable

Operation name	CU (s)	Duration (s)	Users	Billing type
Datallow Gen2 Refresh	112,098.54	7,015.48	1	Billable
Total	112,098.54	7,015.48	1	

Operation name	CU (s)	Duration (s)	Users	Billing type
SQL Endpoint Query	9.66	2.54	1	Billable
Total	9.66	2.54	1	

Operation name	CU (s)	Duration (s)	Users	Billing type
Warehouse Query	8.60	2.20	1	Billable
Cool Lake Compute	0.00	0.00	1	Billable
Total	8.60	2.21	2	

Operation name	CU (s)	Duration (s)	Users	Billing type
Query	1.52	12.08	1	Billable
Dataset On-Demand Refr.	3.50	0.64	1	Billable
Total	7.02	12.72	1	

Operation name	CU (s)	Duration (s)	Users	Billing type
Query	2.51	7.24	1	Billable
Dataset On-Demand Refr.	2.50	0.77	1	Billable
Total	5.01	8.00	1	

Operation name	CU (s)	Duration (s)	Users	Billing type
OneLake Compute	0.07	0.13	1	Billable
High Scale Dataflow Com...	0.00	0.00	1	Billable
High Scale Dataflow Com...	0.00	0.00	1	Billable
Total	0.07	0.13	1	

The High Scale Dataflow Compute Meter recorded negligible activity. Standard Compute meter for Dataflow Gen2 refresh operations consumes 112,098.540 Compute Units (CUs). It's important to consider that other operations, including Warehouse Query, SQL Endpoint Query, and Dataset On-Demand Refresh, constitute detailed aspects of Dataflow Gen2 implementation that are currently transparent and necessary for their respective operations. However, these operations will be concealed in future updates and should be disregarded when estimating costs for Dataflow Gen2.

ⓘ Note

Although reported as a metric, the actual duration of the run isn't relevant when calculating the effective CU hours with the Fabric Metrics App since the CU seconds metric it also reports already accounts for its duration.

Metric	Standard Compute	High Scale Compute
Total CU seconds	112,098.54 CU seconds	0 CU seconds
Effective CU-hours billed	$112,098.54 / (60*60) = 31.14$ CU hours	$0 / (60*60) = 0$ CU hours

Total run cost at \$0.18/CU hour = $(31.14 \text{ CU-hours}) * (\$0.18/\text{CU hour}) \approx \5.60

Next steps

- Data pipelines pricing for Data Factory in Microsoft Fabric
- Dataflow Gen2 pricing for Data Factory in Microsoft Fabric
- Pricing example scenarios

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#) | [Ask the community](#)

Quickstart: Create your first dataflow to get and transform data

Article • 11/15/2023

Dataflows are a self-service, cloud-based, data preparation technology. In this article, you create your first dataflow, get data for your dataflow, then transform the data and publish the dataflow.

Prerequisites

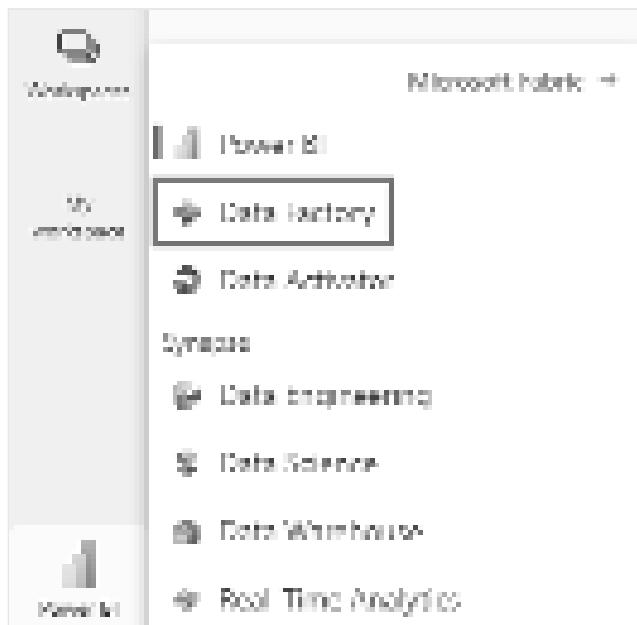
The following prerequisites are required before you start:

- A Microsoft Fabric tenant account with an active subscription. Create a free account ↗.
- Make sure you have a Microsoft Fabric enabled Workspace: Create a workspace.

Create a dataflow

In this section, you're creating your first dataflow.

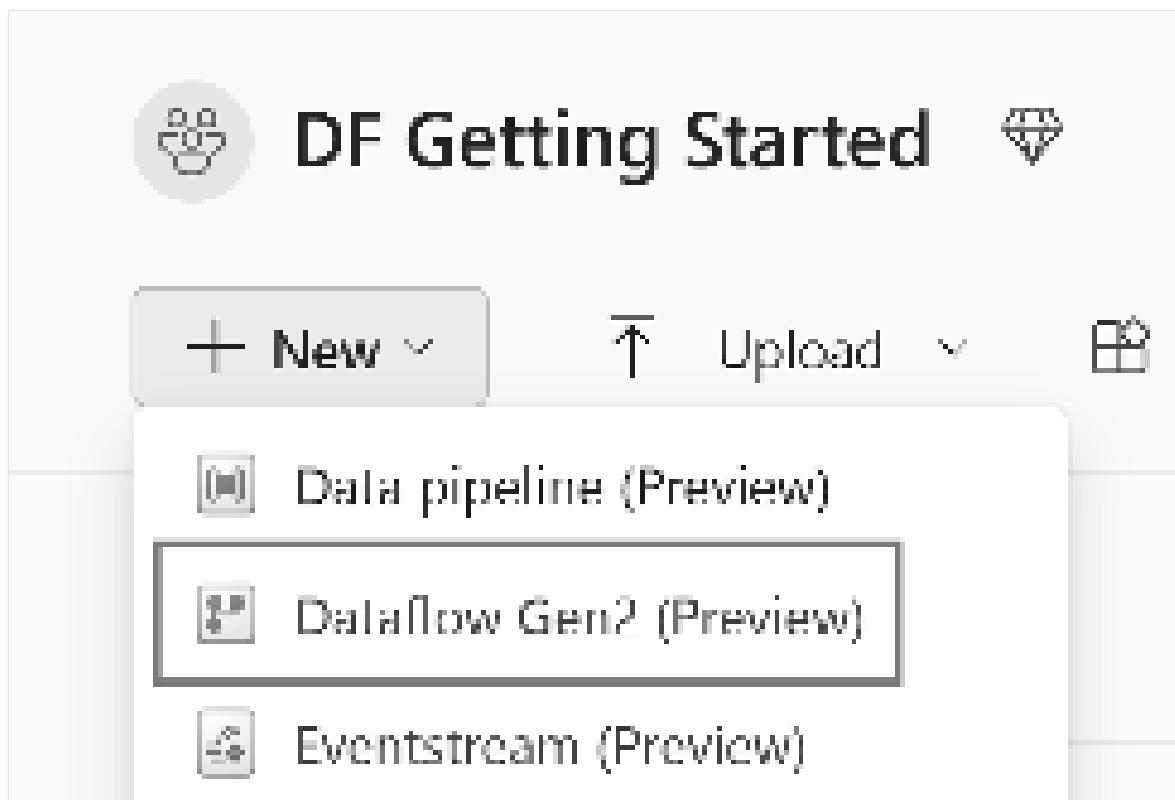
1. Switch to the **Data factory** experience.



2. Navigate to your Microsoft Fabric workspace.



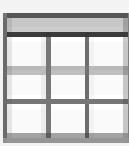
3. Select New, and then select Dataflow Gen2.



Get data

Let's get some data! In this example, you're getting data from an OData service. Use the following steps to get data in your dataflow.

1. In the dataflow editor, select Get data and then select More.

Home**Transform****Add column**Get
data ▾Enter
data

Options

Manage
parameters ▾

Parameters



Excel workbook



Dataflows



SQL Server database



Text/CSV



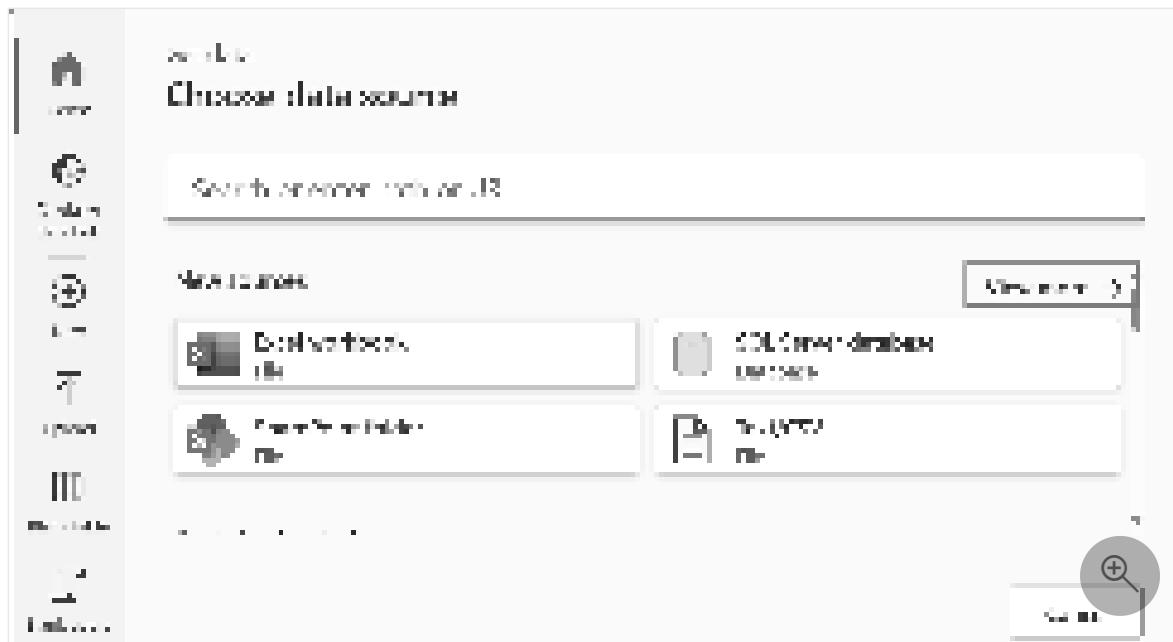
Web page



Blank query

More...

2. In Choose data source, select View more.



3. In **New source**, select **Other > OData** as the data source.



4. Enter the URL <https://services.odata.org/v4/northwind/northwind.svc/>, and then select **Next**.



5. Select the **Orders** and **Customers** tables, and then select **Create**.

You can learn more about the get data experience and functionality at [Getting data overview](#).

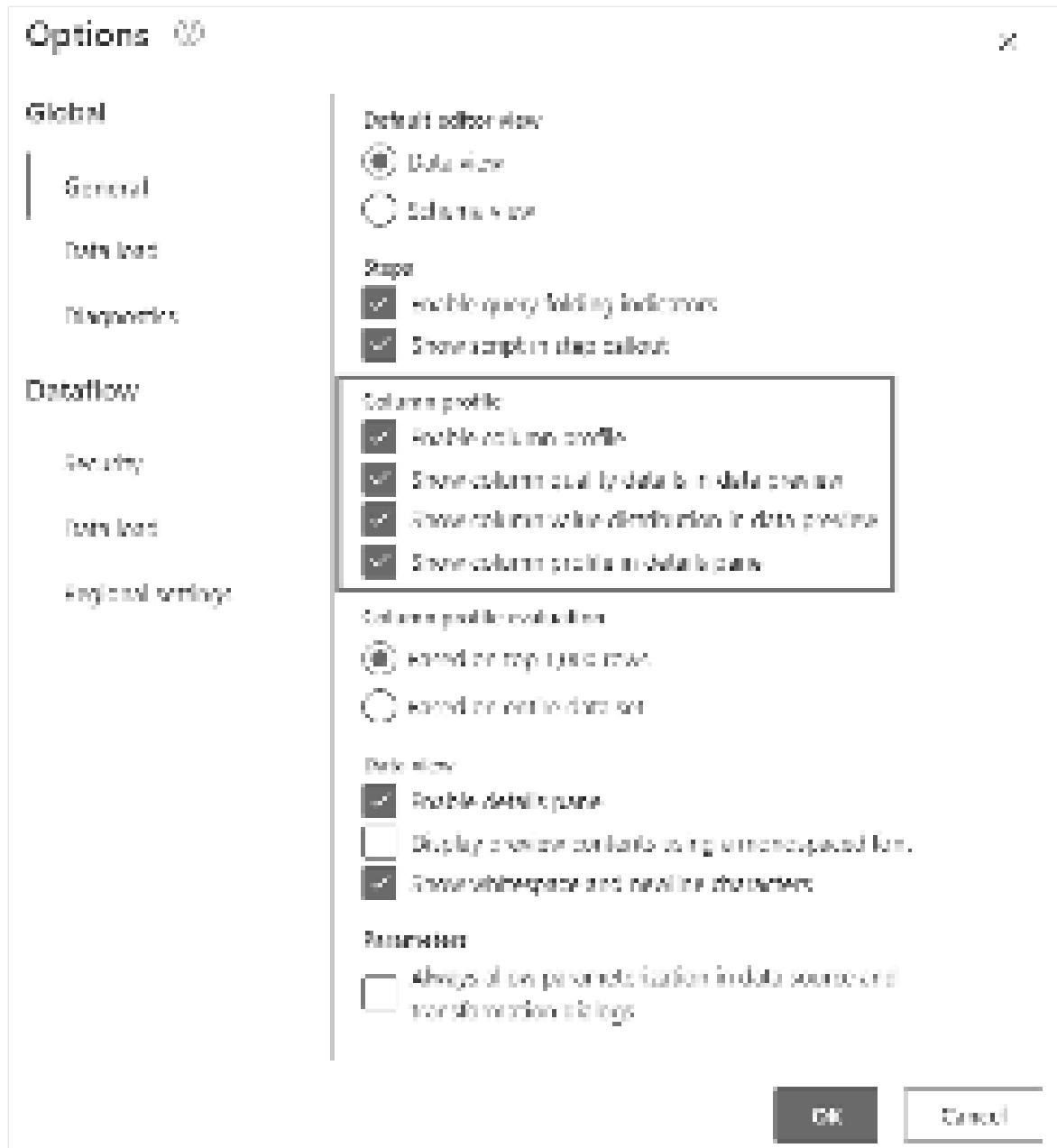
Apply transformations and publish

You have now loaded your data into your first dataflow, congratulations! Now it's time to apply a couple of transformations in order to bring this data into the desired shape.

You're going to be doing this task from the Power Query editor. You can find a detailed overview of the Power Query editor at [The Power Query user interface](#).

Follow these steps to apply transformations and publish:

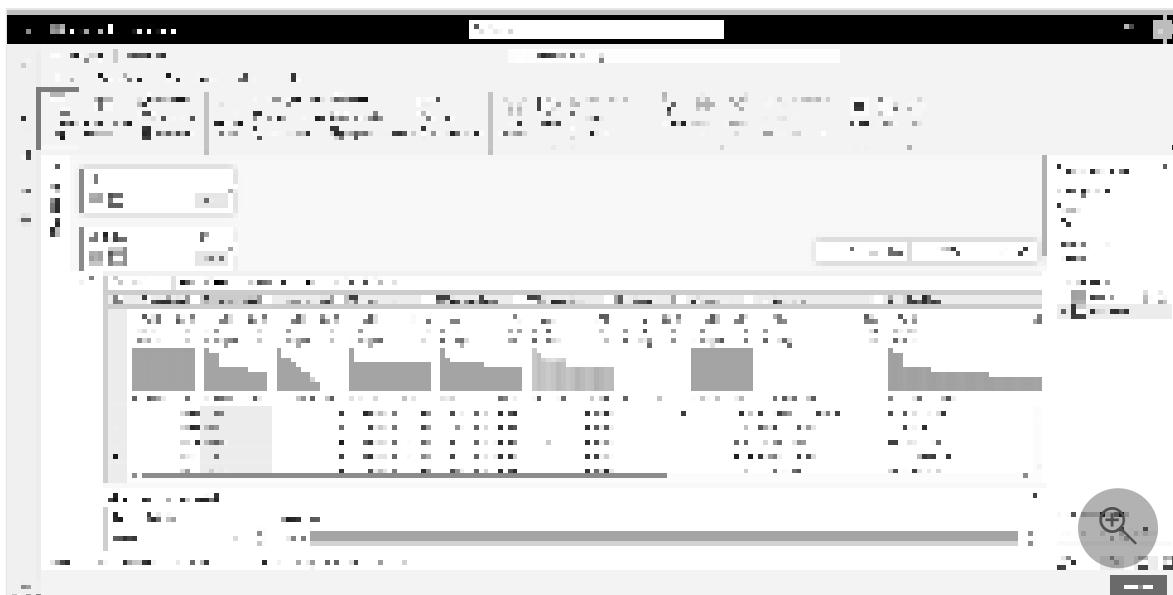
1. Ensure that the Data Profiling tools are enabled by navigating to **Home > Options > Global Options**.



Also make sure you've enabled the diagram view using the options under the **View** tab in the Power Query editor ribbon, or by selecting the diagram view icon on the lower right side of the Power Query window.



2. Within the Orders table, you calculate the total number of orders per customer. To achieve this goal, select the **CustomerID** column in the data preview and then select **Group By** under the **Transform** tab in the ribbon.



3. You perform a count of rows as the aggregation within **Group By**. You can learn more about **Group By** capabilities at [Grouping or summarizing rows](#).

Group by

Specify the column to group by and the desired output.

Basic Advanced

Group by:

CustomerID

New column name:

Count

Operation:

Count rows

Column:

Use fuzzy grouping

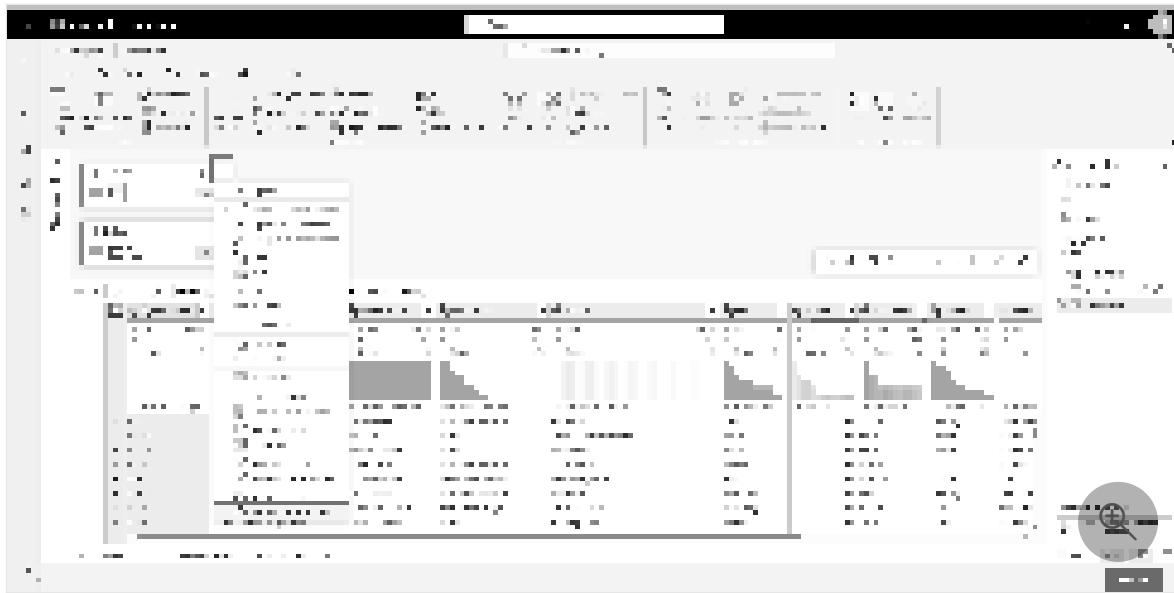
Fuzzy group options

OK

Cancel

- After grouping data in the Orders table, we'll obtain a two-column table with **CustomerID** and **Count** as the columns.

CustomerID	Count
1	10
2	10
3	10
4	10
5	10
6	10
7	10
8	10
9	10
10	10
11	10
12	10
13	10
14	10
15	10
16	10
17	10
18	10
19	10
20	10
21	10
22	10
23	10
24	10
25	10
26	10
27	10
28	10
29	10
30	10
31	10
32	10
33	10
34	10
35	10
36	10
37	10
38	10
39	10
40	10
41	10
42	10
43	10
44	10
45	10
46	10
47	10
48	10
49	10
50	10
51	10
52	10
53	10
54	10
55	10
56	10
57	10
58	10
59	10
60	10
61	10
62	10
63	10
64	10
65	10
66	10
67	10
68	10
69	10
70	10
71	10
72	10
73	10
74	10
75	10
76	10
77	10
78	10
79	10
80	10
81	10
82	10
83	10
84	10
85	10
86	10
87	10
88	10
89	10
90	10
91	10
92	10
93	10
94	10
95	10
96	10
97	10
98	10
99	10
100	10
101	10
102	10
103	10
104	10
105	10
106	10
107	10
108	10
109	10
110	10
111	10
112	10
113	10
114	10
115	10
116	10
117	10
118	10
119	10
120	10
121	10
122	10
123	10
124	10
125	10
126	10
127	10
128	10
129	10
130	10
131	10
132	10
133	10
134	10
135	10
136	10
137	10
138	10
139	10
140	10
141	10
142	10
143	10
144	10
145	10
146	10
147	10
148	10
149	10
150	10
151	10
152	10
153	10
154	10
155	10
156	10
157	10
158	10
159	10
160	10
161	10
162	10
163	10
164	10
165	10
166	10
167	10
168	10
169	10
170	10
171	10
172	10
173	10
174	10
175	10
176	10
177	10
178	10
179	10
180	10
181	10
182	10
183	10
184	10
185	10
186	10
187	10
188	10
189	10
190	10
191	10
192	10
193	10
194	10
195	10
196	10
197	10
198	10
199	10
200	10
201	10
202	10
203	10
204	10
205	10
206	10
207	10
208	10
209	10
210	10
211	10
212	10
213	10
214	10
215	10
216	10
217	10
218	10
219	10
220	10
221	10
222	10
223	10
224	10
225	10
226	10
227	10
228	10
229	10
230	10
231	10
232	10
233	10
234	10
235	10
236	10
237	10
238	10
239	10
240	10
241	10
242	10
243	10
244	10
245	10
246	10
247	10
248	10
249	10
250	10
251	10
252	10
253	10
254	10
255	10
256	10
257	10
258	10
259	10
260	10
261	10
262	10
263	10
264	10
265	10
266	10
267	10
268	10
269	10
270	10
271	10
272	10
273	10
274	10
275	10
276	10
277	10
278	10
279	10
280	10
281	10
282	10
283	10
284	10
285	10
286	10
287	10
288	10
289	10
290	10
291	10
292	10
293	10
294	10
295	10
296	10
297	10
298	10
299	10
300	10
301	10
302	10
303	10
304	10
305	10
306	10
307	10
308	10
309	10
310	10
311	10
312	10
313	10
314	10
315	10
316	10
317	10
318	10
319	10
320	10
321	10
322	10
323	10
324	10
325	10
326	10
327	10
328	10
329	10
330	10
331	10
332	10
333	10
334	10
335	10
336	10
337	10
338	10
339	10
340	10
341	10
342	10
343	10
344	10
345	10
346	10
347	10
348	10
349	10
350	10
351	10
352	10
353	10
354	10
355	10
356	10
357	10
358	10
359	10
360	10
361	10
362	10
363	10
364	10
365	10
366	10
367	10
368	10
369	10
370	10
371	10
372	10
373	10
374	10
375	10
376	10
377	10
378	10
379	10
380	10
381	10
382	10
383	10
384	10
385	10
386	10
387	10
388	10
389	10
390	10
391	10
392	10
393	10
394	10
395	10
396	10
397	10
398	10
399	10
400	10
401	10
402	10
403	10
404	10
405	10
406	10
407	10
408	10
409	10
410	10
411	10
412	10
413	10
414	10
415	10
416	10
417	10
418	10
419	10
420	10
421	10
422	10
423	10
424	10
425	10
426	10
427	10
428	10
429	10
430	10
431	10
432	10
433	10
434	10
435	10
436	10
437	10
438	10
439	10
440	10
441	10
442	10
443	10
444	10
445	10
446	10
447	10
448	10
449	10
450	10
451	10
452	10
453	10
454	10
455	10
456	10
457	10
458	10
459	10
460	10
461	10
462	10
463	10
464	10
465	10
466	10
467	10
468	10
469	10
470	10
471	10
472	10
473	10
474	10
475	10
476	10
477	10
478	10
479	10
480	10
481	10
482	10
483	10
484	10
485	10
486	10
487	10
488	10
489	10
490	10
491	10
492	10
493	10
494	10
495	10
496	10
497	10
498	10
499	10
500	10
501	10
502	10
503	10
504	10
505	10
506	10
507	10
508	10
509	10
510	10
511	10
512	10
513	10
514	10
515	10
516	10
517	10
518	10
519	10
520	10
521	10
522	10
523	10
524	10
525	10
526	10
527	10
528	10
529	10
530	10
531	10
532	10
533	10
534	10
535	10
536	10
537	10
538	10
539	10
540	10
541	10
542	10
543	10
544	10
545	10
546	10
547	10
548	10</



6. Configure the Merge operation as shown in the following screenshot by selecting **CustomerID** as the matching column in both tables. Then select **Ok**.

Merge

Select tables and matching columns to create a merged table.

Left table for merge:

Customers

 #	 CustomerName	 CompanyName	 ContactName	 Address
A0001	Alfreds Futterkiste	Manzana, 8	Janet Lampman	Carrera 23A, 10
A0002	Ana Trujillo Emparedados y Repostería	Avda. de la Constitución 222	Manuel Serrano	Avda. Reina Victoria 32
A0003	Anatolio Goya	Sierras del Sol 989	Patricia Berman	Champlain St. Apt. 1E
A0004	Antonio Moreno Taquería	El Mante 222	Ramon Flores	123 Ramon

Right table for merge:

Orders

 OrderID	 Payment
10247	5
10248	6
10249	7
10250	8
10251	9

Join Type:



Use key matching instead of merge

> [Fuzzy matching options](#)

 The selected matches will always come from the left table

OK

Cancel

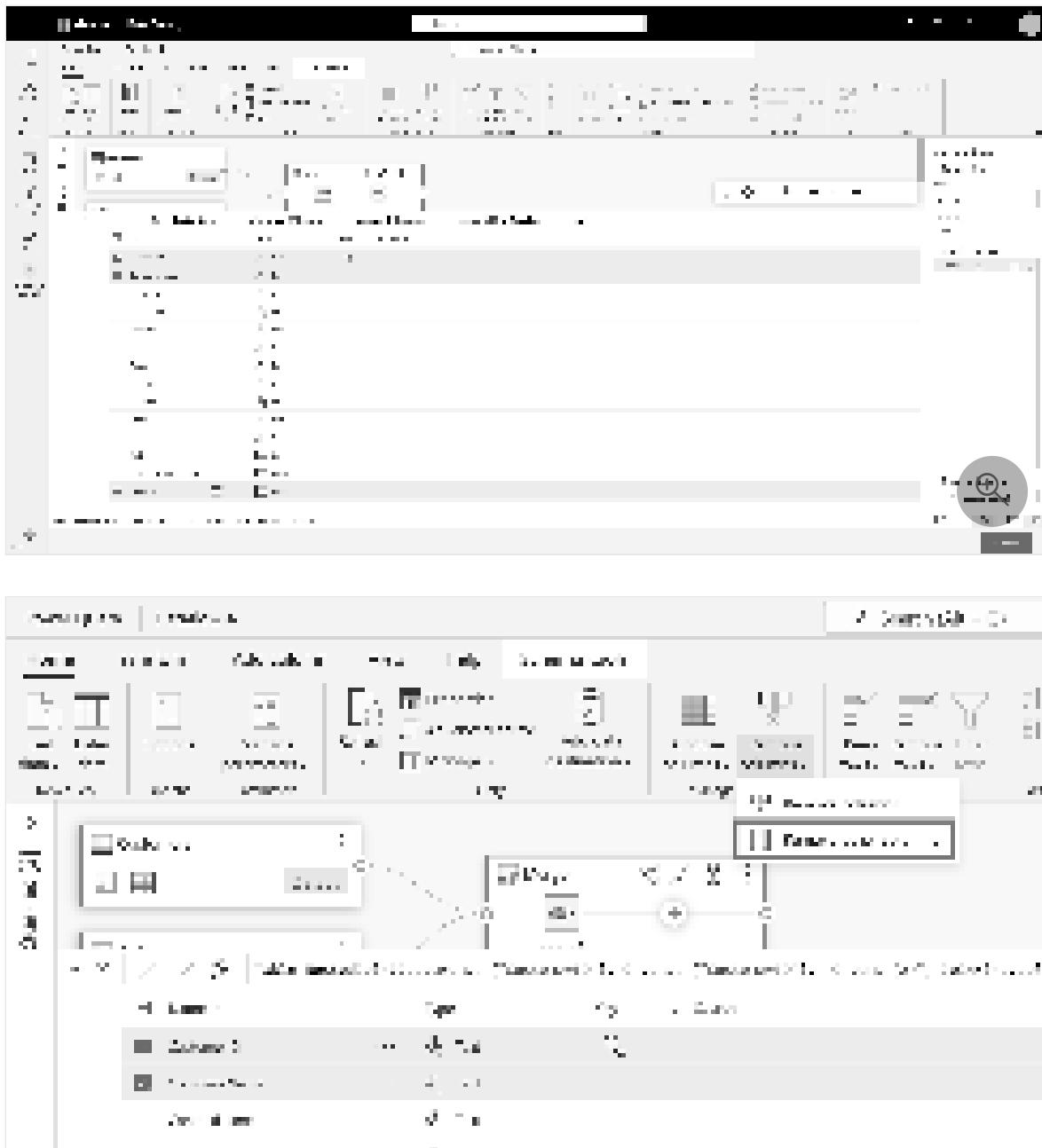
7. Upon performing the **Merge queries as new** operation, you obtain a new query with all columns from the Customers table and one column with nested data from the Orders table.



8. In this example, you're only interested in a subset of columns in the Customers table. You select those columns by using the schema view. Enable the schema view within the toggle button on the bottom-right corner of the dataflows editor.



9. The schema view provides a focused view into a table's schema information, including column names and data types. Schema view has a set of schema tools available through a contextual ribbon tab. In this scenario, you select the **CustomerID**, **CompanyName**, and **Orders (2)** columns, then select the **Remove columns** button, and then select **Remove other columns** in the **Schema tools** tab.



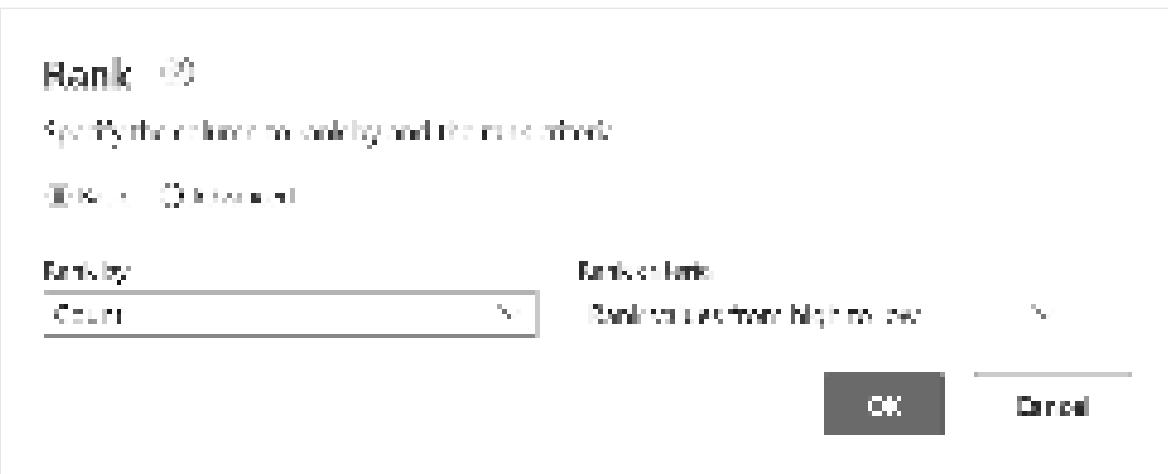
10. The **Orders (2)** column contains nested information resulting from the merge operation you performed a few steps ago. Now, switch back to the data view by selecting the **Show data view** button next to the **Show schema view** button in the bottom-right corner of the UI. Then use the **Expand Column** transformation in the **Orders (2)** column header to select the **Count** column.



11. As the final operation, you want to rank your customers based on their number of orders. Select the **Count** column and then select the **Rank column** button under the **Add Column** tab in the ribbon.



12. Keep the default settings in **Rank Column**. Then select **OK** to apply this transformation.



13. Now rename the resulting query as **Ranked Customers** using the **Query settings** pane on the right side of the screen.

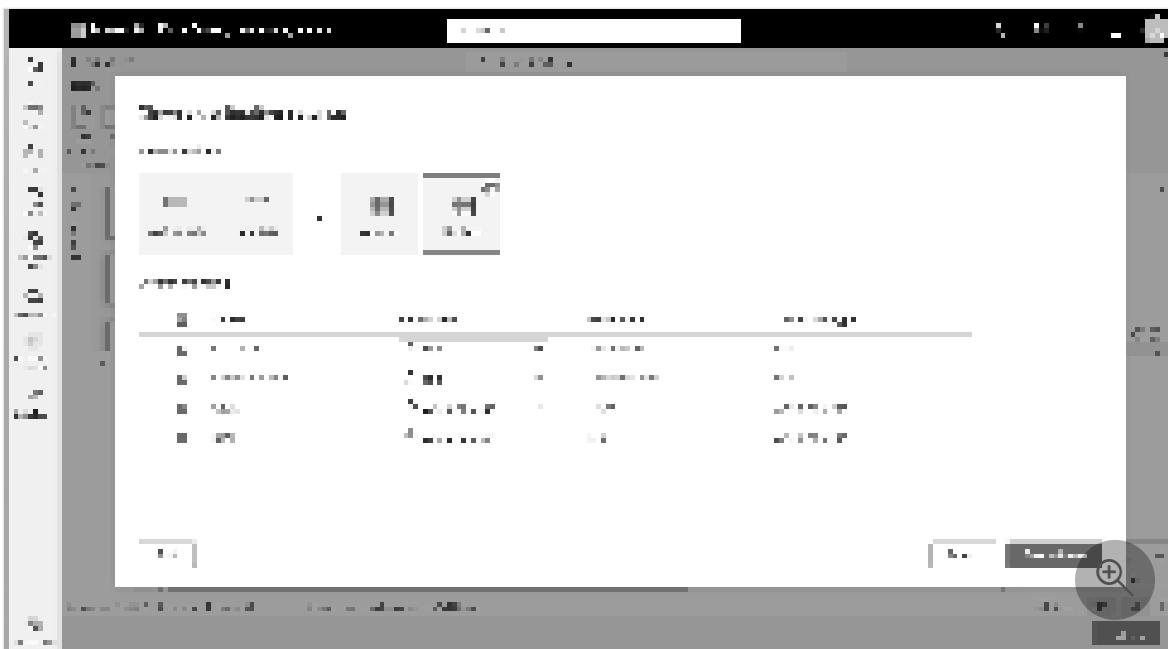
A screenshot of the Microsoft Power BI desktop application. The main area shows a data grid with several columns of data. On the left, there's a 'Data' pane with a tree view of the data source. On the right, there's a 'Query settings' pane with tabs for 'General', 'Transform', and 'Advanced'. The 'General' tab is selected, showing options like 'Name' (set to 'Ranked'), 'Format', and 'Type'. A magnifying glass icon is visible at the bottom right of the screen.

14. You've finished transforming and combining your data. So, you now configure its output destination settings. Select **Choose data destination** at the bottom of the **Query settings** pane.



15. For this step, you can configure an output to your lakehouse if you have one available, or skip this step if you don't. Within this experience, you're able to configure the destination lakehouse and table for your query results, in addition to the update method (Append or Replace).





16. Your dataflow is now ready to be published. Review the queries in the diagram view, and then select **Publish**.



You're now returned to the workspace. A spinner icon next to your dataflow name indicates publishing is in progress. Once the publishing completes, your dataflow is ready to refresh!

ⓘ Important

When the first Dataflow Gen2 is created in a workspace, Lakehouse and Warehouse items are provisioned along with their related SQL analytics endpoint and semantic models. These items are shared by all dataflows in the workspace and are required for Dataflow Gen2 to operate, shouldn't be deleted, and aren't intended to be used directly by users. The items are an

implementation detail of Dataflow Gen2. The items aren't visible in the workspace, but might be accessible in other experiences such as the Notebook, SQL analytics endpoint, Lakehouse, and Warehouse experiences. You can recognize the items by their prefix in the name. The prefix of the items is 'DataflowsStaging'.

17. In your workspace, select the **Schedule Refresh** icon.

Name	Type
My First Dataflow	Dataflow

18. Turn on the scheduled refresh, select **Add another time**, and configure the refresh as shown in the following screenshot.

Schedule

Don't give a refresh schedule

Create a data refresh schedule to automatically refresh the data available in the workspace. Learn more

Refresh frequency

Every

Time zone

1:00 AM (Central - Interval Time)

Time

4:00 AM

Add another time

Send refresh failure notifications to:

Everyone (team)

Everyone (organization)

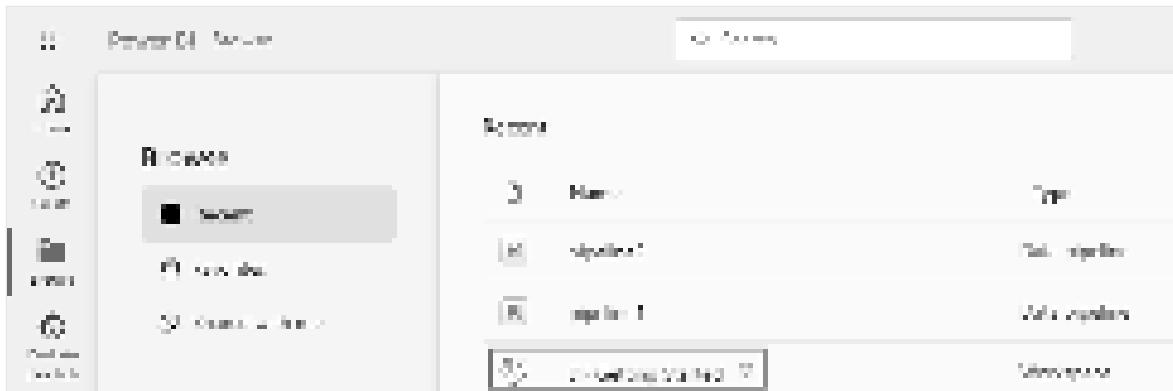
Everyone (global)

Apply **Cancel**

Clean up resources

If you're not going to continue to use this dataflow, delete the dataflow using the following steps:

1. Navigate to your Microsoft Fabric workspace.



2. Select the vertical ellipsis next to the name of your dataflow and then select **Delete**.



3. Select **Delete** to confirm the deletion of your dataflow.

Delete dataflow

X

Are you sure you want to permanently delete Northwind Sales Dataflow? Content from this dataflow that's included in any datasets, reports, dashboards, or tiles will be deleted.

Delete

Cancel

Next steps

The dataflow in this sample shows you how to load and transform data in Dataflow Gen2. You learned how to:

- ✓ Create a Dataflow Gen2.
- ✓ Transform data.
- ✓ Configure destination settings for transformed data.
- ✓ Run and schedule your data pipeline.

Advance to the next article to learn how to create your first data pipeline.

[Quickstart: Create your first data pipeline to copy data](#)

Feedback

Was this page helpful?

Yes

No

Provide product feedback  | Ask the community 

Quickstart: Create your first pipeline to copy data

Article • 11/15/2023

In this quickstart, you build a data pipeline to move a Sample dataset to the Lakehouse. This experience shows you a quick demo about how to use pipeline copy activity and how to load data into Lakehouse.

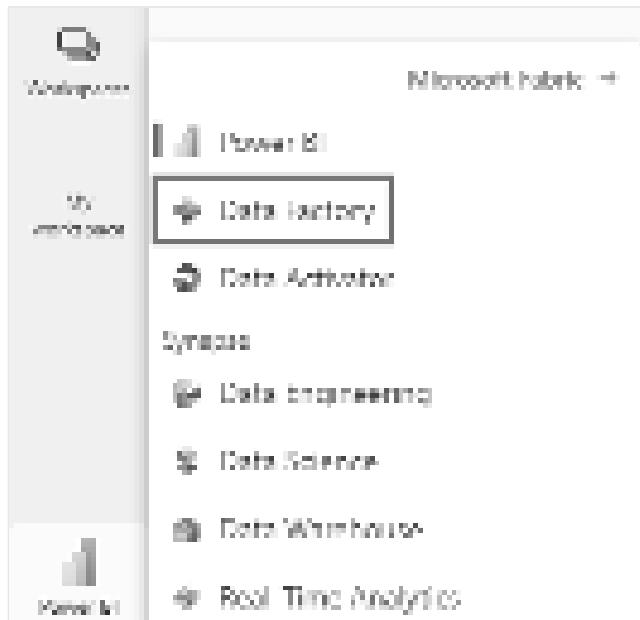
Prerequisites

To get started, you must complete the following prerequisites:

- A Microsoft Fabric tenant account with an active subscription. Create a free account ↗.
- Make sure you have a Microsoft Fabric enabled Workspace: Create a workspace.

Create a data pipeline

1. Navigate to Power BI ↗.
2. Select the Power BI icon in the bottom left of the screen, then select **Data factory** to open homepage of Data Factory.



3. Navigate to your Microsoft Fabric workspace. If you created a new workspace in the prior Prerequisites section, use this one.



4. Select **Data pipeline** and then input a pipeline name to create a new pipeline.

Data Factory - DF Getting Started

DF Getting Started

+ New

Data pipeline

Dashboard

Event hubs

Favorites

Log Analytics

Machine learning

Notebooks

Replays (Preview)

Report

Run Job Definition

Variables

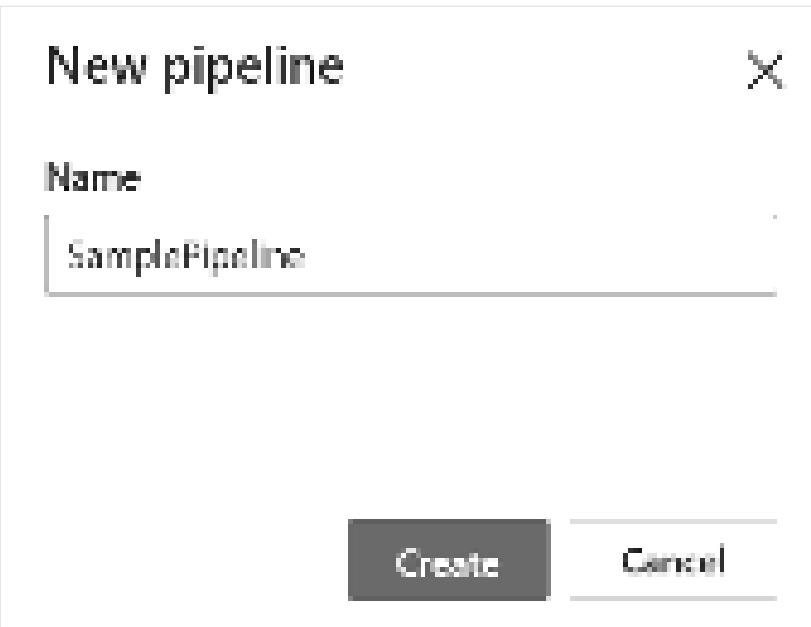
More options

Import from

Import notebook

The screenshot shows the Microsoft Azure Data Factory 'DF Getting Started' interface. On the left is a vertical sidebar with icons for Home, + New, Dashboard, Event hubs, Favorites, Log Analytics, Machine learning, Notebooks, Replays (Preview), Report, Run Job Definition, Variables, More options, Import from, and Import notebook. The main panel has a title 'DF Getting Started' with a sunburst icon. Below it is a search bar with placeholder text 'Type and search...'. A list of items follows, with 'Data pipeline' highlighted by a red box:

- Data pipeline
- Dashboard
- Event hubs
- Favorites
- Log Analytics
- Machine learning
- Notebooks
- Replays (Preview)
- Report
- Run Job Definition
- Variables

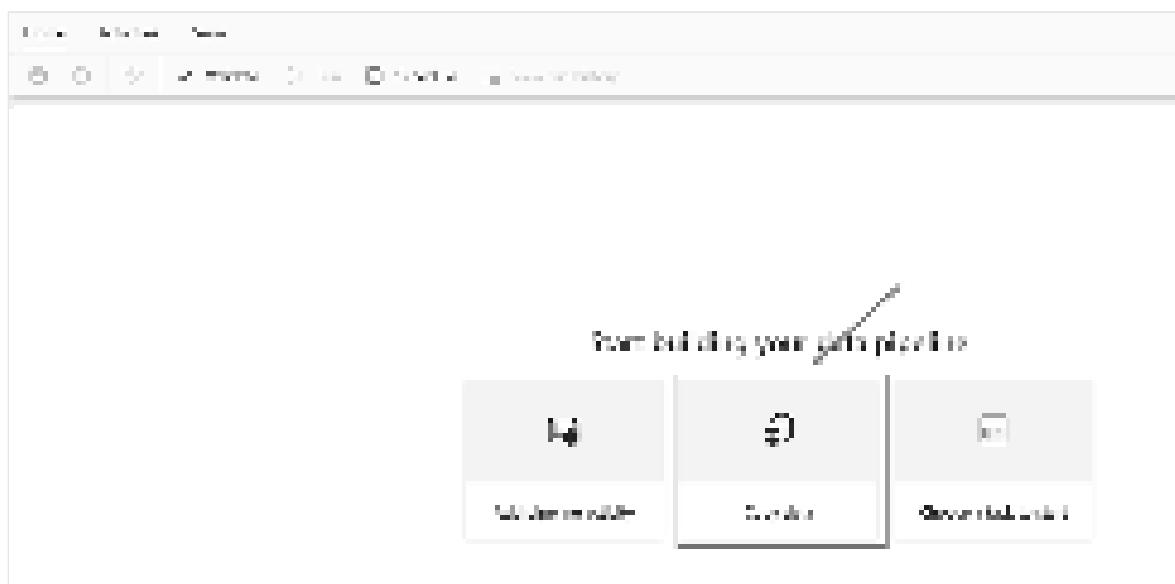


Copy data using pipeline

In this session, you start to build your first pipeline by following below steps about copying from a sample dataset provided by pipeline into Lakehouse.

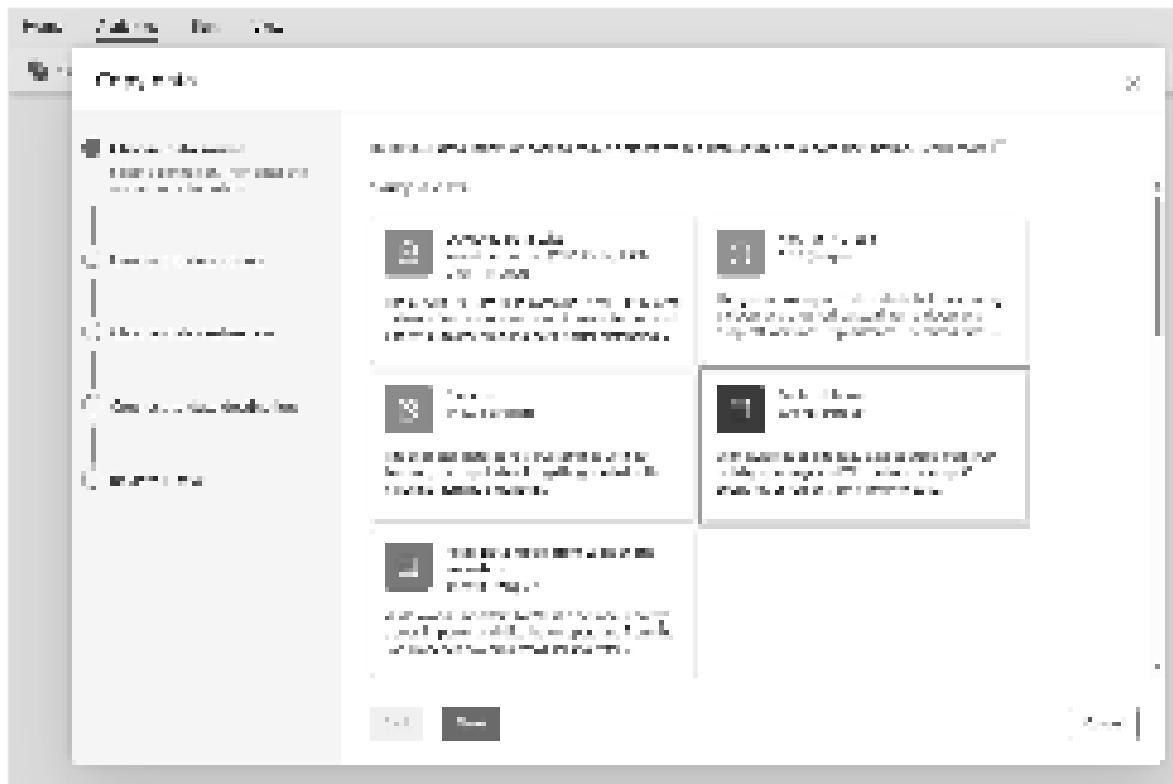
Step 1: Start with the Copy assistant

1. After selecting **Copy data** on the canvas, the **Copy assistant** tool will be opened to get started.

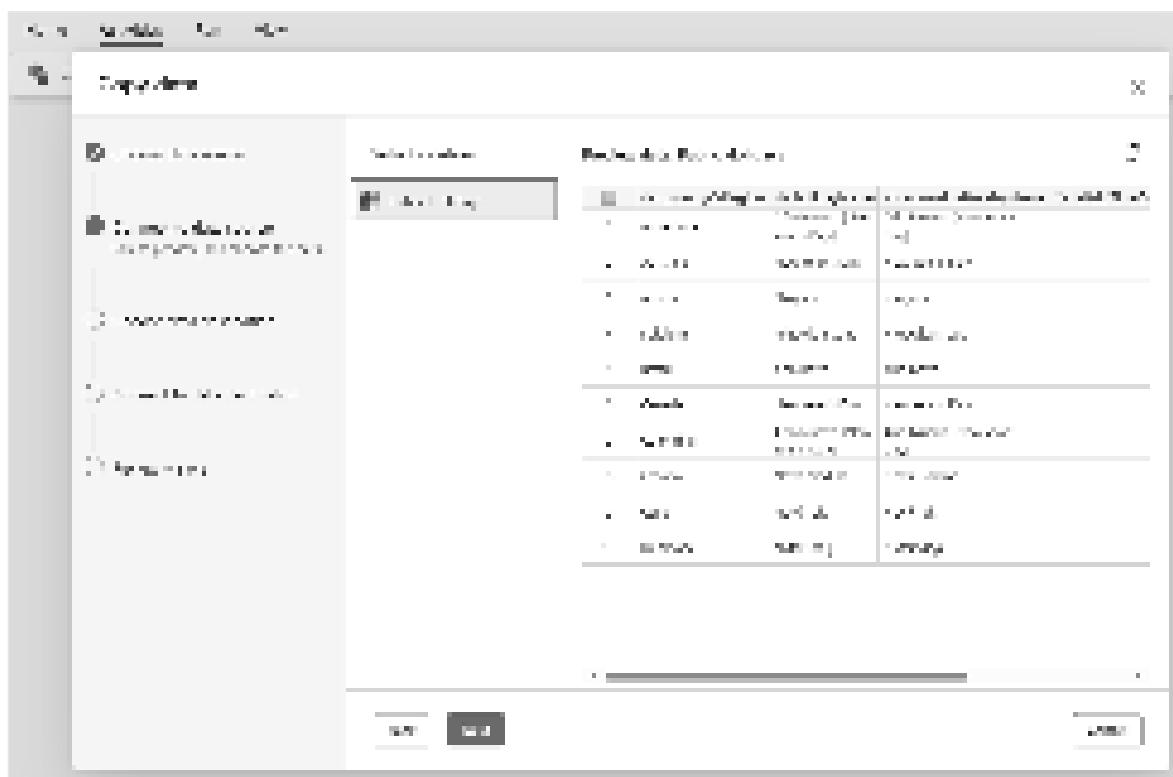


Step 2: Configure your source

1. Choose the **Public Holidays** sample data, and then select **Next**.

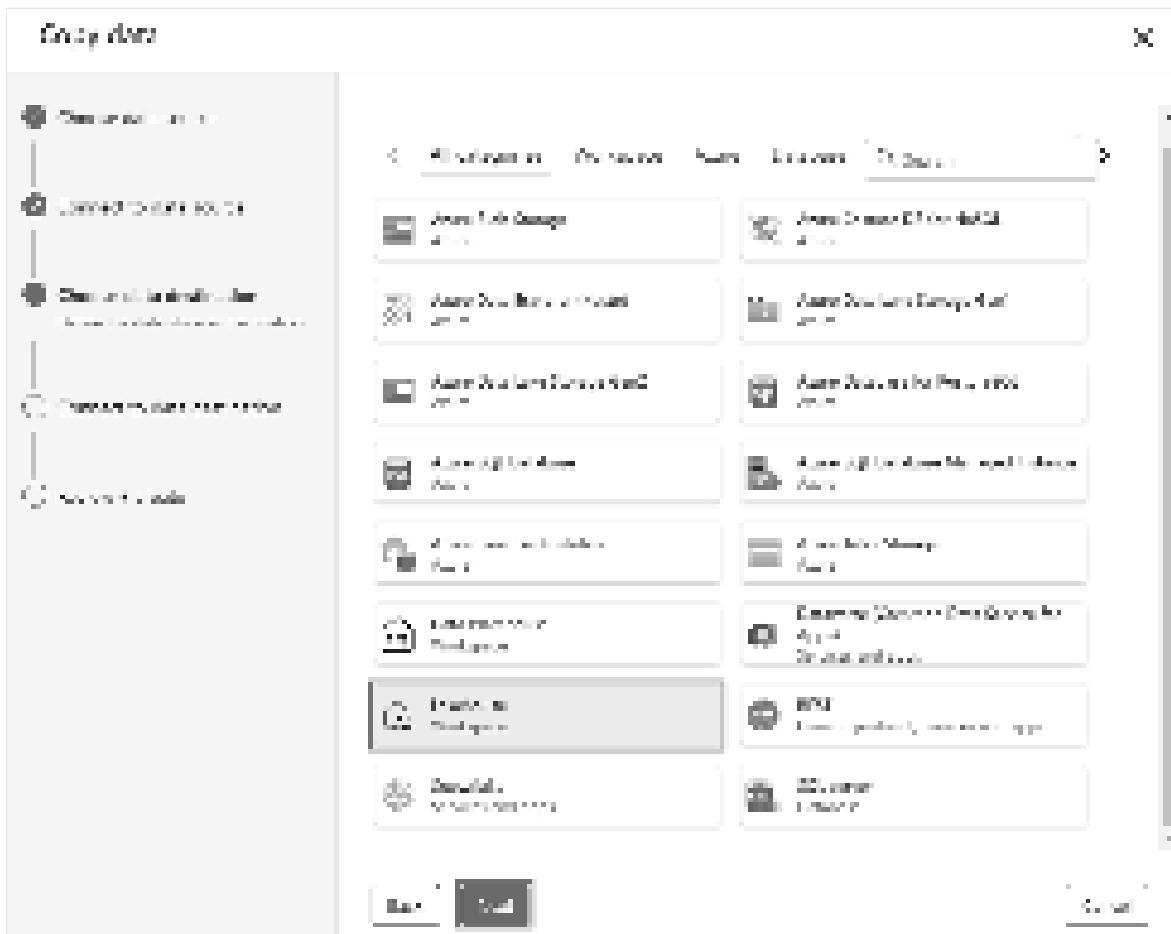


2. On the **Connect to data source** page of the assistant, the preview for the **Public Holidays** sample data is displayed, and then click **Next**.



Step 3: Configure your destination

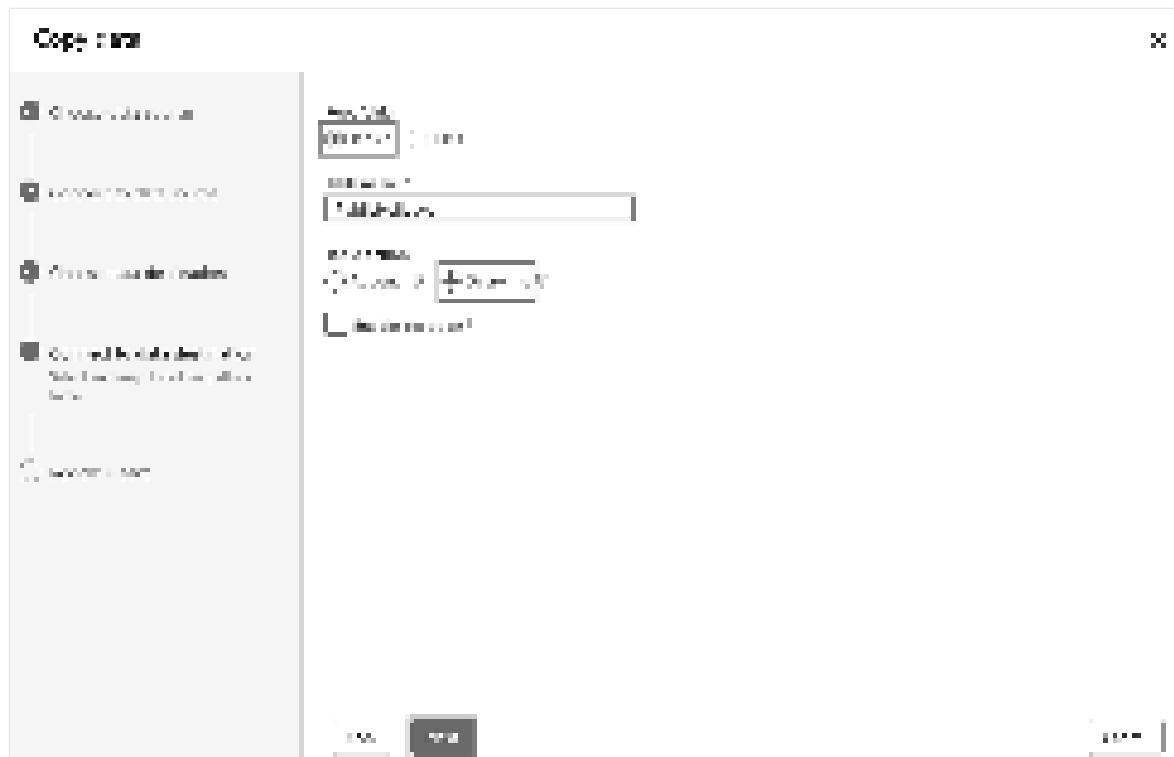
1. Select **Lakehouse** and then **Next**.



2. Select **Create new Lakehouse**, and enter a **Lakehouse name**, then select **Next**.

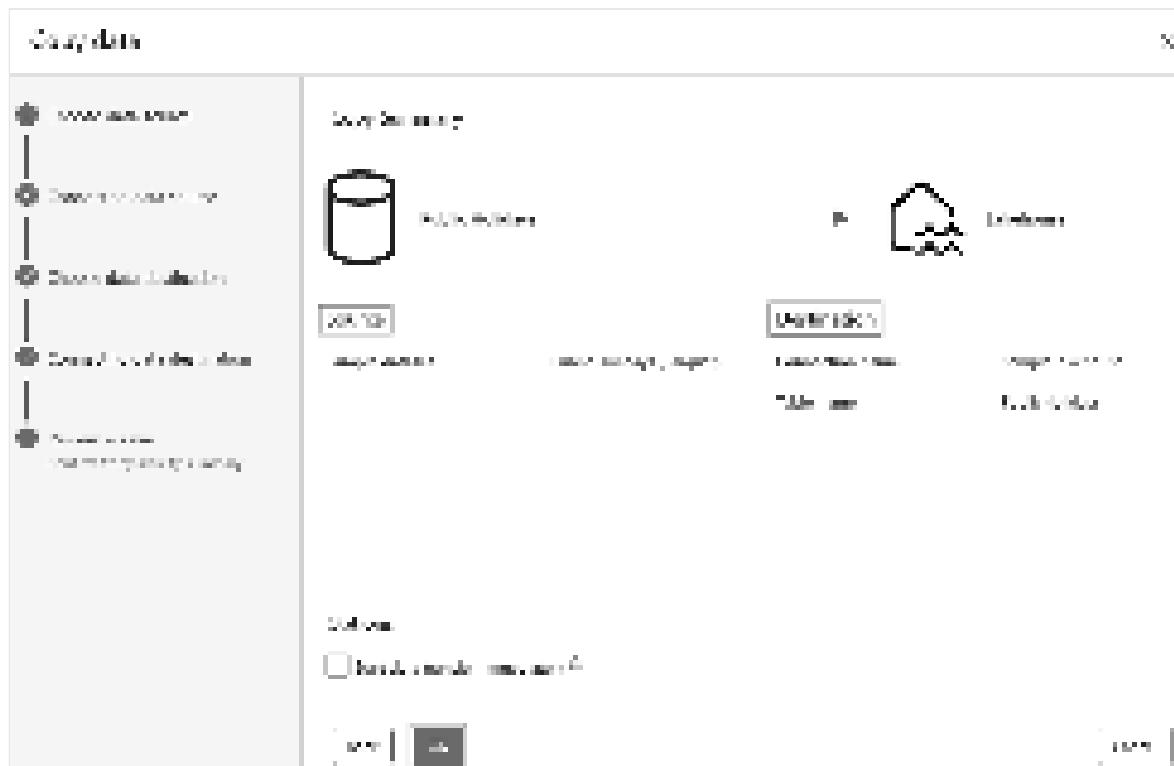


3. Configure and map your source data to the destination Lakehouse table. Select **tables** and provide a **Table name** under **Root folder**, then choose the **Overwrite** option for **Table action**, and select **Next**.



Step 4: Review and create your copy activity

1. Review your copy activity settings in the previous steps and select OK to finish. Or you can revisit the previous steps in the tool to edit your settings, if needed.



2. The Copy activity is added to your new data pipeline canvas. All settings including advanced settings for the activity are available in the tabs below the pipeline canvas when the created **Copy data** activity is selected.

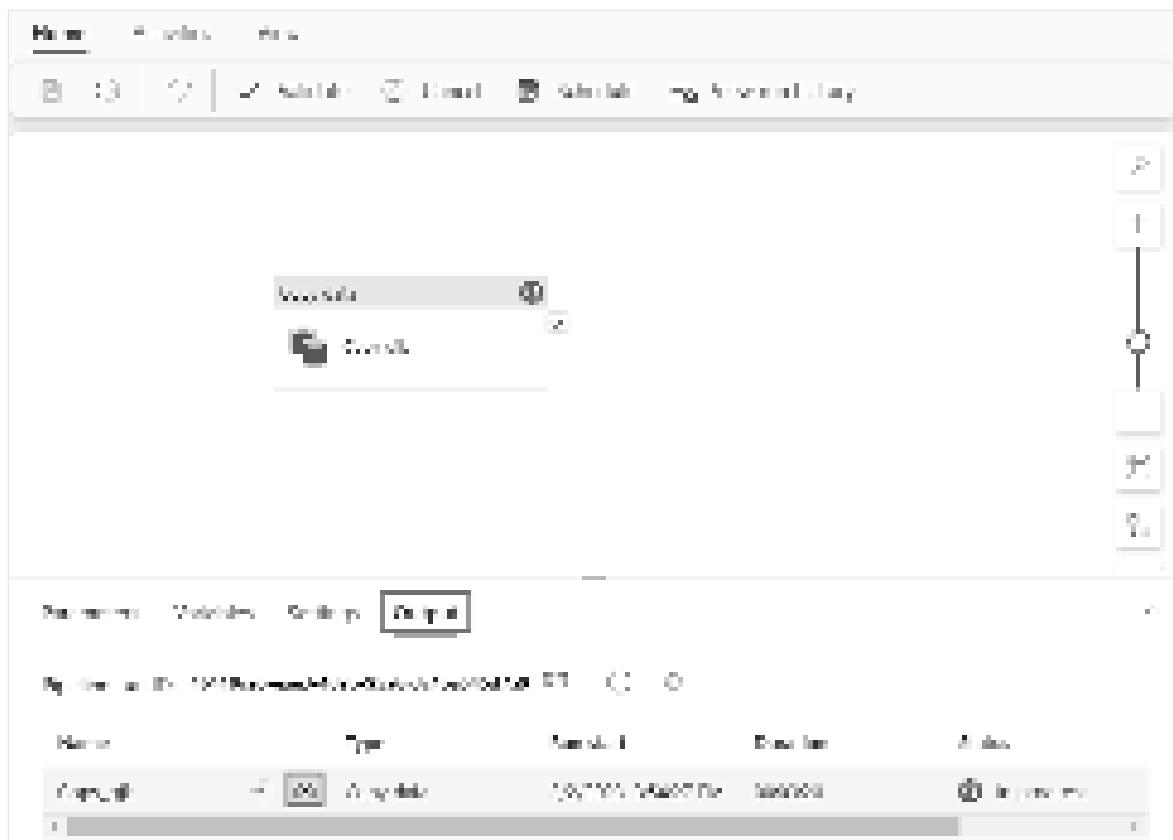
The screenshot shows the Azure Data Factory interface. At the top, there's a navigation bar with tabs: Home (selected), Activities, Run, View. Below the navigation bar are several icons: Refresh, Save, Delete, Run, Validate, Schedule, View run history, and Copy data. A modal dialog titled "Copy data" is open, showing a tree view with a single node labeled "Copy job". The main panel below has tabs: General, Source, Destination, Mapping, Settings. Under General, the Name is set to "Copy job" and the Description is empty. The Timeout is set to "10 minutes" and the Retry count is "0". There's also a "Advanced" section which is collapsed.

Run and schedule your data pipeline

1. Switch to the **Home** tab and select **Run**. A confirmation dialog is displayed. Then select **Save and run** to start the activity.

The screenshot shows the Azure Data Factory interface with a confirmation dialog box overlaid. The dialog title is "Save and run?" and it contains the message: "You have unsaved changes. Pressing cancel will cancel the pipeline without saving it first." At the bottom of the dialog are two buttons: "Cancel" and "Save and run".

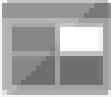
2. You can monitor the running process and check the results on the **Output** tab below the pipeline canvas. Select the run details button (with the glasses icon highlighted) to view the run details.



3. The run details show how much data was read and written and various other details about the run.

Copy data details

Copy_2021

Source	Destination
 Azure Blob Storage	 Database
Data read: 0	Data written: 0
File read: 1	File written: 1
Row read: 65,507	Rows written: 60,557

Status:  Succeeded

Start time: 14/02/2021, 2:17:40 PM

Pipeline run activity ID: 0600711ac-65b4-42a0-9d04-47d59a17

Throughput: 62,755 KB/s

Total duration: 00:00:11

[Duration breakdown](#)

[Advanced](#)

[Close](#)

4. You can also schedule the pipeline to run with a specific frequency as required.

Below is an example scheduling the pipeline to run every 15 minutes.



Next steps

The pipeline in this sample shows you how to copy sample data to Lakehouse. You learned how to:

- ✓ Create a data pipeline.
- ✓ Copy data with the Copy Assistant.
- ✓ Run and schedule your data pipeline.

Next, advance to learn more about monitoring your pipeline runs.

[How to monitor pipeline runs in Microsoft Fabric](#)

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback ↗](#) | [Ask the community ↗](#)

Quickstart: Move and transform data with dataflows and data pipelines

Article • 12/13/2023

In this tutorial, you discover how the dataflow and data pipeline experience can create a powerful and comprehensive Data Factory solution.

Prerequisites

To get started, you must have the following prerequisites:

- A tenant account with an active subscription. Create a free account [↗](#).
- Make sure you have a Microsoft Fabric enabled Workspace: Create a workspace that isn't the default My Workspace.
- An Azure SQL database with table data.
- A Blob Storage account.

Dataflows compared to pipelines

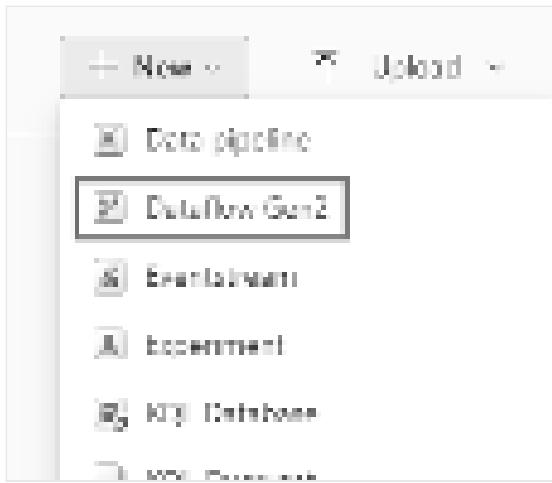
Dataflows Gen2 enable you to leverage a low-code interface and 300+ data and AI-based transformations to easily clean, prep, and transform data with more flexibility than any other tool. Data Pipelines enable rich out-of-the-box data orchestration capabilities to compose flexible data workflows that meet your enterprise needs. In a pipeline, you can create logical groupings of activities that perform a task, which might include calling a Dataflow to clean and prep your data. While there is some functionality overlap between the two, the choice of which to use for a specific scenario depends on whether you require the full richness of pipelines or can use the simpler but more limited capabilities of dataflows. For more details, refer to the Fabric decision guide

Transform data with dataflows

Follow these steps to set up your dataflow.

Step 1: Create a dataflow

1. Choose your Fabric enabled workspace, and then select **New**. Then select **Dataflow Gen2**.

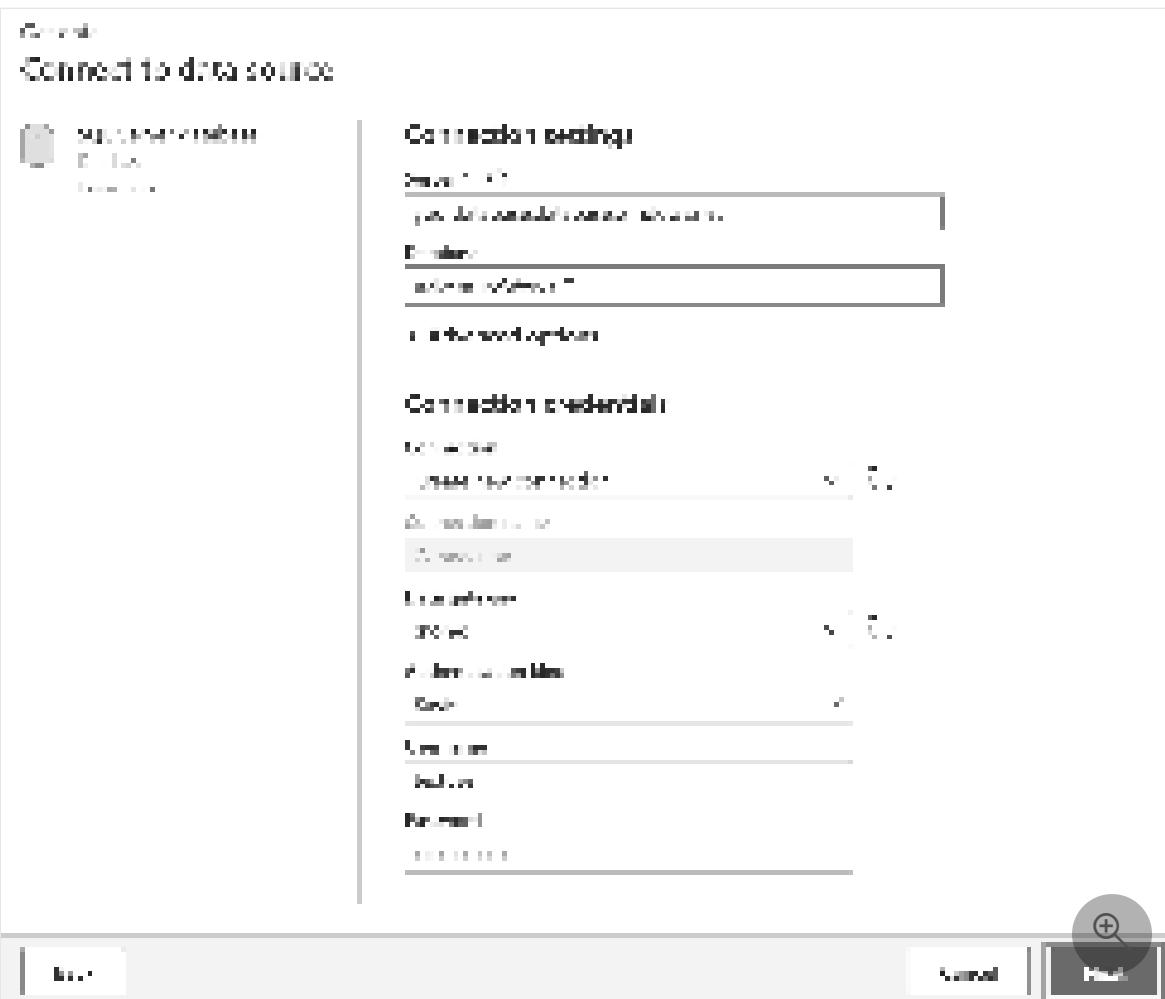


2. The dataflow editor window appears. Select the **Import from SQL Server** card.



Step 2: Get data

1. On the **Connect to data source** dialog presented next, enter the details to connect to your Azure SQL database, then select **Next**. For this example, you use the **AdventureWorksLT** sample database configured when you set up the Azure SQL database in the prerequisites.

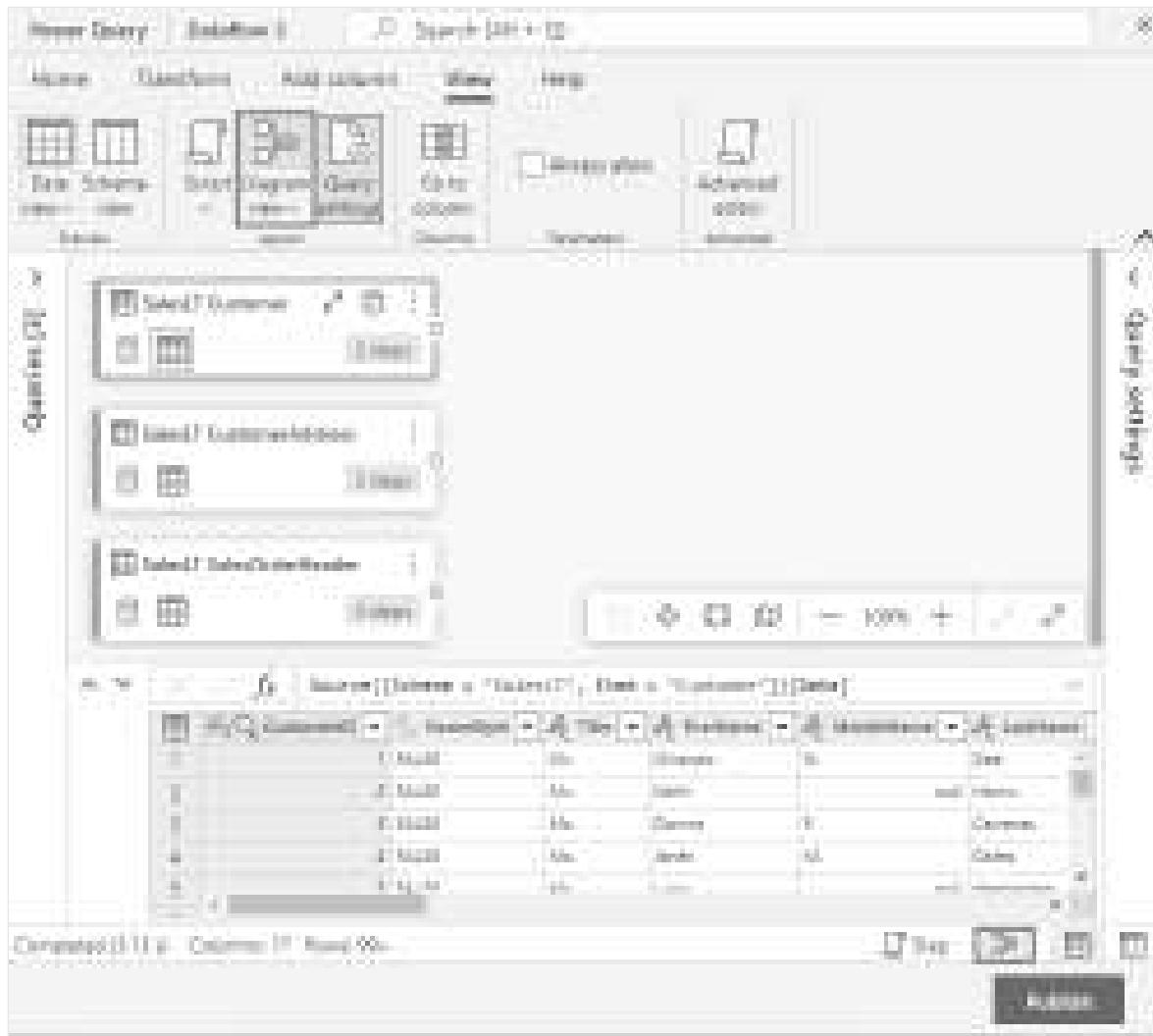


2. Select the data you'd like to transform and then select **Create**. For this quickstart, select **SalesLT.Customer** from the **AdventureWorksLT** sample data provided for Azure SQL DB, and then the **Select related tables** button to automatically include two other related tables.

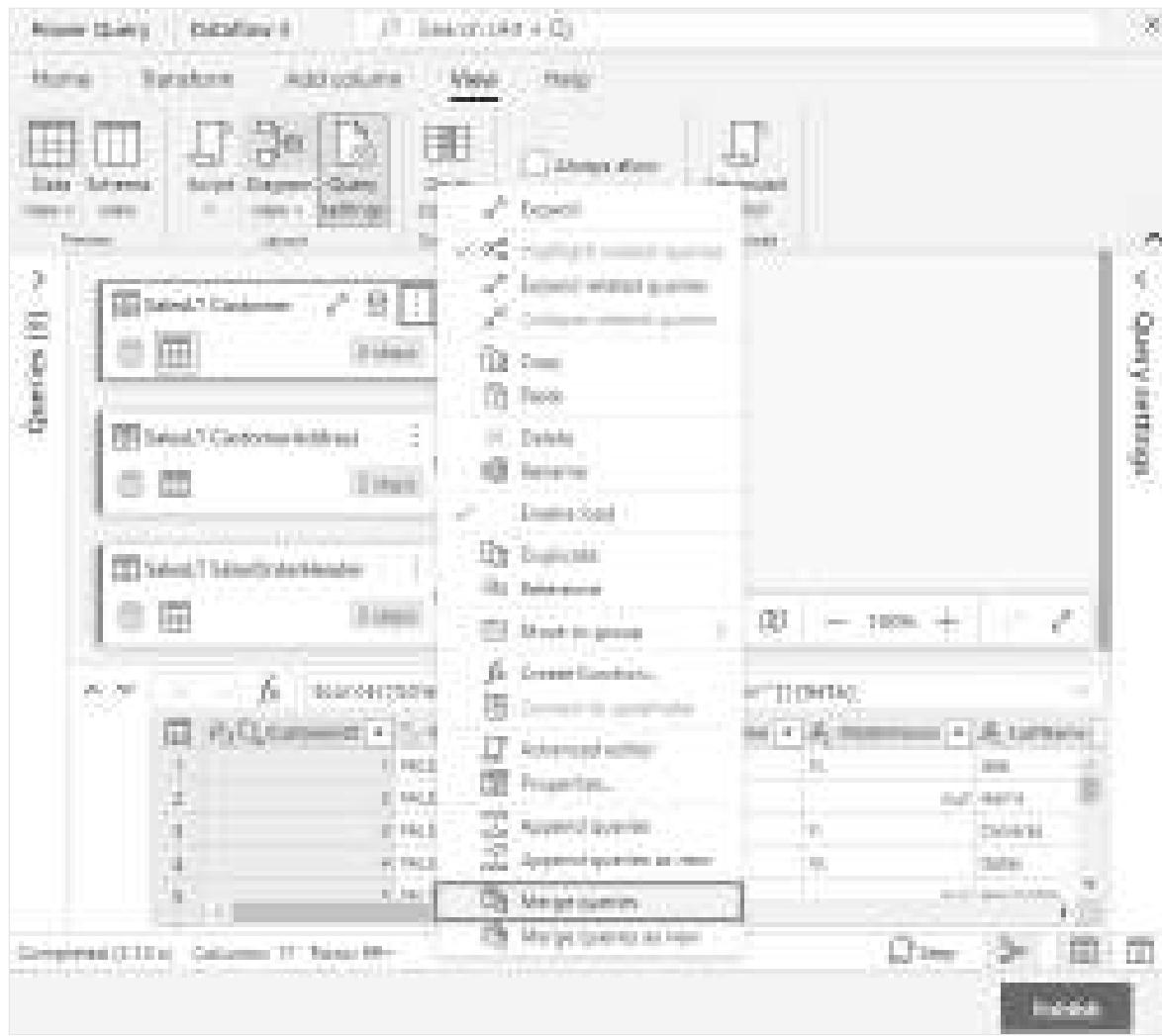
The screenshot shows the 'Select data' dialog box. On the left, a tree view lists tables and views under 'Adventure WorksLT'. The 'SalesLT.Customer' table is selected and highlighted. To its right is a preview grid showing data from the 'Customer' table. The grid has columns: 'CustomerID', 'CustomerType', 'Name', 'Address', 'City', 'StateProvince', 'PostalCode', 'CountryRegionCode', 'Phone', 'Fax', 'Email', 'SupportRepID', 'SalesLT.CustomerID', 'SalesLT.CustomerType', 'SalesLT.Name', 'SalesLT.Address', 'SalesLT.City', 'SalesLT.StateProvince', 'SalesLT.PostalCode', 'SalesLT.CountryRegionCode', 'SalesLT.Phone', 'SalesLT.Fax', 'SalesLT.Email', 'SalesLT.SupportRepID'. At the bottom of the dialog are 'Transform data' and 'Create' buttons.

Step 3: Transform your data

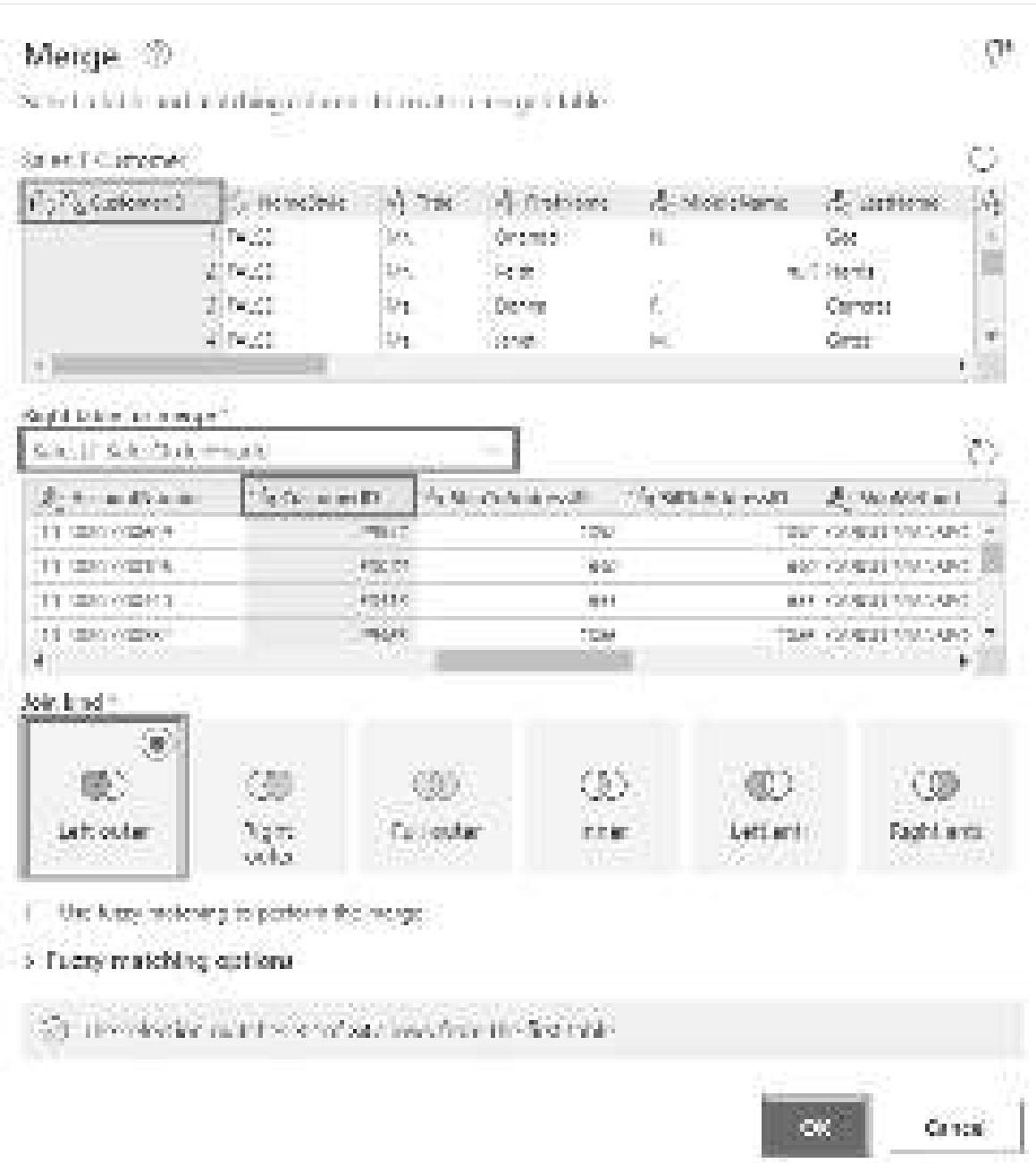
1. If it isn't selected, select the **Diagram view** button along the status bar at the bottom of the page, or select **Diagram view** under the **View** menu at the top of the Power Query editor. Either of these options can toggle the diagram view.



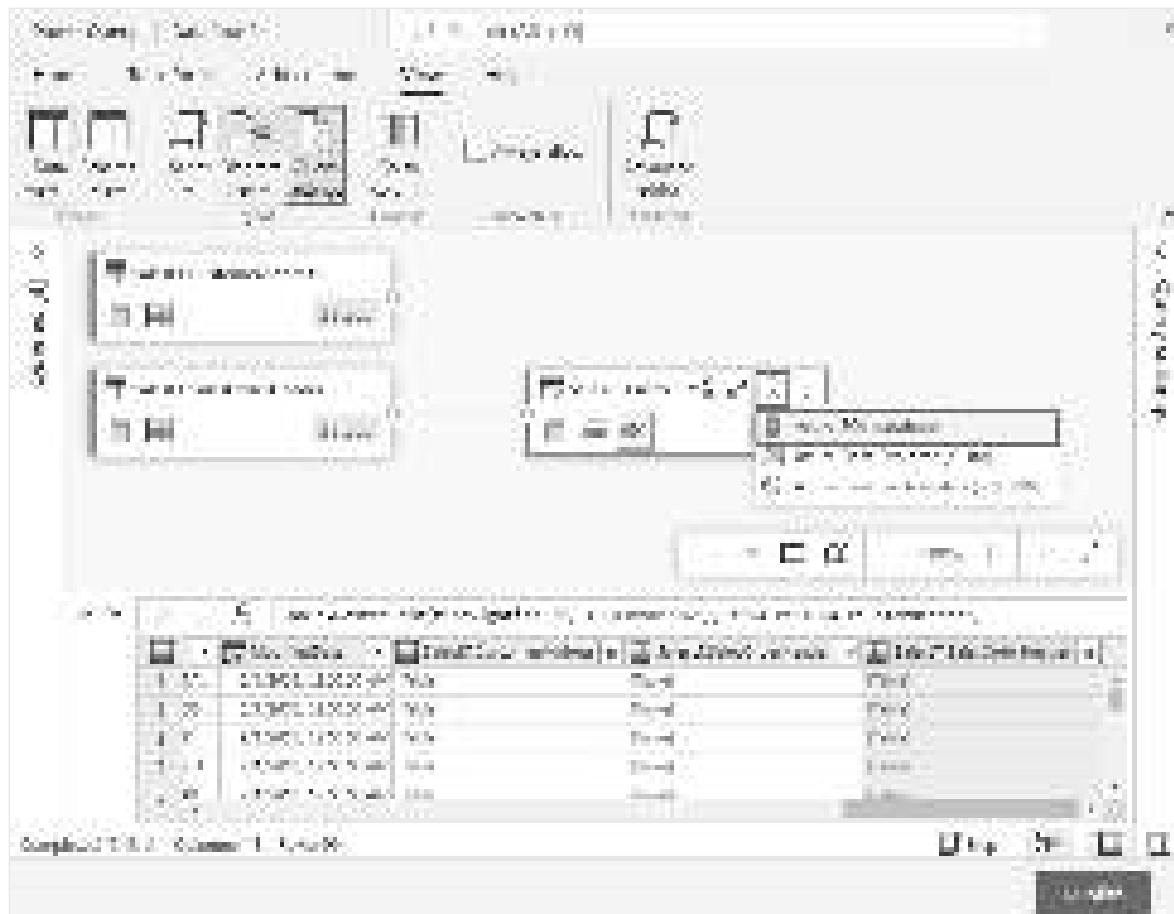
2. Right-click your **SalesLT Customer** query, or select the vertical ellipsis on the right of the query, then select **Merge queries**.



3. Configure the merge by selecting the **SalesOrderHeader** table as the right table for the merge, the **CustomerID** column from each table as the join column, and **Left outer** as the join kind. Then select **Ok** when to add the merge query.

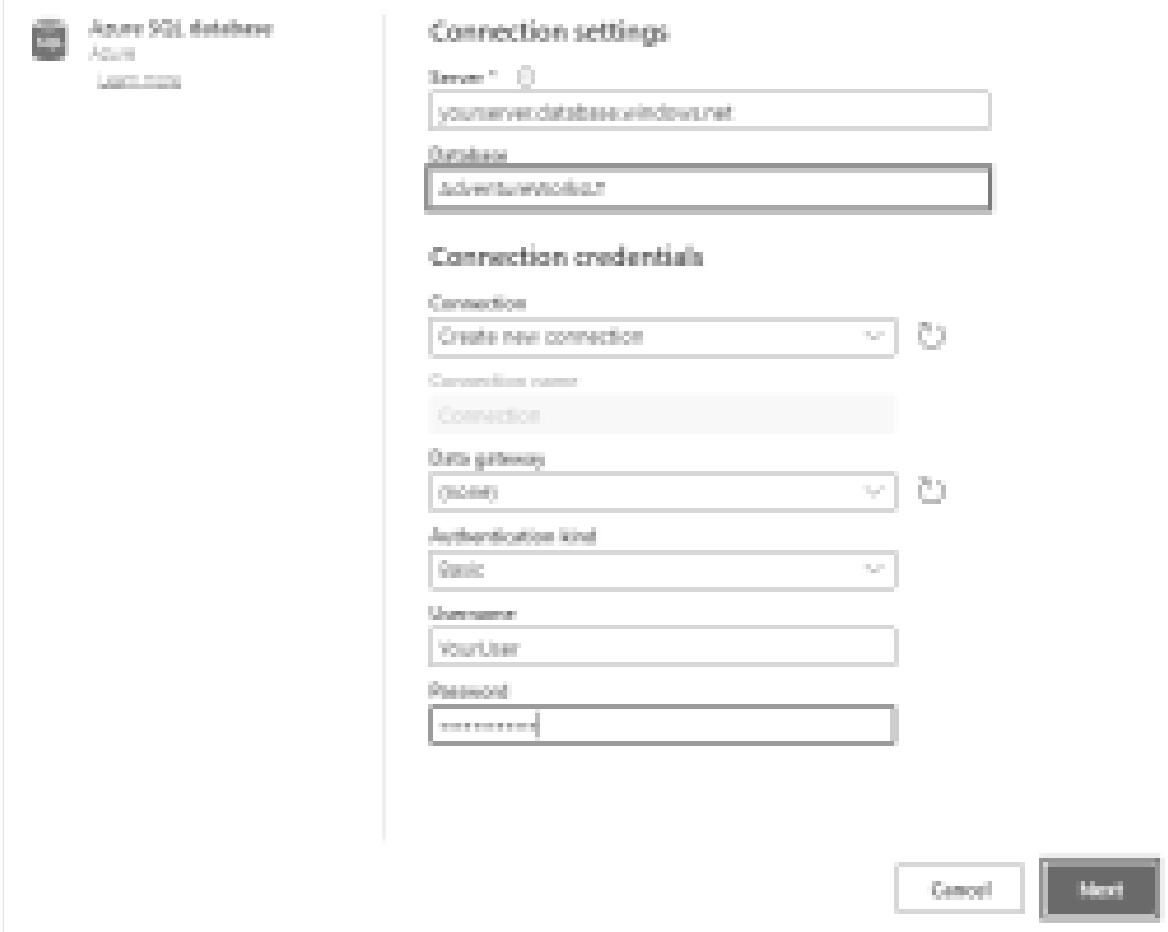


4. Select the **Add data destination** button, which looks like a database symbol with an arrow above it, from the new merge query you just created. Then select **Azure SQL database** as the destination type.



5. Provide the details for your Azure SQL database connection where the merge query is to be published. In this example, you can use the **AdventureWorksLT** database we used as the data source for the destination too.

Connect to data destination



6. Choose a database to store the data, and provide a table name, then select **Next**.



7. You can leave the default settings on the **Choose destination settings** dialog, and just select **Save settings** without making any changes here.

Choose destination settings

Update method



Column mapping

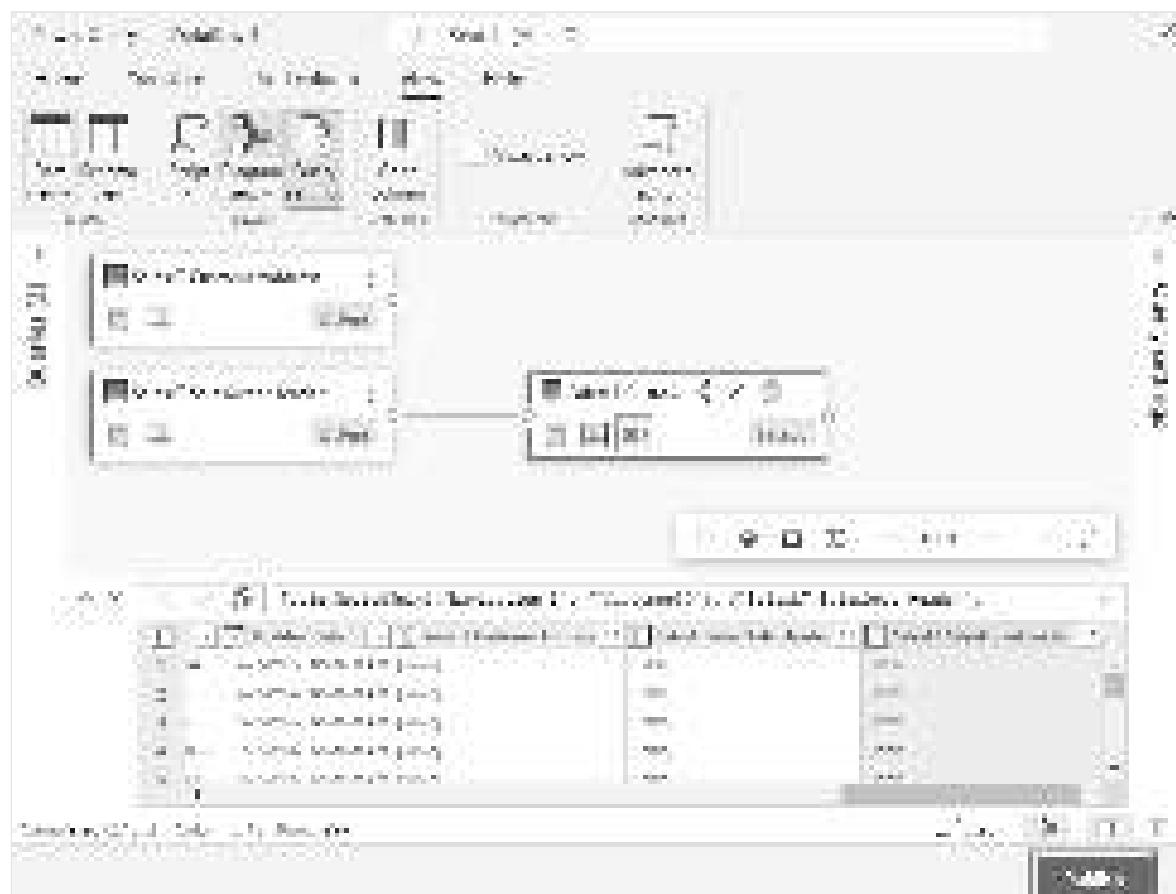
Source	Destination	Type
CustomerID	CustomerID	Whole Number
NameStyle	NameStyle	True/False
Title	Title	Type
FirstName	FirstName	Text
MiddleName	MiddleName	Text
LastName	LastName	Text
Suffix	Suffix	Text
CompanyName	CompanyName	Text

Back

Cancel

Save settings

8. Select Publish back on the dataflow editor page, to publish the dataflow.

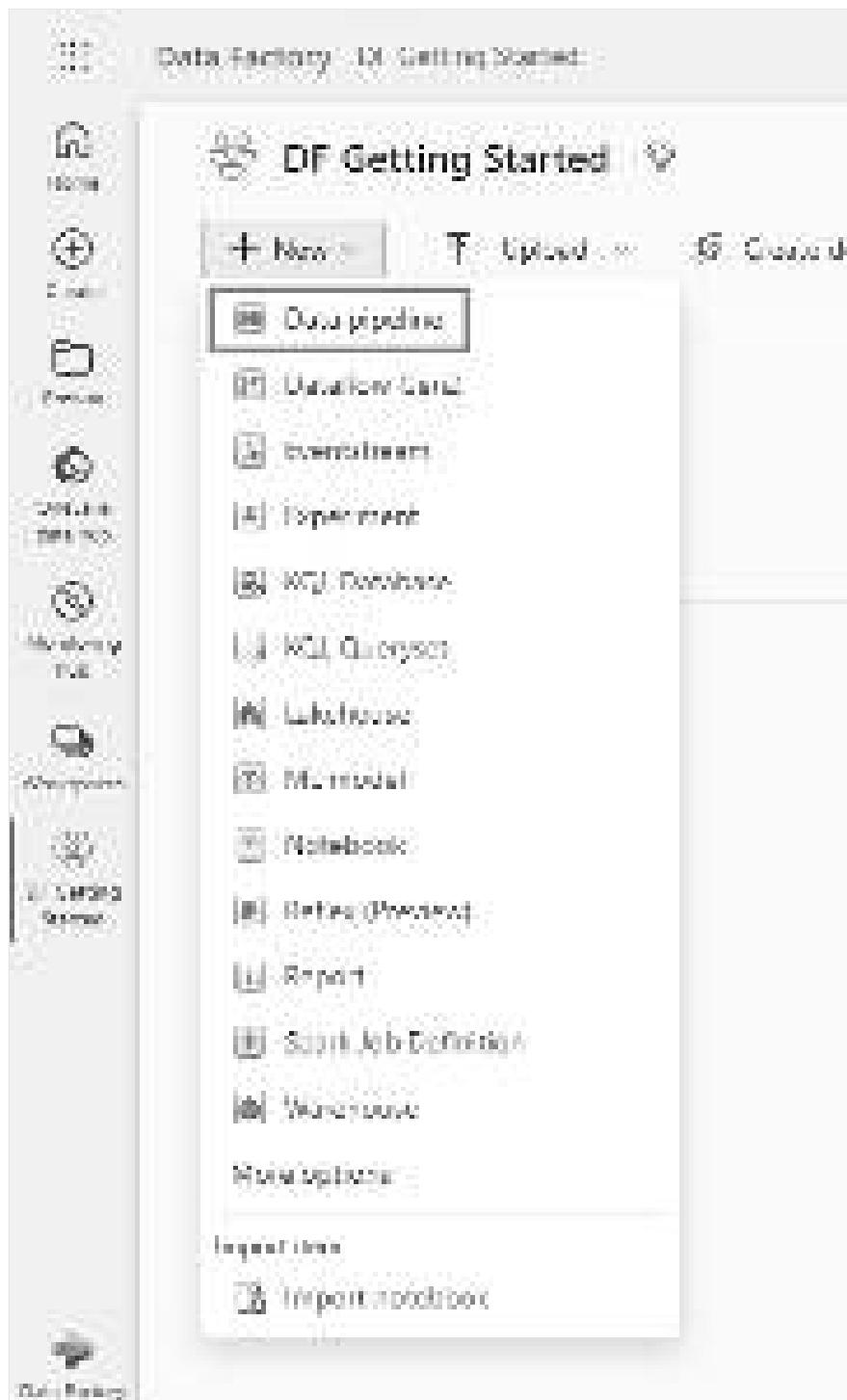


Move data with data pipelines

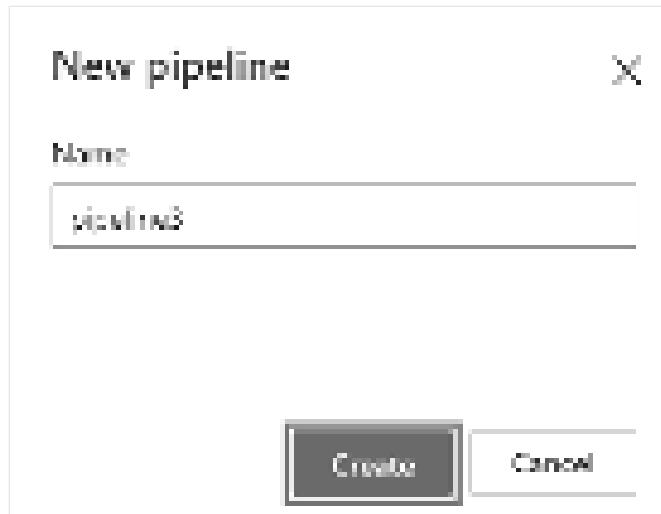
Now that you created a Dataflow Gen2, you can operate on it in a pipeline. In this example, you copy the data generated from the dataflow into text format in an Azure Blob Storage account.

Step 1: Create a new data pipeline

1. From your workspace, select **New**, and then select **Data pipeline**.

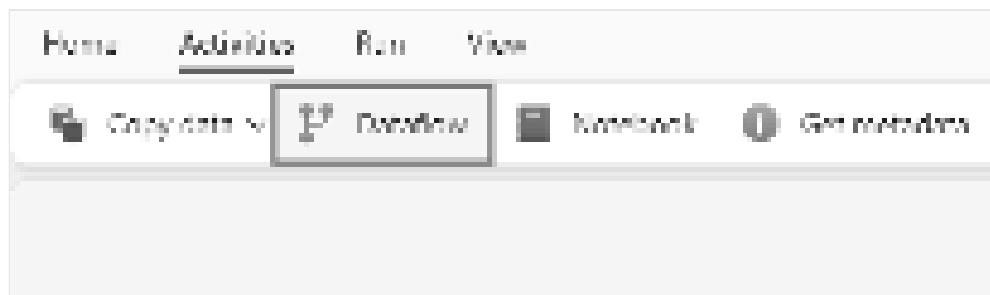


2. Name your pipeline then select **Create**.

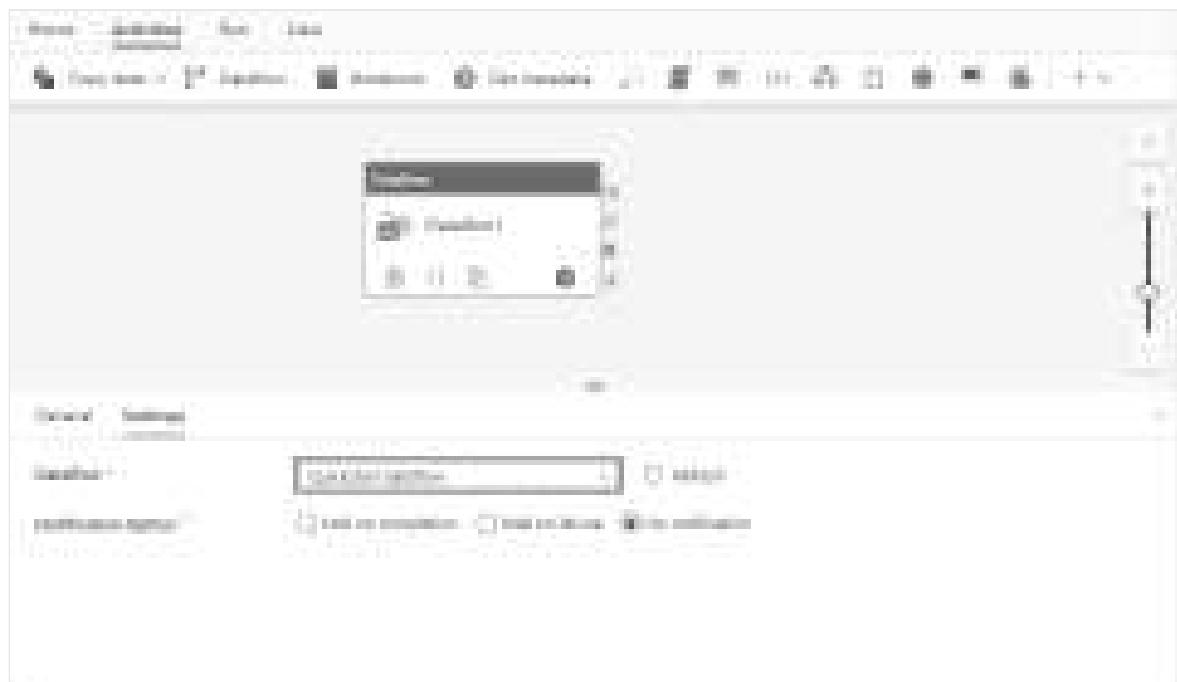


Step 2: Configure your dataflow

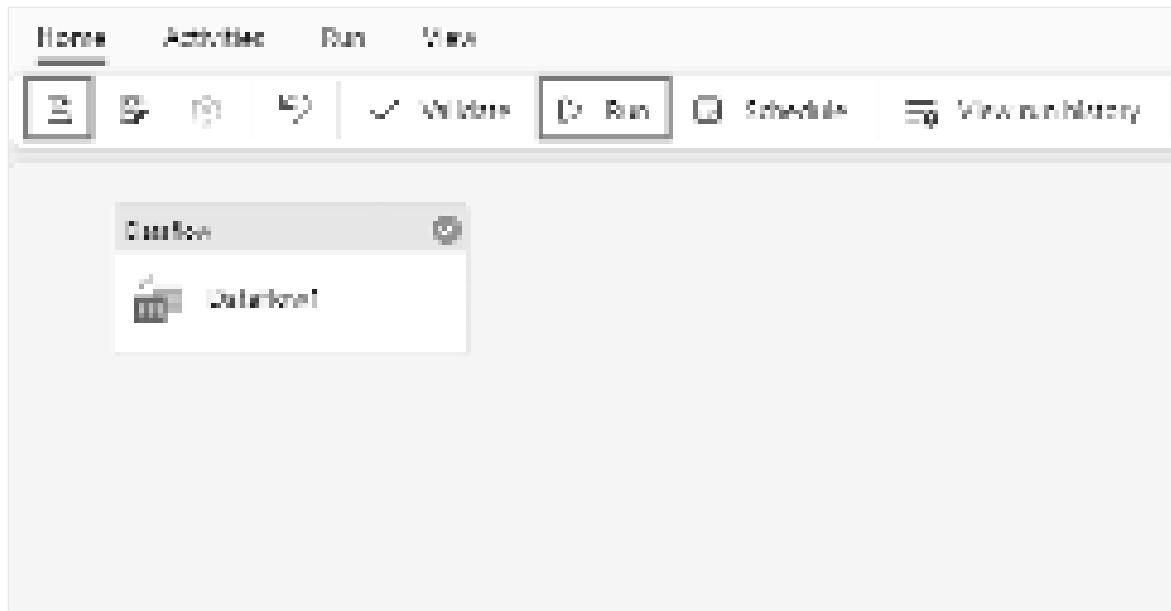
1. Add a new dataflow activity to your data pipeline by selecting **Dataflow** in the **Activities** tab.



2. Select the dataflow on the pipeline canvas, and then the **Settings** tab. Choose the dataflow you created previously from the drop-down list.

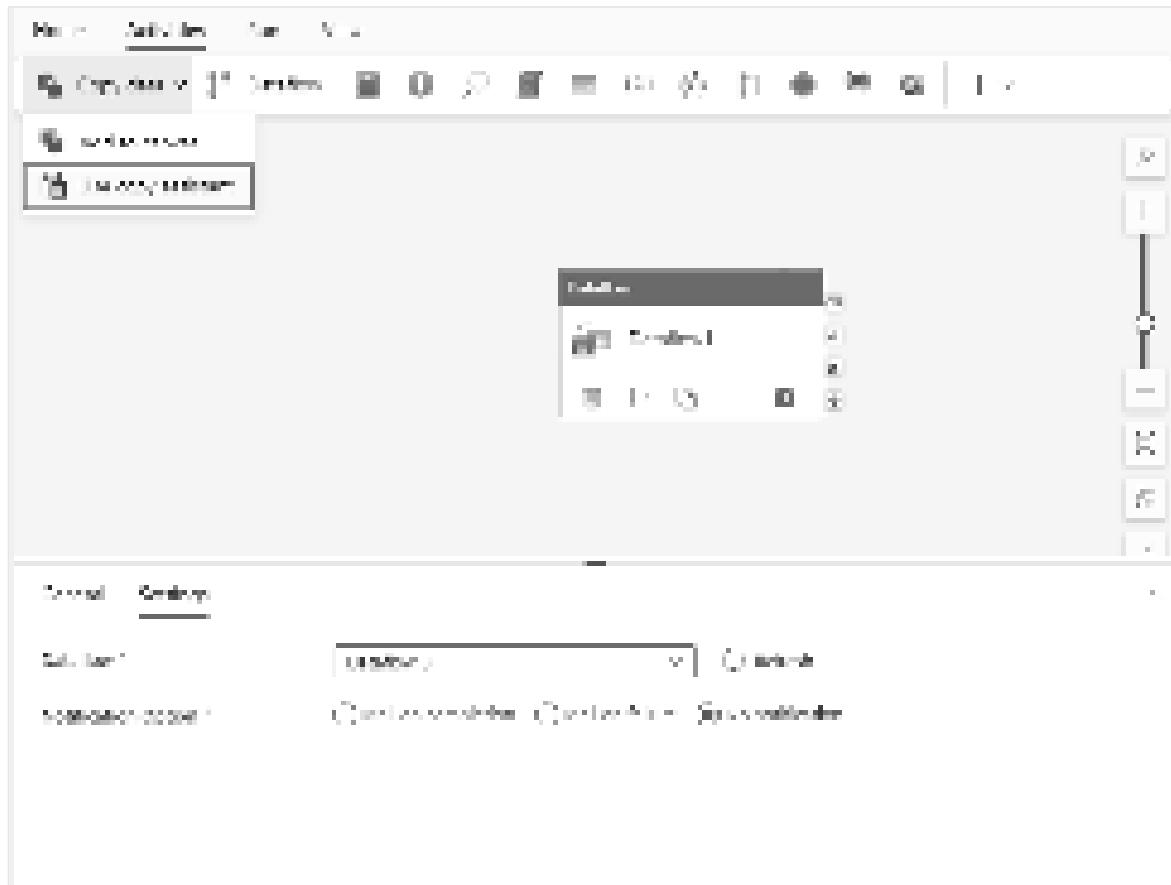


3. Select **Save**, and then **Run** to run the dataflow to initially populate its merged query table you designed in the prior step.



Step 3: Use the copy assistant to add a copy activity

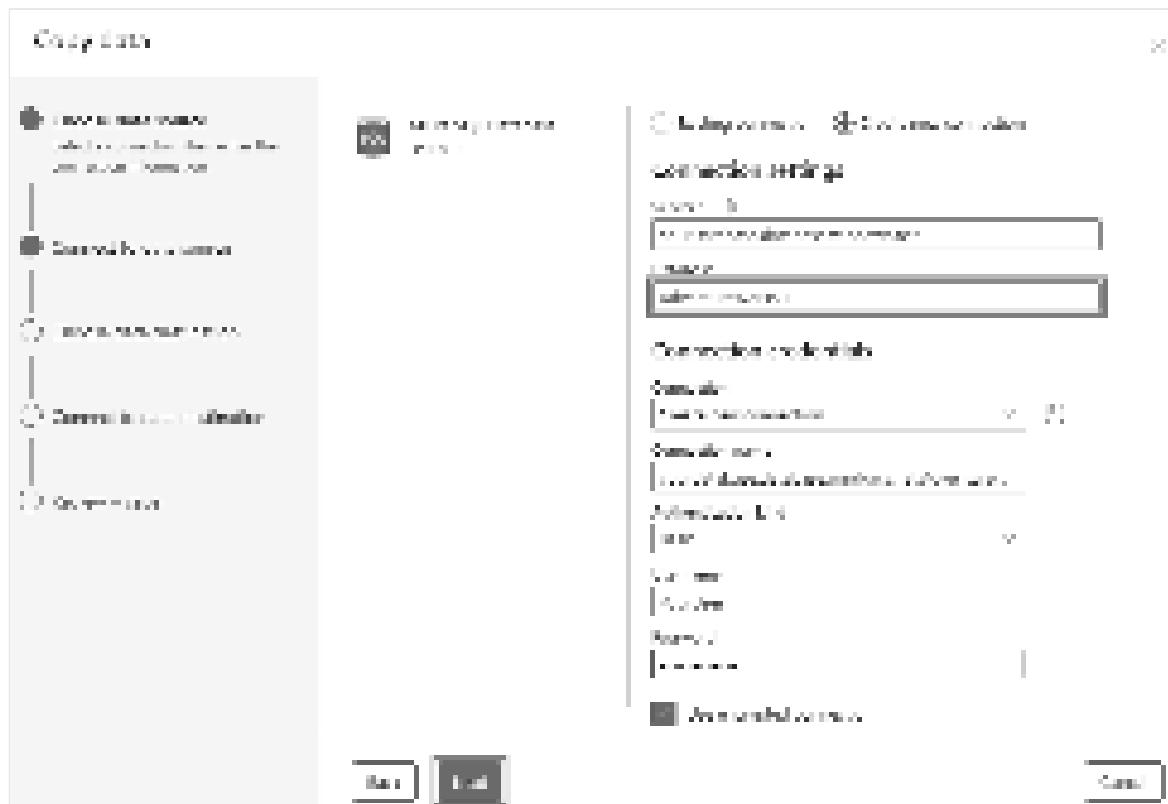
1. Select **Copy data** on the canvas to open the **Copy Assistant** tool to get started. Or select **Use copy assistant** from the **Copy data** drop down list under the **Activities** tab on the ribbon.



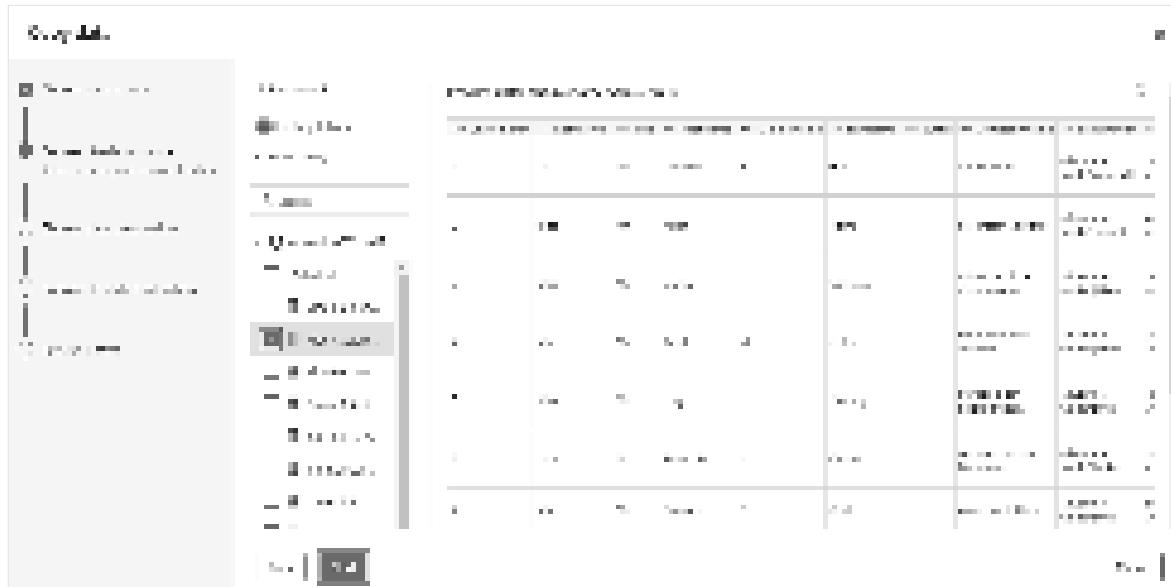
2. Choose your data source by selecting a data source type. In this tutorial, you use the Azure SQL Database used previously when you created the dataflow to generate a new merge query. Scroll down below the sample data offerings and select the **Azure** tab under **Data sources**, then **Azure SQL Database**. Then select **Next** to continue.



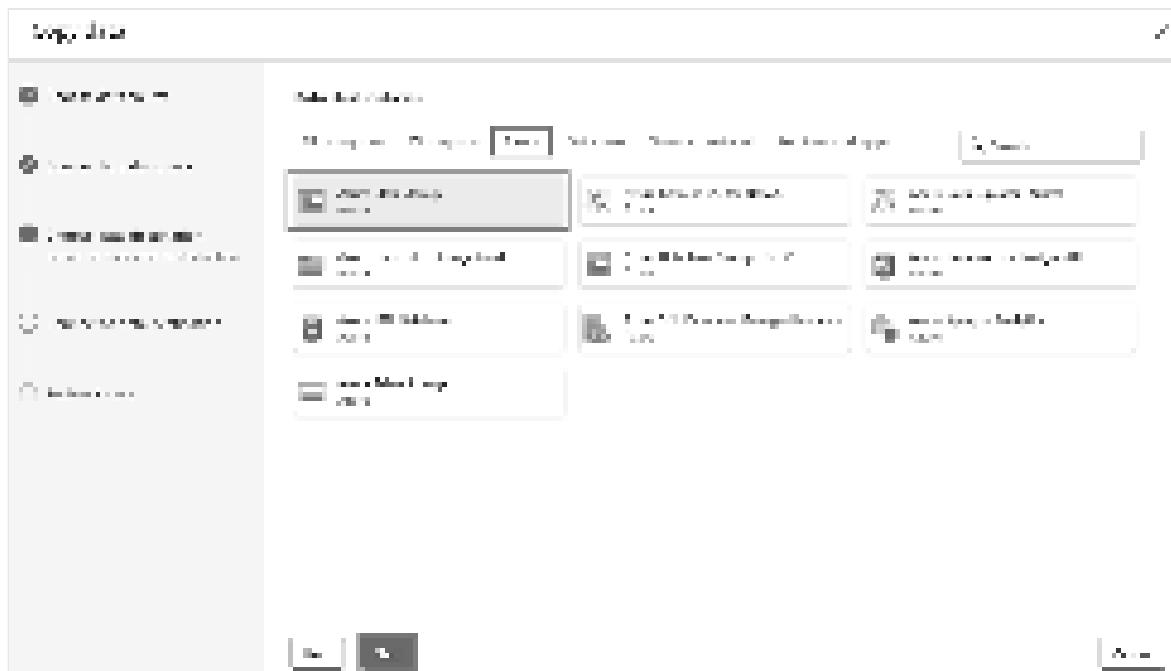
3. Create a connection to your data source by selecting **Create new connection**. Fill in the required connection information on the panel, and enter the AdventureWorksLT for the database, where we generated the merge query in the dataflow. Then select **Next**.



4. Select the table you generated in the dataflow step earlier, and then select **Next**.



5. For your destination, choose **Azure Blob Storage** and then select **Next**.

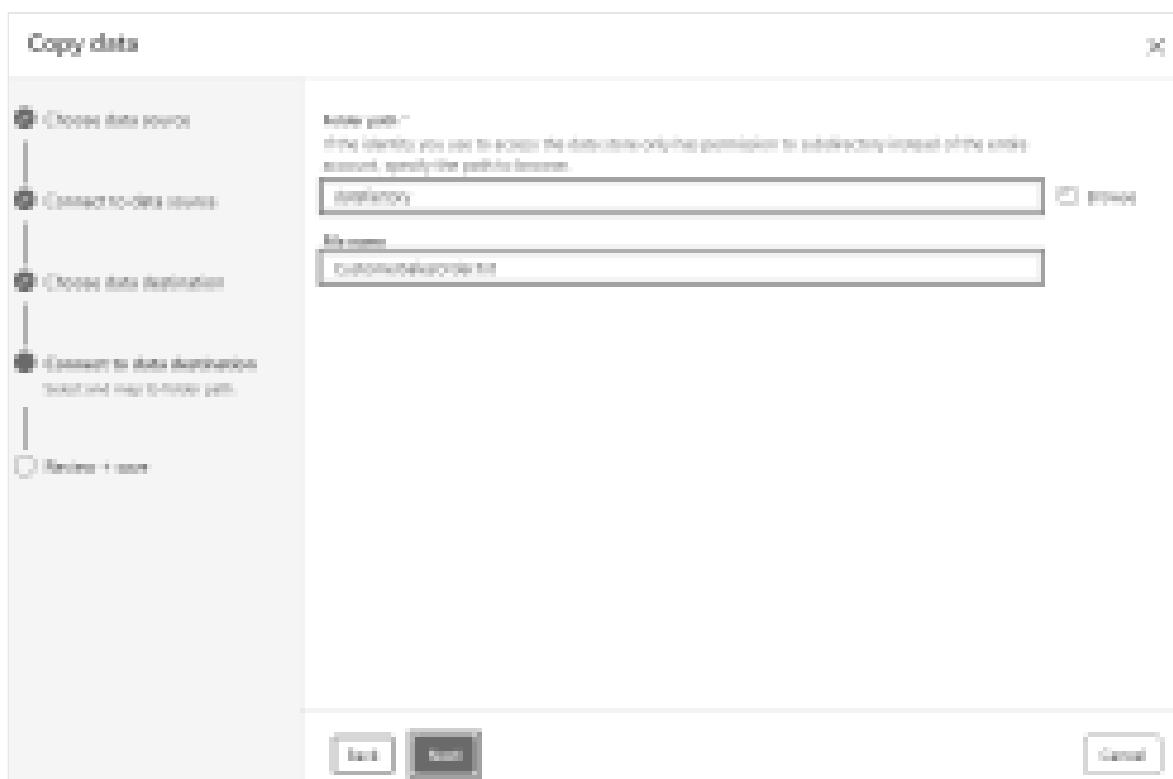


6. Create a connection to your destination by selecting **Create new connection**.

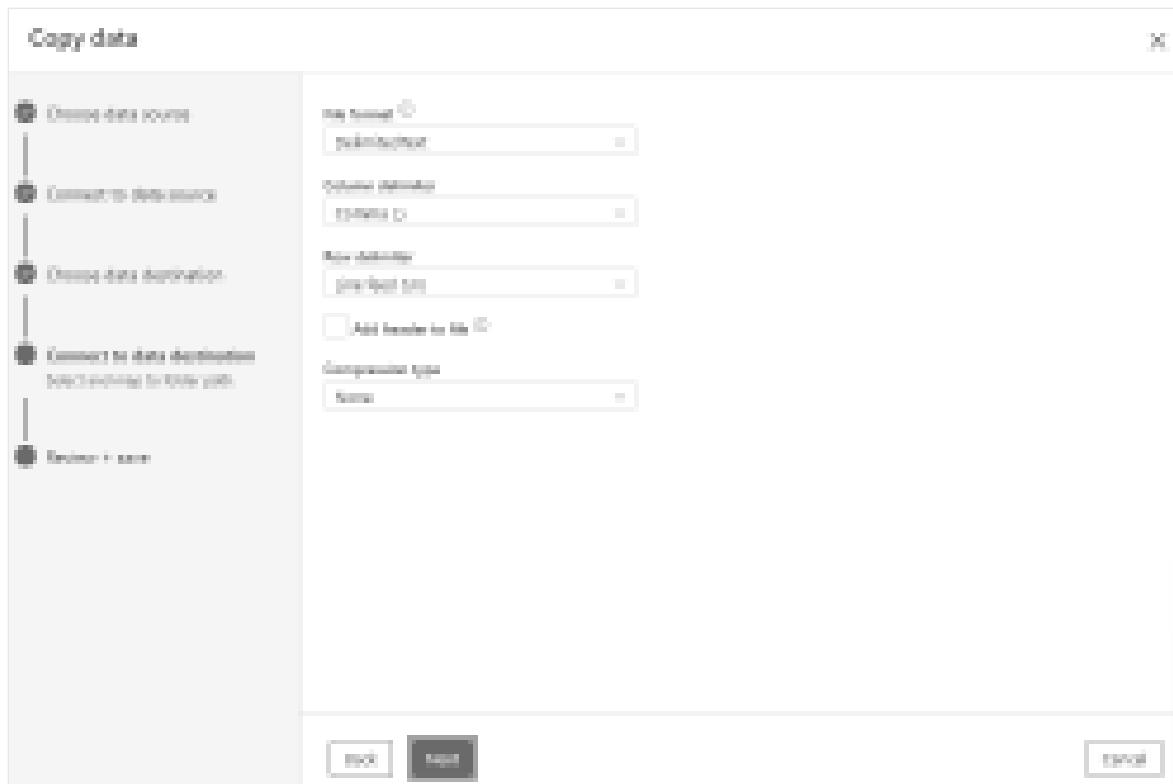
Provide the details for your connection, then select **Next**.



7. Select your **Folder path** and provide a **File name**, then select **Next**.



8. Select **Next** again to accept the default file format, column delimiter, row delimiter and compression type, optionally including a header.

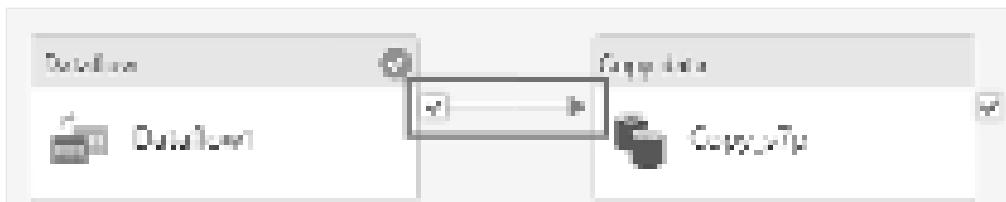


9. Finalize your settings. Then, review and select **OK** to finish the process.

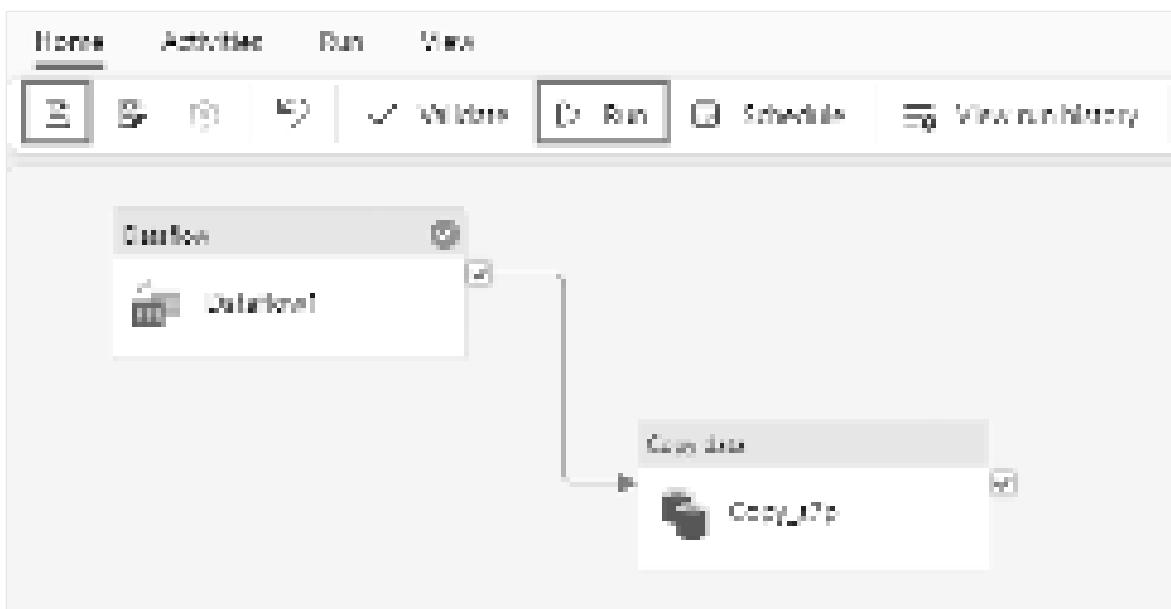


Step 5: Design your data pipeline and save to run and load data

1. To run the **Copy** activity after the **Dataflow** activity, drag from **Succeeded** on the **Dataflow** activity to the **Copy** activity. The **Copy** activity only runs after the **Dataflow** activity has succeeded.



2. Select **Save** to save your data pipeline. Then select **Run** to run your data pipeline and load your data.



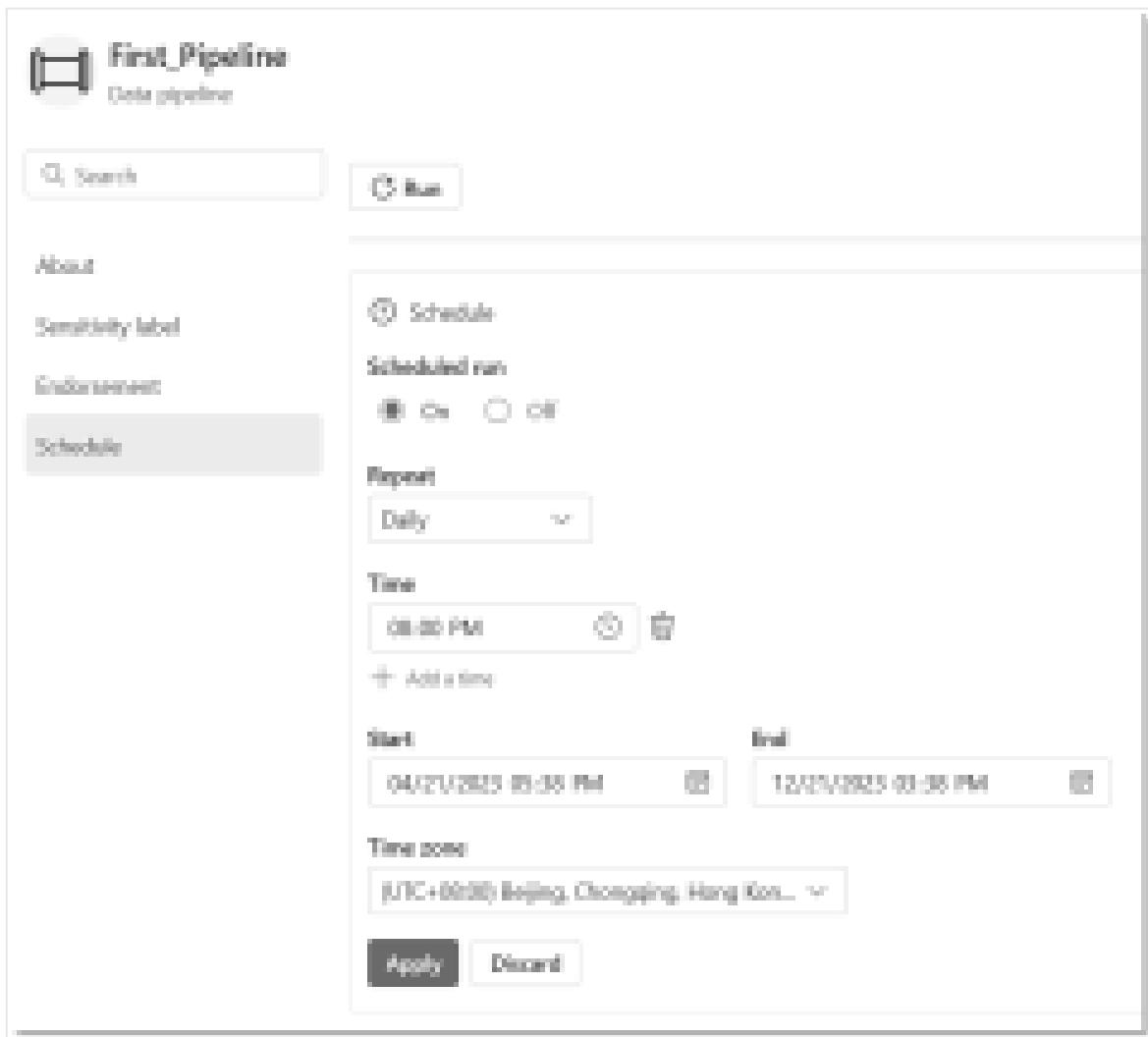
Schedule pipeline execution

Once you finish developing and testing your pipeline, you can schedule it to execute automatically.

1. On the **Home** tab of the pipeline editor window, select **Schedule**.



2. Configure the schedule as required. The example here schedules the pipeline to execute daily at 8:00 PM until the end of the year.



Next steps

This sample shows you how to create and configure a Dataflow Gen2 to create a merge query and store it in an Azure SQL database, then copy data from the database into a text file in Azure Blob Storage. You learned how to:

- ✓ Create a dataflow.
- ✓ Transform data with the dataflow.
- ✓ Create a data pipeline using the dataflow.
- ✓ Order the execution of steps in the pipeline.
- ✓ Copy data with the Copy Assistant.
- ✓ Run and schedule your data pipeline.

Next, advance to learn more about monitoring your pipeline runs.

[How to monitor pipeline runs in Microsoft Fabric](#)

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Microsoft Fabric decision guide: copy activity, dataflow, or Spark

Article • 05/23/2023

Use this reference guide and the example scenarios to help you in deciding whether you need a copy activity, a dataflow, or Spark for your workloads using Microsoft Fabric.

ⓘ Important

Microsoft Fabric is in preview.

Copy activity, dataflow, and Spark properties

	Pipeline copy activity	Dataflow Gen 2	Spark
Use case	Data lake and data warehouse migration, data ingestion, lightweight transformation	Data ingestion, data transformation, data wrangling, data profiling	Data ingestion, data transformation, data processing, data profiling
Primary developer persona	Data engineer, data integrator	Data engineer, data integrator, business analyst	Data engineer, data scientist, data developer
Primary developer skill set	ETL, SQL, JSON	ETL, M, SQL	Spark (Scala, Python, Spark SQL, R)
Code written	No code, low code	No code, low code	Code
Data volume	Low to high	Low to high	Low to high
Development interface	Wizard, canvas	Power query	Notebook, Spark job definition
Sources	30+ connectors	150+ connectors	Hundreds of Spark libraries

	Pipeline copy activity	Dataflow Gen 2	Spark
Destinations	18+ connectors	Lakehouse, Azure SQL database, Azure Data explorer, Azure Synapse analytics	Hundreds of Spark libraries
Transformation complexity	Low: lightweight - type conversion, column mapping, merge/split files, flatten hierarchy	Low to high: 300+ transformation functions	Low to high: support for native Spark and open-source libraries

Review the following three scenarios for help with choosing how to work with your data in Fabric.

Scenario1

Leo, a data engineer, needs to ingest a large volume of data from external systems, both on-premises and cloud. These external systems include databases, file systems, and APIs. Leo doesn't want to write and maintain code for each connector or data movement operation. He wants to follow the medallion layers best practices, with bronze, silver, and gold. Leo doesn't have any experience with Spark, so he prefers the drag and drop UI as much as possible, with minimal coding. And he also wants to process the data on a schedule.

The first step is to get the raw data into the bronze layer lakehouse from Azure data resources and various third party sources (like Snowflake Web, REST, AWS S3, GCS, etc.). He wants a consolidated lakehouse, so that all the data from various LOB, on-premises, and cloud sources reside in a single place. Leo reviews the options and selects **pipeline copy activity** as the appropriate choice for his raw binary copy. This pattern applies to both historical and incremental data refresh. With copy activity, Leo can load Gold data to a data warehouse with no code if the need arises and pipelines provide high scale data ingestion that can move petabyte-scale data. Copy activity is the best low-code and no-code choice to move petabytes of data to lakehouses and warehouses from varieties of sources, either ad-hoc or via a schedule.

Scenario2

Mary is a data engineer with a deep knowledge of the multiple LOB analytic reporting requirements. An upstream team has successfully implemented a solution to migrate multiple LOB's historical and incremental data into a common lakehouse. Mary has been tasked with cleaning the data, applying business logics, and loading it into multiple destinations (such as Azure SQL DB, ADX, and a lakehouse) in preparation for their respective reporting teams.

Mary is an experienced Power Query user, and the data volume is in the low to medium range to achieve desired performance. Dataflows provide no-code or low-code interfaces for ingesting data from hundreds of data sources. With dataflows, you can transform data using 300+ data transformation options, and write the results into multiple destinations with an easy to use, highly visual user interface. Mary reviews the options and decides that it makes sense to use **Dataflow Gen 2** as her preferred transformation option.

Scenario3

Adam is a data engineer working for a large retail company that uses a lakehouse to store and analyze its customer data. As part of his job, Adam is responsible for building and maintaining the data pipelines that extract, transform, and load data into the lakehouse. One of the company's business requirements is to perform customer review analytics to gain insights into their customers' experiences and improve their services.

Adam decides the best option is to use **Spark** to build the extract and transformation logic. Spark provides a distributed computing platform that can process large amounts of data in parallel. He writes a Spark application using Python or Scala, which reads structured, semi-structured, and unstructured data from OneLake for customer reviews and feedback. The application cleanses, transforms, and writes data to Delta tables in the lakehouse. The data is then ready to be used for downstream analytics.

Next steps

- How to copy data using copy activity
- Quickstart: Create your first dataflow to get and transform data
- How to create an Apache Spark job definition in Fabric

How to access on-premises data sources in Data Factory for Microsoft Fabric

Article • 11/15/2023

Data Factory for Microsoft Fabric is a powerful cloud-based data integration service that allows you to create, schedule, and manage workflows for various data sources. In scenarios where your data sources are located on-premises, Microsoft provides the On-Premises Data Gateway to securely bridge the gap between your on-premises environment and the cloud. This document will guide you through the process of accessing on-premises data sources within Data Factory for Microsoft Fabric using the On-Premises Data Gateway.

Create an on-premises data gateway

1. An on-premises data gateway is a software application designed to be installed within a local network environment. It provides a means to directly install the gateway onto your local machine. For detailed instructions on how to download and install the on-premises data gateway, please refer to [Install an on-premises data gateway](#).



On-premises data gateway

Status

Data exchange

My services

Services

Actions

Your gateway is set up

Open a link to view information on your gateway

Gateway address: <http://192.168.1.100:8080>

Address is available

Download

 Help us improve the experience and connect by sending user information

Email

Send the survey questionnaire

Sign in

Close

2. Log in using your user account to access the on-premises data gateway, after which it will be prepared for utilization.



On-premises data gateway

- Links**
- [New connection](#)
- [Discover](#)
- [Run SQL](#)
- [Overview](#)
- [Gateway keys](#)

💡 The gateway `Demogateway02` is online and ready to be used.

Discover new services → [Discover now](#)

A new version is available [Download](#)

💡 It's highly recommended to update your gateway system to the latest version.

[Read the release information](#)

Logic Apps, Azure Functions, Power BI Services
using Cloud API

[Create a service in Azure](#)

**Power Apps, Power Automate,
Power BI Services**
using Cloud API

(+) Add API

Power BI
Using Cloud API

(+) Add API

[Close](#)

Create a connection for your on-premises data source

1. Navigate to the admin portal and select the settings button (an icon that looks like a gear) at the top right of the page. Then choose Manage connections and gateways from the dropdown menu that appears.

Settings

X

Preferences

General →

Notifications →

Item settings →

Developer settings →

Resources and extensions

Manage personal storage →

Power BI settings →

Manage connections and gateways →

Manage embed codes →

Azure Analysis Services migrations →

2. On the **New connection** dialog that appears, select **On-premises** and then provide your gateway cluster, along with the associated resource type and relevant

information.

New connection X

On-premises Virtual network Cloud

Gateway cluster name *
DemoGateway02

Connection name *
SQLServerConnection

Connection type *
SQL Server

Server *
Ex: testcore.contoso.com

Database *
Ex: ContosoDB

Authentication

Create **Close**

Connect your on-premises data source to a Dataflow Gen2 in Data Factory for Microsoft Fabric

1. Go to your workspace and create a Dataflow Gen2.



① Note

Please be aware that the Fabric pipeline currently does not offer support for on-premises data sources. However, you can implement a workaround by initially transferring the data to a cloud storage using a Dataflow Gen2.

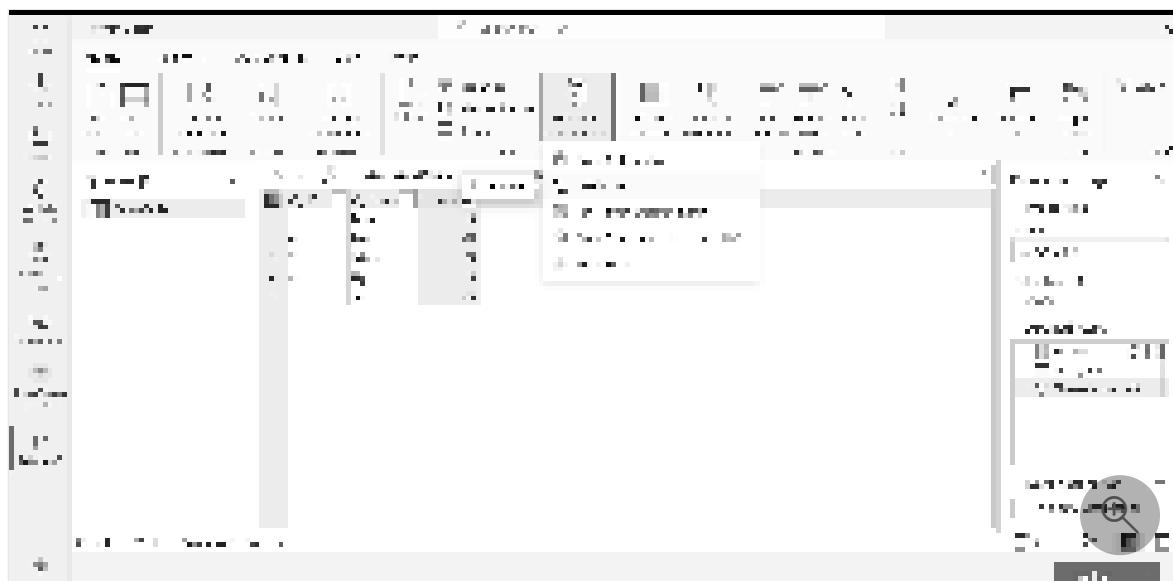
2. Add a new source to the dataflow and select the connection established in the previous step.



3. You can use the Dataflow Gen2 to perform any necessary data transformations based on your requirements.



4. Use the **Add data destination** button on the **Home** tab of the Power Query editor to add a destination for your data from the on-premises source.



5. Publish the Dataflow Gen2.



You have now created a Dataflow Gen2 to load data from an on-premises data source into a cloud destination.

Using on-premises data in a pipeline

Fabric pipelines can leverage on-premises data gateway for on-premises data access and data movement. Although Fabric pipeline currently does not offer support for on-premises data sources directly, you can implement a workaround by initially transferring the data to cloud storage using a Dataflow Gen2 as described in this article, and then accessing the cloud storage from the pipeline to work with the data.

Next steps

- On-premises data gateway considerations for output destinations
 - Known issues with the on-premises data gateway
-

Feedback

Was this page helpful?



Provide product feedback ↗ | Ask the community ↗

How to monitor activity in Microsoft Fabric

Article • 11/15/2023

There are two ways to monitor activity in Data Factory for Microsoft Fabric. The service supports monitoring for both pipeline runs and dataflows.

Monitor data pipeline runs

Refer to How to monitor data pipeline runs for details on how to monitor your data pipelines.

Monitor dataflows

Refer to View refresh history and monitor your dataflows for details on how to monitor your dataflows.

Next steps

To get started with Microsoft Fabric, go to Quickstart: Create your first Dataflow Gen2 to get and transform data.

Feedback

Was this page helpful?



Provide product feedback [↗](#) | Ask the community [↗](#)

Migrate to Data Factory in Microsoft Fabric

Article • 11/15/2023

Data Factory in Microsoft Fabric brings Power Query and Azure Data Factory together into a modern trusted data integration experience that empowers data and business professionals to extract, load, and transform data for their organization. In addition, powerful data orchestration capabilities enable you to build simple to complex data workflows that orchestrate the steps needed for your data integration needs.

Key concepts in Data Factory in Microsoft Fabric

- **Get data and transformation** - Dataflow Gen2 is an evolution of dataflows in Power BI. Dataflow Gen2 is rearchitected to use Fabric compute engines for data processing and transformation. This enables them to ingest and transform data at any scale.
- **Data orchestration** - Using data pipelines already familiar to users of Azure Data Factory (ADF), Microsoft Fabric enables the same orchestration capabilities offered in ADF. As part of the GA release of Fabric, data pipelines support most of the activities available in ADF.

Refer to this list of activities that are a part of data pipelines in Fabric. The SSIS activity will be added to data pipelines by Q2 CY2024.

- **Enterprise-ready data movement** - Whether it's small data movement or petabyte-scale, Data Factory provides a serverless and intelligent data movement platform that enables you to move data between diverse data sources and data destinations reliably. With support for 170+ connectors, Data Factory in Microsoft Fabric enables you to move data between multiclouds, data sources on-premises, and within virtual networks (VNet). Intelligent throughput optimization enables the data movement platform to automatically detect the size of the compute needed for data movement.

Migrate from Azure Data Factory (ADF)

To enable customers to upgrade to Microsoft Fabric from Azure Data Factory (ADF), we support the following features:

- **Data pipeline activities** - We support most of the activities that you already use in ADF to Data Factory in Fabric. In addition, we have added new activities for notifications, for example, the Teams and Outlook activities. Refer to this list of activities that are available in Data Factory in Fabric.
- **OneLake/Lakehouse connector in Azure Data Factory** - For many ADF customers, you can now integrate with Microsoft Fabric, and bring data into the Fabric OneLake[↗].
- **Azure Data Factory Mapping Dataflow to Fabric** - We provide this guide for ADF customers considering building new data transformations in Fabric.

In addition, for customers considering migrating their ADF mapping dataflows to Fabric, you can apply sample code from the Fabric Customer Advisory Team (Fabric CAT) to convert mapping dataflows to Spark code. Find out more at [Mapping dataflows to Microsoft Fabric[↗]](#).

As part of the Data Factory in Microsoft Fabric roadmap, we are working towards the preview of the following by Q2 CY2024:

- **Mounting of ADF in Fabric** - This feature will enable customers to mount their existing ADF in Microsoft Fabric. All ADF pipelines will work as-is, and continue running on Azure, while enabling you to explore Fabric and work out a more comprehensive upgrade plan.
- **Upgrade from ADF pipelines to Fabric** - We are working with customers and the community to learn how we can best support upgrades of data pipelines from ADF to Fabric. As part of this, we will deliver an upgrade experience that empowers you to test your existing data pipelines in Fabric using mounting and upgrading the data pipelines.

Migrate Power BI Dataflow Gen1 to Dataflow Gen2 in Fabric

Dataflow Gen2 in Fabric provides many advantages and new capabilities compared to dataflows (Gen1) in Power BI:

- High-Scale Get Data ("Fast Copy")
- High-Scale Data Transformations (using Fabric Lakehouse SQL engine)
- More Output Destinations: Azure SQL DB, Lakehouse, Warehouse, SharePoint, KQL Databases, and more
- Enhanced Refresh History & Monitoring experience
- Enhanced Authoring and Publish experiences.

We encourage customers to start trying out Dataflow Gen2, either to recreate existing Dataflow Gen1 scenarios or to try out new ones. Early feedback on Dataflow Gen2 will help us evolve and mature product capabilities.

We have a few options for customers to recreate your Gen1 dataflows as Dataflow Gen2:

- Export Dataflow Gen1 queries and import them into Dataflow Gen2: You can now export queries in both the Dataflows and Dataflow Gen2 authoring experiences and save them to PQT files that you can then import into Dataflow Gen2. For more information, see [Use the export template feature](#).
- Copy and paste in Power Query: If you have a dataflow in Power BI or Power Apps, you can copy your queries and paste them in the editing experience of your Dataflow Gen2 artifact. This functionality allows you to migrate your dataflow to Gen2 without having to rewrite your queries. For more information, see [Copy and paste existing dataflows \(Gen1\) queries](#).

Also refer to the following article for further considerations: [Differences between Dataflow Gen1 and Gen2](#)

Next steps

- Pipeline activities supported in Microsoft Fabric
- Guide to Dataflow Gen2 for ADF mapping dataflow users

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Move queries from Dataflow Gen1 to Dataflow Gen2

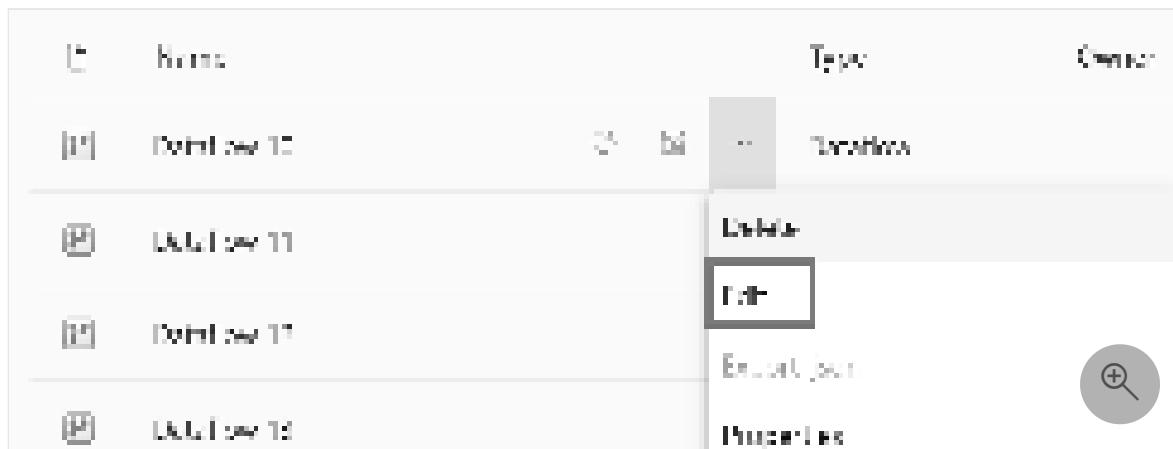
Article • 11/15/2023

Dataflow Gen2 is the new generation of dataflows. However, many existing dataflows have been created over the years using the original version of dataflows (now called *Dataflow Gen1* and listed as the **Dataflow** type in your workspace). This article describes how you can reuse your older Dataflow Gen1 dataflows by importing Dataflow Gen1 dataflows into Dataflow Gen2 using the export template feature. It also includes information on how to copy the Gen1 queries and paste them into a new Dataflow Gen2.

Use the export template feature

To use the export template feature:

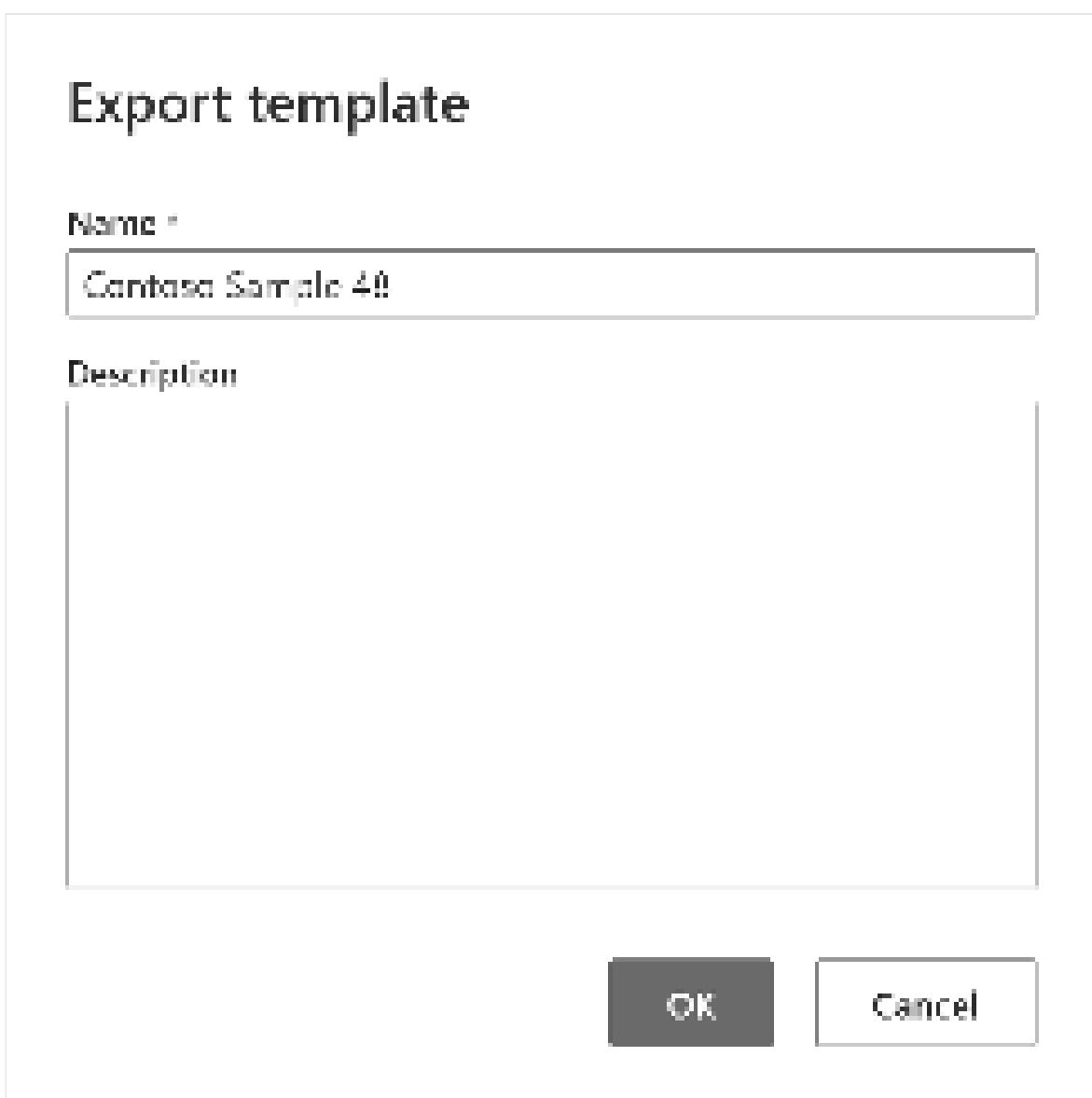
1. From your Power BI or Data Factory workspace, select the ellipsis next to an existing Dataflow Gen1 and then select **Edit**.



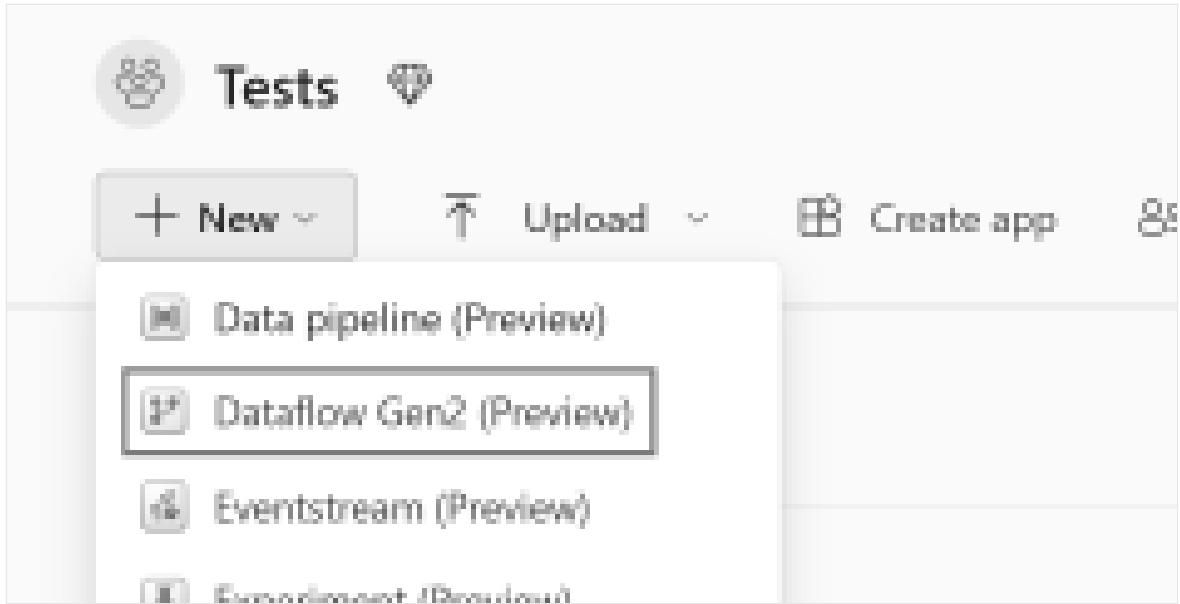
2. In the **Home** tab of the Power Query editor, select **Export template**.



3. In **Export template**, enter the name you want to call this template in **Name**.
Optionally, you can add a description for this template in **Description**.



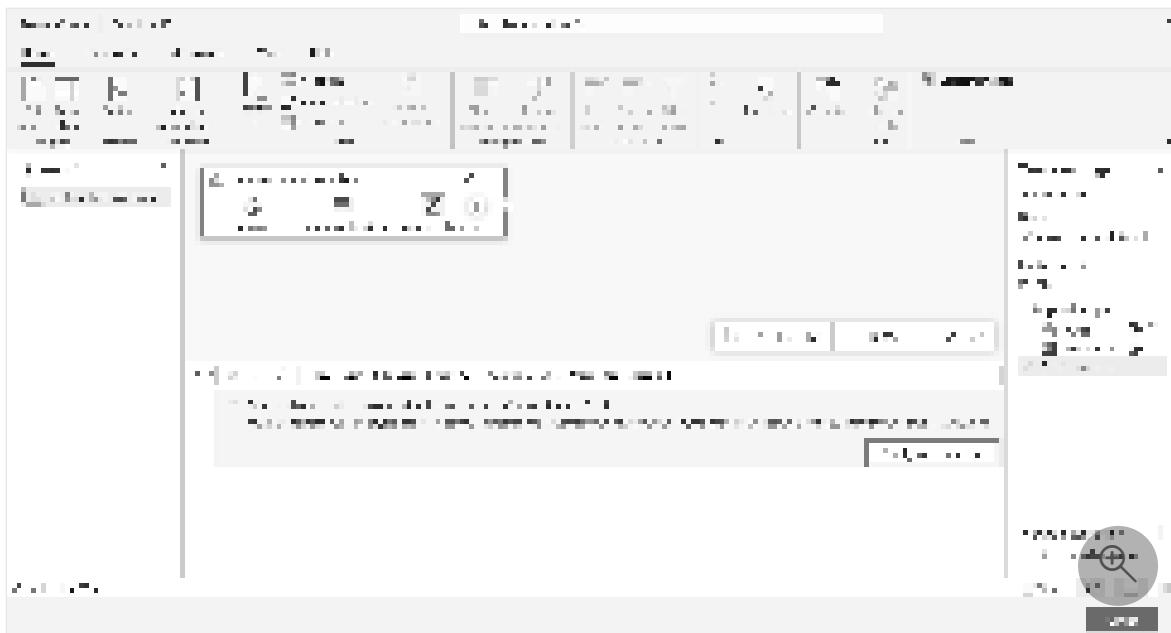
4. Select **OK** to save the template. The template is saved in your default Downloads folder.
5. From your Data Factory workspace, select **New**, and then select **Dataflow Gen2**.



6. From the current view pane of the Power Query editor, select **Import from a Power Query template**.



7. In the **Open** dialog box, browse to your default Downloads folder and select the .pqt file you saved in the previous steps. Then select **Open**.
8. The template is then imported to your Dataflow Gen2. You might be required to enter your credentials at this time. If so, select **Configure connection** and enter your credentials. Then select **Connect**.



Your Dataflow Gen1 has now been imported to Dataflow Gen2.

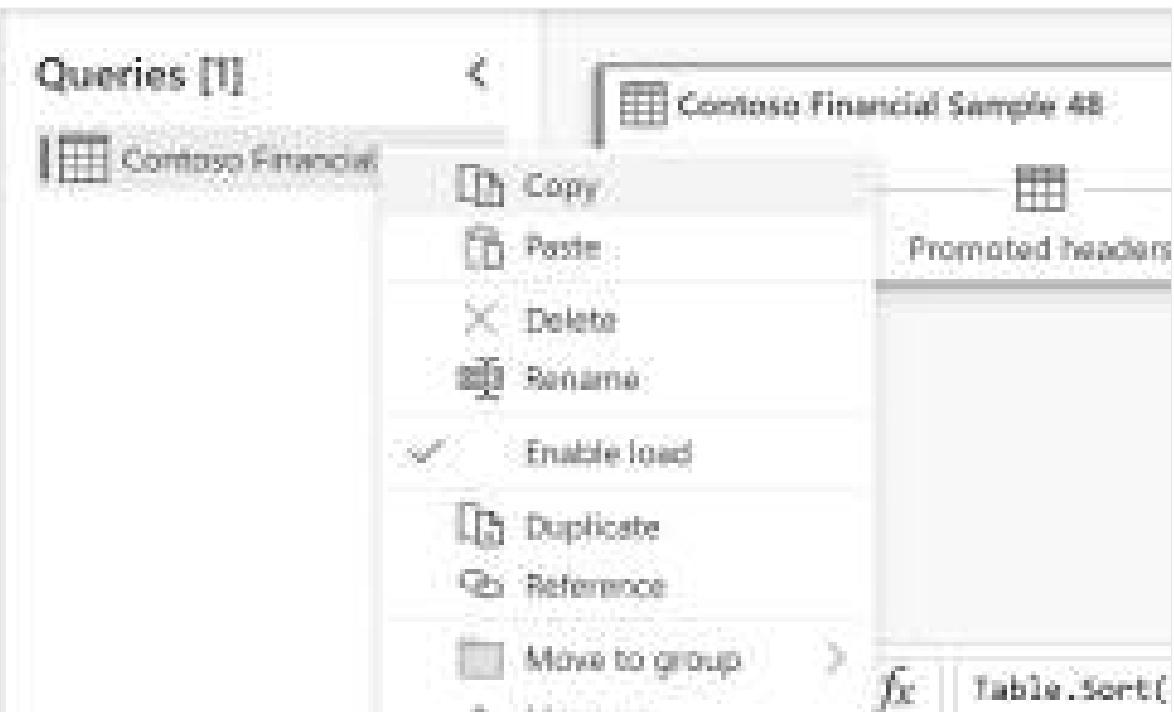
Copy and paste existing Dataflow Gen1 queries

To copy existing Dataflow Gen1 queries:

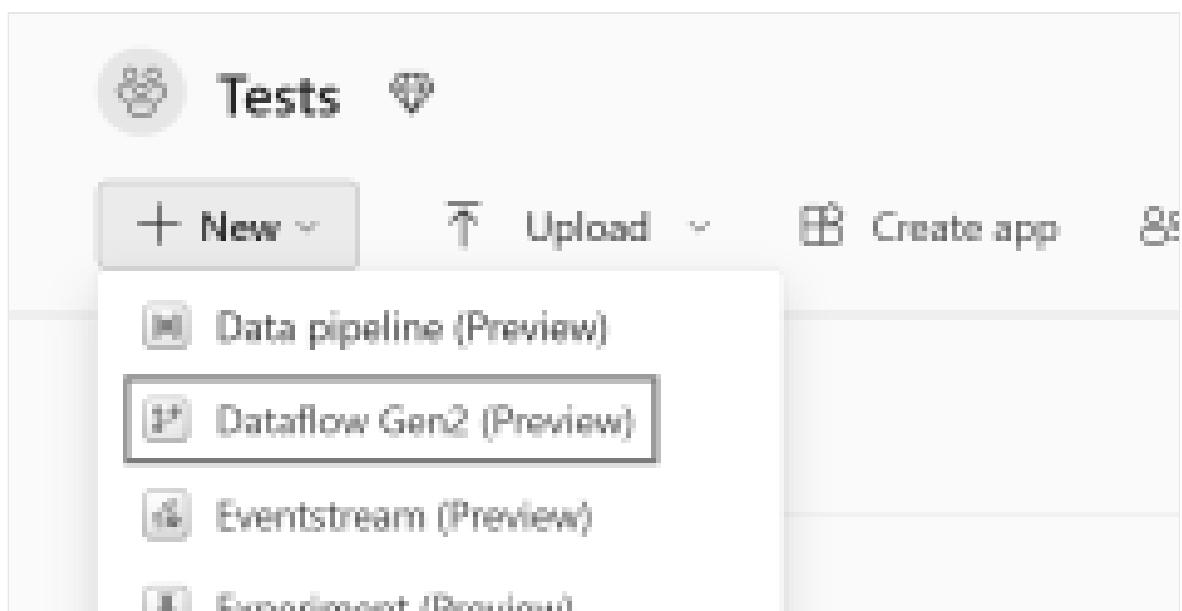
1. From your Power BI or Data Factory workspace, select the ellipsis next to an existing Dataflow Gen1 and then select **Edit**.



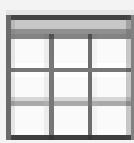
2. In Power Query, select the query or queries you want to copy. If there are multiple queries or folders (called *groups* in Power Query) you want to copy, select **Ctrl** as you select the queries or folders you want to copy. Then either select **Ctrl+C** or right-click in the selection and select **Copy**.



3. Open an existing Dataflow Gen2, or create a new Dataflow Gen2 in Data Factory.
To create a new dataflow, open an existing workflow and select **New > Dataflow Gen2**.



4. In the Power Query editor, select **Get data > Blank query**.

Home**Transform****Add column**Get
data ▾Enter
data

Options

Manage
parameters ▾

Excel workbook



Dataflows



SQL Server database



Text/CSV



Web page



Blank query

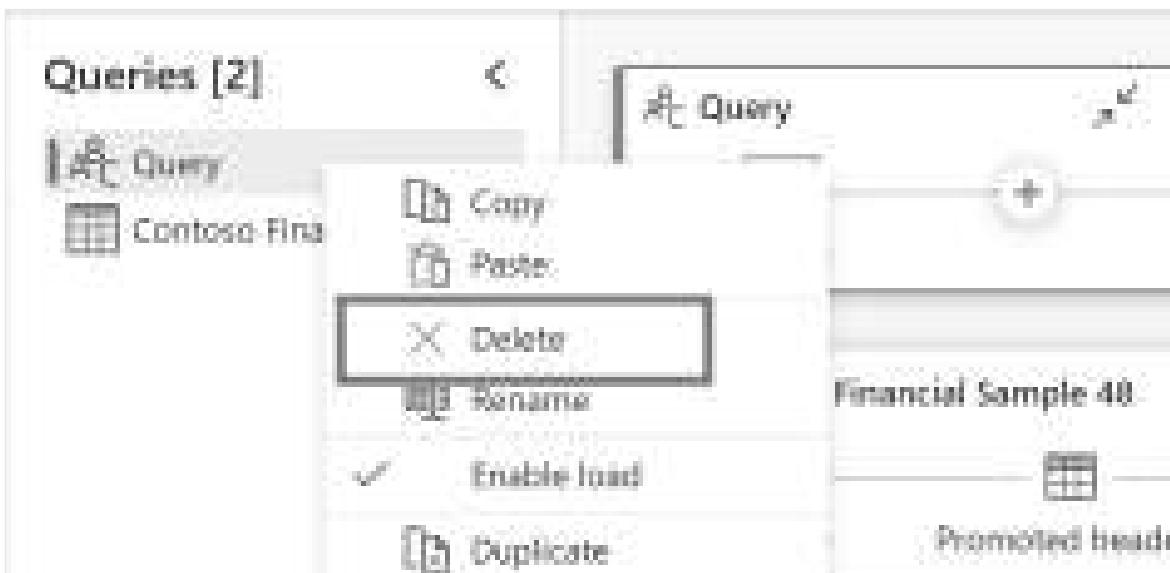
More...

Parameters

5. In the **Blank query** dialog box, select **Next**.
6. Select inside the **Queries** pane of the Power Query editor, and then select **Ctrl+V** to paste the query.
7. You might need to add your credentials before you can use the pasted queries. If a **Credentials are required to connect to the Web source.** message appears, select **Configure connection**, and then enter your credentials.



8. Select **Connect** to connect to your data.
9. Once you've connected to your data, right-click the initial blank query, and then select **Delete** to remove the empty query.



Next steps

- Differences between Dataflow Gen1 and Gen2 in Microsoft Fabric

Feedback

Was this page helpful?

Yes

No

Provide product feedback [↗](#) | Ask the community [↗](#)

Data Factory end-to-end scenario: introduction and architecture

Article • 11/15/2023

This tutorial helps you accelerate the evaluation process for Data Factory in Microsoft Fabric by providing a step-by-step guidance for a full data integration scenario within one hour. By the end of this tutorial, you understand the value and key capabilities of Data Factory and know how to complete a common end-to-end data integration scenario.

Overview: Why Data Factory in Microsoft Fabric?

This section helps you understand the role of Fabric generally, and the role Data Factory plays within it.

Understand the value of Microsoft Fabric

Microsoft Fabric provides a one-stop shop for all the analytical needs for every enterprise. It covers a complete spectrum of services including data movement, data lake, data engineering, data integration and data science, real time analytics, and business intelligence. With Fabric, there's no need to stitch together different services from multiple vendors. Instead, your users enjoy an end-to-end, highly integrated, single, and comprehensive product that is easy to understand, onboard, create, and operate.

Understand the value of Data Factory in Microsoft Fabric

Data Factory in Fabric combines the ease-of-use of Power Query with the scale and power of Azure Data Factory. It brings the best of both products together into a unified experience. The goal is to make sure Data Integration in Factory works well for both citizen and professional data developers. It provides low-code, AI-enabled data preparation and transformation experiences, petabyte-scale transformation, hundreds of connectors with hybrid, multicloud connectivity. Purview provides governance, and the service features enterprise scale Data/Op commitments, CI/CD, application lifecycle management, and monitoring.

Introduction - Understand three key features of Data Factory

- Data ingestion: The Copy activity in pipelines lets you move petabyte-scale data from hundreds of data sources into your data Lakehouse for further processing.
- Data transformation and preparation: Dataflow Gen2 provides a low-code interface for transforming your data using 300+ data transformations, with the ability to load the transformed results into multiple destinations such as Azure SQL databases, Lakehouse, and more.
- End-to-end integration flow automation: Pipelines provide orchestration of activities that include Copy, Dataflow, and Notebook activities, and more. This lets you manage activities all in one place. Activities in a pipeline can be chained together to operate sequentially, or they can operate independently in parallel.

In this end-to-end data integration use case, you learn:

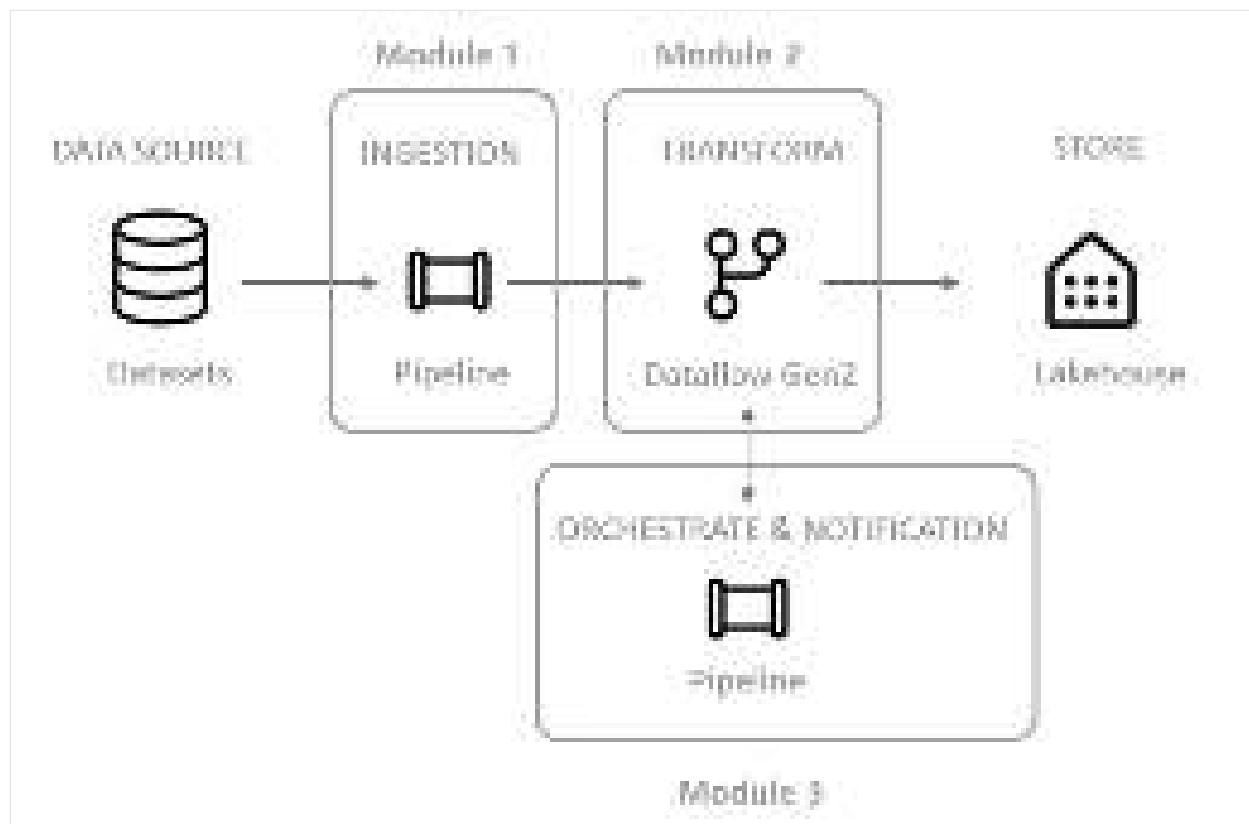
- How to ingest data using the copy assistant in a pipeline
- How to transform the data using a dataflow either with a no-code experience, or by writing your own code to process the data with a Script or Notebook activity
- How to automate the entire end-to-end data integration flow using a pipeline with triggers and flexible control flow activities.

Architecture

In the next 50 minutes, you're tasked with completing an end-to-end data integration scenario. This includes ingesting raw data from a source store into the Bronze table of a Lakehouse, processing all the data, moving it to the Gold table of the data Lakehouse, sending an email to notify you once all the jobs are complete, and finally, setting up the entire flow to run on a scheduled basis.

The scenario is divided into three modules:

- Module 1: Create a pipeline with Data Factory to ingest raw data from a Blob storage to a Bronze table in a data Lakehouse.
- Module 2: Transform data with a dataflow in Data Factory to process the raw data from your Bronze table and move it to a Gold table in the data Lakehouse.
- Module 3: Complete your first data integration journey to send an email to notify you once all the jobs are complete, and finally, setup the entire flow to run on a scheduled basis.



You use the sample dataset **NYC-Taxi** as the data source for the tutorial. After you finish, you'll be able to gain insight into daily discounts on taxi fares for a specific period of time using Data Factory in Microsoft Fabric.

Next steps

In this introduction to our end-to-end tutorial for your first data integration using Data Factory in Microsoft Fabric, you learned:

- ✓ The value and role of Microsoft Fabric
- ✓ The value and role of Data Factory in Fabric
- ✓ Key features of Data Factory
- ✓ What you will learn in this tutorial

Continue to the next section now to create your data pipeline.

Module 1: Create a pipeline with Data Factory

Feedback

Was this page helpful?

Yes

No

Provide product feedback | Ask the community

Module 1: Create a pipeline with Data Factory

Article • 11/15/2023

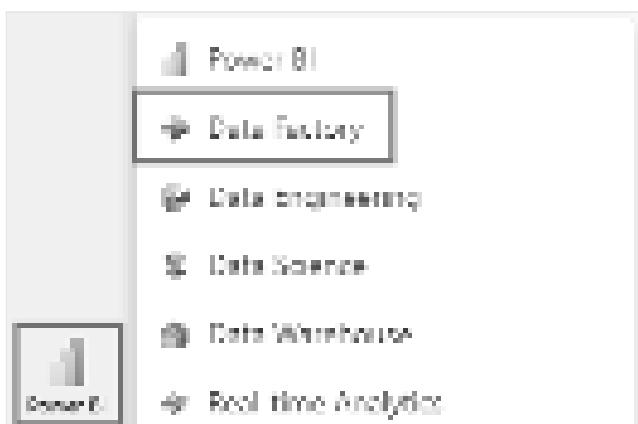
This module takes 10 minutes, ingesting raw data from the source store into the Bronze table of a data Lakehouse using the Copy activity in a pipeline.

The high-level steps in module 1 are as follows:

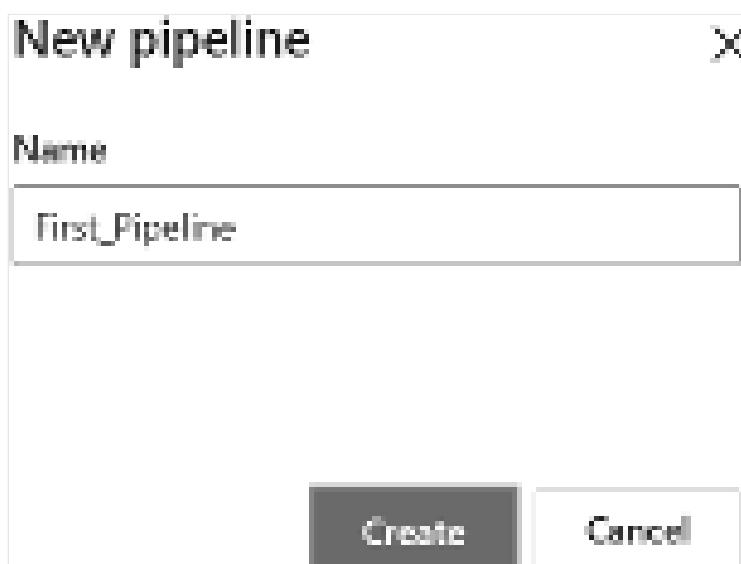
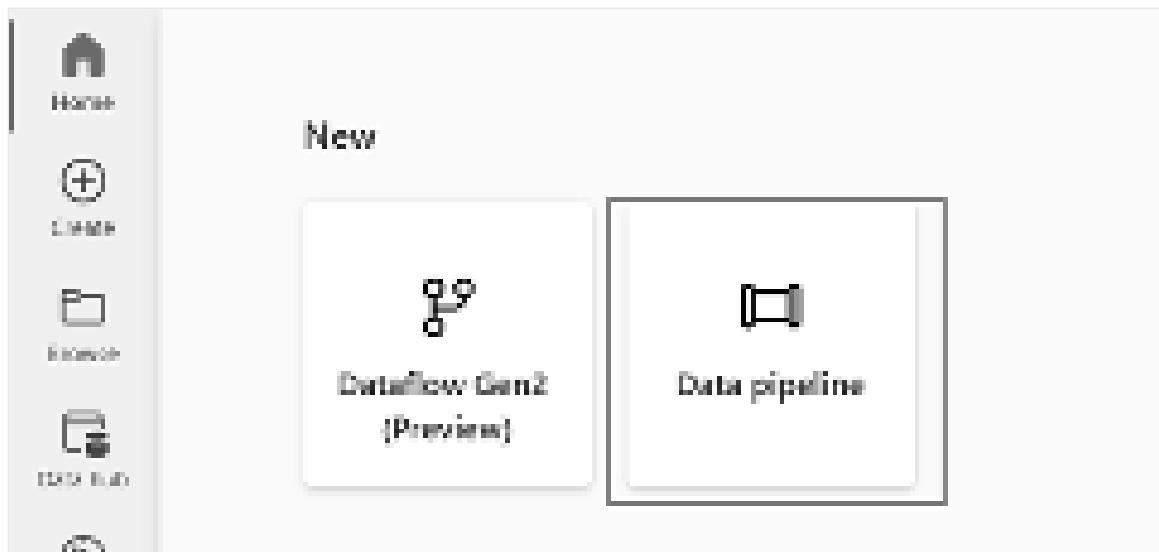
1. Create a data pipeline.
2. Use a Copy Activity in the pipeline to load sample data into a data Lakehouse.

Create a data pipeline

1. A Microsoft Fabric tenant account with an active subscription is required. Create a free account [↗](#).
2. Make sure you have a Microsoft Fabric enabled Workspace: Create a workspace.
3. Sign into Power BI [↗](#).
4. Select the default Power BI icon at the bottom left of the screen, and switch to the **Data Factory** experience.



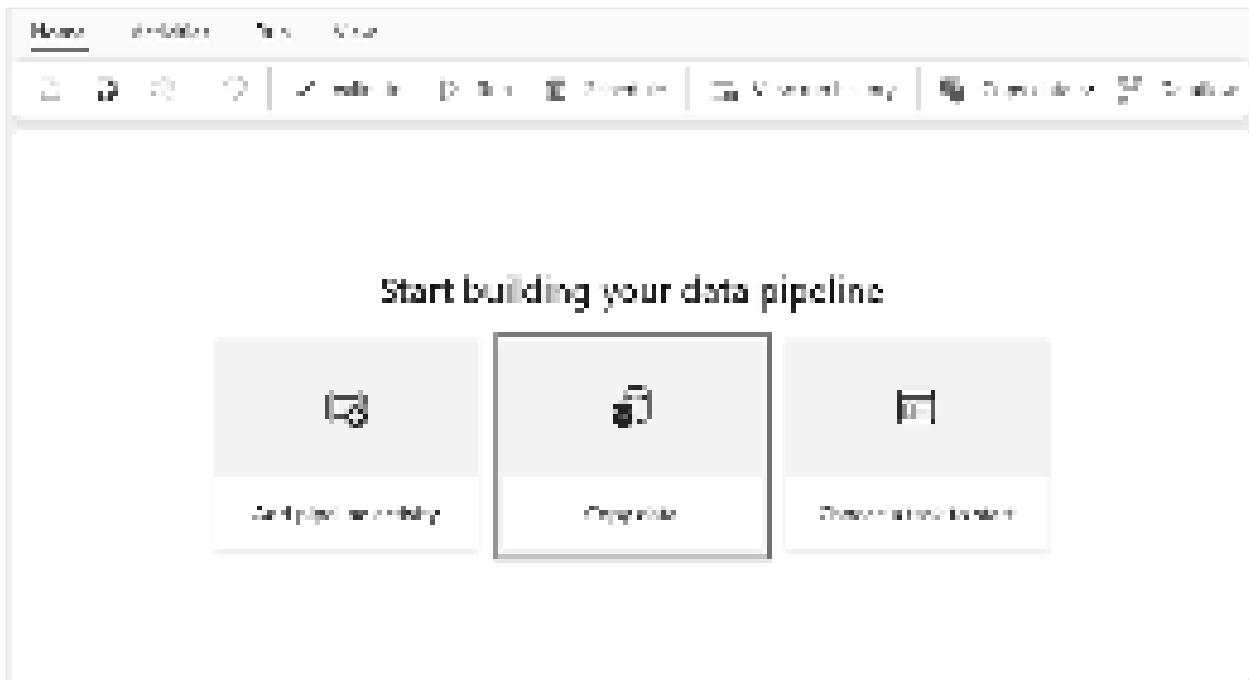
5. Select **Data pipeline** and provide a pipeline name. Then select **Create**.



Use a Copy activity in the pipeline to load sample data to a data Lakehouse

Step 1: Use the copy assistant to configure a copy activity.

Select **Copy data** to open the copy assistant tool.



Step 2: Configure your settings in the copy assistant.

1. The **Copy data** dialog is displayed with the first step, **Choose data source**, highlighted. Scroll down if necessary to the **Data sources** section, and select the **Azure Blob Storage** data source type. Then select **Next**.



2. In the next step, select **Create new connection** and then provide the URL for the blob storage hosting the sample data provided for this tutorial, at <https://nyctaxisample.blob.core.windows.net/sample>. The authentication kind is **Anonymous**. Select **Next** after providing the URL.



3. The **Connect to data source** step appears, and initially, you see an error **Unable to list files**, because permissions have only been granted to the **sample** folder in the blob storage. Provide the folder name, **sample**, and select **Retry**.



(!) Note

The blob storage folder is case sensitive and should be in all lower case.

4. The blob storage browser appears next. Select the **NYC-Taxi-Green-2015-01.parquet** file, and wait for the data preview to appear. Then select **Next**.



5. For the **Choose data destination** step of the copy assistant, select **Lakehouse** and then **Next**.

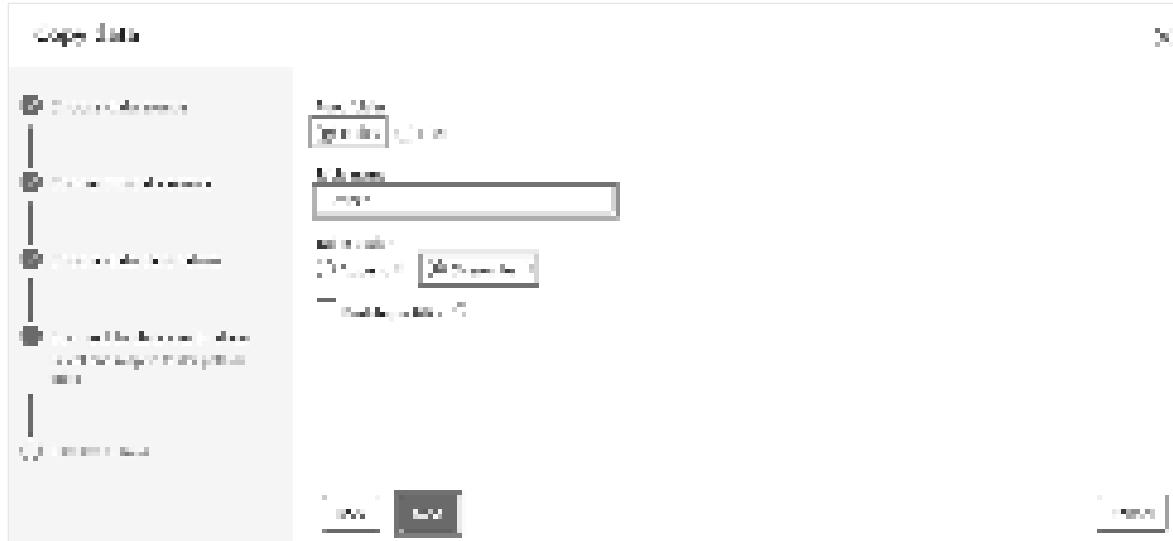


6. Select **Create new Lakehouse** on the data destination configuration page that appears, and enter a name for the new Lakehouse. Then select **Next** again.



7. Now configure the details of your Lakehouse destination on the **Select and map to folder path or table**. page. Select **Tables** for the **Root folder**, provide a table name,

and choose the **Overwrite** action. Don't check the **Enable partition** checkbox that appears after you select the **Overwrite** table action.

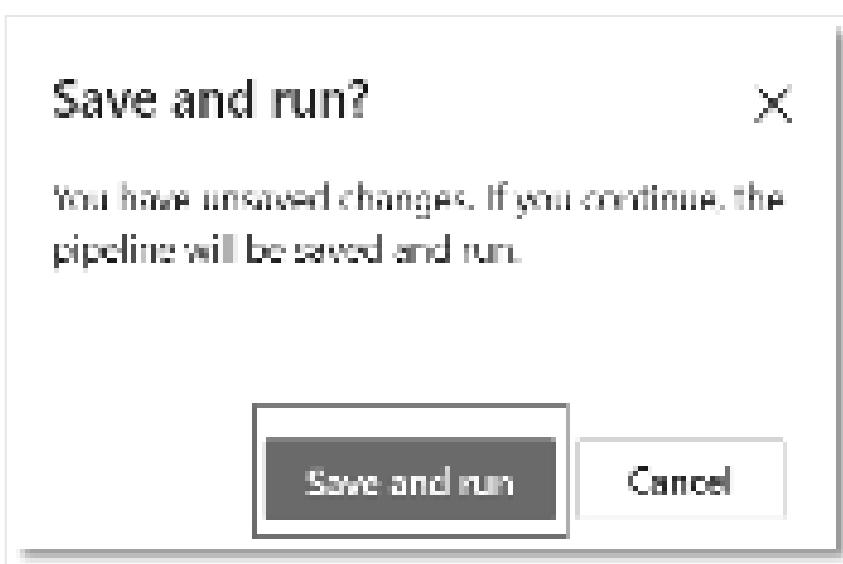
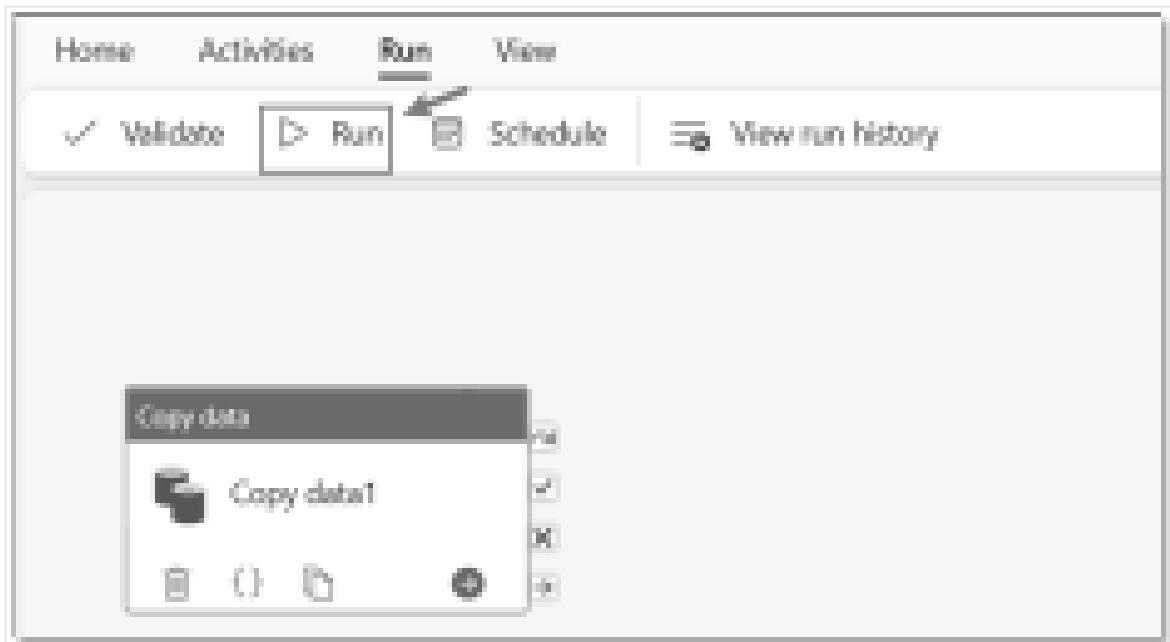


- Finally, on the **Review + save** page of the copy data assistant, review the configuration. For this tutorial, uncheck the **Start data transfer immediately** checkbox, since we run the activity manually in the next step. Then select **OK**.



Step 3: Run and view the results of your Copy activity.

- Select the **Run** tab in the pipeline editor. Then select the **Run** button, and then **Save and run** at the prompt, to run the Copy activity.



2. You can monitor the run and check the results on the **Output** tab below the pipeline canvas. Select the run details button (the "glasses" icon that appears when you hover over the running pipeline run) to view the run details.

Run ID	Run Type	Run Status	Run Details
Copy data1	Run	Succeeded	View details

3. The run details show 1,508,501 rows read and written.

Copy data details

Copy detail

Source	Destination
 Azure Blob Storage	 Lakehouse
Data read: 1,185 GB	Data written: 37,977 MB
Files read: 1	Files written: 1
Rows read: 1,506,501	Rows written: 1,506,501

States: Success | Succeeded

Start time: 4/24/2023, 4:22:58 PM

Pipeline run activity ID:	1750629-415b-4938-97d3-59e04094cf52
Throughput:	23.896 MB/s
Total duration:	00:00:56

[Duration breakdown](#)

[Advanced](#)

[Close](#)

4. Expand the **Duration breakdown** section to see the duration of each stage of the Copy activity. After reviewing the copy details, select **Close**.

Copy data details

Start time	4/26/2023, 4:22:58 PM
Pipeline run activity ID	17506209-1d5b-4938-97d3-59ec40f4cf52
Throughput	23.698 MB/s
Total duration	00:00:56

Duration breakdown

Start time	4/26/2023, 4:22:59 PM
Used DBs	4
Used parallel copies	1
Queue	[Progress Bar]
Transfer	[Progress Bar]
Reading from...	[Progress Bar]
Writing to sink	[Progress Bar]

> Advanced

Next steps

In this first module to our end-to-end tutorial for your first data integration using Data Factory in Microsoft Fabric, you learned how to:

- ✓ Create a data pipeline.
- ✓ Add a Copy activity to your pipeline.
- ✓ Use sample data and create a data Lakehouse to store the data to a new table.
- ✓ Run the pipeline and view its details and duration breakdown.

Continue to the next section now to create your dataflow.

Module 2: Transform data with a dataflow in Data Factory

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Module 2: Transform data with a dataflow in Data Factory

Article • 11/15/2023

This module takes about 25 minutes to create a dataflow, apply transformations, and move the raw data from the Bronze table into a Gold Lakehouse table.

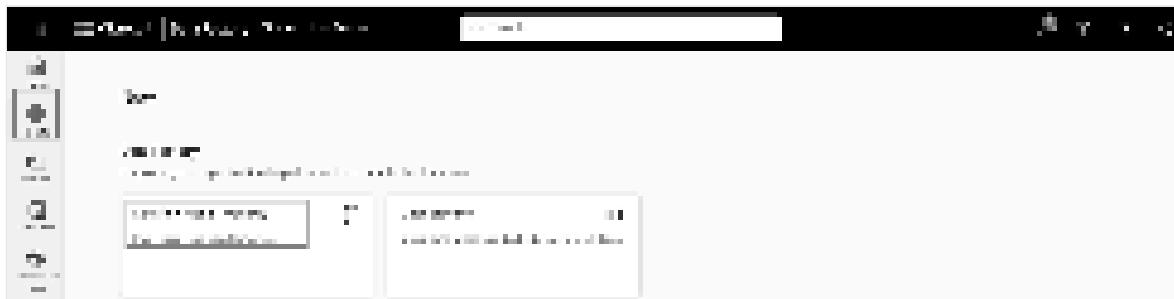
With the raw data loaded into your Bronze Lakehouse table from the last module, you can now prepare that data and enrich it by combining it with another table that contains discounts for each vendor and their trips during a particular day. This final Gold Lakehouse table is loaded and ready for consumption.

The high-level steps in the dataflow are as follows:

- Get raw data from the Lakehouse table created by the Copy activity in Module 1: Create a pipeline with Data Factory.
- Transform the data imported from the Lakehouse table.
- Connect to a CSV file containing discounts data.
- Transform the discounts data.
- Combine trips and discounts data.
- Load the output query into the Gold Lakehouse table.

Get data from a Lakehouse table

1. From the sidebar, select **Create**, and then **Dataflow Gen2** to create a new dataflow gen2.



2. From the new dataflow menu, select **Get data**, and then **More....**



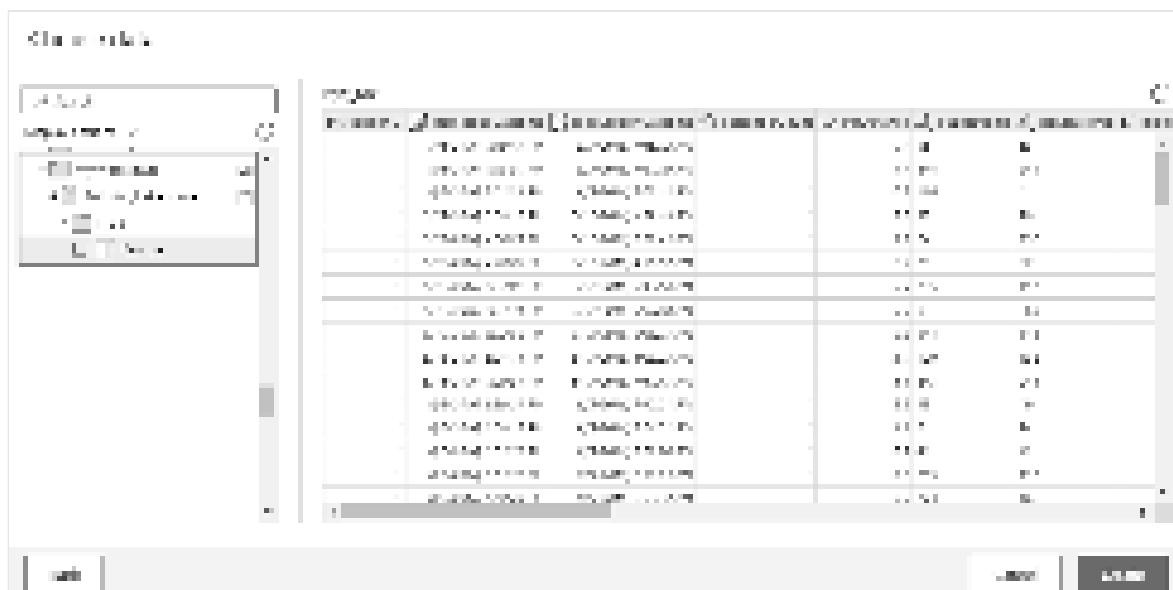
3. Search for and select the **Lakehouse** connector.



4. The **Connect to data source** dialog appears, and a new connection is automatically created for you based on the currently signed in user. Select **Next**.

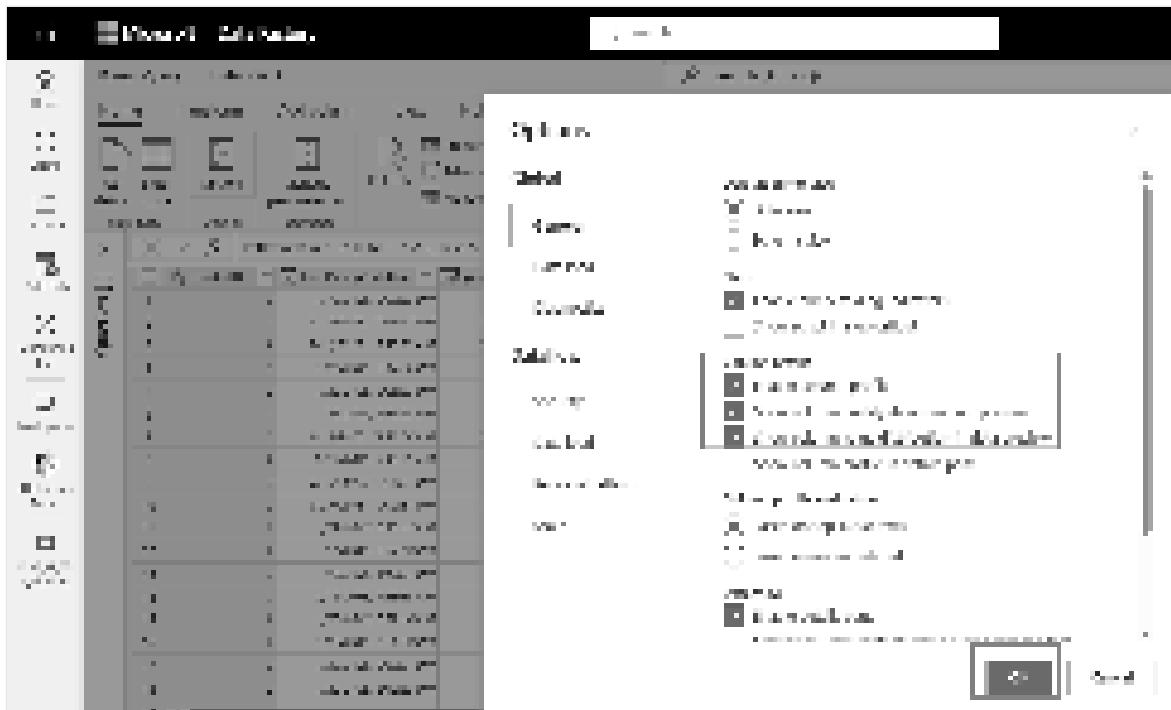


5. The **Choose data** dialog is displayed. Use the navigation pane to find the Lakehouse you created for the destination in the prior module, and select the **Tutorial_Lakehouse** data table.



6. *(Optional)* Once your canvas is populated with the data, you can set **column profile** information, as this is useful for data profiling. You can apply the right transformation and target the right data values based on it.

To do this, select **Options** from the ribbon pane, then select the first three options under **Column profile**, and then select **OK**.



Transform the data imported from the Lakehouse

1. Select the data type icon in the column header of the second column, **IcepPickupDatetime**, to display a dropdown menu and select the data type from the menu to convert the column from the **Date/Time** to **Date** type.

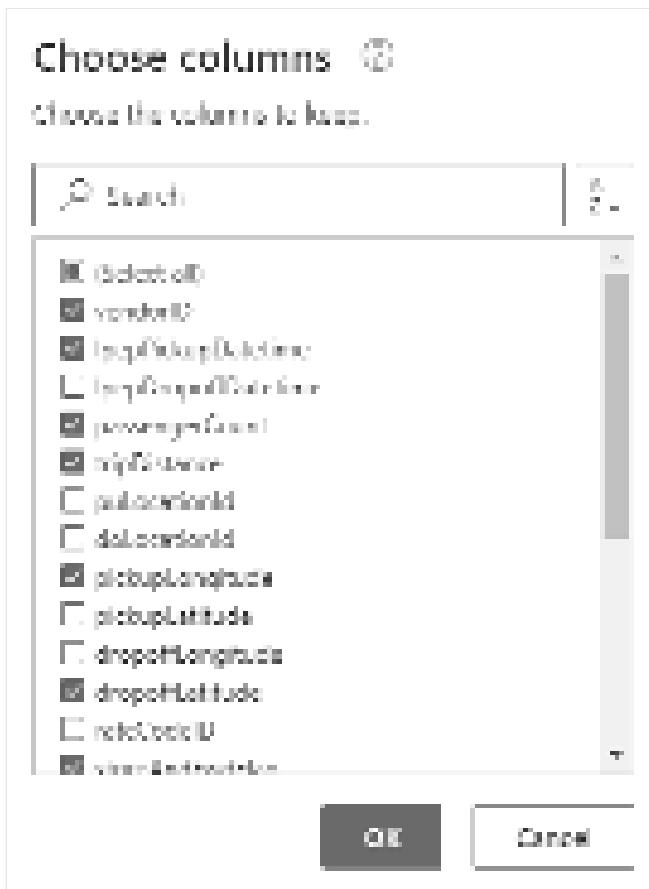
A screenshot of the Microsoft Power BI ribbon interface. The 'Data Type' section of the ribbon is highlighted. A dropdown menu is open under 'Date', showing various options like 'Decimal number', 'Currency', 'Whole number', and 'Percentage'. The 'Date' option is currently selected, indicated by a yellow background and a checked checkbox icon. To the right of the ribbon, there is a preview pane showing a table with columns for 'Date' and 'Value'. The table contains several rows of date values such as '1/1/2015 12:04:07 PM' and '1/1/2015 12:05:07 PM'. The preview pane also shows summary statistics: 'Count: 988 unique' and 'Distinct: 8 unique'.

- 2. (Optional) On the Home tab of the ribbon, select the Choose columns option from the Manage columns group.*

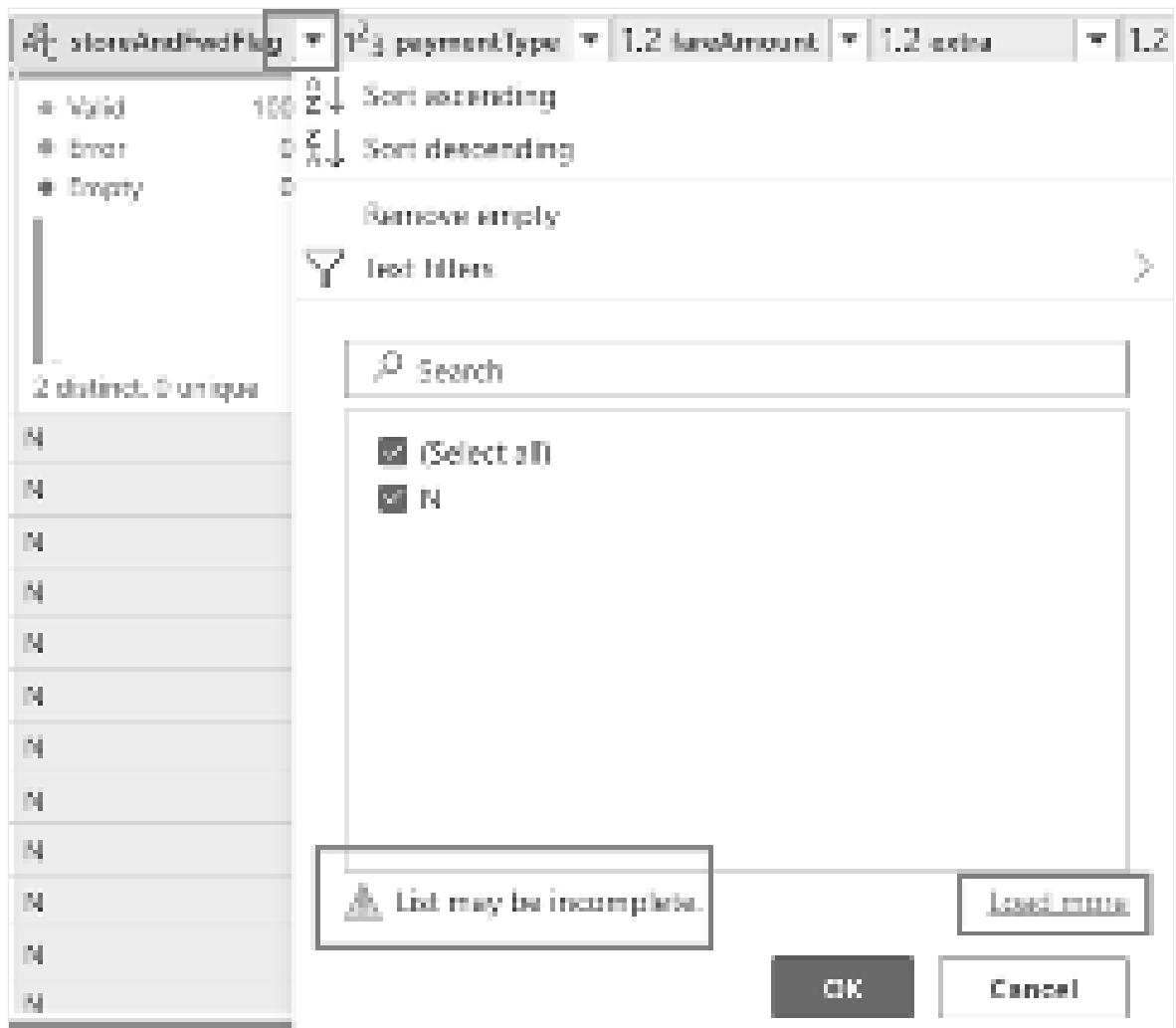


3. (Optional) On the **Choose columns** dialog, deselect some columns listed here, then select **OK**.

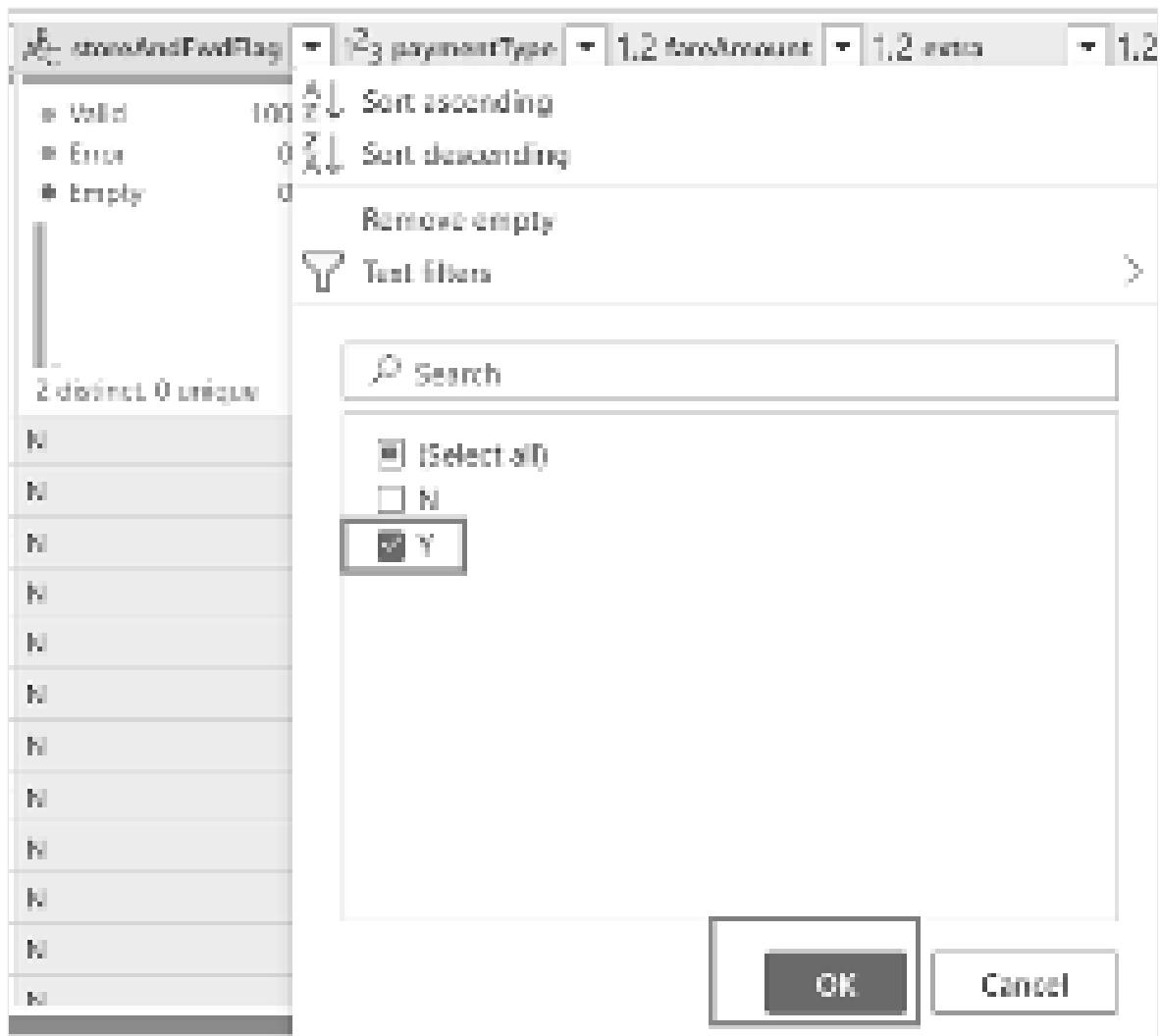
- lpepDropoffDatetime
- puLocationId
- doLocationId
- pickupLatitude
- dropoffLongitude
- rateCodeID



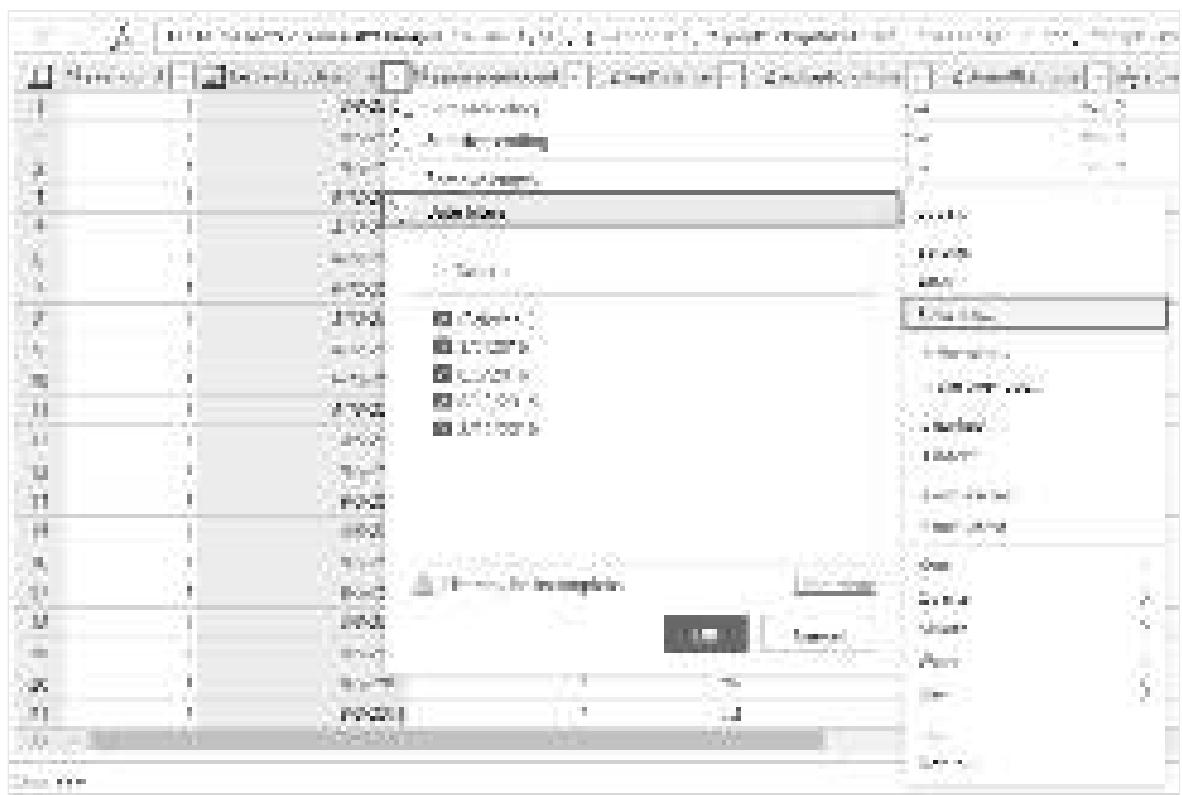
4. Select the **storeAndFwdFlag** column's filter and sort dropdown menu. (If you see a warning **List may be incomplete**, select **Load more** to see all the data.)



5. Select 'Y' to show only rows where a discount was applied, and then select **OK**.



6. Select the **Icep_Pickup_Datetime** column sort and filter dropdown menu, then select **Date filters**, and choose the **Between...** filter provided for Date and Date/Time types.



7. In the **Filter rows** dialog, select dates between January 1, 2015, and January 31, 2015, then select **OK**.



Connect to a CSV file containing discount data

Now, with the data from the trips in place, we want to load the data that contains the respective discounts for each day and VendorID, and prepare the data before combining it with the trips data.

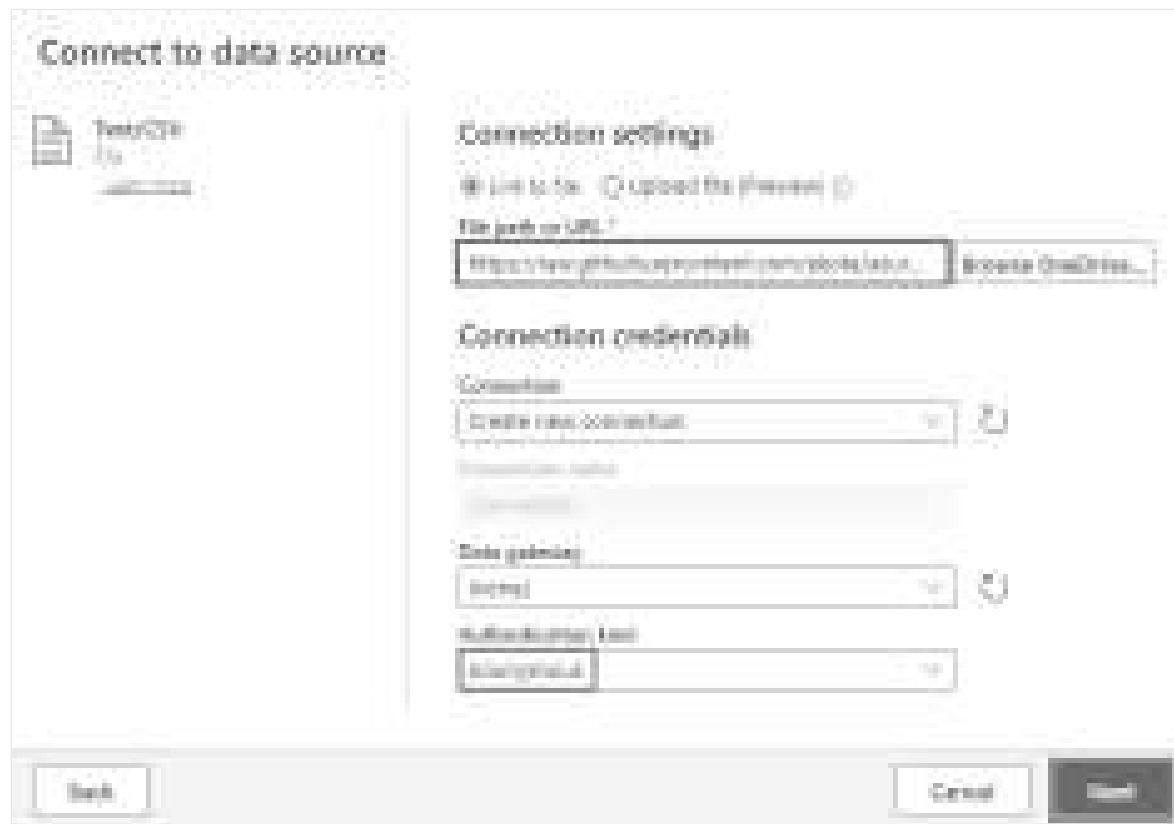
1. From the **Home** tab in the dataflow editor menu, select the **Get data** option, and then choose **Text/CSV**.



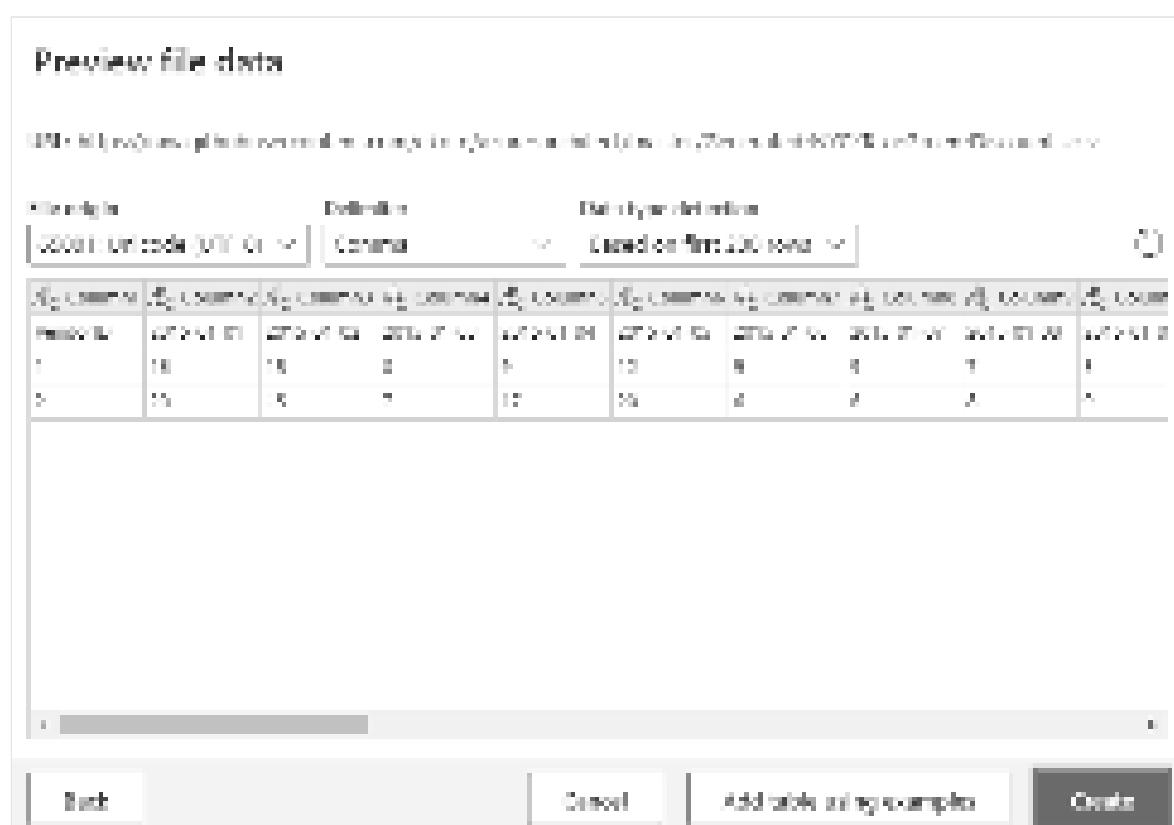
2. On the **Connect to data source** dialog, provide the following details:

- **File path or URL** - <https://raw.githubusercontent.com/ekote/azure-architect/master/Generated-NYC-Taxi-Green-Discounts.csv>
- **Authentication kind** - Anonymous

Then select **Next**.



3. On the Preview file data dialog, select Create.



Transform the discount data

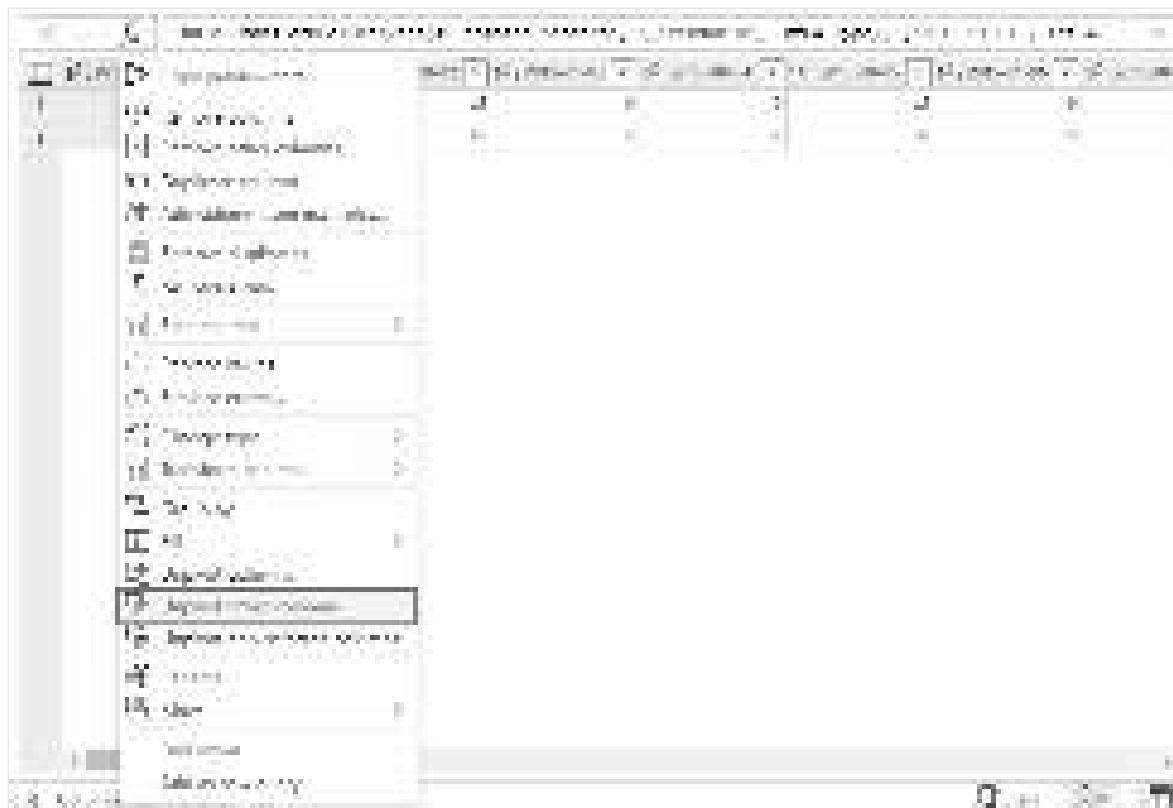
1. Reviewing the data, we see the headers appear to be in the first row. Promote them to headers by selecting the table's context menu at the top left of the preview grid area to select **Use first row as headers**.



! Note

After promoting the headers, you can see a new step added to the **Applied steps** pane at the top of the dataflow editor to the data types of your columns.

2. Right-click the **VendorID** column, and from the context menu displayed, select the option **Unpivot other columns**. This allows you to transform columns into attribute-value pairs, where columns become rows.



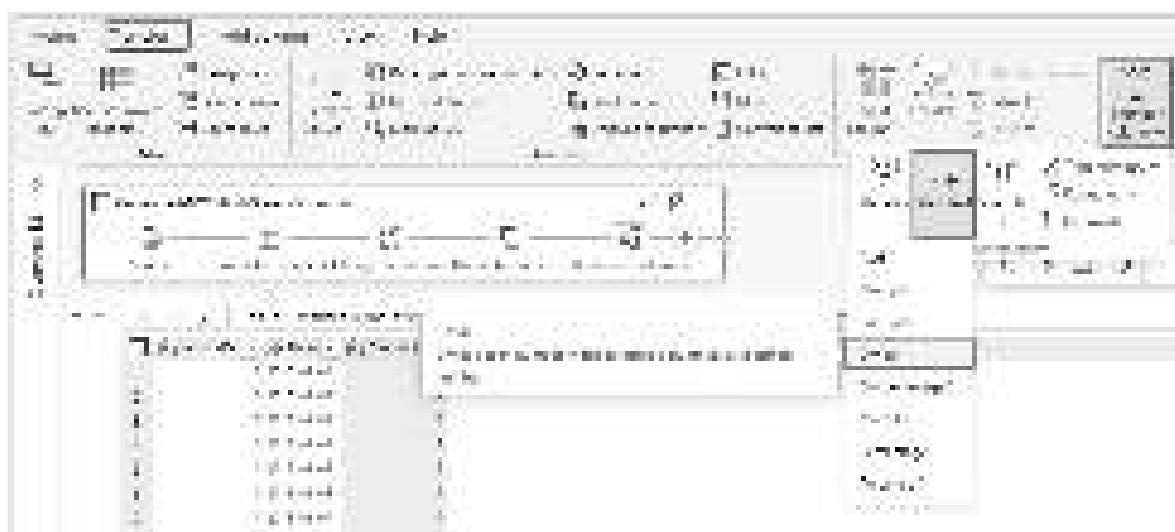
3. With the table unpivoted, rename the **Attribute** and **Value** columns by double-clicking them and changing **Attribute** to **Date** and **Value** to **Discount**.

	Attribute	Value
1	2015-01-01	10
2	2015-01-02	15
3	2015-01-03	0
4	2015-01-04	2
5	2015-01-05	17
6	2015-01-06	2
7	2015-01-07	2
8	2015-01-08	7

4. Change the data type of the Date column by selecting the data type menu to the left of the column name and choosing Date.



5. Select the **Discount** column and then select the **Transform** tab on the menu. Select **Number column**, and then select **Standard numeric transformations** from the submenu, and choose **Divide**.



6. On the **Divide** dialog, enter the value 100.



Combine trips and discounts data

The next step is to combine both tables into a single table that has the discount that should be applied to the trip, and the adjusted total.

1. First, toggle the **Diagram view** button so you can see both of your queries.

The screenshot shows the Power BI Desktop interface. At the top, the ribbon has tabs: Home, Transform, Add column, View, and Help. Under the Transform tab, there are icons for 'Get data', 'New query', 'Create model', 'Data', 'Text', 'Image', and 'Import from database'. Below the ribbon, there's a sidebar with 'Queries' and a 'Diagram view' button. The main workspace contains two queries: 'Discounts' and 'nyc_taxi'. The 'nyc_taxi' query is currently selected, showing its data grid. The data grid has columns: 'id', 'date', and 'total'. The first few rows of data are:

id	date	total
1	2011-01-01	0.00
2	2011-01-01	0.00
3	2011-01-01	0.00
4	2011-01-01	0.00
5	2011-01-01	0.00
6	2011-01-01	0.00
7	2011-01-01	0.00
8	2011-01-01	0.00
9	2011-01-01	0.00
10	2011-01-01	0.00

2. Select the **nyc_taxi** query, and on the **Home** tab, Select the **Combine** menu and choose **Merge queries**, then **Merge queries as new**.

The screenshot shows the Microsoft Power BI Data Editor interface. At the top, there's a ribbon with tabs like Home, Transform, Add columns, View, and Help. Below the ribbon, there are several icons for data operations such as Select, Sort, Filter, Merge, and Remove. A 'Merge' button is highlighted with a red box. In the center, there are two tables: 'Fare Details' (with columns like VendorID, PULocationID, and RateID) and 'Generated-NYC-Taxi-Green-Discounts' (with columns like VendorID, RateID, and Date). On the right side, there's a 'Merge' dialog box with three buttons: 'Merge without', 'Merge quickly', and 'Merge columns automatically'. The main workspace shows the two tables side-by-side.

3. On the **Merge** dialog, select **Generated-NYC-Taxi-Green-Discounts** from the **Right table for merge** drop down, and then select the "light bulb" icon on the top right of the dialog to see the suggested mapping of columns between the two tables.

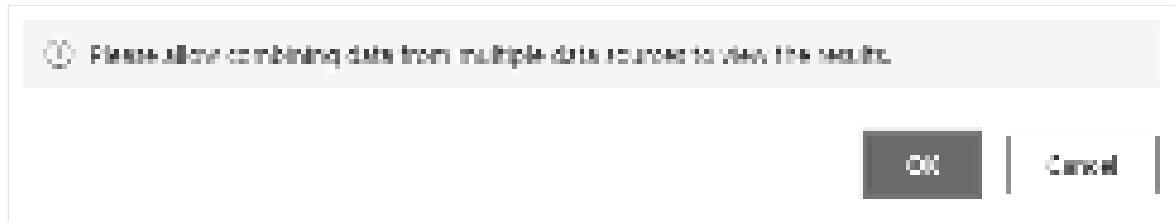
This screenshot shows the 'Merge' dialog box. The 'Left table for merge' dropdown is set to 'Fare Details'. The 'Right table for merge' dropdown is set to 'Generated-NYC-Taxi-Green-Discounts'. The 'Suggested' tab is selected. It displays two sets of column mappings:

- From table: Fare Details** (VendorID, PULocationID, RateID) mapped to **To table: Generated-NYC-Taxi-Green-Discounts** (VendorID, RateID, Date).
- From table: Generated-NYC-Taxi-Green-Discounts** (VendorID, RateID, Date) mapped to **To table: Fare Details** (VendorID, PULocationID, RateID).

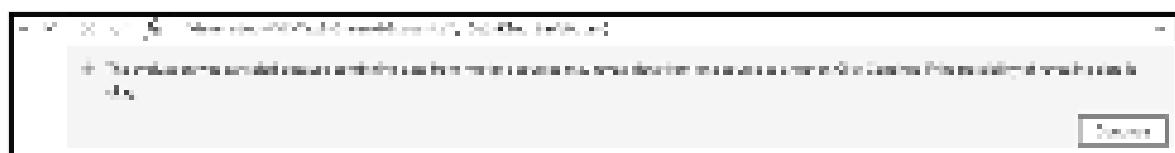
Below the mappings, there are six circular icons representing different merge operations: 'Merge without', 'Merge quickly', 'Merge columns automatically', 'Merge rows automatically', 'Merge rows quickly', and 'Merge rows without'. There are also two checkboxes at the bottom: 'Always matching the previous row' and 'Priority matching by date'.

Choose each of the two suggested column mappings, one at a time, mapping the VendorID and date columns from both tables. When both mappings are added, the matched column headers are highlighted in each table.

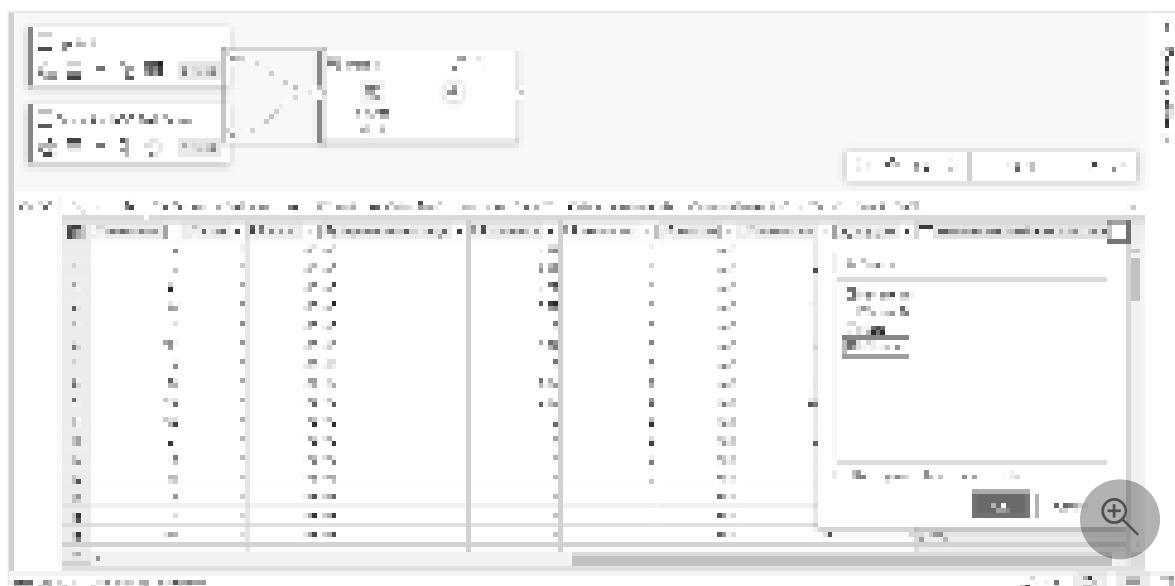
4. A message is shown asking you to allow combining data from multiple data sources to view the results. Select **OK** on the Merge dialog.



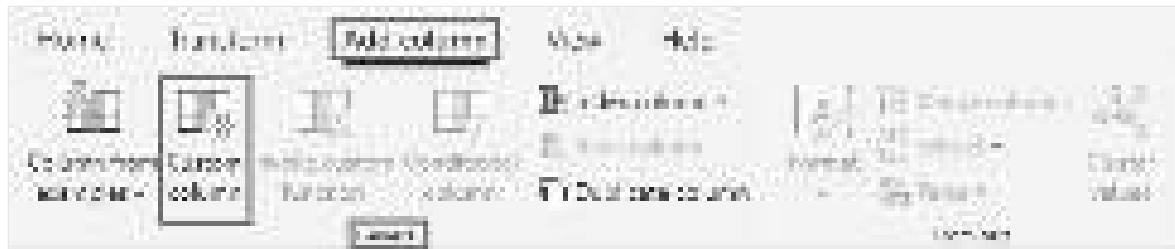
5. In the table area, you'll initially see a warning that "The evaluation was canceled because combining data from multiple sources may reveal data from one source to another. Select continue if the possibility of revealing data is okay." Select **Continue** to display the combined data.



6. Notice how a new query was created in Diagram view showing the relationship of the new Merge query with the two queries you previously created. Looking at the table pane of the editor, scroll to the right of the Merge query column list to see a new column with table values is present. This is the "Generated NYC Taxi-Green-Discounts" column, and its type is **[Table]**. In the column header there's an icon with two arrows going in opposite directions, allowing you to select columns from the table. Deselect all of the columns except **Discount**, and then select **OK**.



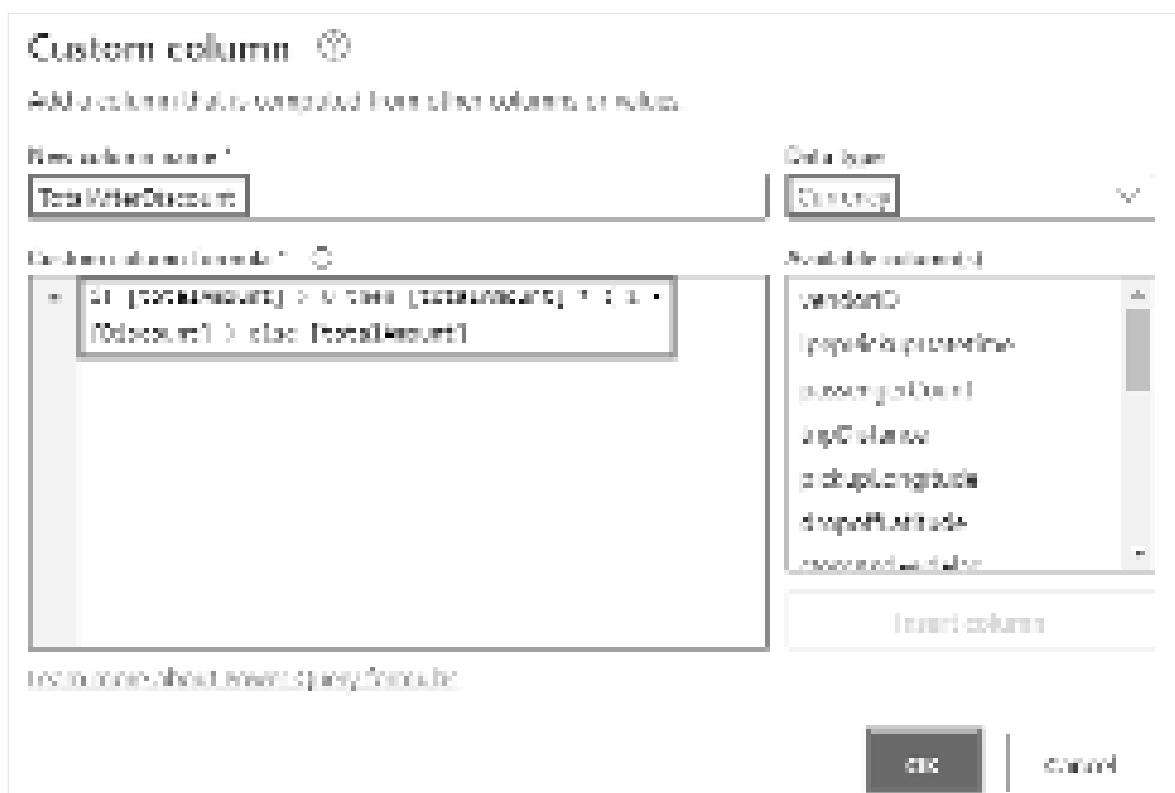
7. With the discount value now at the row level, we can create a new column to calculate the total amount after discount. To do so, select the **Add column** tab at the top of the editor, and choose **Custom column** from the **General** group.



8. On the **Custom column** dialog, you can use the Power Query formula language (also known as M) to define how your new column should be calculated. Enter *TotalAfterDiscount* for the **New column name**, select **Currency** for the **Data type**, and provide the following M expression for the **Custom column formula**:

```
if [totalAmount] > 0 then [totalAmount] * ( 1 -[Discount] ) else [totalAmount]
```

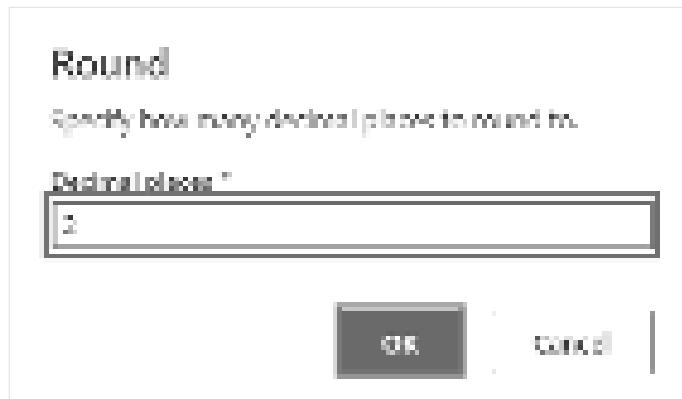
Then select **OK**.



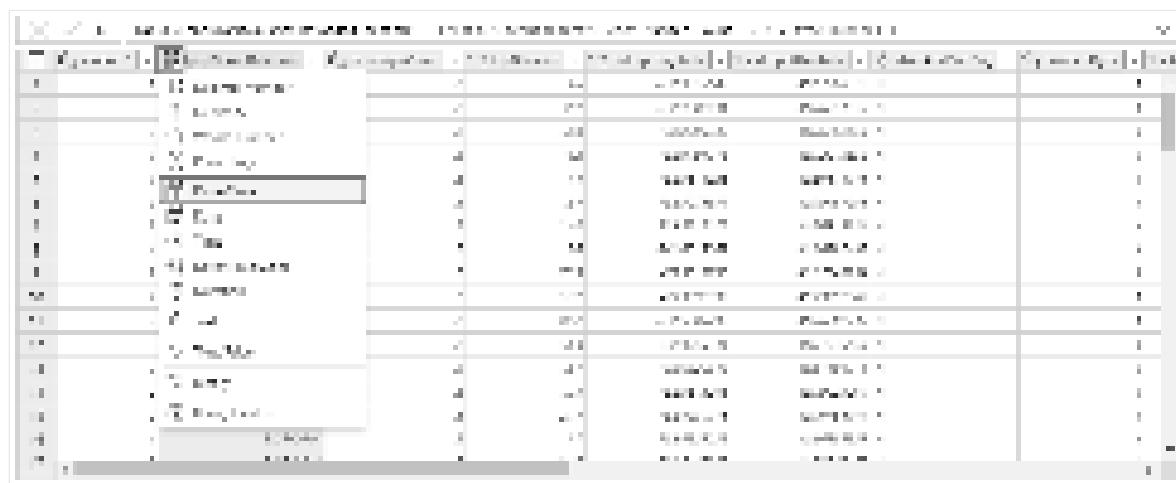
9. Select the newly create **TotalAfterDiscount** column and then select the **Transform** tab at the top of the editor window. On the **Number column group**, select the **Rounding** drop down and then choose **Round....**



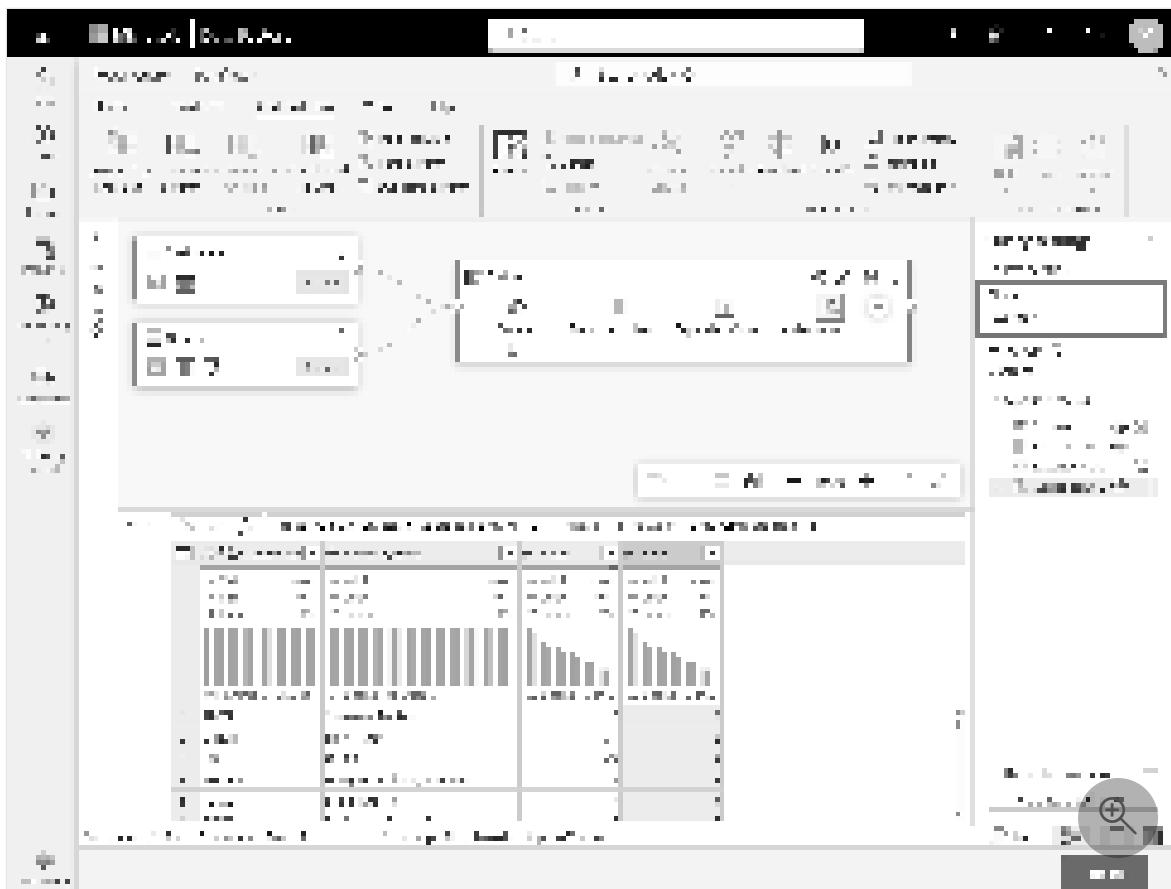
10. On the Round dialog, enter 2 for the number of decimal places and then select OK.



11. Change the data type of the lpepPickupDatetime from Date to Date/Time.



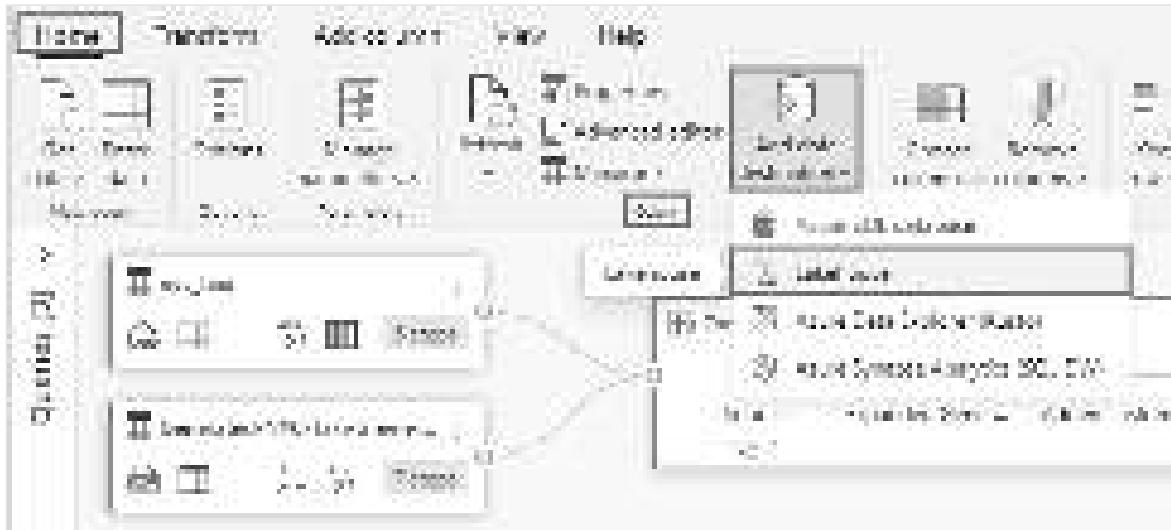
12. Finally, expand the **Query settings** pane from the right side of the editor if it isn't already expanded, and rename the query from **Merge to Output**.



Load the output query to a table in the Lakehouse

With the output query now fully prepared and with data ready to output, we can define the output destination for the query.

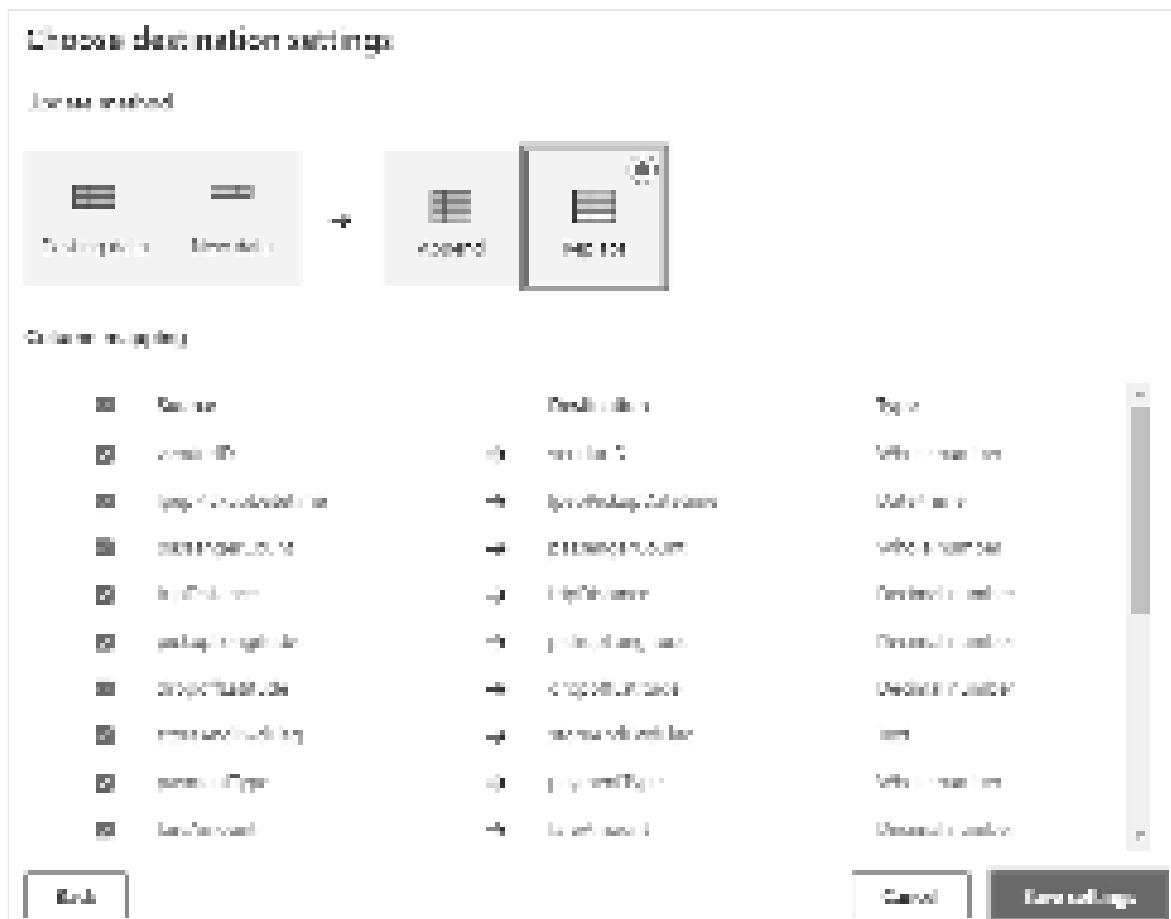
1. Select the **Output** merge query created previously. Then select the **Home** tab in the editor, and **Add data destination** from the **Query** grouping, to select a **Lakehouse** destination.



2. On the **Connect to data destination** dialog, your connection should already be selected. Select **Next** to continue.
3. On the **Choose destination target** dialog, browse to the Lakehouse where you wish to load the data and name the new table *nyc_taxi_with_discounts*, then select **Next** again.



4. On the **Choose destination settings** dialog, leave the default Replace update method, double check that your columns are mapped correctly, and select **Save settings**.



5. Back in the main editor window, confirm that you see your output destination on the **Query settings** pane for the **Output** table, and then select **Publish**.

ⓘ Important

When the first Dataflow Gen2 is created in a workspace, Lakehouse and Warehouse items are provisioned along with their related SQL analytics endpoint and semantic models. These items are shared by all dataflows in the workspace and are required for Dataflow Gen2 to operate, shouldn't be deleted, and aren't intended to be used directly by users. The items are an implementation detail of Dataflow Gen2. The items aren't visible in the workspace, but might be accessible in other experiences such as the Notebook, SQL-endpoint, Lakehouse, and Warehouse experiences. You can recognize the items by their prefix in the name. The prefix of the items is 'DataflowsStaging'.

6. *(Optional)* On the workspace page, you can rename your dataflow by selecting the ellipsis to the right of the dataflow name that appears after you select the row, and choosing **Properties**.

The screenshot shows the 'My workspace' interface. At the top, there are buttons for 'New', 'Upload', and a search bar labeled 'Filter by keyword'. Below is a table listing three dataflows:

#	Name	Type	Owner
1	Dataflow1	Data pipeline	Shared
2	Dataflow2	Dataflow Gen2	Shared
3	Dataflow3		

A context menu is open over the 'Dataflow1' row, with the following options:

- Delete
- Edit
- Export JSON
- Properties** (highlighted)
- Refresh history
- Settings
- View lineage

7. Select the refresh icon for the dataflow after selecting its row, and when complete, you should see your new Lakehouse table created as configured in the **Data destination** settings.

	Name	Type	Owner	Last modified
11	query1	Table: query1	Team Lakehouse	—
12	Dataflow 1	Pipeline: Dataflow	Team Lakehouse	—
13	Initial query	Table: Initial query	Team Lakehouse	—
14	new_query_in_lakehouse	Table: new_query_in_lakehouse	Team Lakehouse	—

8. Check your Lakehouse to view the new table loaded there.

Next steps

In this second module to our end-to-end tutorial for your first data integration using Data Factory in Microsoft Fabric, you learned how to:

- ✓ Create a new Dataflow Gen2.
- ✓ Import and transform sample data.
- ✓ Import and transform text/CSV data.
- ✓ Merge data from both data sources into a new query.
- ✓ Transform data and generate new columns in a query.
- ✓ Configure an output destination source for a query.
- ✓ Rename and refresh your new dataflow.

Continue to the next section now to integrate your data pipeline.

Module 3: Automate and send notifications with Data Factory

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Module 3: Automate and send notifications with Data Factory

Article • 11/15/2023

You'll complete this module in 10 minutes to send an email notifying you when all the jobs in a pipeline are complete, and configure it to run on a scheduled basis.

In this module you learn how to:

- Add an Office 365 Outlook activity to send the output of a Copy activity by email.
- Add schedule to run the pipeline.
- *(Optional)* Add a dataflow activity into the same pipeline.

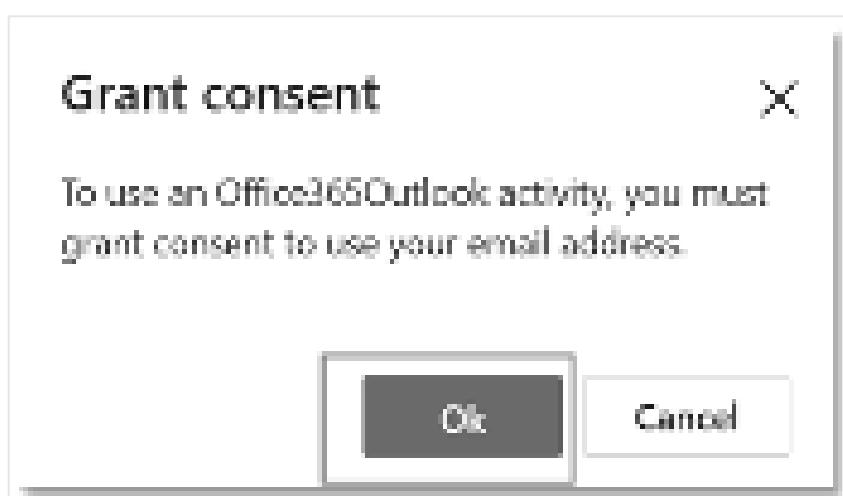
Add an Office 365 Outlook activity to your pipeline

We use the pipeline you created in Module 1: Create a pipeline in Data Factory.

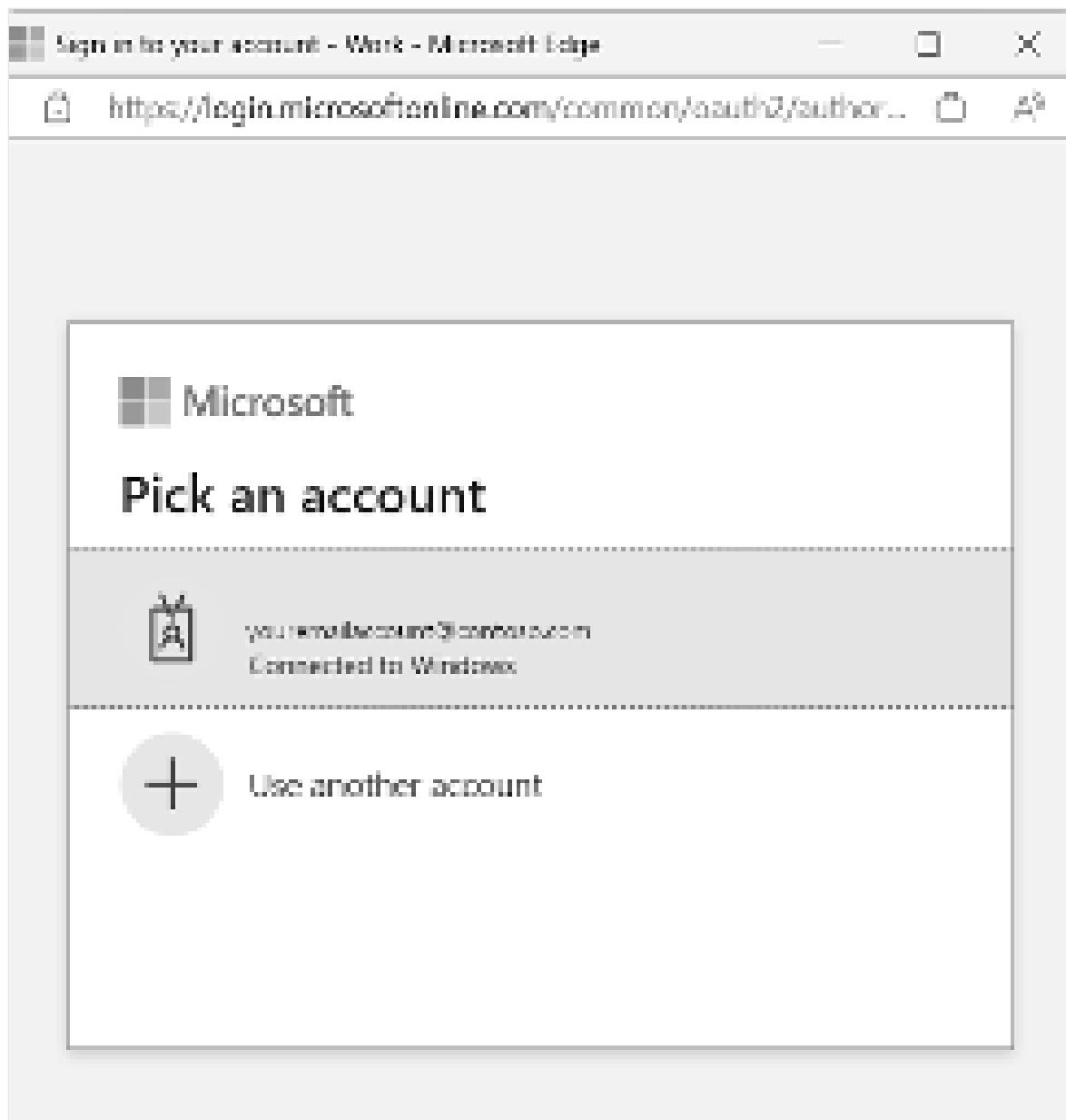
1. Select the **Activities** tab in the pipeline editor and find the Office Outlook activity.



2. Select **OK** to grant consent to use your email address.



3. Select the email address you want to use.



ⓘ Note

The service doesn't currently support personal email. You must use an enterprise email address.

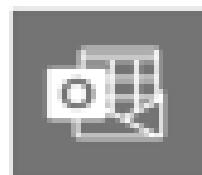
4. Select **Allow access** to confirm.



Microsoft

Confirmation required

You are about to provide access to



Office 365 Outlook

to a connection created by user 12145678-1214-1234-1234-121456789012.

[Allow access](#)

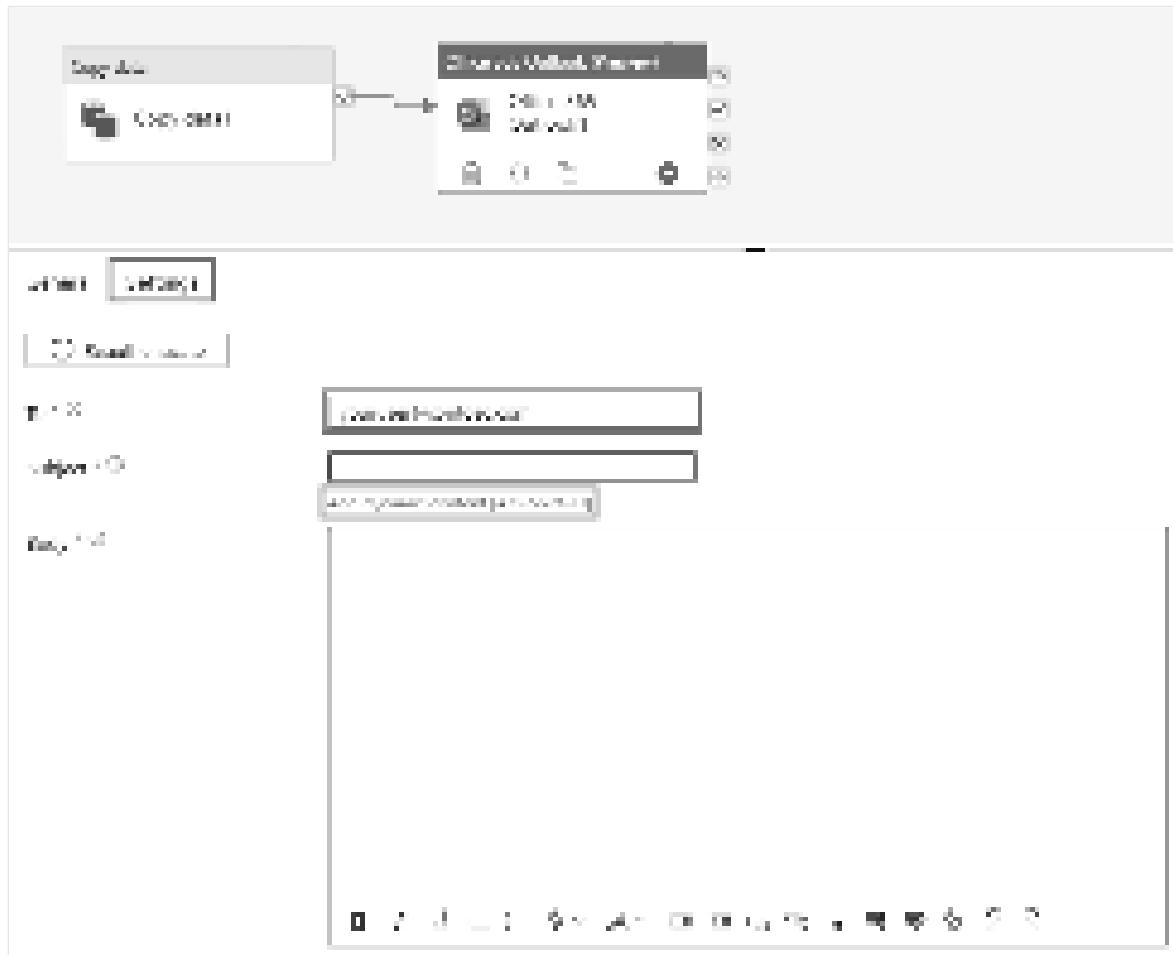
[Cancel](#)

5. Select and drag the **On success** path (a green checkbox on the top right side of the activity in the pipeline canvas) from your Copy activity to your new Office 365 Outlook activity.



6. Select the Office 365 Outlook activity from the pipeline canvas, then select the **Settings** tab of the property area below the canvas to configure the email.

- Enter your email address in the **To** section. If you want to use several addresses, use ; to separate them.
- For the **Subject**, select the field so that the **Add dynamic content** option appears, and then select it to display the pipeline expression builder canvas.



7. The **Pipeline expression builder** dialog appears. Enter the following expression, then select **OK**:

```
@concat('DI in an Hour Pipeline Succeeded with Pipeline Run Id', pipeline().RunId)
```

Pipeline expression builder

Add dynamic content here using any combination of expressions, functions and system variables.

Preview (0) In an Azure Pipeline associated with Pipeline Run ID: pipeline().RunId)

← Back to pipeline

Activity outputs

Parameters

System variables

Functions

Variables

Search

Copy result

Copy result activity output

Copy data1

Copy data1 pipeline value value (predicted)

OK

Cancel

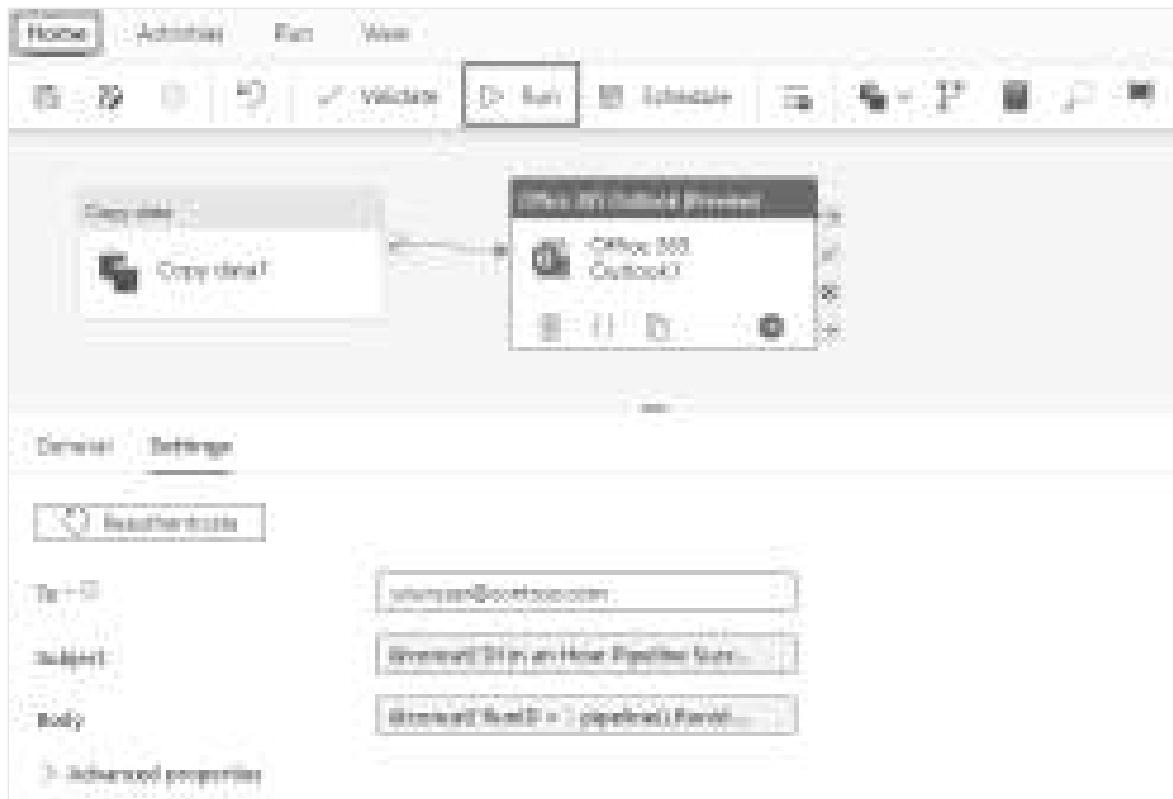
- For the **Body**, select the field again and choose the **Add dynamic content** option when it appears below the text area. Add the following expression again in the **Pipeline expression builder** dialog that appears, then select **OK**:

```
@concat('RunID = ', pipeline().RunId, ' ; ', 'Copied rows ', activity('Copy data1').output.rowsCopied, ' ; ', 'Throughput ', activity('Copy data1').output.throughput)
```

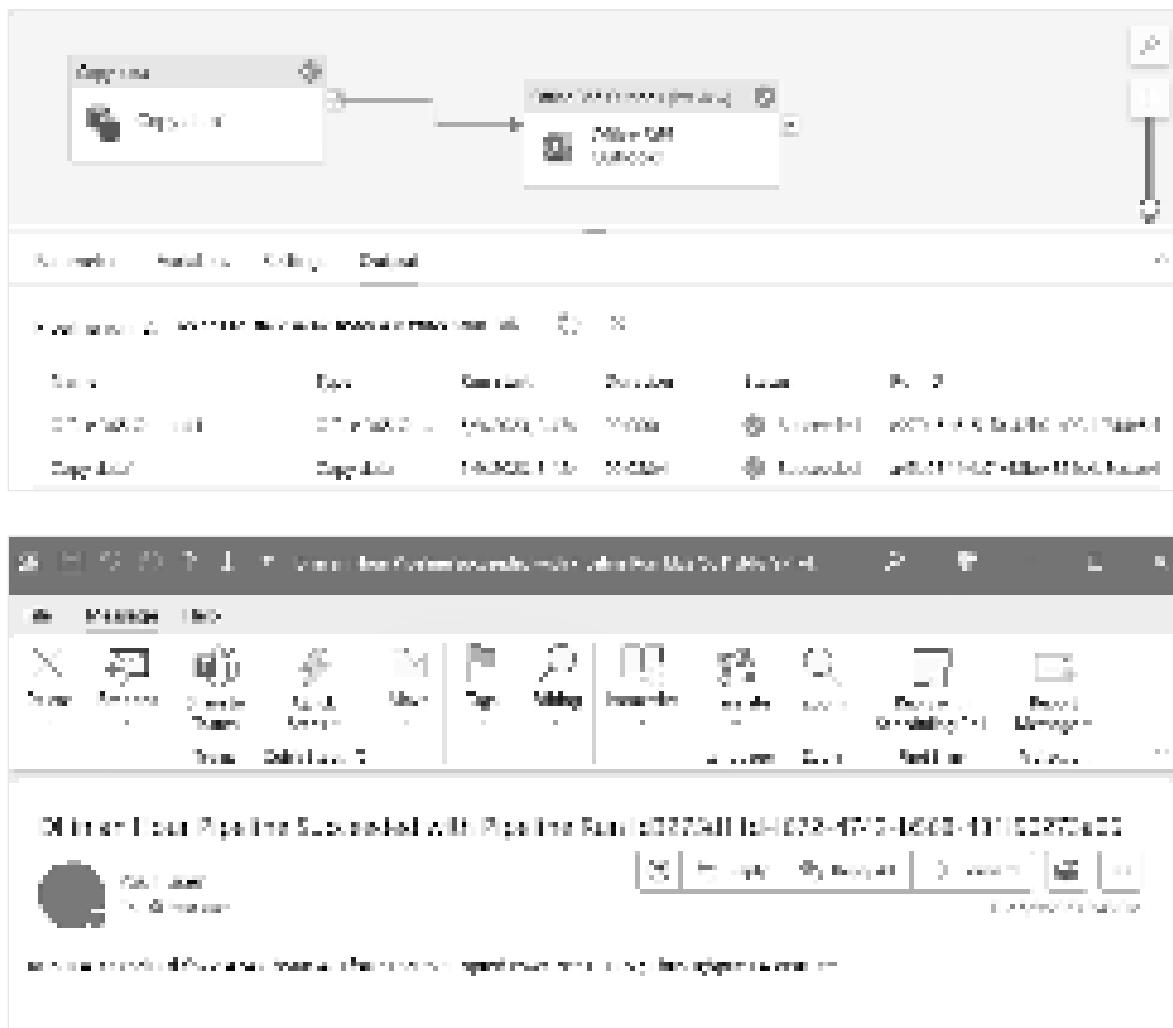
⚠ Note

Replace **Copy data1** with the name of your own pipeline copy activity.

- Finally select the **Home** tab at the top of the pipeline editor, and choose **Run**. Then select **Save and run** again on the confirmation dialog to execute these activities.



10. After the pipeline runs successfully, check your email to find the confirmation email sent from the pipeline.



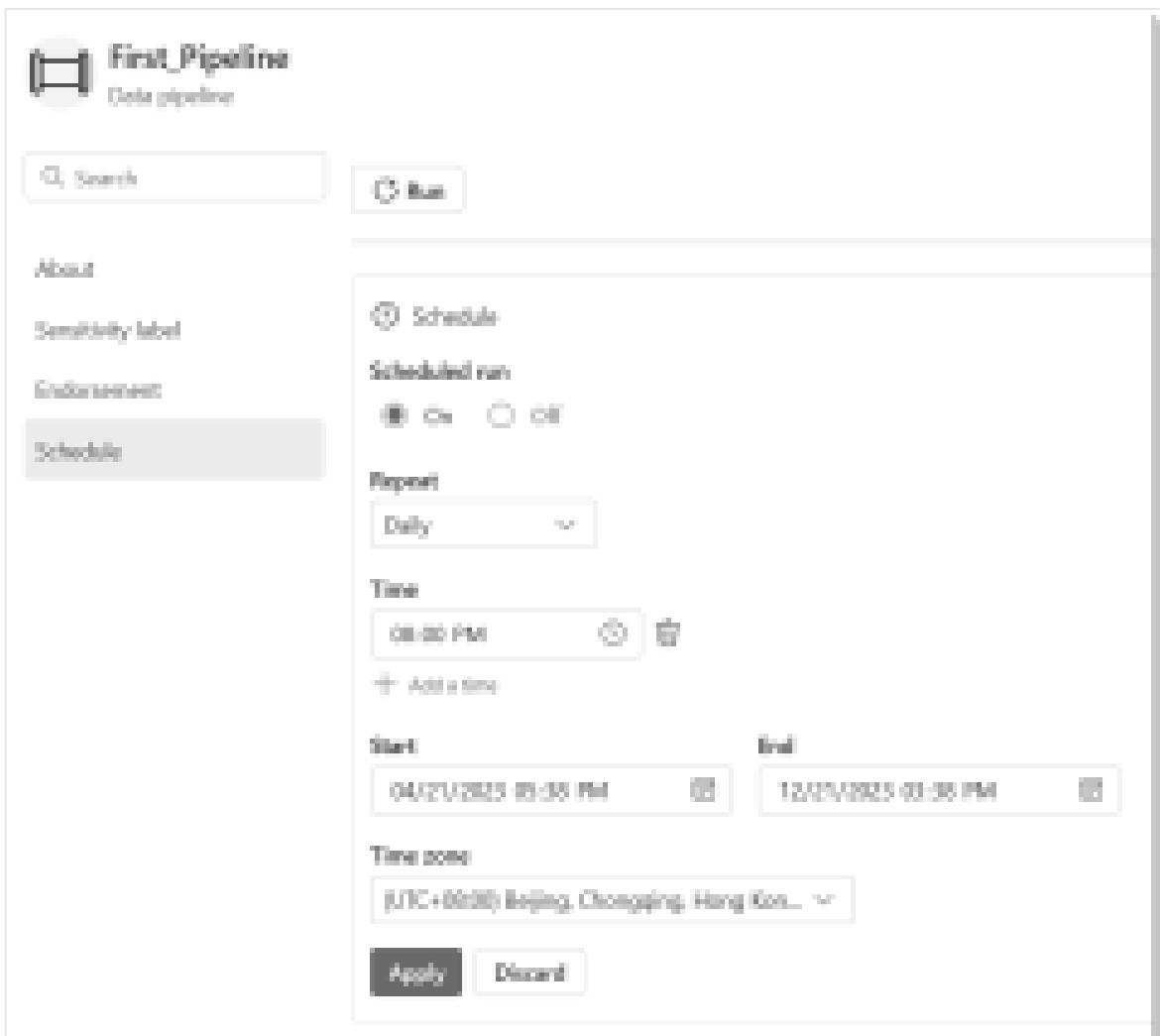
Schedule pipeline execution

Once you finish developing and testing your pipeline, you can schedule it to execute automatically.

1. On the **Home** tab of the pipeline editor window, select **Schedule**.



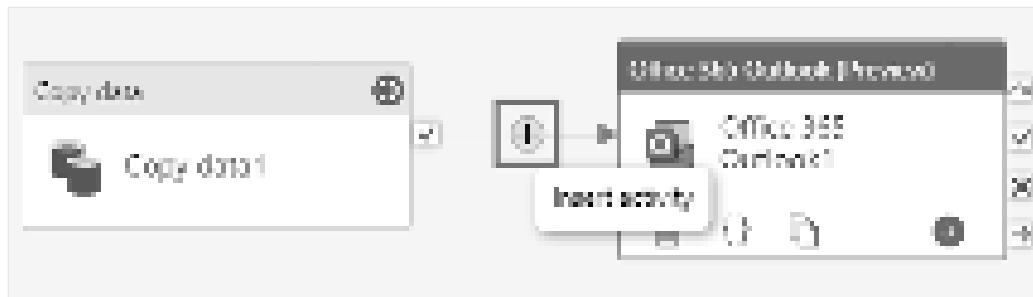
2. Configure the schedule as required. The example here schedules the pipeline to execute daily at 8:00 PM until the end of the year.



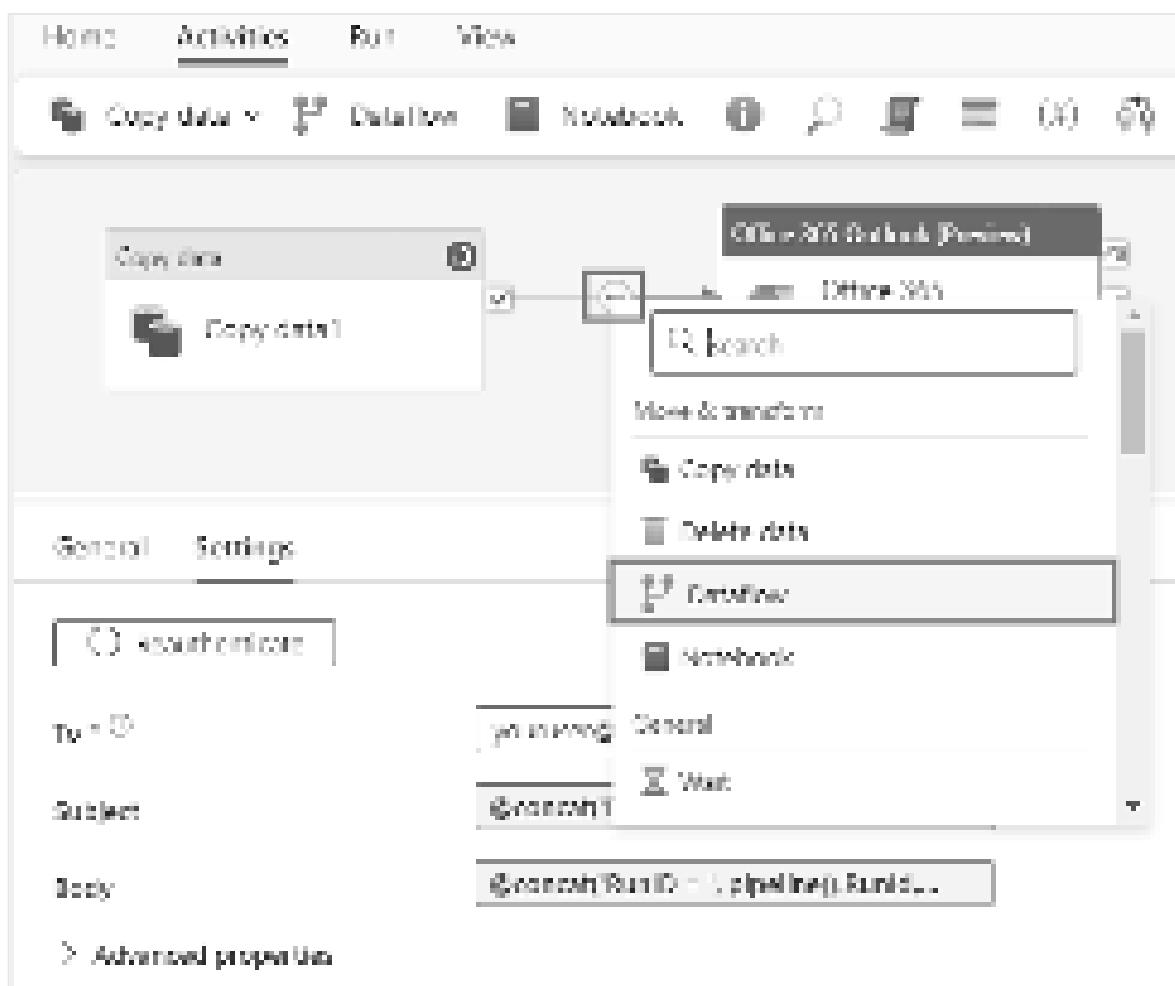
(Optional) Add a Dataflow activity to the pipeline

You can also add the dataflow you created in Module 2: Create a dataflow in Data Factory into the pipeline.

1. Hover over the green line connecting the Copy activity and the Office 365 Outlook activity on your pipeline canvas, and select the + button to insert a new activity.



2. Choose Dataflow from the menu that appears.



3. The newly created Dataflow activity is inserted between the Copy activity and the Office 365 Outlook activity, and selected automatically, showing its properties in

the area below the canvas. Select the **Settings** tab on the properties area, and then select your dataflow created in Module 2: Create a dataflow in Data Factory.



Next steps

In this third module to our end-to-end tutorial for your first data integration using Data Factory in Microsoft Fabric, you learned how to:

- ✓ Use a Copy activity to ingest raw data from a source store into a table in a data Lakehouse.
- ✓ Use a Dataflow activity to process the data and move it into a new table in the Lakehouse.
- ✓ Use an Office 365 Outlook activity to send an email notifying you once all the jobs are complete.
- ✓ Configure the pipeline to run on a scheduled basis.
- ✓ *(Optional)* Insert a Dataflow activity in an existing pipeline flow.

Now that you completed the tutorial, learn more about how to monitor pipeline runs:

[Monitor pipeline runs](#)

Feedback

Was this page helpful?

Yes

No

Provide product feedback | Ask the community

Connector overview

Article • 11/15/2023

Data Factory in Microsoft Fabric offers a rich set of connectors that allow you to connect to different types of data stores. You can take advantage of those connectors to transform data in dataflows or move a PB-level of dataset with high-scale in a data pipeline.

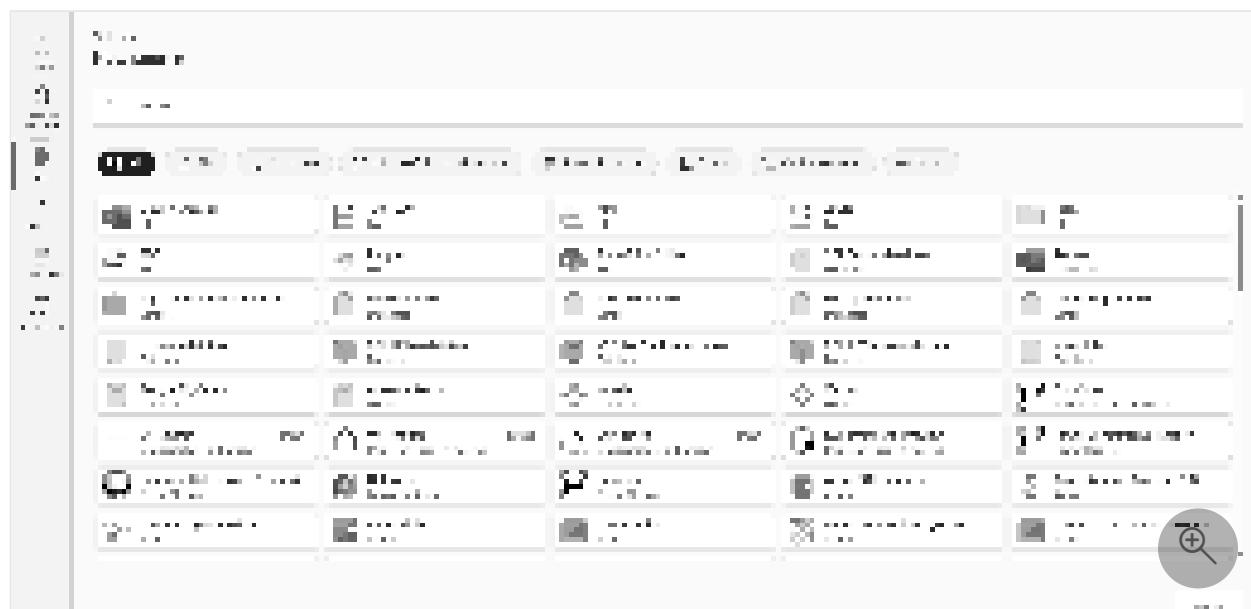
Prerequisites

Before you can set up a connection in Dataflow Gen2 or a data pipeline, the following prerequisites are required:

- A Microsoft Fabric tenant account with an active subscription. Create an account for free.
- A Microsoft Fabric enabled Workspace. Create a workspace.

Supported data connectors in dataflows

Dataflow Gen2 provide data ingestion and transformation capabilities over a wide range of data sources. These data sources include various types of files, databases, online, cloud, and on-premises data sources. There are greater than 145 different data connectors, which are accessible from the dataflows authoring experience within the get data experience.



For a comprehensive list of all currently supported data connectors, go to Dataflow Gen2 connectors in Microsoft Fabric.

The following connectors are currently available for output destinations in Dataflow Gen2:

- Azure Data Explorer
- Azure SQL
- Data Warehouse
- Lakehouse

Supported data stores in data pipeline

Data Factory in Microsoft Fabric supports data stores in a data pipeline through the Copy, Lookup, Get Metadata, Delete, Script, and Stored Procedure activities. For a list of all currently supported data connectors, go to Data pipeline connectors in Microsoft Fabric.

 **Note**

Currently, a pipeline on managed VNet and on-premises data access with a gateway aren't supported in Data Factory for Microsoft Fabric.

Next steps

- How to copy data using copy activity
- Data source management

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

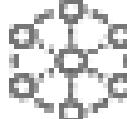
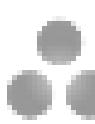
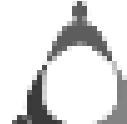
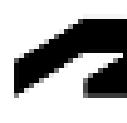
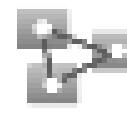
Dataflow Gen2 connectors in Microsoft Fabric

Article • 11/15/2023

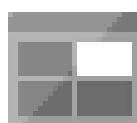
The following table contains a list of all the connectors currently available for Data Factory Dataflow Gen2 in Microsoft Fabric. For those connectors that have a reference page, a link is provided under the connector icon and name.

The connectors are listed in alphabetical order in separate tables for each letter in the alphabet. Use the **In this article** list at the top of this article to go to any of the alphabetized tables.

A

Connectors				
				
Access Database By Microsoft	Acterys (Beta) By Acterys	Actian (Beta) By Actian	Adobe Analytics By Microsoft	Amazon Athena By Amazon
				
Amazon OpenSearch Service (Beta) By Amazon	Amazon Redshift By Microsoft	Anaplan By Anaplan	Asana (Beta) By Asana	Assemble Views By Autodesk
				
AtScale cubes By Microsoft	Autodesk Construction Cloud By Autodesk	Automation Anywhere By Automation Anywhere	Automy Data Analytics (Beta) By ACEROYALTY	Azure Analysis Services database By Microsoft

Connectors



Azure Blob
Storage
By Microsoft



Azure Cosmos
DB v2 (Beta)
By Microsoft



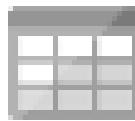
Azure Cost
Management
By Microsoft



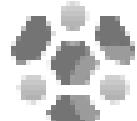
Azure
Databricks
By Databricks



Azure Data
Explorer
(Kusto)
By Microsoft



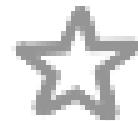
Azure Data Lake
Storage Gen2
By Microsoft



Azure HDInsight
(HDFS)
By Microsoft



Azure
HDInsight
on AKS Trino
(Beta)
By Microsoft



Azure
HDInsight
Spark
By Microsoft



Azure Synapse
Analytics
(SQL DW)
By Microsoft



Azure Synapse
Analytics
workspace
(Beta)
By Microsoft



Azure SQL
database
By Microsoft



Azure Table
Storage
By Microsoft



Azure Time
Series Insights
(Beta)
By Microsoft

B

Connectors



BI Connector
By Guidanz



BitSight Security
Ratings
By BitSight



Bloomberg Data
and Analytics
By Bloomberg



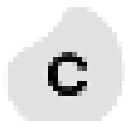
BQE Core
By BQE

C

Connectors



CData Connect
Cloud
By CData



Celonis EMS (Beta)
By Celonis



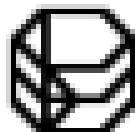
Cherwell (Beta)
By Cherwell



CloudBluePSA
(Beta)
By CloudBluePSA



Cognite Data
Fusion
By Cognite



Common Data
Service (legacy)
By Microsoft

D

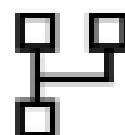
Connectors



Data Virtuality
LDW
By Data Virtuality



Databricks
By Databricks



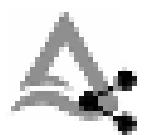
Dataflows
By Microsoft



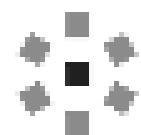
Datamarts
By Microsoft



Dataverse
By Microsoft



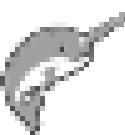
Delta Sharing
By Databricks



Denodo
By Denodo



Digital
Construction
Works Insights
By Digital
Construction
Works



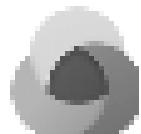
Dremio Cloud
By Dremio



Dremio Software
By Dremio

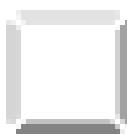
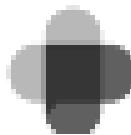


Dynamics 365
Business Central
By Microsoft

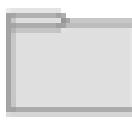


Dynamics 365
Customer Insights
(Beta)
By Microsoft

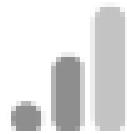
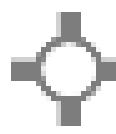
E

Connectors				
				
Eduframe Reporting (Beta) By Drieam	Emigo Data Source By Sagra	Emplifi Metrics (Beta) By Emplifi	Entersoft Business Suite (Beta) By Entersoft	EQuIS By EarthSoft

F

Connectors				
				
FactSet Analytics By FactSet	FactSet RMS (Beta) By FactSet	FHIR By Microsoft	Folder By Microsoft	Funnel By Funnel

G

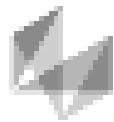
Connectors			
			
Google Analytics By Microsoft	Google BigQuery By Microsoft	Google BigQuery (Azure AD) (Beta) By Microsoft	Google Sheets By Microsoft

H

Connectors



HDInsight
Interactive Query
By Microsoft



Hexagon PPM
Smart API
By Hexagon PPM



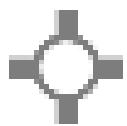
Hive LLAP
By Microsoft

I

Connectors



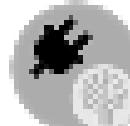
IBM DB2
database
By Microsoft



Impala
By Microsoft



Indexima
By Indexima



Industrial
App Store
By Intelligent Plant



Information
Grid (Beta)
By Luminis



InterSystems
IRIS (Beta)
By Intersystems



Intune Data
Warehouse (Beta)
By Microsoft

J

Connectors



Jamf Pro
(Beta)
By Jamf



Jethro
(Beta)
By JethroData



JSON
By Microsoft

K

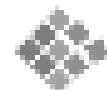
Connectors



Kognitwin
By Kongsberg



KQL Database
By Microsoft



Kyligence
By Kyligence

L

Connectors



Lakehouse
(Beta)
By Microsoft



Linkar PICK
Style/MultiValue
Databases
(Beta)
By Kosday Solutions

M

Connectors



MariaDB
By MariaDB



MarkLogic
By MarkLogic



Microsoft
Exchange
Online
By Microsoft



Microsoft Teams
Personal
Analytics
(Beta)
By Microsoft



MicroStrategy
for Power BI
By
MicroStrategy

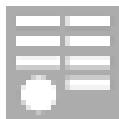
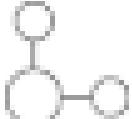


MongoDB
Atlas
SQL interface
(Beta)
By MongoDB

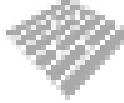
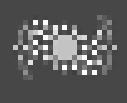
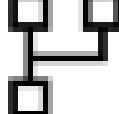


MySQL
database
By Microsoft

O

Connectors		
		
OData By Microsoft	ODBC By Microsoft	OpenSearch Project (Beta) By OpenSearch

P

Connectors				
				
Parquet By Microsoft	Palantir Foundry By Palantir	Paxata By Paxata	PDF By Microsoft	Planview Enterprise Architecture By Planview
				
Planview IdeaPlace By Planview	Planview Portfolios By Planview	Planview Projectplace By Planview	PostgreSQL database By Microsoft	Power BI dataflows (Legacy) By Microsoft
				
Product Insights (Beta) By Microsoft	Profisee By Profisee			

Q

Connectors



QubolePresto (Beta)
By Qubole



Quickbase
By Quick Base

R

Connectors



Roamler (Beta)
By Roamler

S

Connectors



**Salesforce
Objects**
By Microsoft



**Salesforce
Reports**
By Microsoft



**SAP BW
Application
Server**
By Microsoft



**SAP BW
Message
Server**
By Microsoft



**SAP HANA
database**
By Microsoft



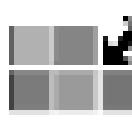
**SharePoint
folder**
By Microsoft



SharePoint list
By Microsoft



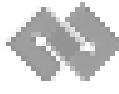
**SharePoint
Online
list**
By Microsoft



**Shortcuts
Business
Insights (Beta)**
By Shortcuts



SingleStore (Beta)
By SingleStore



**SIS-CC SDMX
(Beta)**
By SIS-CC



SitelImprove
By SitelImprove



Smartsheet
By Microsoft



Snowflake
By Microsoft



**Socialbakers Metrics
(Beta)**
By Emplifi

Connectors



SoftOneBI
(Beta)
By SoftOne
SolarWinds
Service
Desk (Beta)
By SolarWinds



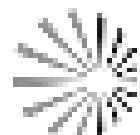
Solver
By BI360



Spark
By Microsoft



SQL Server
database
By Microsoft



Starburst
Enterprise
By Starburst
Data



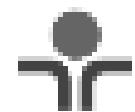
SumTotal
By SumTotal



SurveyMonkey
By
SurveyMonkey

T

Connectors



TeamDesk (Beta)
By ForeSoft



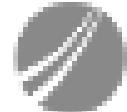
Tenforce (Smart)List
By Tenforce



Teradata database
By Microsoft



Text/CSV
By Microsoft



TIBCO(R) Data
Virtualization
By TIBCO

U

Connectors



Usercube (Beta)
By Usercube

V

Connectors



Vena
By Vena



Vertica
By Microsoft



Vessel Insight
By Kongsberg



Viva Insights
By Microsoft

W

Connectors



Warehouse (Beta)
By Microsoft



Web API
By Microsoft



Web page
By Microsoft



Webtrends
Analytics
(Beta)
By Microsoft



Witivio (Beta)
By Witivio



Wrike (Beta)
By Wrike

X

Connectors



XML
By Microsoft

Z

Connectors



Zoho Creator
By Zoho



Zucchetti HR
Infinity (Beta)
By Zucchetti

Feedback

Was this page helpful?

Yes

No

Provide product feedback [↗](#) | Ask the community [↗](#)

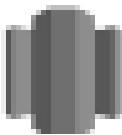
Data pipeline connectors in Microsoft Fabric

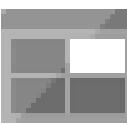
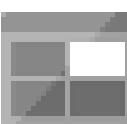
Article • 11/15/2023

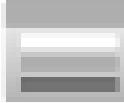
The following table contains a list of all the data pipeline connectors currently available for Data Factory in Microsoft Fabric. A checkmark indicates the connector is currently supported in the listed service; an X indicates that the connector isn't currently supported in the listed service.

The connectors are listed in alphabetical order in separate tables for each letter in the alphabet. Use the **In this article** list at the top of this article to go to any of the alphabetized tables.

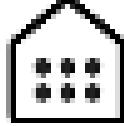
A

Connector	Category	Copy source/ destination	Lookup	Get metadata	Delete	Script	Stored procedure
 Amazon RDS for SQL Server	Database	✓/✗	✓	✓	✗	✗	✓
 Amazon Redshift	Database	✓/✗	✓	✗	✗	✗	✗
 Amazon S3	File	✓/✗	✓	✓	✓	✗	✗
 Amazon S3 Compatible	File	✓/✗	✓	✓	✓	✗	✗

Connector	Category	Copy source/ destination	Lookup	Get metadata	Delete	Script	Stored procedure
	Azure	✓ / ✓	✓	✓	✓	✗	✗
Azure Blob Storage							
	Azure	✓ / ✓	✓	✗	✗	✗	✗
Azure Cosmos DB for NoSQL							
	Azure	✓ / ✓	✓	✗	✗	✗	✗
Azure Database for PostgreSQL							
	Azure	✓ / ✓	✓	✗	✗	✗	✗
Azure Data Explorer							
	Azure	✓ / ✓	✓	✓	✓	✗	✗
Azure Data Lake Storage Gen1							
	Azure	✓ / ✓	✓	✓	✓	✗	✗
Azure Data Lake Storage Gen2							
	Azure	✓ / ✓	✓	✓	✗	✓	✓
Azure							

Connector	Category	Copy source/ destination	Lookup	Get metadata	Delete	Script	Stored procedure
Synapse Analytics							
	Azure	✓ / ✓	✓	✓	✗	✓	✓
Azure SQL database							
	Azure	✓ / ✓	✓	✓	✗	✓	✓
Azure SQL Managed Instance							
	Azure	✓ / ✓	✓	✗	✗	✗	✗
Azure Table Storage							

D

Connector	Category	Copy source/ destination	Lookup	Get metadata	Delete	Script	Stored procedure
	Workspace	✓ / ✓	✓	✓	✗	✓	✓
Data Warehouse							
	Services and apps	✓ / ✓	✓	✗	✗	✗	✗
Dataverse							

Connector	Category	Copy source/ destination	Lookup	Get metadata	Delete	Script	Stored procedure
	Services and apps	✓ / ✓	✓	✗	✗	✗	✗

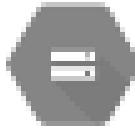
Dynamics CRM

F

Connector	Category	Copy source/ destination	Lookup	Get metadata	Delete	Script	Stored procedure
	File	✓ / ✗	✓	✓	✓	✗	✗

FTP

G

Connector	Category	Copy source/ destination	Lookup	Get metadata	Delete	Script	Stored procedure
	File	✓ / ✗	✓	✓	✓	✗	✗

Google Cloud Storage

H

Connector	Category	Copy source/ destination	Lookup	Get metadata	Delete	Script	Stored procedure
 HTTP	Generic protocol	✓ / ✗	✓	✗	✗	✗	✗

K

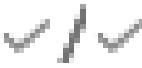
Connector	Category	Copy source/ destination	Lookup	Get metadata	Delete	Script	Stored procedure
 KQL Database	Workspace	✓ / ✓	✓	✗	✗	✗	✗

L

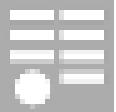
Connector	Category	Copy source/ destination	Lookup	Get metadata	Delete	Script	Stored procedure
 Lakehouse	Workspace	✓ / ✓	✗	✗	✓	✗	✗

M

Connector	Category	Copy source/ destination	Lookup	Get metadata	Delete	Script	Stored procedure
 Microsoft	Services and apps	✓ / ✗	✗	✗	✗	✗	✗

Connector	Category	Copy source/ destination	Lookup	Get metadata	Delete	Script	Stored procedure
356 (Preview)							
	NoSQL	 	 				
MongoDB							
	NoSQL	 					
MongoDB Atlas							

O

Connector	Category	Copy source/ destination	Lookup	Get metadata	Delete	Script	Stored procedure
	Generic protocol	 					
OData							

P

Connector	Category	Copy source/ destination	Lookup	Get metadata	Delete	Script	Stored procedure
	Database	 					
PostgreSQL							

R

Connector	Category	Copy source/ destination	Lookup	Get metadata	Delete	Script	Stored procedure
	Generic protocol	✓ / ✓	✗	✗	✗	✗	✗

REST

S

Connector	Category	Copy source/ destination	Lookup	Get metadata	Delete	Script	Stored procedure
	File	✓ / ✓	✓	✓	✓	✗	✗
SFTP							
	Services and apps	✓ / ✗	✓	✗	✗	✗	✗
SharePoint Online list							
	Services and apps	✓ / ✓	✓	✗	✗	✓	✗
Snowflake							
	Database	✓ / ✓	✓	✓	✗	✓	✓
SQL Server							

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback ↗ | Ask the community ↗

Access database connector overview

Article • 11/15/2023

The Access database connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to an Access database in Dataflow Gen2, go to Set up your Access database connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support an Access database in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your Access database connection

Article • 12/06/2023

This article outlines the steps to create an Access database connection.

Supported authentication types

The Access database connector supports the following authentication types for copy and Dataflow Gen2 respectively.

 Expand table

Authentication type	Copy	Dataflow Gen2
Organizational account	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to an Access database. The following links provide the specific Power Query connector information you need to connect to an Access database in Dataflow Gen2:

- To get started using the Access database connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- Be sure to install or set up any Access database prerequisites before connecting to the Access database connector.
- To connect to the Access database connector from Power Query, go to [Connect to Access database from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support Access database in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Amazon RDS for SQL Server connector overview

Article • 12/08/2023

The Amazon RDS for SQL Server connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Data pipeline

The Amazon RDS for SQL Server connector supports the following capabilities in Data pipeline:

 Expand table

Supported capabilities	Gateway	Authentication
Copy activity (source/-)	None	Basic
Lookup activity	None	Basic
GetMetadata activity	None	Basic
Stored procedure activity	None	Basic

To learn more about the copy activity configuration for Amazon RDS for SQL Server in Data pipeline, go to [Configure in a data pipeline copy activity](#).

Feedback

Was this page helpful?



Yes



No

Provide product feedback  | Ask the community 

How to configure Amazon RDS for SQL Server in copy activity

Article • 12/08/2023

This article outlines how to use the copy activity in a data pipeline to copy data from Amazon RDS for SQL Server.

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

The following properties are supported for Amazon RDS for SQL Server under the **Source** tab of a copy activity.



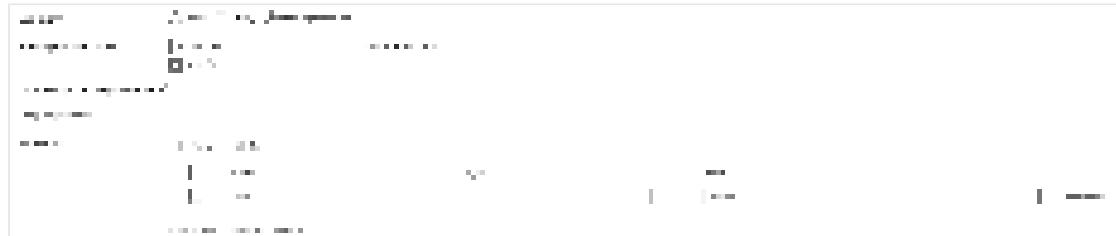
The following properties are required:

- **Data store type:** Select **External**.
- **Connection:** Select an Amazon RDS for SQL Server connection from the connection list. If the connection doesn't exist, then create a new Amazon RDS for SQL Server connection by selecting **New**.
- **Connection type:** Select **Amazon RDS for SQL Server**.
- **Use query:** Specify the way to read data. You can choose **Table**, **Query**, or **Stored procedure**. The following list describes the configuration of each setting:
 - **Table:** Read data from the table specified. Select your source table from the drop-down list or select **Edit** to enter it manually.
 - **Query:** Specify the custom SQL query to read data. An example is `select * from MyTable`. Or select the pencil icon to edit in code editor.



- **Stored procedure:** Use the stored procedure that reads data from the source table. The last SQL statement must be a SELECT statement in the stored procedure.

- **Stored procedure name:** Select the stored procedure or specify the stored procedure name manually when selecting the **Edit** to read data from the source table.
- **Stored procedure parameters:** Specify values for stored procedure parameters. Allowed values are name or value pairs. The names and casing of parameters must match the names and casing of the stored procedure parameters. You can select **Import parameters** to get your stored procedure parameters.



Under **Advanced**, you can specify the following fields:

- **Query timeout (minutes):** Specify the timeout for query command execution, default is 120 minutes. If a parameter is set for this property, allowed values are timespan, such as "02:00:00" (120 minutes).
- **Isolation level:** Specifies the transaction locking behavior for the SQL source. The allowed values are: **Read committed**, **Read uncommitted**, **Repeatable read**, **Serializable**, **Snapshot**. If not specified, the database's default isolation level is used. Refer to **IsolationLevel Enum** for more details.



- **Partition option:** Specify the data partitioning options used to load data from Amazon RDS for SQL Server. Allowed values are: **None** (default), **Physical partitions of table**, and **Dynamic range**. When a partition option is enabled (that is, not **None**), the degree of parallelism to concurrently load data from Amazon RDS for SQL Server is controlled by **Degree of copy parallelism** in copy activity settings tab.
 - **None:** Choose this setting to not use a partition.
 - **Physical partitions of table:** When using a physical partition, the partition column and mechanism are automatically determined based on your physical table definition.

- **Dynamic range:** When using query with parallel enabled, the range partition parameter (`?AdfDynamicRangePartitionCondition`) is needed. Sample query: `SELECT * FROM <TableName> WHERE ?AdfDynamicRangePartitionCondition.`
- **Partition column name:** Specify the name of the source column in `integer` or `date/datetime` type (`int`, `smallint`, `bigint`, `date`, `smalldatetime`, `datetime`, `datetime2`, or `datetimeoffset`) that's used by range partitioning for parallel copy. If not specified, the index or the primary key of the table is auto-detected and used as the partition column.
If you use a query to retrieve the source data, hook `?` `AdfDynamicRangePartitionCondition` in the WHERE clause. For an example, see the Parallel copy from SQL database section.
- **Partition upper bound:** Specify the maximum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering the rows in table. All rows in the table or query result will be partitioned and copied. If not specified, copy activity auto detect the value. For an example, see the Parallel copy from SQL database section.
- **Partition lower bound:** Specify the minimum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering the rows in table. All rows in the table or query result will be partitioned and copied. If not specified, copy activity auto detect the value. For an example, see the Parallel copy from SQL database section.
- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.

Note the following points:

- If **Query** is specified for source, the copy activity runs this query against the Amazon RDS for SQL Server source to get the data. You also can specify a stored procedure by specifying **Stored procedure name** and **Stored procedure parameters** if the stored procedure takes parameters.
- When using stored procedure in source to retrieve data, note if your stored procedure is designed as returning different schema when different parameter value is passed in, you may encounter failure or see unexpected result when importing schema from UI or when copying data to SQL database with auto table creation.

Mapping

For Mapping tab configuration, go to Configure your mappings under mapping tab.

Settings

For **Settings** tab configuration, go to Configure your other settings under settings tab.

Parallel copy from SQL database

The Amazon RDS for SQL Server connector in copy activity provides built-in data partitioning to copy data in parallel. You can find data partitioning options on the **Source** tab of the copy activity.

When you enable partitioned copy, copy activity runs parallel queries against your Amazon RDS for SQL Server source to load data by partitions. The parallel degree is controlled by the **Degree of copy parallelism** in the copy activity settings tab. For example, if you set **Degree of copy parallelism** to four, the service concurrently generates and runs four queries based on your specified partition option and settings, and each query retrieves a portion of data from your Amazon RDS for SQL Server.

You are suggested to enable parallel copy with data partitioning especially when you load large amount of data from your Amazon RDS for SQL Server. The following are suggested configurations for different scenarios. When copying data into file-based data store, it's recommended to write to a folder as multiple files (only specify folder name), in which case the performance is better than writing to a single file.

 Expand table

Scenario	Suggested settings
Full load from large table, with physical partitions.	<p>Partition option: Physical partitions of table.</p> <p>During execution, the service automatically detects the physical partitions, and copies data by partitions.</p> <p>To check if your table has physical partition or not, you can refer to this query.</p>
Full load from large table, without physical partitions, while with an integer or datetime column for data partitioning.	<p>Partition options: Dynamic range partition.</p> <p>Partition column (optional): Specify the column used to partition data. If not specified, the primary key column is used.</p> <p>Partition upper bound and partition lower bound (optional): Specify if you want to determine the partition stride. This is not for filtering the rows in table, all rows in the table will be partitioned and copied. If not specified, copy activity auto detects the values and it can take long time depending on MIN and MAX values. It is recommended to provide upper bound and lower bound.</p> <p>For example, if your partition column "ID" has values range from 1 to 100, and you set the lower bound as 20 and the upper bound as 80, with parallel copy as 4, the service retrieves data by 4 partitions - IDs in range <=20, [21, 50], [51, 80], and >=81, respectively.</p>

Scenario	Suggested settings
Load a large amount of data by using a custom query, without physical partitions, while with an integer or date/datetime column for data partitioning.	<p>Partition options: Dynamic range partition.</p> <p>Query: <code>SELECT * FROM <TableName> WHERE ?</code> <code>AdfDynamicRangePartitionCondition AND <your_additional_where_clause></code>.</p> <p>Partition column: Specify the column used to partition data.</p> <p>Partition upper bound and partition lower bound (optional): Specify if you want to determine the partition stride. This is not for filtering the rows in table, all rows in the query result will be partitioned and copied. If not specified, copy activity auto detect the value.</p> <p>During execution, the service replaces <code>?AdfRangePartitionColumnName</code> with the actual column name and value ranges for each partition, and sends to Amazon RDS for SQL Server.</p> <p>For example, if your partition column "ID" has values range from 1 to 100, and you set the lower bound as 20 and the upper bound as 80, with parallel copy as 4, the service retrieves data by 4 partitions- IDs in range <code><=20</code>, <code>[21, 50]</code>, <code>[51, 80]</code>, and <code>>=81</code>, respectively.</p> <p>Here are more sample queries for different scenarios:</p> <ol style="list-style-type: none"> 1. Query the whole table: <code>SELECT * FROM <TableName> WHERE ?AdfDynamicRangePartitionCondition</code> 2. Query from a table with column selection and additional where-clause filters: <code>SELECT <column_list> FROM <TableName> WHERE ?</code> <code>AdfDynamicRangePartitionCondition AND <your_additional_where_clause></code> 3. Query with subqueries: <code>SELECT <column_list> FROM (<your_sub_query>) AS T WHERE ?</code> <code>AdfDynamicRangePartitionCondition AND <your_additional_where_clause></code> 4. Query with partition in subquery: <code>SELECT <column_list> FROM (SELECT <your_sub_query_column_list> FROM <TableName> WHERE ?AdfDynamicRangePartitionCondition) AS T</code>

Best practices to load data with partition option:

1. Choose distinctive column as partition column (like primary key or unique key) to avoid data skew.
2. If the table has built-in partition, use partition option **Physical partitions of table** to get better performance.

Sample query to check physical partition

SQL

```
SELECT DISTINCT s.name AS SchemaName, t.name AS TableName, pf.name AS PartitionFunctionName, c.name AS ColumnName, iif(pf.name is null, 'no', 'yes') AS HasPartition
FROM sys.tables AS t
LEFT JOIN sys.objects AS o ON t.object_id = o.object_id
```

```

LEFT JOIN sys.schemas AS s ON o.schema_id = s.schema_id
LEFT JOIN sys.indexes AS i ON t.object_id = i.object_id
LEFT JOIN sys.index_columns AS ic ON ic.partition_ordinal > 0 AND ic.index_id =
i.index_id AND ic.object_id = t.object_id
LEFT JOIN sys.columns AS c ON c.object_id = ic.object_id AND c.column_id =
ic.column_id
LEFT JOIN sys.partition_schemes ps ON i.data_space_id = ps.data_space_id
LEFT JOIN sys.partition_functions pf ON pf.function_id = ps.function_id
WHERE s.name='[your schema]' AND t.name = '[your table name]'
```

If the table has physical partition, you would see "HasPartition" as "yes" like the following.

SchemaName	TableName	PartitionFunctionName	ColumnName	HasPartition
dbo	AdventureworksLT	RangePartition	RegionID	Yes

Table summary

See the following table for the summary and more information for the Amazon RDS for SQL Server copy activity.

Source information

 Expand table

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	< your connection >	Yes	connection
Connection type	Your connection type. Select Amazon RDS for SQL Server.	Amazon RDS for SQL Server	Yes	/
Use query	The custom SQL query to read data.	<ul style="list-style-type: none"> • Table • Query • Stored procedure 	Yes	/
Table	Your source data table.	< name of your destination table>	No	schema table
Query	The custom SQL query to read data.	< your query >	No	sqlReaderQuery

Name	Description	Value	Required	JSON script property
Stored procedure name	This property is the name of the stored procedure that reads data from the source table. The last SQL statement must be a SELECT statement in the stored procedure.	< stored procedure name >	No	sqlReaderStoredProcedureName
Stored procedure parameter	These parameters are for the stored procedure. Allowed values are name or value pairs. The names and casing of parameters must match the names and casing of the stored procedure parameters.	< name or value pairs >	No	storedProcedureParameters
Query timeout	The timeout for query command execution.	timespan (the default is 120 minutes)	No	queryTimeout
Isolation level	Specifies the transaction locking behavior for the SQL source.	<ul style="list-style-type: none"> • Read committed • Read uncommitted • Repeatable read • Serializable • Snapshot 	No	isolationLevel: <ul style="list-style-type: none"> • ReadCommitted • ReadUncommitted • RepeatableRead • Serializable • Snapshot
Partition option	The data partitioning options used to load data from Amazon RDS for SQL Server.	<ul style="list-style-type: none"> • None (default) • Physical partitions of table • Dynamic range 	No	partitionOption: <ul style="list-style-type: none"> • None (default) • PhysicalPartitionsOfTable • DynamicRange
Partition column name	<p>The name of the source column in integer or date/datetime type (<code>int</code>, <code>smallint</code>, <code>bigint</code>, <code>date</code>, <code>smalldatetime</code>, <code>datetime</code>, <code>datetime2</code>, or <code>datetimeoffset</code>)</p> <p>that's used by range partitioning for parallel copy. If not specified, the index or the primary key of the table is auto-detected and used as the partition column. If you use a query to retrieve the source data, hook ? <code>AdfDynamicRangePartitionCondition</code> in the WHERE clause.</p>	< your partition column names >	No	partitionColumnName

Name	Description	Value	Required	JSON script property
Partition upper bound	The maximum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering the rows in table. All rows in the table or query result will be partitioned and copied. If not specified, copy activity auto detect the value.	< your partition upper bound >	No	partitionUpperBound
Partition lower bound	The minimum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering the rows in table. All rows in the table or query result will be partitioned and copied. If not specified, copy activity auto detect the value.	< your partition lower bound >	No	partitionLowerBound
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.	• Name • Value	No	additionalColumns: • name • value

Next steps

- Amazon RDS for SQL Server overview

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Amazon Redshift connector overview

Article • 11/15/2023

The Amazon Redshift connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to Amazon Redshift in Dataflow Gen2, go to Set up your Amazon Redshift connection.

Support in data pipelines

The Amazon Redshift connector supports the following capabilities in data pipelines:

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	Amazon Redshift
Lookup activity	None	Amazon Redshift

To learn more about the copy activity configuration for Amazon Redshift in data pipelines, go to Configure in a data pipeline copy activity.

Feedback

Was this page helpful?



Yes



No

Provide product feedback ↗ | Ask the community ↗

Set up your Amazon Redshift connection

Article • 12/06/2023

This article outlines the steps to create an Amazon Redshift connection.

Supported authentication types

The Amazon Redshift connector supports the following authentication types for copy and Dataflow Gen2 respectively.

[+] Expand table

Authentication type	Copy	Dataflow Gen2
Amazon Redshift	n/a	✓
Microsoft Account	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Amazon Redshift. The following links provide the specific Power Query connector information you need to connect to Amazon Redshift in Dataflow Gen2:

- To get started using the Amazon Redshift connector in Dataflow Gen2, go to Get data from Data Factory in Microsoft Fabric.
- Be sure to install or set up any Amazon Redshift prerequisites before connecting to the Amazon Redshift connector.
- To connect to the Amazon Redshift connector from Power Query, go to Connect to Amazon Redshift data from Power Query Online.

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Configure Amazon Redshift in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in data pipelines to copy data from Amazon Redshift.

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

Go to **Source** tab to configure your copy activity source. See the following content for the detailed configuration.



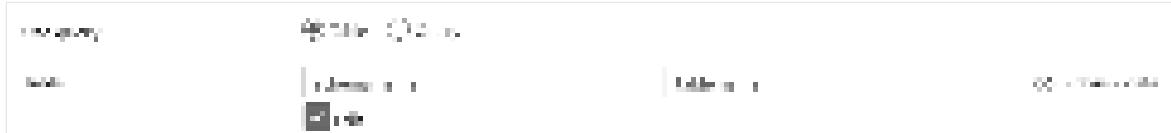
The following three properties are **required**:

- **Data store type:** Select **External**.

- **Connection:** Select an Amazon Redshift connection from the connection list. If no connection exists, then create a new Amazon Redshift connection by selecting **New**.
- **Use query:** Select from **Table or Query**.

If you select **Table**:

- **Table:** Specify the name of the table. Select the table from the drop-down list or enter the name manually by selecting **Edit**.



If you select **Query**:

- **Query:** Specify the custom query to read data. For example: `select * from MyTable.`



Under **Advanced**, you can specify the following fields:

- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.

Mapping

For **Mapping** tab configuration, see Configure your mappings under mapping tab.

Settings

For **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

See the following table for the summary and more information of the Amazon Redshift copy activity.

Source information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	< your Amazon Redshift connection >	Yes	connection
Use query	The way to read data. Apply Table to read data from the specified table or apply Query to read data using queries.	<ul style="list-style-type: none"> • Table • Query 	Yes	<ul style="list-style-type: none"> • typeProperties (under <code>typeProperties</code> -> <code>source</code>) - schema - table • query
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.	<ul style="list-style-type: none"> • Name • Value 	No	additionalColumns: <ul style="list-style-type: none"> • name • value

Next steps

- Amazon Redshift connector overview
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Amazon S3 connector overview

Article • 12/22/2023

This Amazon S3 connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

Data Factory in Microsoft Fabric doesn't currently support Amazon S3 in Dataflow Gen2.

Support in Data pipeline

The Amazon S3 connector supports the following capabilities in Data pipeline:

 Expand table

Supported capabilities	Gateway	Authentication
Copy activity (source/-)	None	Basic
Lookup activity	None	Basic
GetMetadata activity	None	Basic
Delete activity	None	Basic

To learn about how to connect to Amazon S3 data in Data pipeline, go to Set up your Amazon S3 connection.

To learn about the copy activity configuration for Amazon S3 in Data pipeline, go to Configure Amazon S3 in a copy activity.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your Amazon S3 connection

Article • 11/15/2023

This article outlines the steps to create an Amazon S3 connection.

Supported authentication types

The Amazon S3 connector supports the following authentication types for copy and Dataflow Gen2 respectively.

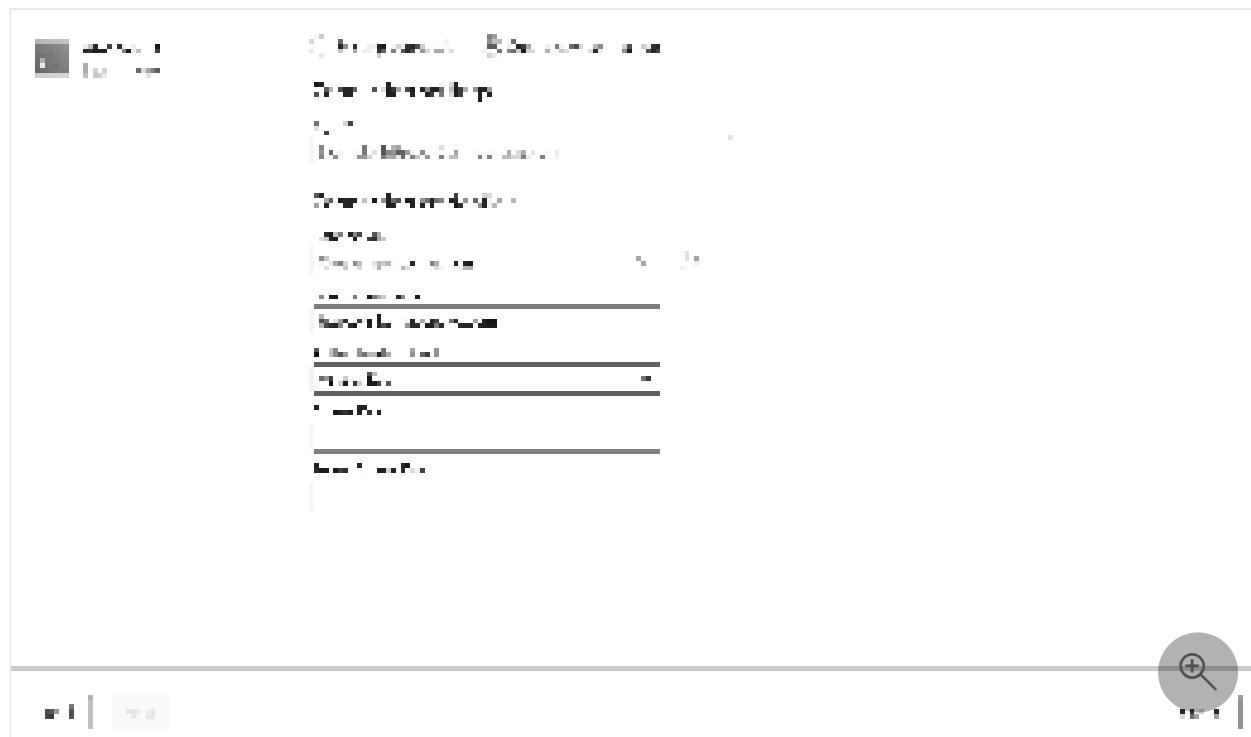
Authentication type	Copy	Dataflow Gen2
Basic	✓	n/a

Set up your connection in Dataflow Gen2

The Amazon S3 connector isn't currently supported in Dataflow Gen2.

Set up your connection in a data pipeline

Browse to the [New connection page](#) for the Data Factory pipeline to configure the connection details and create the connection.



You have two ways to browse to this page:

- In copy assistant, browse to this page after selecting the connector.
- In pipeline, browse to this page after selecting + New in Connection section and selecting the connector.

Step 1: Specify the new connection name, type, data source path

Connection settings

Url
Example: <https://s3.amazonaws.com>

Connection credentials

Connection
Create new connection

Connection name

Authentication kind
Access Key

Access Key Id

Secret Access Key

In the **New connection** pane, specify the following field:

- **Url:** Specify the account endpoint URL of your Amazon S3 data.

Step 2: Select and set your authentication

Under **Authentication method**, select your authentication method from the drop-down list and complete the related configuration. The Amazon S3 connector supports the following authentication types:

- Basic

Authentication

Authentication method *

Basic

Basic

Basic authentication

- **Username:** The user name to use to access the Amazon S3 endpoint.
- **Password:** The password for the specified username.

Authentication

Authentication method *

Basic

Username *

< Your username >

Password *

< Your password >

Step 3: Specify the privacy level that you want to apply

In the **General** tab, under **Privacy level**, select the privacy level that you want apply. Three privacy levels are supported. For more information, go to [privacy levels](#).

Step 4: Create your connection

Select **Create**. Your creation is successfully tested and saved if all the credentials are correct. If the credentials aren't correct, the creation fails with errors.



Table summary

The following table contains connector properties that are supported in pipeline copy.

Name	Description	Required	Property	Copy
Connection name	A name for your connection.	Yes		✓
Connection type	Select Amazon S3 for your connection type.	Yes		✓
Data source path	Enter your account endpoint URL of your Amazon S3.	Yes		✓
Authentication	Go to Authentication	Yes		Go to Authentication

Authentication

The following table contains properties for the supported authentication type.

Name	Description	Required	Property	Copy
Basic				✓
- Username	The user name to use to access the Amazon S3 endpoint.	Yes		
- Password	The password for specified username.	Yes		

Next steps

- Configure in a data pipeline copy activity

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Configure Amazon S3 in a copy activity

Article • 12/22/2023

This article outlines how to use the copy activity in a data pipeline to copy data from and to Amazon S3.

Required permissions

To copy data from Amazon S3, make sure you've been granted the following permissions for Amazon S3 object operations: `s3:GetObject` and `s3:GetObjectVersion`.

In addition, `s3>ListAllMyBuckets` and `s3>ListBucket/s3:GetBucketLocation` permissions are required for operations like testing connection and browsing from root.

For the full list of Amazon S3 permissions, go to [Specifying Permissions in a Policy](#) on the AWS site ↗.

Supported format

Amazon S3 supports the following file formats. Refer to each article for format-based settings.

- Avro format
- Binary format
- Delimited text format
- Excel format
- JSON format
- ORC format
- Parquet format
- XML format

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

The following properties are supported for Amazon S3 under the **Source** tab of a copy activity.



The following properties are required:

- **Data store type:** Select **External**.
 - **Connection:** Select an Amazon S3 connection from the connection list. If no connection exists, then create a new Amazon connection by selecting **New**.
 - **Connection type:** Select **Amazon S3** for your connection type.
 - **File path type:** You can choose **File path**, **Prefix**, **Wildcard file path**, or **List of files** as your file path type. The configuration for each setting is:
 - **File path:** If you choose this type, the data can be copied from the given bucket or the given bucket and folder path specified.
 - **Prefix:** If you choose this type, specify the **Bucket** and **Prefix**.
 - **Bucket:** Specify the S3 bucket name. It is required.
 - **Prefix:** Specify the configured prefix for the S3 key name under the given bucket to filter source S3 files. S3 keys whose names start with

`bucket/this_prefix` are selected. It utilizes S3's service-side filter, which provides better performance than a wildcard filter.

When you use prefix and choose to copy to file-based destination with preserving hierarchy, note the sub-path after the last "/" in prefix will be preserved. For example, you have source `bucket/folder/subfolder/file.txt`, and configure prefix as `folder/sub`, then the preserved file path is `subfolder/file.txt`.



- **Wildcard file path:** If you choose this type, specify the **Bucket** and **Wildcard paths**.
 - **Bucket:** Specify the S3 bucket name. It is required.
 - **Wildcard paths:** Specify the folder or file path with wildcard characters under your given bucket to filter your source folders or files.

Allowed wildcards are: `*` (matches zero or more characters) and `?` (matches zero or single character). Use `\` to escape if your folder name has wildcard or this escape character inside. See more examples in Folder and file filter examples.



Wildcard folder path: The folder path with wildcard characters under the given bucket to filter source folders.

Wildcard file name: The file name with wildcard characters under the given bucket and folder path (or wildcard folder path) to filter source files.

- **List of files:** If you choose this type, specify the **Folder path** and **Path to file list** to indicates to copy a given file set. Point to a text file that includes a list of files you want to copy, one file per line, which is the relative path to the path configured. For more examples, go to File list examples.



- **Folder path:** Specify the path to the folder under given bucket. It is required.
- **Path to file list:** Specify the path of the text file that includes a list of files you want to copy.
- **Recursively:** Specify whether the data is read recursively from the subfolders or only from the specified folder. When **Recursively** is selected and the destination is a file-based store, an empty folder or subfolder isn't copied or created at the destination. This property is selected by default and doesn't apply when you configure **Path to file list**.
- **File format:** Select the file format applied from the drop-down list. Select **Settings** to configure the file format. For settings of different file formats, refer to articles in Supported format for detailed information.

Under **Advanced**, you can specify the following fields:

- **Filter by last modified:** Files are filtered based on the last modified dates that you specified. This property doesn't apply when you configure your file path type as **List of files**.
 - **Start time (UTC):** The files are selected if their last modified time is greater than or equal to the configured time.
 - **End time (UTC):** The files are selected if their last modified time is less than the configured time.

When **Start time (UTC)** has datetime value but **End time (UTC)** is NULL, it means the files whose last modified attribute is greater than or equal with the datetime value will be selected. When **End time (UTC)** has datetime value but **Start time (UTC)** is NULL, it means the files whose last modified attribute is less than the datetime value will be selected. The properties can be NULL, which means no file attribute filter will be applied to the data.

- **Enable partition discovery:** Specify whether to parse the partitions from the file path and add them as additional source columns. It is unselected by default and not supported when you use binary file format.
 - **Partition root path:** When partition discovery is enabled, specify the absolute root path in order to read partitioned folders as data columns.
If it is not specified, by default,
 - When you use file path or list of files on source, partition root path is the path that you configured.
 - When you use wildcard folder filter, partition root path is the sub-path before the first wildcard.

- When you use prefix, partition root path is sub-path before the last "/".

For example, assuming you configure the path as

`root/folder/year=2020/month=08/day=27`:

- If you specify partition root path as `root/folder/year=2020`, copy activity will generate two more columns month and day with value "08" and "27" respectively, in addition to the columns inside the files.
- If partition root path is not specified, no extra column will be generated.



- Max concurrent connection:** The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.
- Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.

Mapping

For **Mapping** tab configuration, go to Configure your mappings under mapping tab. If you choose Binary as your file format, mapping will not be supported.

Settings

For the **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

The following tables contain more information about the copy activity in Amazon S3.

Source information

[] Expand table

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/

Name	Description	Value	Required	JSON script property
Connection	Your connection to the source data store.	<your Amazon S3 connection>	Yes	connection
Connection type	Select a type for your connection.	Amazon S3	Yes	/
File path type	The file path type used to get source data.	<ul style="list-style-type: none"> • File path • Prefix • Wildcard file path • List of files 	Yes	<ul style="list-style-type: none"> • folderPath, fileName • prefix • wildcardFolderPath, wildcardFileName • fileListPath
File path	The folder/file path to the source file.	< file path>	Yes for File path	bucketName fileName folderpath
Bucket	The S3 bucket name.	<your bucket name>	Yes	bucketName
Prefix	The configured prefix for the S3 key name under the given bucket to filter source S3 files.	<your prefix>	No	prefix
Wildcard paths	The folder/file path with wildcard characters under the configured bucket to filter source folders/files.	< your wildcard file path >	Yes for Wildcard file name	wildcardFolderPath wildcardFileName
Path to file list	Indicates to copy a given file set. Point to a text file that includes a list of files you want to copy, one file per line.	< file list path >	No	fileListPath
File format	The file format for your source data. For the information of different file formats, refer to articles in Supported format for detailed information.	/	Yes	/
Recursively	Indicates whether the data is read recursively from the subfolders or	selected (default) or unselect	No	recursive

Name	Description	Value	Required	JSON script property
	<p>only from the specified folder. Note that when Recursively is selected and the destination is a file-based store, an empty folder or subfolder isn't copied or created at the destination. This property doesn't apply when you configure Path to file list.</p>			
Filter by last modified	<p>The files with last modified time in the range [Start time, End time) will be filtered for further processing. The time will be applied to UTC time zone in the format of <code>yyyy-mm-ddThh:mm:ss.fffZ</code>.</p> <p>These properties can be skipped which means no file attribute filter will be applied. This property doesn't apply when you configure your file path type as List of files.</p>	datetime	No	modifiedDatetimeStart modifiedDatetimeEnd
Enable partition discovery	Indicates whether to parse the partitions from the file path and add them as additional source columns.	selected or unselected (default)	No	enablePartitionDiscovery: true or false (default)
Max concurrent connection	The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.	<max concurrent connections>	No	maxConcurrentConnections
Additional columns	Add additional data columns to store	<ul style="list-style-type: none"> • Name • Value 	No	additionalColumns: <ul style="list-style-type: none"> • name

Name	Description	Value	Required	JSON script property
	source files' relative path or static value. Expression is supported for the latter.			• value

Next steps

Set up your Amazon S3 connection

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Amazon S3 Compatible connector overview

Article • 11/15/2023

This Amazon S3 Compatible connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

Data Factory in Microsoft Fabric doesn't currently support Amazon S3 Compatible connectors in Dataflow Gen2.

Support in data pipelines

The Amazon S3 Compatible connector supports the following capabilities in data pipelines:

Supported capabilities	Gateway	Authentication
Copy activity (source/-)	None	Access Key
Lookup activity	None	Access Key
GetMetadata activity	None	Access Key
Delete activity	None	Access Key

To learn more about the copy activity configuration for Amazon S3 Compatible in data pipelines, go to [Configure in a data pipeline copy activity](#).

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Configure Amazon S3 Compatible in a copy activity

Article • 11/24/2023

This article outlines how to use the copy activity in a data pipeline to copy data from Amazon S3 Compatible.

Required permissions

To copy data from Amazon S3 Compatible, make sure you've been granted the following permissions for Amazon S3 Compatible object operations: `s3:GetObject` and `s3:GetObjectVersion`.

In addition, `s3>ListAllMyBuckets` and `s3>ListBucket/s3:GetBucketLocation` permissions are required for operations like testing connection and browsing from root.

For the full list of Amazon S3 Compatible permissions, go to [Specifying Permissions in a Policy](#) on the AWS site.

Supported format

Amazon S3 Compatible supports the following file formats. Refer to each article for format-based settings.

- Avro format
- Binary format
- Delimited text format
- Excel format
- JSON format
- ORC format
- Parquet format
- XML format

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General

- Source
- Mapping
- Settings

General

Refer to the General settings guidance to configure the **General** settings tab.

Source

The following properties are supported for Amazon S3 Compatible under the **Source** tab of a copy activity.



The following properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select an Amazon S3 Compatible connection from the connection list. If no connection exists, then create a new Amazon S3 Compatible connection by selecting **New**.
- **Connection type:** Select **Amazon S3 Compatible** for your connection type.
- **File path type:** You can choose **File path**, **Prefix**, **Wildcard file path**, or **List of files** as your file path type. The configuration for each setting is:
 - **File path:** If you choose this type, the data can be copied from the given bucket or the given bucket and folder path specified.
 - **Prefix:** If you choose this type, specify the **Bucket** and **Prefix**.

- **Bucket:** Specify the S3 Compatible Storage bucket name. It is required.
- **Prefix:** Specify the configured prefix for the S3 key name under the given bucket to filter source S3 files. S3 keys whose names start with `bucket/this_prefix` are selected. It utilizes S3's service-side filter, which provides better performance than a wildcard filter.

When you use prefix and choose to copy to file-based destination with preserving hierarchy, note the sub-path after the last "/" in prefix will be preserved. For example, you have source `bucket/folder/subfolder/file.txt`, and configure prefix as `folder/sub`, then the preserved file path is `subfolder/file.txt`.



- **Wildcard file path:** If you choose this type, specify the **Bucket** and **Wildcard paths**.
 - **Bucket:** Specify the S3 Compatible Storage bucket name. It is required.
 - **Wildcard paths:** Specify the folder or file path with wildcard characters under your given bucket to filter your source folders or files.

Allowed wildcards are: `*` (matches zero or more characters) and `?` (matches zero or single character). Use `\` to escape if your folder name has wildcard or this escape character inside. See more examples in Folder and file filter examples.



Wildcard folder path: The folder path with wildcard characters under the given bucket to filter source folders.

Wildcard file name: The file name with wildcard characters under the given bucket and folder path (or wildcard folder path) to filter source files.

- **List of files:** If you choose this type, specify the **Folder path** and **Path to file list** to indicates to copy a given file set. Point to a text file that includes a list of files you want to copy, one file per line, which is the relative path to the path configured. For more examples, go to File list examples.



- **Folder path:** Specify the path to the folder under given bucket. It is required.
- **Path to file list:** Specify the path of the text file that includes a list of files you want to copy.
- **File format:** Select the file format applied from the drop-down list. Select **Settings** to configure the file format. For settings of different file formats, refer to articles in Supported format for detailed information.
- **Recursively:** Specify whether the data is read recursively from the subfolders or only from the specified folder. When **Recursively** is selected and the destination is a file-based store, an empty folder or subfolder isn't copied or created at the destination. This property is selected by default and doesn't apply when you configure **Path to file list**.

Under **Advanced**, you can specify the following fields:

- **Filter by last modified:** Files are filtered based on the last modified dates that you specified. This property doesn't apply when you configure your file path type as **List of files**.
 - **Start time (UTC):** The files are selected if their last modified time is greater than or equal to the configured time.
 - **End time (UTC):** The files are selected if their last modified time is less than the configured time.

When **Start time (UTC)** has datetime value but **End time (UTC)** is NULL, it means the files whose last modified attribute is greater than or equal with the datetime value will be selected. When **End time (UTC)** has datetime value but **Start time (UTC)** is NULL, it means the files whose last modified attribute is less than the datetime value will be selected. The properties can be NULL, which means no file attribute filter will be applied to the data.

- **Enable partition discovery:** Specify whether to parse the partitions from the file path and add them as additional source columns. It is unselected by default and not supported when you use binary file format.
 - **Partition root path:** When partition discovery is enabled, specify the absolute root path in order to read partitioned folders as data columns.

If it is not specified, by default,

- When you use file path or list of files on source, partition root path is the path that you configured.
- When you use wildcard folder filter, partition root path is the sub-path before the first wildcard.
- When you use prefix, partition root path is sub-path before the last "/".

For example, assuming you configure the path as

`root/folder/year=2020/month=08/day=27`:

- If you specify partition root path as `root/folder/year=2020`, copy activity will generate two more columns month and day with value "08" and "27" respectively, in addition to the columns inside the files.
- If partition root path is not specified, no extra column will be generated.



- **Max concurrent connection:** The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.
- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.

Mapping

For **Mapping** tab configuration, go to Configure your mappings under mapping tab. If you choose Binary as your file format, mapping will not be supported.

Settings

For the **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

The following tables contain more information about the copy activity in Amazon S3 Compatible.

Source information

[Expand table](#)

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	<your Amazon S3 Compatible connection>	Yes	connection
Connection type	Select a type for your connection.	Amazon S3 Compatible	Yes	/
File path type	The file path type used to get source data.	<ul style="list-style-type: none">• File path• Prefix• Wildcard file path• List of files	Yes	<ul style="list-style-type: none">• folderPath, fileName• prefix• wildcardFolderPath, wildcardFileName• fileListPath
File path	The folder/file path to the source file.	< file path>	Yes for File path	bucketName fileName folderpath
Bucket	The S3 Compatible Storage bucket name.	<your bucket name>	Yes	bucketName
Prefix	The configured prefix for the S3 key name under the given bucket to filter source S3 files.	<your prefix>	No	prefix
Wildcard paths	The folder/file path with wildcard characters under the configured bucket to filter source folders/files.	< your wildcard file path >	Yes for Wildcard file name	wildcardFolderPath wildcardFileName
Path to file list	Indicates to copy a given file set. Point to a text file that includes a list of files you want to copy, one file per line.	< file list path >	No	fileListPath
File format	The file format for your source data. For the information of different file formats, refer to articles in Supported	/	Yes	/

Name	Description	Value	Required	JSON script property
	format for detailed information.			
Recursively	Indicates whether the data is read recursively from the subfolders or only from the specified folder. Note that when Recursively is selected and the destination is a file-based store, an empty folder or subfolder isn't copied or created at the destination. This property doesn't apply when you configure Path to file list .	selected (default) or unselect	No	recursive
Filter by last modified	The files with last modified time in the range [Start time, End time) will be filtered for further processing. The time will be applied to UTC time zone in the format of <code>yyyy-mm-ddThh:mm:ss.fffZ</code> . These properties can be skipped which means no file attribute filter will be applied. This property doesn't apply when you configure your file path type as List of files.	datetime	No	modifiedDatetimeStart modifiedDatetimeEnd
Enable partition discovery	Indicates whether to parse the partitions from the file path and add them as additional source columns.	selected or unselected (default)	No	enablePartitionDiscovery: true or false (default)
Max concurrent connection	The upper limit of concurrent connections established to the data store during the activity run. Specify a value	<max concurrent connections>	No	maxConcurrentConnections

Name	Description	Value	Required	JSON script property
	only when you want to limit concurrent connections.			
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.	<ul style="list-style-type: none">• Name• Value	No	additionalColumns: <ul style="list-style-type: none">• name• value

Next steps

- Amazon S3 Compatible overview
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Azure Analysis Services connector overview

Article • 11/15/2023

The Azure Analysis Services connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to Azure Analysis Services in Dataflow Gen2, go to Set up your Azure Analysis Services connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support Azure Analysis Services in data pipelines.

Feedback

Was this page helpful?



[Provide product feedback ↗](#) | [Ask the community ↗](#)

Set up your Azure Analysis Services connection

Article • 11/15/2023

This article outlines the steps to create an Azure Analysis Services connection.

Supported authentication types

The Azure Analysis Services connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Organizational account	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Azure Analysis Services. The following links provide the specific Power Query connector information you need to connect to Azure Analysis Services in Dataflow Gen2:

- To get started using the Azure Analysis Services connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- Be sure to install or set up any Azure Analysis Services prerequisites before connecting to the Azure Analysis Services connector.
- To connect to the Azure Analysis Services connector from Power Query, go to [Connect to Azure Analysis Services database from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support Azure Analysis Services in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Azure Blob Storage connector overview

Article • 11/15/2023

The Azure Blob Storage connector is supported in Data Factory in Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to Azure Blob Storage in Dataflow Gen2, go to Set up your connection in Dataflow Gen2.

Support in data pipelines

The Azure Blob Storage connector supports the following capabilities in data pipelines.

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	Anonymous Key OAuth2 Service principal Shared Access Signature (SAS)
Lookup activity	None	Anonymous Key OAuth2 Service principal Shared Access Signature (SAS)
GetMetadata activity	None	Anonymous Key OAuth2 Service principal Shared Access Signature (SAS)

To learn about how to connect to Azure Blob Storage in data pipelines, go to Set up your Azure Blob Storage connection.

To learn about the copy activity configuration for Azure Blob Storage in data pipelines, go to Configure Azure Blob Storage in a copy activity.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your Azure Blob Storage connection

Article • 11/15/2023

Azure Blob Storage is Microsoft's object storage solution for the cloud. Blob storage is optimized for storing massive amounts of unstructured data. This article outlines the steps to create an Azure Blob Storage connection.

Supported authentication types

The Azure Blob Storage connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Anonymous	✓	✓
Account key	✓	✓
Shared Access Signature (SAS)	✓	✓
Organizational account	✓	✓
Service principal	✓	✓

Set up your connection in Dataflow Gen2

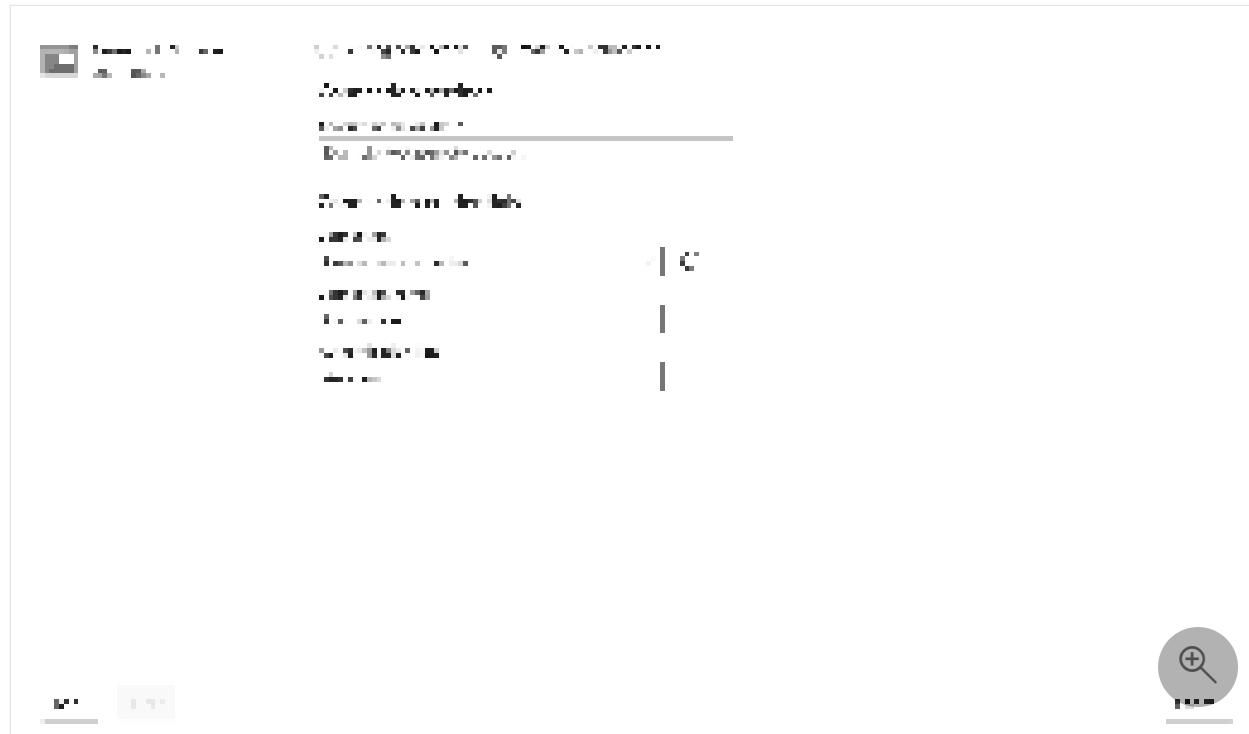
Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Azure Blobs. The following links provide the specific Power Query connector information you need to connect to Azure Blobs in Dataflow Gen2:

- To get started using the Azure Blobs connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- Be sure to install or set up any Azure Blobs prerequisites before connecting to the Azure Blobs connector.
- To connect to the Azure Blobs connector from Power Query, go to [Connect to Azure Blob Storage from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Browse to the New connection page for the data factory pipeline to configure the connection details and create the connection.



You have two ways to browse to this page:

- In copy assistant, browse to this page after selecting the connector.
- In pipeline, browse to this page after selecting + New in Connection section and selecting the connector.

Step 1: Specify the account name or URL, connection and connection name

A screenshot of the 'New connection' page showing the 'Connection settings' and 'Connection credentials' sections. The 'Connection settings' section contains a field labeled 'Account name or URL *' with the value 'testblobstorageaccount'. The 'Connection credentials' section contains a 'Connection' dropdown menu set to 'Create new connection' with a refresh icon next to it, and a 'Connection name' input field containing 'AzureBlobStorage'.

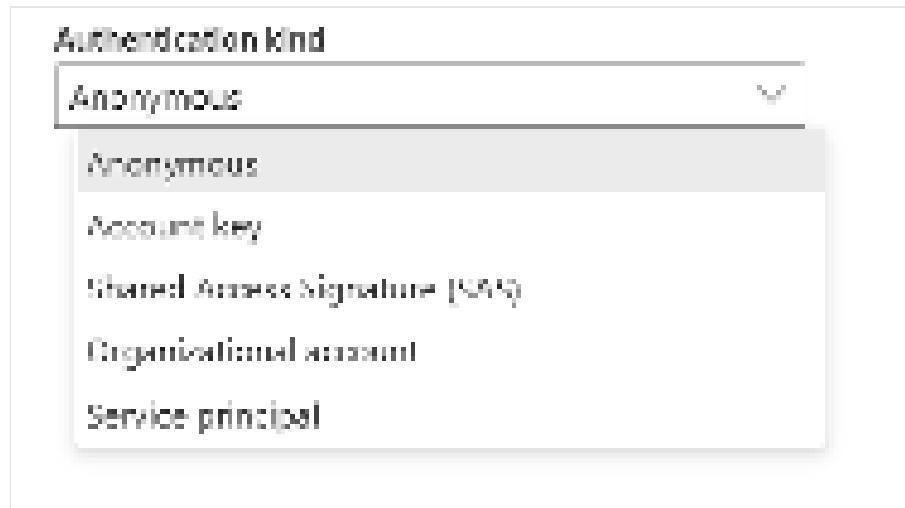
In the **New connection** pane, specify the following fields:

- **Account name or URL:** Specify your Azure Blob Storage account name or URL. Browse to the **Endpoints** section in your storage account and the blob service endpoint is the account URL.
- **Connection:** Select **Create new connection**.
- **Connection name:** Specify a name for your connection.

Step 2: Select and set your authentication kind

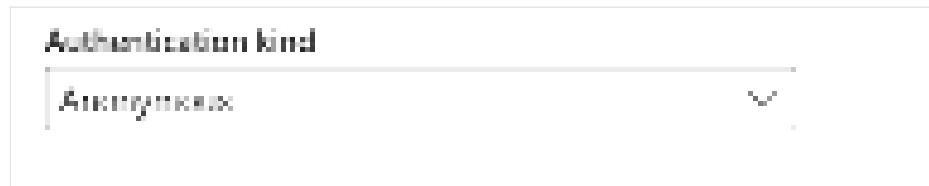
Under **Authentication kind**, select your authentication kind from the drop-down list and complete the related configuration. The Azure Blob Storage connector supports the following authentication types:

- Anonymous
- Account key
- Shared Access Signature (SAS)
- Organizational account
- Service principal



Anonymous authentication

Select **Anonymous** under **Authentication kind**.



Account key authentication

Specify the account key of your Azure Blob Storage. Go to your Azure Blob Storage account interface, browse to the **Access key** section, and get your account key.

The screenshot shows a dropdown menu labeled 'Authentication kind'. The option 'Account key' is selected and highlighted in blue. Below the dropdown is a text input field labeled 'Account key' containing a series of asterisks ('*****').

Shared Access Signature (SAS) authentication

Specify the shared access signature token (SAS token) to the storage resources, such as a blob or container.

The screenshot shows a dropdown menu labeled 'Authentication kind'. The option 'Shared Access Signature (SAS)' is selected and highlighted in blue. Below the dropdown is a text input field labeled 'SAS token' containing a series of asterisks ('*****').

If you don't have a SAS token, switch to **Shared access signature** in your Azure Blob Storage account interface. Under **Allowed resource types**, select **Service**. Then select **Generate SAS and connection string**. You can get your SAS token from the **SAS token** that's displayed.

The shared access signature is a URI that encompasses in its query parameters all the information necessary for authenticated access to a storage resource. To access storage resources with the shared access signature, the client only needs to pass in the shared access signature to the appropriate constructor or method.

For more information about shared access signatures, go to Shared access signatures: Understand the shared access signature model.

Organizational account authentication

Select **Sign in**, which displays the sign in interface. Enter your account and password to sign in your organizational account. After signing in, go back to the **New connection** page.



Service principal authentication

You need to specify the tenant ID, service principal client ID and service principal key when using this authentication.

A screenshot of a configuration form for 'Service principal' authentication. It includes three text input fields: 'Tenant ID' containing '<your tenant ID>', 'Service principal client ID' containing '<your application's client ID>', and 'Service principal Key' containing '*****'.

- **Tenant ID:** Specify the tenant information (domain name or tenant ID) under which your application resides. Retrieve it by hovering over the upper-right corner of the Azure portal.
- **Service principal client ID:** Specify the application's client ID.
- **Service principal Key:** Specify your application's key.

To use service principal authentication, follow these steps:

1. Register an application entity in Azure Active Directory (Azure AD) by following Authorize access to blobs using Azure Active Directory. Make note of these values, which you use to define the connection:
 - Tenant ID
 - Application ID
 - Application key
2. Grant the service principal proper permission in Azure Blob Storage. For more information on the roles, go to Assign an Azure role for access to blob data.

- As source, in Access control (IAM), grant at least the **Storage Blob Data Reader** role.
- As destination, in Access control (IAM), grant at least the **Storage Blob Data Contributor** role.

Step 3: Create your connection

Select **Create** to create your connection. Your creation is successfully tested and saved if all the credentials are correct. If not correct, the creation fails with errors.

Table summary

The following table contains the properties for data pipeline connection creation.

Name	Description	Required	Property	Copy
Account name or URL	Azure Blob Storage account name or endpoint.	Yes		✓
Connection	Select Create new connection .	Yes		✓
Connection name	A name for your connection.	Yes		✓
Authentication kind	Go to Authentication.	Yes	Go to Authentication.	

Authentication

The properties in the following table are the supported authentication types.

Name	Description	Required	Property	Copy
Anonymous				✓
Account key				✓
- Account key	The account key of the Azure Blob Storage.	Yes		
Shared Access Signature (SAS)				✓
- SAS token	The shared access signature token to the Storage resources, such as a blob or container.	Yes		

Name	Description	Required	Property	Copy
Organizational account				✓
Service principal				✓
- Tenant ID	The tenant information (domain name or tenant ID).	Yes		
- Service principal client ID	The application's client ID.	Yes		
- Service principal Key	The application's key.	Yes		

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback [↗](#) | Ask the community [↗](#)

Configure Azure Blob Storage in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in a data pipeline to copy data from and to Azure Blob Storage.

Supported format

Azure Blob Storage supports the following file formats. Refer to each article for format-based settings.

- Avro format
- Binary format
- Delimited text format
- Excel format
- JSON format
- ORC format
- Parquet format
- XML format

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Destination
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

The following properties are supported for Azure Blob Storage under the **Source** tab of a copy activity.



The following properties are required:

- **Data store type:** Select **External**.
 - **Connection:** Select an Azure Blob Storage connection from the connection list. If no connection exists, then create a new Azure Blob Storage connection by selecting **New**.
 - **File path:** Select **Browse** to choose the file that you want to copy, or fill in the path manually.
 - **File settings:** Select **File settings** to configure the file format. For settings of different file formats, refer to articles in Supported format for detailed information.

Under **Advanced**, you can specify the following fields:

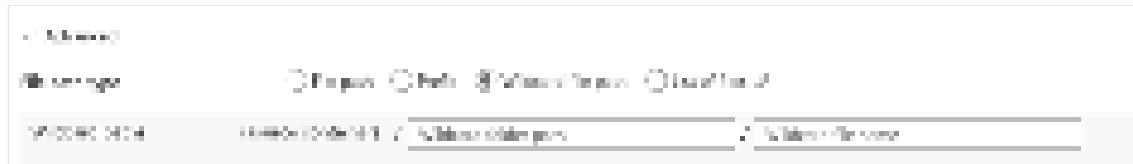
- **File path type:** You can choose **File path**, **Prefix**, **Wildcard file path**, **List of files** as your file path type. The configuration of each setting is:
 - **File path:** If you choose this type, the data can be copied from the given container or folder/file path specified previously.
 - **Prefix:** Prefix for the blob name under the given container configured to filter source blobs. Blobs whose names start with `container/this_prefix` are selected. It utilizes the service-side filter for blob storage.

When you use **Prefix** and choose to copy to a file-based destination with preserving hierarchy, the subpath after the last "/" in the prefix is preserved. For example, you have a source `container/folder/subfolder/file.txt`, and configure the prefix as `folder/sub`, then the preserved file path is `subfolder/file.txt`.



- **Wildcard file path:** Specify the folder or file path with wildcard characters under your given blob container to filter your source folders or files.

Allowed wildcards are `*` (matches zero or more characters) and `?` (matches zero or single character). Use `\` to escape if your folder name has a wildcard or this escape character inside. For more examples, go to [Folder and file filter examples](#).



Wildcard folder path: Specify the folder path with wildcard characters under your given container to filter source folders.

Wildcard file name: Specify the file name with wildcard characters under your given container and folder path (or wildcard folder path) to filter source files.

- **List of files:** Indicates a given file set to copy to. In **Path to file list**, enter or browse to a text file that includes a list of files you want to copy, one file per line, which is the relative path to each file.

When you're using this option, don't specify a file name. For more examples, go to [File list examples](#).



- **Recursively:** If this checkbox is selected, all files in the input folder and its subfolders are processed recursively. If you unselect the checkbox, just the ones in the selected folder are processed. This setting is disabled when a single file is selected.
- **Delete files after completion:** If this checkbox is selected, the binary files are deleted from source store after successfully moving to the destination store. The file deletion is per file, so when copy activity fails, you'll notice that some files have

already been copied to the destination and deleted from the source, while others are still remaining in the source store.

! Note

This property is only valid in a binary files copy scenario.

- **Max concurrent connections:** This property indicates the upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.

Destination

The following properties are supported for Azure Blob Storage under the **Destination** tab of a copy activity.



The following properties are **required**:

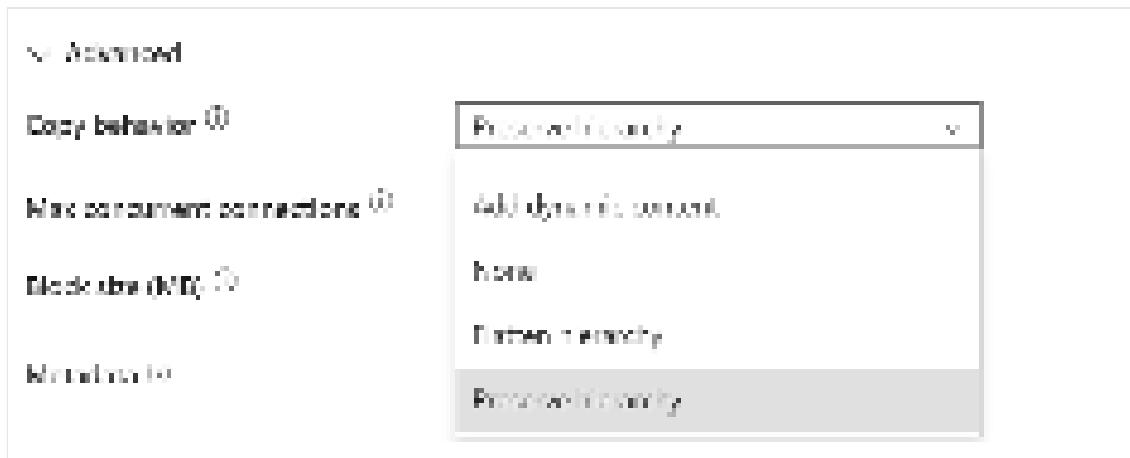
- **Data store type:** Select **External**.
- **Connection:** Select an Azure Blob Storage connection from the connection list. If the connection doesn't exist, then create a new Azure Blob Storage connection by selecting **New**.
- **File path:** Select **Browse** to choose the file that you want to copy or fill in the path manually.
- **File settings:** Select **File settings** to configure the file format. For settings of different file formats, refer to articles in Supported format for detailed information.

Under **Advanced**, you can specify the following fields:

- **Copy behavior:** Defines the copy behavior when the source is files from a file-based data store. You can choose **Add dynamic content**, **None**, **FlattenHierarchy**,

or **Preserve hierarchy** from the drop-down list.

- **Add dynamic content:** To specify an expression for a property value, select **Add dynamic content**. This selection opens the expression builder where you can build expressions from supported system variables, activity output, functions, and user-specified variables or parameters. For information about the expression language, go to Expressions and functions.
- **None:** Choose this selection to not use any copy behavior.
- **Flatten hierarchy:** All files from the source folder are in the first level of the destination folder. The destination files have autogenerated names.
- **Preserve hierarchy:** Preserves the file hierarchy in the target folder. The relative path of source file to source folder is identical to the relative path of target file to target folder.

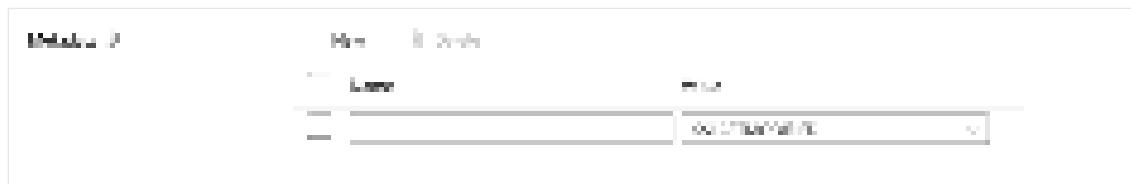


- **Max concurrent connections:** The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.
- **Block size (MB):** Specify the block size, in megabytes, used to write data to block blobs. For more information, go to About block blobs.
- **Metadata:** Set custom metadata when copying to a destination. Each object under the `metadata` array represents an extra column. The `name` defines the metadata key name, and the `value` indicates the data value of that key. If the `preserve` attributes feature is used, the specified metadata will union/overwrite with the source file metadata.

Allowed data values are:

- `$$LASTMODIFIED`: a reserved variable indicates to store the source files' last modified time. Apply to file-based source with binary format only.

- Expression
 - Static value



Mapping

For **Mapping** tab configuration, go to Configure your mappings under mapping tab. If you choose Binary as your file format, mapping won't be supported.

Settings

For **Settings** tab configuration, see [Configure your other settings under settings tab](#).

Table summary

The following tables contain more information about the copy activity in Azure Blob Storage.

Source information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	<your connection>	Yes	connection
File path	The file path of your source data.	<file path of your source>	Yes	container fileName
File path type	The file path type that you want to use.	<ul style="list-style-type: none">• File path• Prefix• Wildcard folder path, Wildcard file name• List of files	No	<ul style="list-style-type: none">• prefix• wildcardFolderPath, wildcardFileName• fileListPath

Name	Description	Value	Required	JSON script property
Recursively	Process all files in the input folder and its subfolders recursively or just the ones in the selected folder. This setting is disabled when a single file is selected.	Selected or unselect	No	recursive
Delete files after completion	The files in the source data store will be deleted right after being moved to the destination store. The file deletion is per file, so when a copy activity fails, you can tell that some files have already been copied to the destination and deleted from source, while others are still in the source store.	Selected or unselect	No	deleteFilesAfterCompletion
Max concurrent connections	The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.	<max concurrent connections>	No	maxConcurrentConnections

Destination information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the destination data store.	<your connection>	Yes	connection

Name	Description	Value	Required	JSON script property
File path	The file path of your destination data.	File path of the source	Yes	container fileName
Copy behavior	Defines the behavior when copying files from one file system, like storage, to the other (for example, from one blob storage to another).	<ul style="list-style-type: none"> None Add dynamic content Flatten hierarchy Preserve hierarchy 	No	copyBehavior
Max concurrent connections	The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.	<max concurrent connections>	No	maxConcurrentConnections
Block size (MB)	Specify the block size in MB when writing data to Azure Blob Storage. Allowed value is between 4 MB and 100 MB.	<block size>	No	blockSizeInMB
Metadata	Set the custom metadata when copy to destination.	<ul style="list-style-type: none"> \$\$LASTMODIFIED Expression Static value 	No	metadata

Next steps

- Set up your Azure Blob Storage connection

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#) | [Ask the community](#)

Azure Cosmos DB for NoSQL connector overview

Article • 11/15/2023

The Azure Cosmos DB for NoSQL connector is supported in Data Factory in Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

Data Factory in Microsoft Fabric doesn't currently support Azure Cosmos DB for NoSQL in Dataflow Gen2.

Support in data pipelines

The Azure Cosmos DB for NoSQL connector supports the following capabilities in data pipelines:

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	Key
Lookup activity	None	Key

To learn about how to connect to Azure Cosmos DB for NoSQL in data pipelines, go to Set up your Azure Cosmos DB for NoSQL connection.

To learn about the copy activity configuration for Azure Cosmos DB for NoSQL in data pipelines, go to Configure Azure Cosmos DB for NoSQL in a copy activity.

Feedback

Was this page helpful?



Yes



No

Provide product feedback | Ask the community

Set up your Azure Cosmos DB for NoSQL connection

Article • 11/15/2023

This article outlines the steps to create an Azure Cosmos DB for NoSQL connection.

Supported authentication types

The Azure Cosmos DB for NoSQL connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Account key	✓	n/a

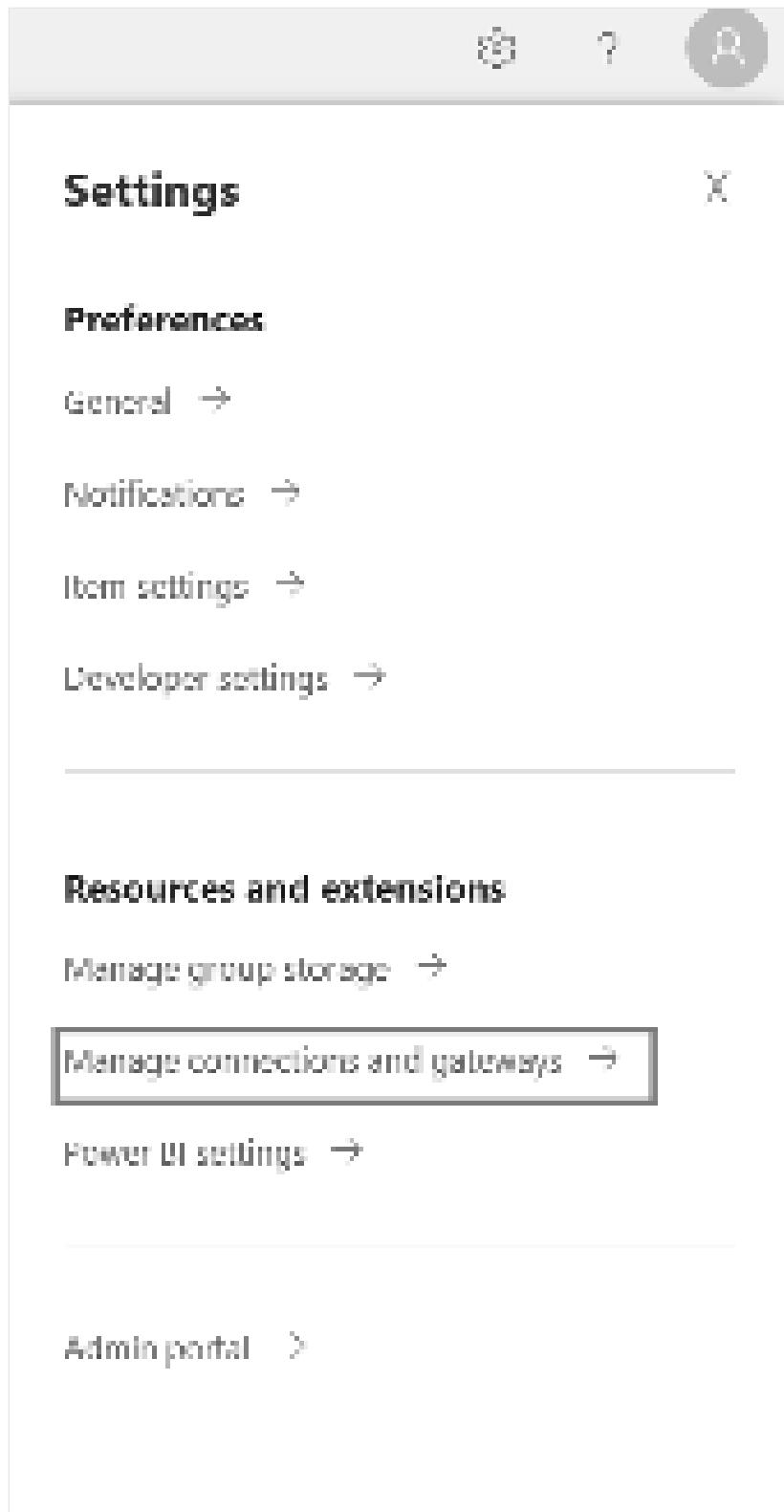
Set up your connection in Dataflow Gen2

The Azure Cosmos DB for NoSQL connector isn't currently supported in Dataflow Gen2.

Set up your connection in a data pipeline

To create a connection in a data pipeline:

1. From the page header in the Microsoft Fabric service, select **Settings**  > **Manage connections and gateways**.



2. Select **New** at the top of the ribbon to add a new data source.



The New connection pane opens on the left side of the page.



Setup connection

Step 1: Specify the new connection name, type, account endpoint and database

New connection



ⓘ Currently, these cloud connections are only supported for Data Pipelines and Kusto. In the future, other artifacts can also make use of the cloud connections. To create personal cloud connections in Datasets, Dataloads, and Dataloops, use the Power Query Online experience in 'Get data'.



Connection name *

Connection type *

Account Endpoint *

Database *

In the **New connection** pane, choose **Cloud**, and specify the following field:

- **Connection name:** Specify a name for your connection.
- **Connection type:** Select **Azure CosmosDB (Data pipeline)** for your connection type.
- **Account Endpoint:** Enter your account endpoint URL of your Azure Cosmos DB for NoSQL.
- **Database:** Enter the database ID of your Azure Cosmos DB for NoSQL.

Step 2: Select and set your authentication

Under **Authentication method**, select your authentication method from the drop-down list and complete the related configuration. The Azure Cosmos DB for NoSQL connector supports the following authentication types:

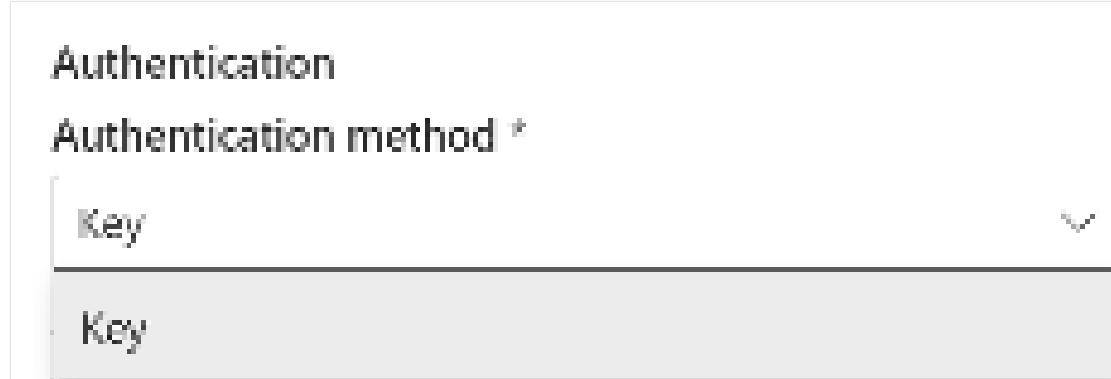
- Key

Authentication

Authentication method *

Key

Key



Key authentication

Account key: Specify the account key of your Azure Cosmos DB for NoSQL connection. Go to your Azure Cosmos DB for NoSQL account interface, browse to the **Keys** section, and get your account key.

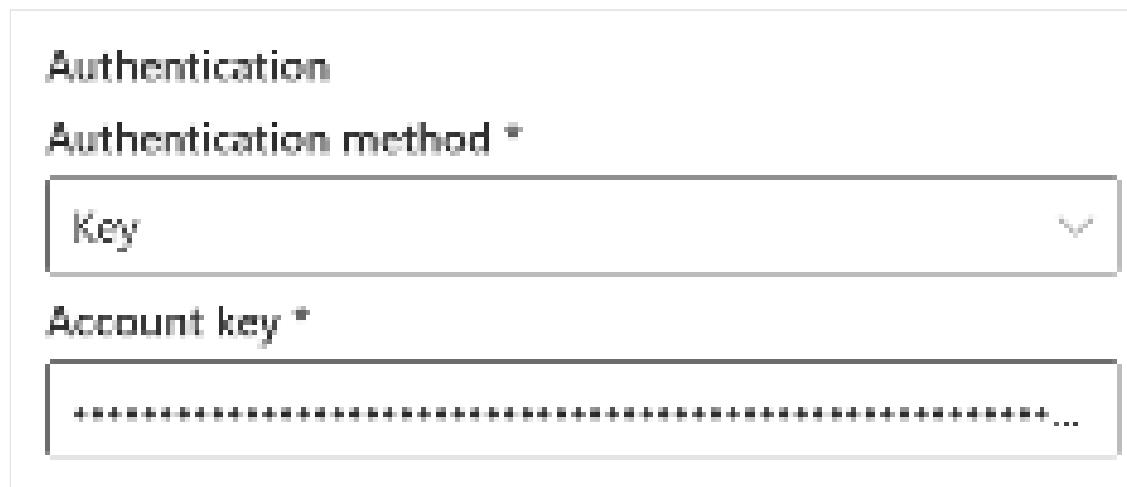
Authentication

Authentication method *

Key

Account key *

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX...



Step 3: Specify the privacy level that you want to apply

In the **General** tab, select the privacy level that you want apply in the **Privacy level** drop-down list. Three privacy levels are supported. For more information, see [privacy levels](#).

Step 4: Create your connection

Select **Create**. Your creation will be successfully tested and saved if all the credentials are correct. If not correct, the creation will fail with errors.

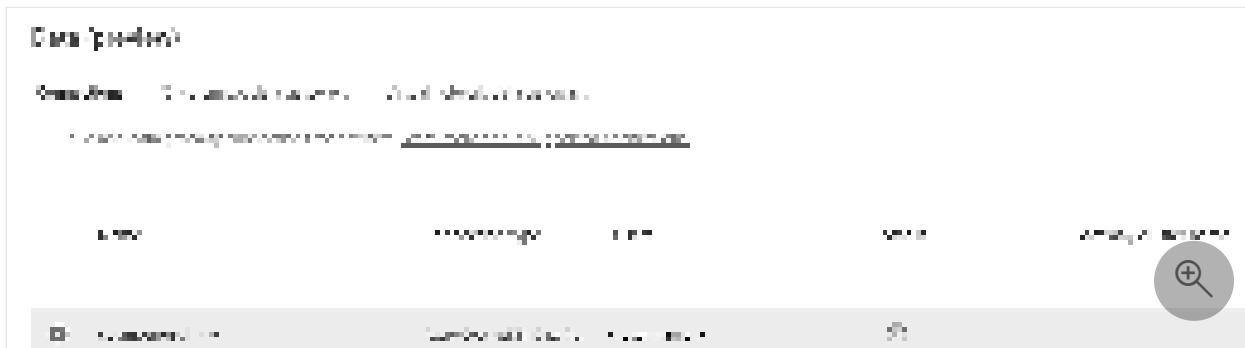


Table summary

The connector properties in the following table are supported in pipeline copy:

Name	Description	Required	Property	Copy
Connection name	A name for your connection.	Yes		✓
Connection type	Select Azure CosmosDB (Data pipeline) for your connection type.	Yes		✓
Account Endpoint	Enter your Azure Cosmos DB for NoSQL account endpoint URL.	Yes		✓
Database	Enter the Azure Cosmos DB for NoSQL database ID.	Yes		✓
Authentication	Go to Authentication.	Yes	Go to Authentication.	Go to Authentication.
Privacy Level	The privacy level that you want to apply. Allowed values are Organizational , Privacy , and Public .	Yes		✓

Authentication

The properties in the following table are the supported authentication types.

Name	Description	Required	Property	Copy
Key				✓

Name	Description	Required	Property	Copy
- Account key	The Azure Cosmos DB for NoSQL account key.	Yes		

Next steps

- Configure Azure Cosmos DB for NoSQL in a copy activity
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback [↗](#) | Ask the community [↗](#)

Configure Azure Cosmos DB for NoSQL in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in a data pipeline to copy data from and to Azure Cosmos DB for NoSQL.

Supported configuration

For the configuration of each tab under the copy activity, go to the following sections respectively.

- General
- Source
- Destination
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

The following properties are supported for Azure Cosmos DB for NoSQL under the **Source** tab of a copy activity.

The screenshot shows the 'Source' tab of the Azure Data Factory Copy Activity configuration pane. The tab navigation bar at the top includes 'General', 'Source' (which is selected and highlighted in blue), 'Destination', 'Mapping', and 'Settings'. The main configuration area contains the following fields:

- Data store type:** Radio buttons for 'Workspace' (selected), 'External', and 'Sample dataset'.
- Connection:** A dropdown menu showing 'cosmosdb' with a small arrow indicating it's a dropdown.
- Container:** A dropdown menu showing 'None' with a small arrow indicating it's a dropdown.
- Filter:** A checkbox labeled 'Filter'.
- Advanced:** A button with a downward arrow.
- User query:** Radio buttons for 'Table' (selected) and 'Query'.
- Page size:** An input field containing '100' with a small info icon.
- Output笙etime:** A checkbox with a checked state.
- Preferred regions:** A plus sign followed by 'None'.
- Additional columns:** A plus sign followed by 'None'.
- Search icon:** A circular icon with a magnifying glass and a plus sign, located in the bottom right corner.

The following three properties are required:

- **Data store type:** Select **External**.
- **Connection:** Select an Azure Cosmos DB for NoSQL connection from the connection list. If no connection exists, then create a new Azure Cosmos DB for NoSQL connection by selecting **New**.
- **Container:** Select the container that you want to use. Select **Edit** to enter the container name manually.

Under **Advanced**, you can specify the following fields:

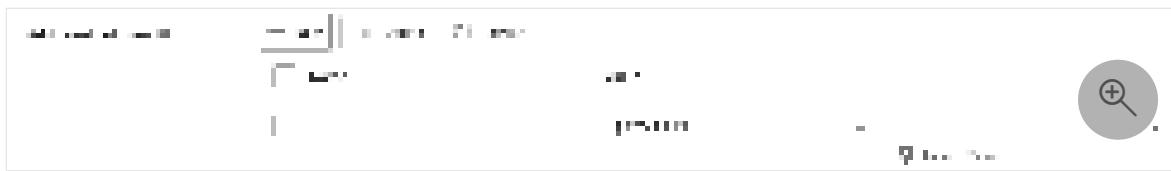
- **Use query:** You can choose either **Table** or **Query** as your use query. The following list describes the configuration of each setting.
 - **Table:** Reads data from the table you specified in **Table**.
 - **Query:** Specifies the Azure Cosmos DB query to read data.



- **Page size:** The number of documents per page of the query result. Default is "-1", which means to use the service side dynamic page size up to 1000.
- **Detect datetime:** Whether to detect datetime from the string values in the documents. Allowed values are: true (default), false.
- **Preferred regions:** The preferred list of regions to connect to when retrieving data from Azure Cosmos DB. Select one preferred region from the drop-down list after selecting **New**.



- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter. For more information, go to Add additional columns during copy.



Destination

The following properties are supported for Azure Cosmos DB for NoSQL under the **Destination** tab of a copy activity.

A screenshot of the Azure portal's destination configuration pane. At the top, there are tabs for General, Source, Destination, Mapping, and Settings. The Destination tab is selected. Below the tabs, there are several configuration fields: 'Data store type' (with options 'File/Share' and 'External' selected), 'Connection' (a dropdown menu showing '1 connection'), 'Container' (a dropdown menu showing 'None' and a 'Browse' button), and 'Advanced' settings which include 'Write behavior' (set to 'Insert'), 'Write batch timeout', 'Write batch size', 'Max concurrent connections' (set to 100), and 'Enable performance metrics analysis'. A search icon is located in the bottom right corner.

The following three properties are required:

- **Data store type:** Select **External**.
- **Connection:** Select an Azure Cosmos DB for NoSQL connection from the connection list.
- **Container:** Select **Browse** to choose the file that you want to copy, or fill in the path manually.

Under **Advanced**, you can specify the following fields:

- **Write behavior:** Defines the write behavior when the destination is files from a file-based data store. You can choose **Add dynamic content**, **Insert**, or **Upsert** from the drop-down list.
 - **Add dynamic content:** Open the **Add dynamic content** pane. This opens the expression builder where you can build expressions from supported system variables, activity output, functions, and user-specified variables or parameters. For information about the expression language, go to Expressions and functions.

- **Insert:** Choose this option if your source data has inserts.
- **Upsert:** The behavior of upsert is to replace the document if a document with the same ID already exists; otherwise, insert the document.



- **Write batch timeout:** Wait time for the batch insert operation to complete before it times out. Allowed values are `Timespan`. An example is `00:30:00` (30 minutes).
- **Write batch size:** Specify the number of rows to insert into the SQL table per batch. The allowed value is an integer (number of rows). By default, the service dynamically determines the appropriate batch size based on the row size.
- **Max concurrent connections:** The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.
- **Disable performance metrics analytics:** This setting controls the collection of metrics, such as DTU, DWU, RU, and so on for copy performance optimization and recommendations. If you're concerned with this behavior, turn off this feature.

Mapping

For the **Mapping** tab configuration, go to [Configure your mappings under mapping tab](#).

Settings

For the **Settings** tab configuration, go to [Configure your other settings under settings tab](#).

Table summary

The following tables contain more information about the copy activity in Azure Cosmos DB for NoSQL.

Source table

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	<your connection>	Yes	connection
Container	The container of your source data.	<container of your source>	Yes	container fileName
Use query	You can choose Table or Query as your use query.	<ul style="list-style-type: none"> • Table • Query 	No	type
Page size	<p>The number of documents per page of the query result. Default is "-1", which means to use the service side dynamic page size up to 1000.</p>	<your Page size>	No	pageSize
Delete datetime	<p>The files on source data store will be deleted right after being moved to the destination store. The file deletion is per file, so when copy activity fails, you'll note that some files have already been copied to the destination and deleted from source while others are still on source store.</p>	Selected or unselect	No	detectDatetime
Preferred regions	The preferred list of regions to connect to when retrieving data from Azure Cosmos DB. Select one preferred region from the drop-down list after selecting New.	<your preferred regions>	No	preferredRegions
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter. For more information, go to Add additional columns during copy.	<max concurrent connections>	No	additionalColumns

Destination table

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the destination data store.	<your connection>	Yes	connection
Container	The container of your destination data.	<container of source>	Yes	container fileName
Write behavior	Describes how to write data to Azure Cosmos DB. Allowed values: insert and upsert. The behavior of upsert is to replace the document if a document with the same ID already exists; otherwise, insert the document.	<ul style="list-style-type: none"> • Add dynamic content • Insert • Upsert 	No	writeBehavior
Write batch timeout	Wait time for the batch insert operation to complete before it times out. Allowed values are Timespan. An example is 00:30:00 (30 minutes).	timespan	No	writeBatchTimeout
Write batch size	The number of rows to insert into the SQL table per batch. The allowed value is integer (number of rows). By default, the service dynamically determines the appropriate batch size based on the row size.	<number of rows > (integer)	No	writeBatchSize
Max concurrent connections	The upper limit of concurrent connections established to the data store during the	<max concurrent connections>	No	maxConcurrentConnections

Name	Description	Value	Required	JSON script property
	activity run. Specify a value only when you want to limit concurrent connections.			
Disable performance metrics analytics	This setting controls collection of metrics such as DTU, DWU, RU, and so on for copy performance optimization and recommendations. If you're concerned with this behavior, turn off this feature.	Selected or unselect	No	disableMetricsCollection

Next steps

- Set up your Azure Cosmos DB for NoSQL connection
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Azure Data Explorer overview

Article • 11/15/2023

This Azure Data Explorer connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to Azure Data Explorer in Dataflow Gen2, go to Set up your connection in Dataflow Gen2.

Support in data pipeline

The Azure Data Explorer connector supports the following capabilities in data pipelines:

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	User Auth
Lookup activity	None	User Auth

To learn more about the copy activity configuration for Azure Data Explorer in data pipelines, go to [Configure in a data pipeline copy activity](#).

Feedback

Was this page helpful?



Yes



No

Provide product feedback | Ask the community

Set up your Azure Data Explorer connection

Article • 11/15/2023

This article outlines the steps to create an Azure Data Explorer connection.

Supported authentication types

The Azure Data Explorer connector supports the following authentication types for copy and Dataflow Gen2 respectively.

[] Expand table

Authentication type	Copy	Dataflow Gen2
Organizational account	✓	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Azure Data Explorer. The following links provide the specific Power Query connector information you need to connect to Azure Data Explorer in Dataflow Gen2:

- To get started using the Azure Data Explorer connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- Be sure to install or set up any Azure Data Explorer prerequisites before connecting to the Azure Data Explorer connector.
- To connect to the Azure Data Explorer connector from Power Query, go to [Connect to Azure Data Explorer from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#) | [Ask the community](#)

Configure Azure Data Explorer in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in a data pipeline to copy data from and to Azure Data Explorer.

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

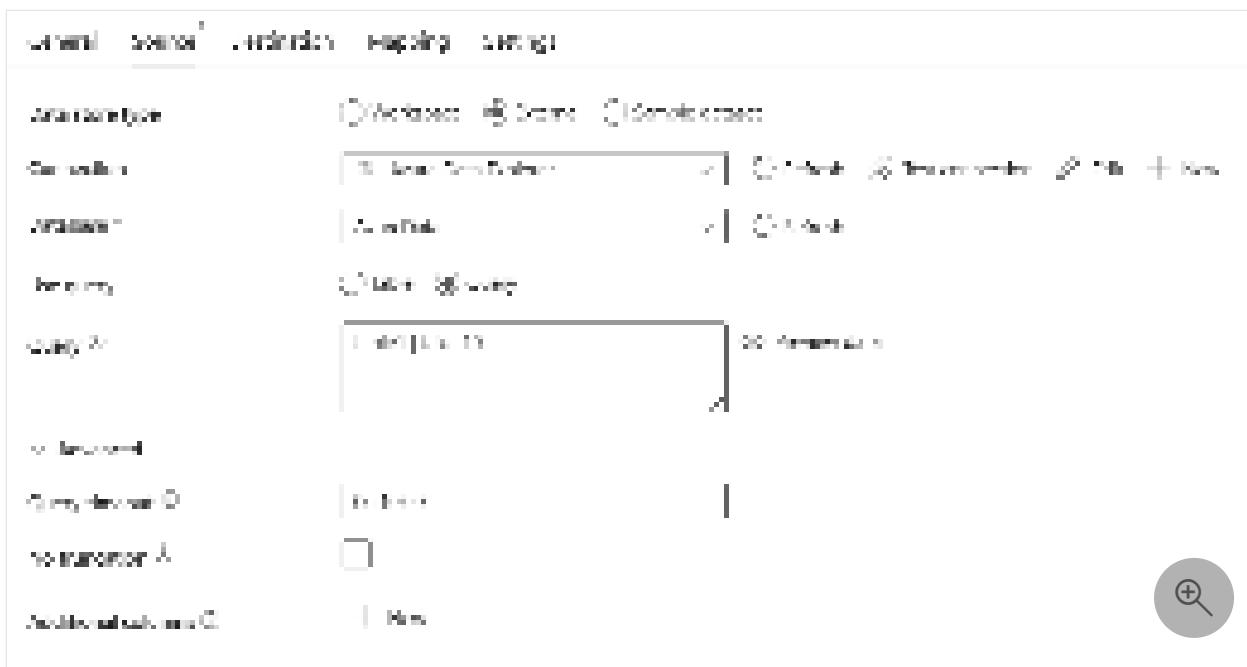
- General
- Source
- Destination
- Mapping
- Settings

General

For **General** tab configuration, go to General.

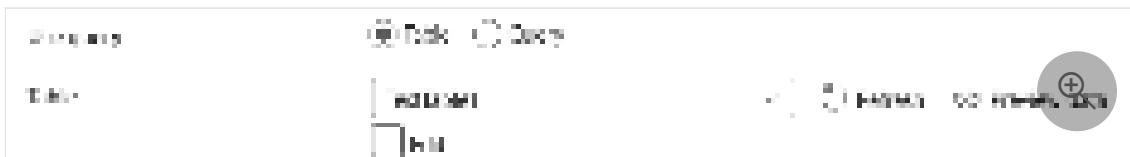
Source

The following properties are supported for Azure Data Explorer under the **Source** tab of a copy activity.



The following properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select an Azure Data Explorer connection from the connection list. If no connection exists, then create a new Azure Data Explorer connection by selecting **New**.
- **Database:** Select your database from the drop-down list.
- **Use query:** Select **Table or Query**.
 - **Table:** Select a table from the drop-down list or select **Edit** to manually enter it to read data.



- **Query:** Specify the read-only request given in a KQL format. Use the custom KQL query as a reference.



Under **Advanced**, you can specify the following fields:

- **Query timeout:** Specify the wait time before the query request times out. Default value is 10 minutes (00:10:00). Allowed max value is 1 hour (01:00:00).
- **No truncation:** Indicates whether to truncate the returned result set. By default result is truncated after 500,000 records or 64 MB. Truncation is strongly recommended for a proper behavior of the activity.
- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.

Destination

The following properties are supported for Azure Data Explorer under the **Destination** tab of a copy activity.



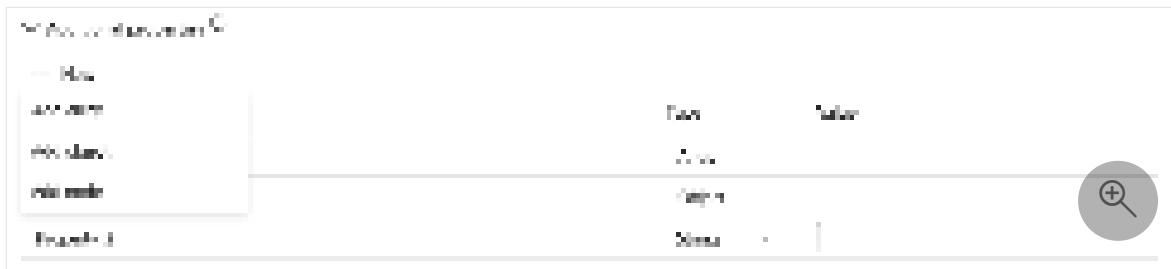
The following properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select an Azure Data Explorer connection from the connection list. If no connection exists, then create a new Azure Data Explorer connection by selecting **New**.
- **Database:** Select your database from the drop-down list.
- **Table:** Select a table from the drop-down list or select **Edit** to manually enter it to write data.

Under **Advanced**, you can specify the following fields:

- **Ingestion mapping name:** Specify the name of a mapping that was pre-created and assigned to an Azure Data Explorer destination table in advance.
- **Additional properties:** A property bag that can be used for specifying any of the ingestion properties that aren't being set already by the Azure Data Explorer

destination. Specifically, it can be useful for specifying ingestion tags. To learn more, go to Azure Data Explorer data ingestion.



Mapping

For **Mapping** tab configuration, go to Configure your mappings under mapping tab.

Settings

For **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

The following tables contain more information about a copy activity in an Azure Data Explorer.

Source information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	< your Azure Data Explorer connection >	Yes	connection
Database	Your database that you use as source.	< your database >	Yes	database
Use query	The way to read data. Apply Table to read data from the specified table or apply Query to read data using queries.	<ul style="list-style-type: none">• Table• Query	No	table query
Query timeout	Specify the wait time before the query request times out.	timespan (the default is	No	queryTimeout

Name	Description	Value	Required	JSON script property
	Allowed maximum value is 1 hour (01:00:00).	00:10:00 - 10 minutes)		
No truncation	Indicates whether to truncate the returned result set. By default, the result is truncated after 500,000 records or 64 MB. Truncation is strongly recommended for a proper behavior of the activity.	select or unselect	No	noTruncation: true or false
Additional columns	Add additional data columns to the store source files' relative path or static value. Expression is supported for the latter.	<ul style="list-style-type: none"> • Name • Value 	No	additionalColumns: <ul style="list-style-type: none"> • name • value

Destination information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the destination data store.	< your Azure Data Explorer connection >	Yes	connection
Database	Your database that you use as destination.	< your database >	Yes	database
Table	Your destination data table to write data.	<your table name>	Yes	table
Ingestion mapping name	The name of a mapping that was pre-created and assigned to Azure Data Explorer destination table in advance.	<your ingestion mapping name>	No	ingestionMappingName
Additional properties	A property bag that can be used for specifying any of the ingestion properties that aren't being set already by the Azure Data Explorer destination. Specifically, it	<ul style="list-style-type: none"> • Name • Type • Value 	No	additionalProperties

Name	Description	Value	Required	JSON script property
	can be useful for specifying ingestion tags. Learn more from Azure Data Explorer data ingestion.			

Next steps

- Azure Data Explorer connector overview
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback [↗](#) | Ask the community [↗](#)

Azure Data Lake Storage Gen1 connector overview

Article • 11/30/2023

The Azure Data Lake Storage Gen1 connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Supported in Data pipeline

The Azure Data Lake Storage Gen1 connector supports the following capabilities in Data pipeline:

 Expand table

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	Service principal
Lookup activity	None	Service principal
GetMetadata activity	None	Service principal
Delete data activity	None	Service principal

To learn more about the copy activity configuration for Azure Data Lake Storage Gen1 in Data pipeline, go to [Configure in a data pipeline copy activity](#).

Feedback

Was this page helpful?



Yes



No

[Provide product feedback](#) | [Ask the community](#)

Configure Azure Data Lake Storage Gen1 in copy activity

Article • 11/30/2023

This article outlines how to use the copy activity in data pipeline to copy data from and to Azure Data Lake Storage Gen1.

Supported format

Azure Data Lake Storage Gen1 supports the following file formats. Refer to each article for format-based settings.

- Avro format
- Binary format
- Delimited text format
- Excel format
- JSON format
- ORC format
- Parquet format
- XML format

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

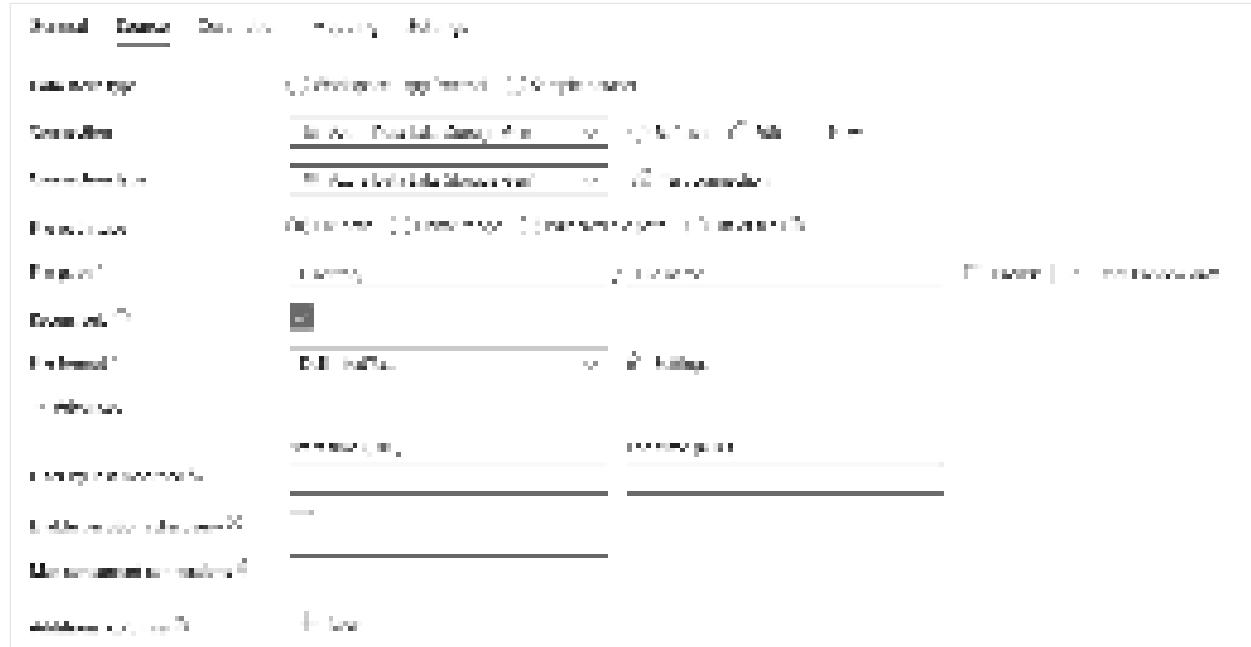
- General
- Source
- Destination
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

The following properties are supported for Azure Data Lake Storage Gen1 under the **Source** tab of a copy activity.



The following properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select an Azure Data Lake Storage Gen1 connection from the connection list. If no connection exists, then create a new Azure Data Lake Storage Gen1 connection by selecting **New**.
- **Connection type:** Select **Azure Data Lake Storage Gen1**.
- **File path type:** You can choose **File path**, **Name range**, **Wildcard file path**, or **List of files** as your file path type. The configuration of each of these settings is:
 - **File path:** If you choose this type, the data can be copied from the specified folder/file path.
 - **Name range:** Retrieve folders/files with names before or after a specific value in alphabetical order. It utilizes the service-side filter for ADLS Gen1, which provides better performance than a wildcard filter. For more examples, got to [Name range filter examples](#).



- **Folder path:** The path to a folder in your source data.

- **List after:** Retrieve the folders/files whose name is after this value alphabetically (exclusive).
- **List before:** Retrieve the folders/files whose name is before this value alphabetically (inclusive).
- **Wildcard file path:** Specify the folder or file path with wildcard characters to filter source folders or files.

Allowed wildcards are: `*` (matches zero or more characters) and `?` (matches zero or single character). Use `\` to escape if your folder name has wildcard or this escape character inside. For more examples, go to [Folder and file filter examples](#).



Wildcard folder path: Specify the folder path with wildcard characters to filter source folders.

Wildcard file name: Specify the file name with wildcard characters under the configured folder/wildcard folder path to filter source files.

- **List of files:** Indicates you want to copy a given file set. Specify **Folder path** and **Path to file list** to point to a text file that includes a list of files you want to copy, one file per line, which is the relative path to the path. For more examples, go to [File list examples](#).



- **Folder path:** Specify the path to a folder. It is required.
- **Path to file list:** Specify the path of the text file that includes a list of files you want to copy.
- **Recursively:** Specify whether the data is read recursively from the subfolders or only from the specified folder. Note that when **Recursively** is selected and the destination is a file-based store, an empty folder or subfolder isn't copied or created at the destination. This property is selected by default and doesn't apply when you configure **Path to file list**.

- **File format:** Select the file format applied from the drop-down list. Select **Settings** to configure the file format. For settings of different file formats, refer to articles in Supported format for detailed information.

Under **Advanced**, you can specify the following fields:

- **Filter by last modified:** Files are filtered based on the last modified dates. This property doesn't apply when you configure your file path type as List of files.
 - **Start time (UTC):** The files are selected if their last modified time is greater than or equal to the configured time.
 - **End time (UTC):** The files are selected if their last modified time is less than the configured time.

When **Start time (UTC)** has datetime value but **End time (UTC)** is NULL, it means the files whose last modified attribute is greater than or equal with the datetime value will be selected. When **End time (UTC)** has datetime value but **Start time (UTC)** is NULL, it means the files whose last modified attribute is less than the datetime value will be selected. The properties can be NULL, which means no file attribute filter will be applied to the data.

- **Enable partition discovery:** Specify whether to parse the partitions from the file path and add them as additional source columns. It is unselected by default and not supported when you use binary file format.
 - **Partition root path:** When partition discovery is enabled, specify the absolute root path in order to read partitioned folders as data columns.

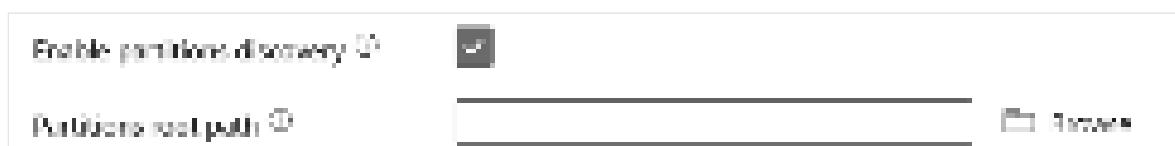
If it is not specified, by default,

 - When you use file path or list of files on source, partition root path is the path that you configured.
 - When you use wildcard folder filter, partition root path is the sub-path before the first wildcard.

For example, assuming you configure the path as

`root/folder/year=2020/month=08/day=27`:

- If you specify partition root path as `root/folder/year=2020`, copy activity will generate two more columns month and day with value "08" and "27" respectively, in addition to the columns inside the files.
- If partition root path is not specified, no extra column will be generated.



- **Max concurrent connections:** This property indicates the upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.
- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.

Destination

The following properties are supported for Azure Data Lake Storage Gen1 under the **Destination** tab of a copy activity.

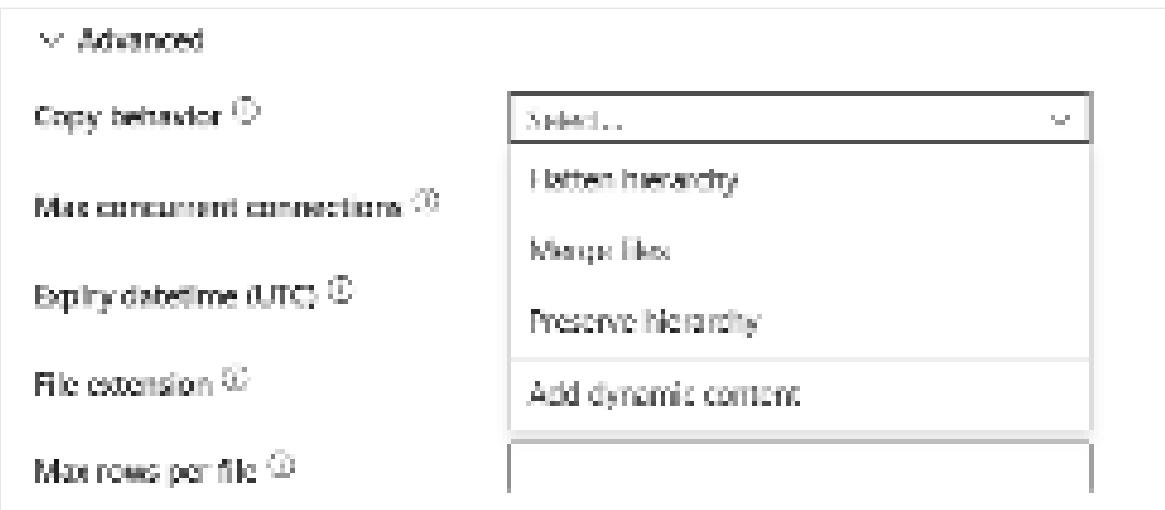


The following properties are required:

- **Data store type:** Select **External**.
- **Connection:** Select an Azure Data Lake Storage Gen1 connection from the connection list. If the connection doesn't exist, then create a new Azure Data Lake Storage Gen1 connection by selecting **New**.
- **Connection type:** Select **Azure Data Lake Storage Gen1**.
- **File path:** Select **Browse** to choose the file that you want to copy or fill in the path manually.
- **File format:** Select the file format applied from the drop-down list. Select **Settings** to configure the file format. For settings of different file formats, refer to articles in Supported format for detailed information.

Under **Advanced**, you can specify the following fields:

- **Copy behavior:** Defines the copy behavior when the source is files from a file-based data store. You can choose a behavior from the drop-down list.



- **Flatten hierarchy:** All files from the source folder are in the first level of the destination folder. The destination files have autogenerated names.
 - **Merge files:** Merges all files from the source folder to one file. If the file name is specified, the merged file name is the specified name. Otherwise, it's an auto-generated file name.
 - **Preserve hierarchy:** Preserves the file hierarchy in the target folder. The relative path of source file to source folder is identical to the relative path of target file to target folder.
- **Max concurrent connections:** The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.
 - **Expiry datetime (UTC):** Specifies the expiry time of the written files. The time is applied to the UTC time in the format of "2020-03-01T08:00:00Z". By default it's NULL, which means the written files are never expired.

Mapping

For **Mapping** tab configuration, go to Configure your mappings under mapping tab. If you choose Binary as your file format, mapping will not be supported.

Settings

For the **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

The following tables contain more information about the copy activity in Azure Data Lake Storage Gen1.

Source information

[] Expand table

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	<your Azure Data Lake Storage Gen1 connection>	Yes	connection
Connection type	Your connection type. Select Azure Data Lake Storage Gen1.	Azure Data Lake Storage Gen1	Yes	/
File path type	The file path type that you want to use.	<ul style="list-style-type: none"> • File path • Name range • Wildcard folder path, Wildcard file name • List of files 	No	<ul style="list-style-type: none"> • folderPath, fileName • listAfter, listBefore • wildcardFolderPath, wildcardFileName • fileListPath
Recursively	Indicates whether the data is read recursively from the subfolders or only from the specified folder. Note that when Recursively is selected and the destination is a file-based store, an empty folder or subfolder isn't copied or created at the destination. This property doesn't apply when you configure Path to file list .	selected (default) or unselect	No	recursive
Filter by last modified	The files with last modified time in the range [Start time, End time) will be filtered for further processing. The time will be applied to UTC time zone in the format of yyyy-mm-	datetime	No	modifiedDatetimeStart modifiedDatetimeEnd

Name	Description	Value	Required	JSON script property
	<p>ddThh:mm:ss.ffffZ.</p> <p>These properties can be skipped which means no file attribute filter will be applied.</p> <p>This property doesn't apply when you configure your file path type as List of files.</p>			
Enable partition discovery	Indicates whether to parse the partitions from the file path and add them as additional source columns.	selected or unselected (default)	No	enablePartitionDiscovery: true or false (default)
Partition root path	When partition discovery is enabled, specify the absolute root path in order to read partitioned folders as data columns.	< your partition root path >	No	partitionRootPath
Max concurrent connections	The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.	<max concurrent connections>	No	maxConcurrentConnections
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.	<ul style="list-style-type: none"> • Name • Value 	No	additionalColumns: <ul style="list-style-type: none"> • name • value

Destination information

[Expand table](#)

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the destination data store.	<your Azure Data Lake Storage Gen1 connection>	Yes	connection
Connection type	Your connection type. Select Azure Data Lake Storage Gen1.	Azure Data Lake Storage Gen1	Yes	/
File path	The file path of your destination data.	< your file path >	Yes	folderPath, fileName
Copy behavior	Defines the copy behavior when the destination is files from a file-based data store.	<ul style="list-style-type: none">• Flatten hierarchy• Merge files• Preserve hierarchy	No	copyBehavior: <ul style="list-style-type: none">• FlattenHierarchy• MergeFiles• PreserveHierarchy
Max concurrent connections	The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.	<max concurrent connections>	No	maxConcurrentConnections
Expiry datetime (UTC)	The expiry time of the written files. The time is applied to the UTC time in the format of "2020-03-01T08:00:00Z". By default it's NULL, which means the written files are never expired.	< your expiry datetime >	No	expiryDatetime

Next steps

- Azure Data Lake Storage Gen1 overview
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Azure Data Lake Storage Gen2 connector overview

Article • 12/22/2023

The Azure Data Lake Storage Gen2 connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to Azure Data Lake Storage Gen2 in Dataflow Gen2, go to [Set up your connection in Dataflow Gen2](#).

Support in Data pipeline

The Azure Data Lake Storage Gen2 connector supports the following capabilities in Data pipeline:

[+] Expand table

Supported capabilities	Gateway	Authentication
Copy activity (Source/Destination)	None	Account Key Organizational account Shared Access Signature (SAS) Service principal
Lookup activity	None	Account Key Organizational account Shared Access Signature (SAS) Service principal
GetMetadata activity	None	Account Key Organizational account Shared Access Signature (SAS) Service principal
Delete data activity	None	Account Key Organizational account Shared Access Signature (SAS) Service principal

To learn about how to connect to Azure Data Lake Storage Gen2 in Data pipeline, go to [Set up your Azure Data Lake Storage Gen2 connection](#).

To learn about the copy activity configuration for Azure Data Lake Storage Gen2 in Data pipeline, go to Configure Azure Data Lake Storage Gen2 in a copy activity.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your Azure Data Lake Storage Gen2 connection

Article • 11/15/2023

This article outlines the steps to create an Azure Date Lake Storage Gen2 connection.

Supported authentication types

The Azure Date Lake Storage Gen2 connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Account key	✓	✓
Organizational account	✓	✓
Service Principal	✓	
Shared Access Signature (SAS)	✓	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Azure Data Lake Storage Gen2. The following links provide the specific Power Query connector information you need to connect to Azure Data Lake Storage Gen2 in Dataflow Gen2:

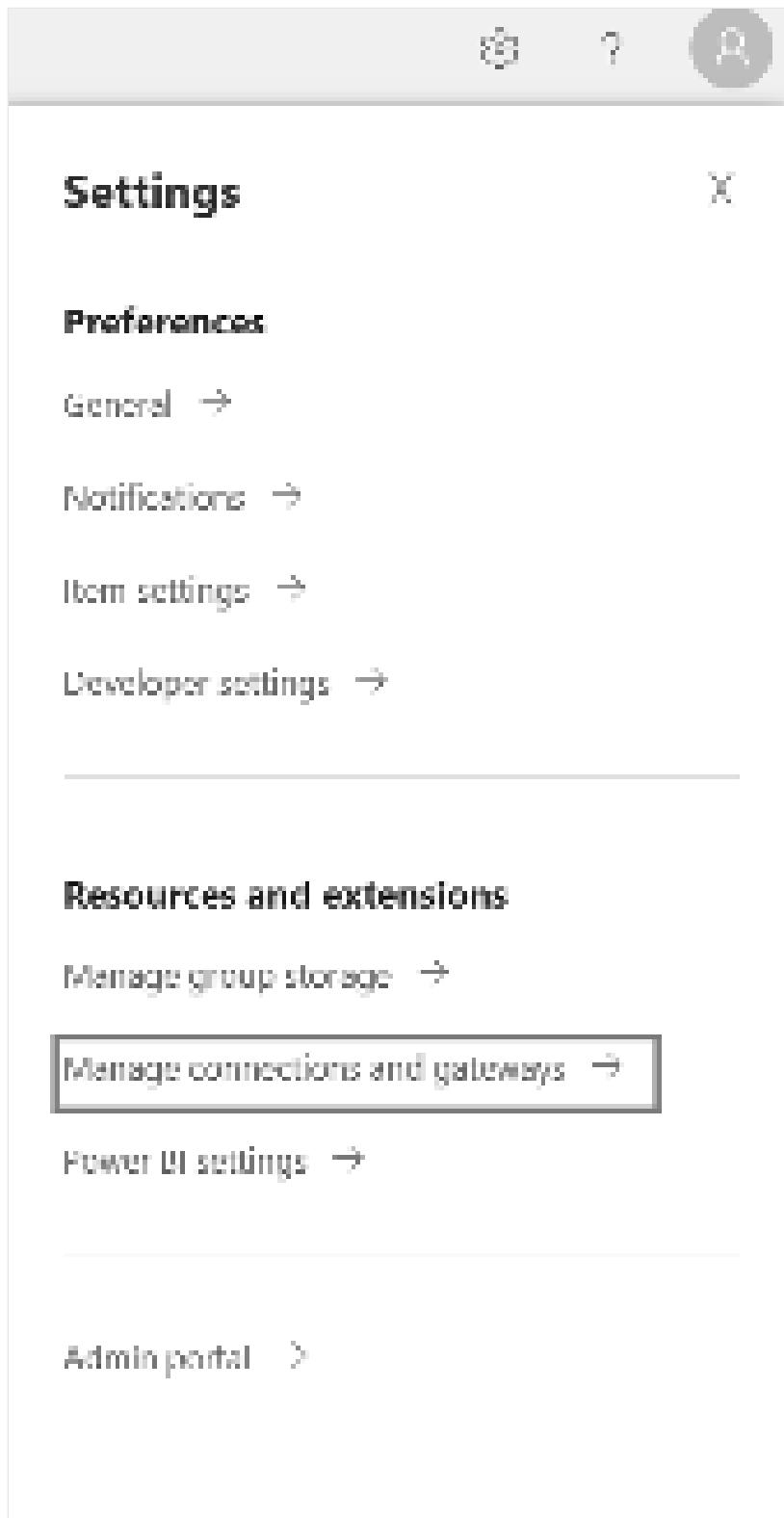
- To get started using the Azure Data Lake Storage Gen2 connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric \(Preview\)](#).
- Be sure to install or set up any Azure Data Lake Storage Gen2 prerequisites before connecting to the Azure Data Lake Storage Gen2 connector.
- To connect to the Azure Data Lake Storage Gen2 connector from Power Query, go to [Connect to Azure Data Lake Storage Gen2 from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

To create a connection in a data pipeline:

1. From the page header in Data Integration service, select **Settings** > **Manage connections and gateways**



2. Select **New** at the top of the ribbon to add a new data source.



The New connection pane shows up on the left side of the page.



Set up your connection

Step 1: Specify the new connection name, type, server and full path

New connection



(1) Currently, these cloud connections are only supported for Data Pipelines and Kusto. In the future, other artifacts can also make use of the cloud connections. To create personal cloud connections in Datasets, Datamarts, and Dataflows, use the Power Query Online experience in "Get data".

The screenshot shows the 'New connection' pane with the following fields filled in:

- Connection name ***: azuredatalakestoragegen2
- Connection type ***: Azure Data Lake Storage Gen2
- Server ***: < Your Server name >
- Full path ***: < Your Container name >

In the **New connection** pane, choose **Cloud**, and specify the following fields:

- **Connection name**: Specify a name for your connection.
- **Connection type**: Select a type for your connection.
- **Server**: Enter your Azure Data Lake Storage Gen2 server name. For example, <https://contosoal1scdm.dfs.core.windows.net>. Specify your Azure Data Lake Storage Gen2 server name. Go to your Azure Data Lake Storage Gen2 account

interface, browse to the **Endpoints** section, and get your Azure Data Lake Storage Gen2.

- **Full path:** Enter the full path to your Azure Data Lake Storage Gen2 container name.

Step 2: Select and set your authentication

Under **Authentication method**, select your authentication from the drop-down list and complete the related configuration. The Azure Data Lake Storage Gen2 connector supports the following authentication types:

- Key
- OAuth2
- Shared Access Signature
- Service Principal



Key authentication

Account key: Specify your Azure Data Lake Storage Gen2 account key. Go to your Azure Data Lake Storage Gen2 account interface, browse to the **Access key** section, and get your account key.

Authentication

Authentication method *

Key

Account key *

OAuth2 authentication

Authentication

Authentication method *

OAuth2

Edit credentials

Open **Edit credentials**. The sign-in interface opens. Enter your account and password to sign in to your account. After signing in, you'll come back to the **New connection** page.

Shared access signature authentication

Authentication

Authentication method *

Shared Access Signature (SAS)

SAS token *

SAS token: Specify the shared access signature token for your Azure Data Lake Storage Gen2 container.

If you don't have a SAS token, switch to **Shared access signature** in your Azure Data Lake Storage Gen2 account interface. Under **Allowed resource types**, select **Container**, and then select **Generate SAS and connection string**. You can get your SAS token from the generated content that appears. The shared access signature is a URI that encompasses in its query parameters all the information necessary for authenticated access to a storage resource. To access storage resources with the shared access signature, the client only needs to pass in the shared access signature to the appropriate constructor or method. For more information about shared access signatures, go to Shared access signatures: Understand the shared access signature model.

Service principal authentication

The screenshot shows the 'Authentication' section of the Azure portal. It includes fields for 'Authentication method' (set to 'Service Principal'), 'Tenant Id' (containing '<your tenant ID>'), 'Service principal ID' (containing '<your application's client ID>'), and 'Service principal key' (containing a redacted password). The entire form is enclosed in a light gray border.

- **Tenant Id:** Specify the tenant information (domain name or tenant ID) under which your application resides. Retrieve it by hovering over the upper-right corner of the Azure portal.
- **Service principal ID:** Specify the application (client) ID.
- **Service principal key:** Specify your application's key.

To use service principal authentication, follow these steps:

1. Register an application entity in Azure Active Directory (Azure AD) by following

Register your application with an Azure AD tenant. Make note of these values, which you use to define the connection:

- Tenant ID
- Application ID
- Application key

2. Grant the service principal proper permission. For examples of how permission works in Azure Data Lake Storage Gen2, go to Access control lists on files and directories.

- **As source**, in Storage Explorer, grant at least **Execute** permission for all upstream folders and the file system, along with **Read** permission for the files to copy. Alternatively, in Access control (IAM), grant at least the **Storage Blob Data Reader** role.
- **As destination**, in Storage Explorer, grant at least **Execute** permission for all upstream folders and the file system, along with **Write** permission for the destination folder. Alternatively, in Access control (IAM), grant at least the **Storage Blob Data Contributor** role.

(!) Note

If you use a UI to author and the service principal isn't set with the "Storage Blob Data Reader/Contributor" role in IAM, when doing a test connection or browsing/navigating folders, choose **Test connection to file path** or **Browse from specified path**, and then specify a path with **Read + Execute** permission to continue.

Step 3: Specify the privacy level you want to apply

In the **General** tab, select the privacy level that you want apply in the **Privacy level** drop-down list. Three privacy levels are supported. For more information, go to General.

General

Privacy level *

Organizational

None

Private

Organizational

Public

Organizational

Step 4: Create your connection

Select **Create**. Your creation is successfully tested and saved if all the credentials are correct. If not correct, the creation fails with errors.



Table summary

The connector properties in the following table are supported in pipeline copy.

Name	Description	Required	Property	Copy
Connection name	A name for your connection.	Yes		✓
Connection type	Select a type for your connection.	Yes		✓
Server	Enter the name of Azure Data Lake Storage Gen2 server, for example, https://contosoадlscdm.dfs.core.windows.net .	Yes		✓
Full path	Enter the full path of your Azure Data Lake Storage Gen2 container name.	Yes		✓
Authentication	Go to Authentication.	Yes	Go to Authentication.	
Privacy Level	The privacy level that you want to apply. Allowed values are Organizational, Privacy, and Public.	Yes		✓

Authentication

The properties in the following table are the supported authentication types.

Name	Description	Required	Property	Copy
Key				✓
- Account key	The Azure Data Lake Storage Gen2 account key.	Yes		
Shared Access Signature (SAS)				✓
- SAS token	Specify the shared access signature token for your Azure Data Lake Storage Gen2 container.	Yes		
Service Principal				✓
- Tenant ID	The tenant information (domain name or tenant ID).	Yes		
- Service Principal ID	The application's client ID.	Yes		
- Service Principal key	The application's key.	Yes		

Next steps

- Configure Azure Data Lake Storage Gen2 in a copy activity
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Configure Azure Data Lake Storage Gen2 in a copy activity

Article • 12/22/2023

This article outlines how to use the copy activity in data pipeline to copy data from and to Azure Data Lake Storage Gen2.

Supported format

Azure Data Lake Storage Gen2 supports the following file formats. Refer to each article for format-based settings.

- Avro format
- Binary format
- Delimited text format
- Excel format
- JSON format
- ORC format
- Parquet format
- XML format

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

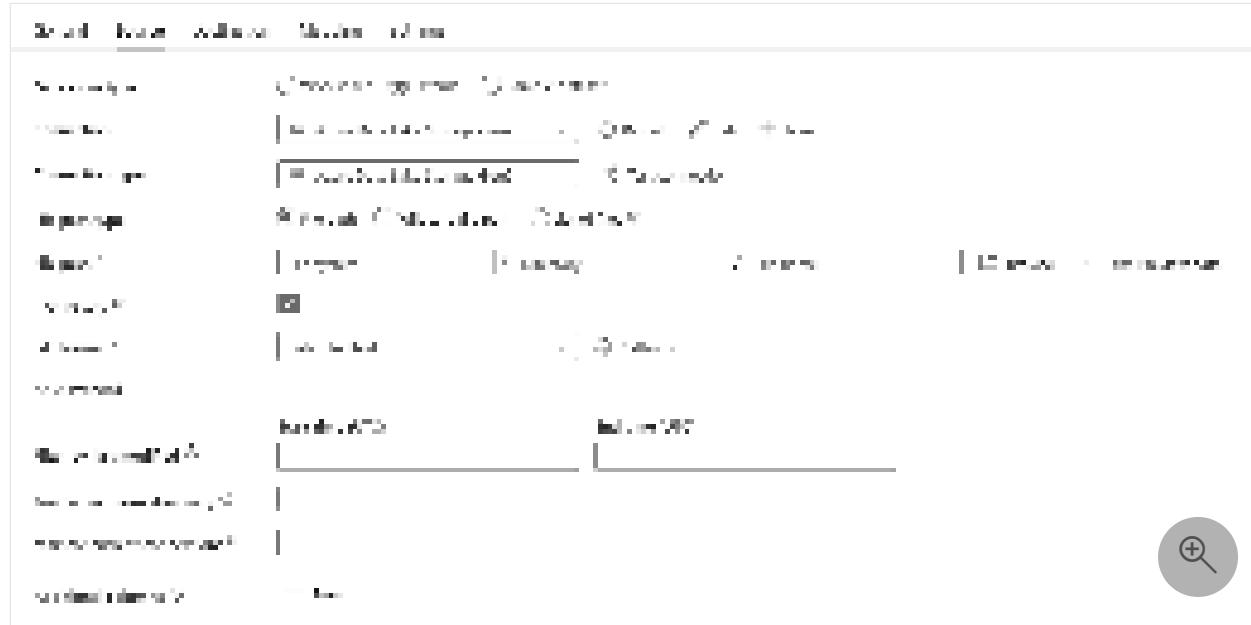
- General
- Source
- Destination
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

The following properties are supported for Azure Data Lake Storage Gen2 under the **Source** tab of a copy activity.



The following properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select an Azure Data Lake Storage Gen2 connection from the connection list. If no connection exists, then create a new Azure Data Lake Storage Gen2 connection by selecting **New**.
- **Connection type:** Select **Azure Data Lake Storage Gen2**.
- **File path type:** You can choose **File path**, **Wildcard file path**, or **List of files** as your file path type. The configuration of each of these settings is:
 - **File path:** If you choose this type, the data can be copied from the specified file system or folder/file path specified previously.
 - **Wildcard file path:** If you choose this type, specify the File system and Wildcard paths.
 - **File system:** The Azure Data Lake Storage Gen2 file system name.
 - **Wildcard paths:** Specify the folder or file path with wildcard characters under the specified file system to filter source folders or files.

Allowed wildcards are: `*` (matches zero or more characters) and `?` (matches zero or single character). Use `\` to escape if your folder name has wildcard or this escape character inside. For more examples, go to [Folder and file filter examples](#).



- **Wildcard folder path:** Specify the folder path with wildcard characters under your specified file system to filter source folders.
- **Wildcard file name:** Specify the file name with wildcard characters under your specified file system + folder path (or wildcard folder path) to filter source files.
- **List of files:** Indicates you want to copy a given file set. Specify **Folder path** and **Path to file list** to point to a text file that includes a list of files you want to copy, one file per line, which is the relative path to the path. For more examples, go to [File list examples](#).



- **Folder path:** Specify the path to a folder under the specified file system. It is required.
- **Path to file list:** Specify the path of the text file that includes a list of files you want to copy.
- **Recursively:** Specify whether the data is read recursively from the subfolders or only from the specified folder. Note that when **Recursively** is selected and the destination is a file-based store, an empty folder or subfolder isn't copied or created at the destination. This property is selected by default and doesn't apply when you configure **Path to file list**.
- **File format:** Select the file format applied from the drop-down list. Select **Settings** to configure the file format. For settings of different file formats, refer to articles in Supported format for detailed information.

Under **Advanced**, you can specify the following fields:

- **Filter by last modified:** Files are filtered based on the last modified dates. This property doesn't apply when you configure your file path type as List of files.
 - **Start time (UTC):** The files are selected if their last modified time is greater than or equal to the configured time.

- **End time (UTC):** The files are selected if their last modified time is less than the configured time.

When **Start time (UTC)** has datetime value but **End time (UTC)** is NULL, it means the files whose last modified attribute is greater than or equal with the datetime value will be selected. When **End time (UTC)** has datetime value but **Start time (UTC)** is NULL, it means the files whose last modified attribute is less than the datetime value will be selected. The properties can be NULL, which means no file attribute filter will be applied to the data.

- **Enable partition discovery:** Specify whether to parse the partitions from the file path and add them as additional source columns. It is unselected by default and not supported when you use binary file format.

- **Partition root path:** When partition discovery is enabled, specify the absolute root path in order to read partitioned folders as data columns.

If it is not specified, by default,

- When you use file path or list of files on source, partition root path is the path that you configured.
- When you use wildcard folder filter, partition root path is the sub-path before the first wildcard.

For example, assuming you configure the path as

`root/folder/year=2020/month=08/day=27`:

- If you specify partition root path as `root/folder/year=2020`, copy activity will generate two more columns month and day with value "08" and "27" respectively, in addition to the columns inside the files.
- If partition root path is not specified, no extra column will be generated.



- **Max concurrent connections:** This property indicates the upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.
- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.

Destination

The following properties are supported for Azure Data Lake Storage Gen2 under the Destination tab of a copy activity.

The screenshot shows the 'Destination' tab of a copy activity configuration. It includes fields for 'File system', 'Container', 'Path', 'Format', 'File format', 'Max concurrent connections', 'Block size (MB)', and 'Metadata'. A search icon is located in the bottom right corner.

The following properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select an Azure Data Lake Storage Gen2 connection from the connection list. If no connection exists, then create a new Azure Data Lake Storage Gen2 connection by selecting **New**.
- **Connection type:** Select **Azure Data Lake Storage Gen2**.
- **File path:** Select **Browse** to choose the file that you want to copy or fill in the path manually.
- **File format:** Select the file format applied from the drop-down list. Select **Settings** to configure the file format. For settings of different file formats, refer to articles in Supported format for detailed information.

Under **Advanced**, you can specify the following fields:

- **Copy behavior:** Defines the copy behavior when the source is files from a file-based data store. You can choose a behavior from the drop-down list.

The screenshot shows the 'Copy behavior' dropdown menu. The options listed are: Select, Flatten hierarchy, Merge files, Preserve hierarchy, and Add dynamic content.

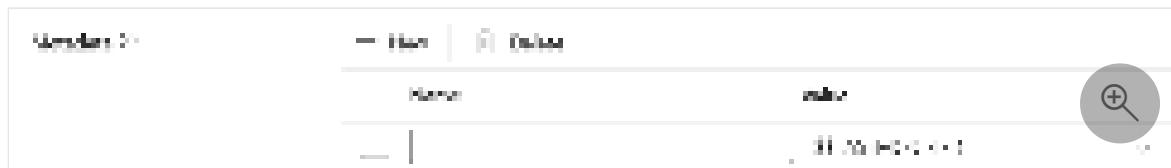
- **Flatten hierarchy:** All files from the source folder are in the first level of the destination folder. The destination files have autogenerated names.
 - **Merge files:** Merges all files from the source folder to one file. If the file name is specified, the merged file name is the specified name. Otherwise, it's an auto-generated file name.
 - **Preserve hierarchy:** Preserves the file hierarchy in the target folder. The relative path of source file to source folder is identical to the relative path of target file to target folder.
- **Max concurrent connections:** The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.
 - **Block size (MB):** Specify the block size in MB used to write data to Azure Data Lake Storage Gen2. For more information, see Block Blobs. Allowed value is between 4 MB and 100 MB.

By default, the block size is automatically determined based on your source store type and data. For non-binary copy into Azure Data Lake Storage Gen2, the default block size is 100 MB so as to fit in at most approximately 4.75-TB data. It may be not optimal when your data is not large. You can explicitly specify a block size, while ensure **Block size (MB)*50000** is big enough to store the data, otherwise copy activity run will fail.

- **Metadata:** Set custom metadata when copying to a destination. Each object under the `metadata` array represents an extra column. The `name` defines the metadata key name, and the `value` indicates the data value of that key. If the `preserve attributes` feature is used, the specified metadata will union/overwrite with the source file metadata.

Allowed data values are:

- `$$LASTMODIFIED`: A reserved variable indicates to store the source files' last modified time. Apply to a file-based source with a binary format only.
- **Expression**
- **Static value**



Mapping

For **Mapping** tab configuration, go to Configure your mappings under mapping tab. If you choose Binary as your file format, mapping isn't supported.

Settings

For the **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

The following tables contain more information about the copy activity in Azure Data Lake Storage Gen2.

Source information

[+] Expand table

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	<your Azure Data Lake Storage Gen2 connection>	Yes	connection
Connection type	Your connection type. Select Azure Data Lake Storage Gen2.	Azure Data Lake Storage Gen2	Yes	/
File path type	The file path type that you want to use.	<ul style="list-style-type: none">• File path• Wildcard folder path, Wildcard file name• List of files	Yes	<ul style="list-style-type: none">• filePath, fileName, fileSystem• wildcardFolderPath, wildcardFileName, fileSystem• filePath, fileName, fileListPath
Recursively	Indicates whether the data is read recursively from the subfolders or only from the specified folder. Note that when Recursively is selected and the destination is	selected (default) or unselect	No	recursive

Name	Description	Value	Required	JSON script property
	a file-based store, an empty folder or subfolder isn't copied or created at the destination. This property doesn't apply when you configure Path to file list .			
Filter by last modified	The files with last modified time in the range [Start time, End time) will be filtered for further processing. The time will be applied to UTC time zone in the format of <code>yyyy-mm-ddThh:mm:ss.fffZ</code> . These properties can be skipped which means no file attribute filter will be applied. This property doesn't apply when you configure your file path type as List of files.	datetime	No	modifiedDatetimeStart modifiedDatetimeEnd
Enable partition discovery	Indicates whether to parse the partitions from the file path and add them as additional source columns.	selected or unselected (default)	No	enablePartitionDiscovery: true or false (default)
Partition root path	When partition discovery is enabled, specify the absolute root path in order to read partitioned folders as data columns.	< your partition root path >	No	partitionRootPath
Max concurrent connections	The upper limit of concurrent connections established to the data store during the	<max concurrent connections>	No	maxConcurrentConnections

Name	Description	Value	Required	JSON script property
	activity run. Specify a value only when you want to limit concurrent connections.			
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.	<ul style="list-style-type: none"> • Name • Value 	No	additionalColumns: <ul style="list-style-type: none"> • name • value

Destination information

Expand table

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the destination data store.	<your Azure Data Lake Storage Gen2 connection>	Yes	connection
Connection type	Your connection type. Select Azure Data Lake Storage Gen2 .	Azure Data Lake Storage Gen2	Yes	/
File path	The file path of your destination data.	< your file path >	Yes	folderPath, fileName, fileSystem
Copy behavior	Defines the copy behavior when the destination is files from a file-based data store.	<ul style="list-style-type: none"> • Flatten hierarchy • Merge files • Preserve hierarchy 	No	copyBehavior: <ul style="list-style-type: none"> • FlattenHierarchy • MergeFiles • PreserveHierarchy
Max concurrent connections	The upper limit of concurrent connections established to the	<max concurrent connections>	No	maxConcurrentConnections

Name	Description	Value	Required	JSON script property
	<p>data store during the activity run.</p> <p>Specify a value only when you want to limit concurrent connections.</p>			
Block size (MB)	<p>Specify the block size in MB when writing data to Azure Data Lake Storage Gen2.</p> <p>Allowed value is between 4 MB and 100 MB.</p>	<block size>	No	blockSizeInMB
Metadata	<p>Set custom metadata when copy to destination.</p>	<ul style="list-style-type: none"> • <code>\$\$LASTMODIFIED</code> • Expression • Static value 	No	metadata

Next steps

- Set up your Azure Data Lake Storage Gen2 connection
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Azure Database for PostgreSQL connector overview

Article • 11/15/2023

The Azure Database for PostgreSQL connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

Data Factory in Microsoft Fabric doesn't currently support Azure Database for PostgreSQL in Dataflow Gen2.

Support in data pipelines

The Azure Database for PostgreSQL connector supports the following capabilities in data pipelines:

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	Basic
Lookup activity	None	Basic

To learn more about the copy activity configuration for Azure Database for PostgreSQL in data pipelines, go to [Configure in a data pipeline copy activity](#).

Feedback

Was this page helpful?



[Provide product feedback](#) | [Ask the community](#)

Configure Azure Database for PostgreSQL in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in Data pipeline to copy data from and to Azure Database for PostgreSQL.

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Destination
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

Go to **Source** tab to configure your copy activity source. See the following content for the detailed configuration.



The following three properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select an Azure Database for PostgreSQL connection from the connection list. If no connection exists, then create a new Azure Database for PostgreSQL connection by selecting **New**.
- **Connection type:** Select **Azure Database for PostgreSQL**.
- **Use query:** Select **Table** to read data from the specified table or select **Query** to read data using queries.

- If you select **Table**:

- **Table:** Select the table from the drop-down list or select **Edit** to manually enter it to read data.



- If you select **Query**:

- **Query:** Specify the custom SQL query to read data. For example: `SELECT * FROM mytable` or `SELECT * FROM "MyTable"`.

Note

In PostgreSQL, the entity name is treated as case-insensitive if not quoted.



Under **Advanced**, you can specify the following fields:

- **Partition option:** Specifies the data partitioning options used to load data from Azure Database for PostgreSQL. When a partition option is enabled (that is, not **None**), the degree of parallelism to concurrently load data from an Azure Database for PostgreSQL is controlled by the **Degree of copy parallelism** in the copy activity settings tab.

If you select **None**, you choose not to use partition.

If you select **Physical partitions of table**:

- **Partition names:** Specify the list of physical partitions that needs to be copied.

If you use a query to retrieve the source data, hook `?AdfTabularPartitionName` in the WHERE clause. For an example, see the Parallel copy from Azure Database for PostgreSQL section.



If you select **Dynamic range**:

- **Partition column name:** Specify the name of the source column **in integer or date/datetime type** (`int`, `smallint`, `bigint`, `date`, `timestamp without time zone`, `timestamp with time zone` or `time without time zone`) that will be used by range partitioning for parallel copy. If not specified, the primary key of the table is auto-detected and used as the partition column.

If you use a query to retrieve the source data, hook `?AdfRangePartitionColumnName` in the WHERE clause. For an example, see the Parallel copy from Azure Database for PostgreSQL section.

- **Partition upper bound:** Specify the maximum value of the partition column to copy data out.

If you use a query to retrieve the source data, hook `?AdfRangePartitionUpbound` in the WHERE clause. For an example, see the Parallel copy from Azure Database for PostgreSQL section..

- **Partition lower bound:** Specify the minimum value of the partition column to copy data out.

If you use a query to retrieve the source data, hook `?AdfRangePartitionLowbound` in the WHERE clause. For an example, see the Parallel copy from Azure Database for PostgreSQL section.



- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.

Destination

Go to **Destination** tab to configure your copy activity destination. See the following content for the detailed configuration.



The following three properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select an Azure Database for PostgreSQL connection from the connection list. If no connection exists, then create a new Azure Database for PostgreSQL connection by selecting **New**.
- **Connection type:** Select **Azure Database for PostgreSQL**.
- **Table:** Select the table from the drop-down list or select **Edit** to manually enter it to write data.

Under **Advanced**, you can specify the following fields:

- **Write method:** Select the method used to write data into Azure Database for PostgreSQL. Select from **Copy command** (default, which is more performant) and **Bulk insert**.
- **Pre-copy script:** Specify a SQL query for the copy activity to execute before you write data into Azure Database for PostgreSQL in each run. You can use this property to clean up the preloaded data.
- **Write batch timeout:** Specify the wait time for the batch insert operation to finish before it times out. The allowed value is timespan. The default value is 00:30:00 (30

minutes).

- **Write batch size:** Specify the number of rows loaded into Azure Database for PostgreSQL per batch. Allowed value is an integer that represents the number of rows. The default value is 1,000,000.

Mapping

For **Mapping** tab configuration, see [Configure your mappings under mapping tab](#).

Settings

For **Settings** tab configuration, go to [Configure your other settings under settings tab](#).

Parallel copy from Azure Database for PostgreSQL

The Azure Database for PostgreSQL connector in copy activity provides built-in data partitioning to copy data in parallel. You can find data partitioning options on the **Source** tab of the copy activity.

When you enable partitioned copy, copy activity runs parallel queries against your Azure Database for PostgreSQL source to load data by partitions. The parallel degree is controlled by the **Degree of copy parallelism** in the copy activity settings tab. For example, if you set **Degree of copy parallelism** to four, the service concurrently generates and runs four queries based on your specified partition option and settings, and each query retrieves a portion of data from your Azure Database for PostgreSQL.

You are suggested to enable parallel copy with data partitioning especially when you load large amount of data from your Azure Database for PostgreSQL. The following are suggested configurations for different scenarios. When copying data into file-based data store, it's recommended to write to a folder as multiple files (only specify folder name), in which case the performance is better than writing to a single file.

Scenario	Suggested settings
Full load from large table, with physical partitions.	Partition option: Physical partitions of table. During execution, the service automatically detects the physical partitions, and copies data by partitions.
Full load from large table, without physical partitions,	Partition options: Dynamic range. Partition column: Specify the column used to partition data. If

Scenario	Suggested settings
while with an integer column for data partitioning.	not specified, the primary key column is used.
Load a large amount of data by using a custom query, with physical partitions.	<p>Partition option: Physical partitions of table.</p> <p>Query: <code>SELECT * FROM ?AdfTabularPartitionName WHERE <your_additional_where_clause>.</code></p> <p>Partition name: Specify the partition name(s) to copy data from. If not specified, the service automatically detects the physical partitions on the table you specified in the PostgreSQL dataset.</p> <p>During execution, the service replaces <code>?AdfTabularPartitionName</code> with the actual partition name, and sends to Azure Database for PostgreSQL.</p>
Load a large amount of data by using a custom query, without physical partitions, while with an integer column for data partitioning.	<p>Partition options: Dynamic range.</p> <p>Query: <code>SELECT * FROM ?AdfTabularPartitionName WHERE ?AdfRangePartitionColumnName <= ?AdfRangePartitionUpbound AND ?AdfRangePartitionColumnName >= ?AdfRangePartitionLowbound AND <your_additional_where_clause>.</code></p> <p>Partition column: Specify the column used to partition data. You can partition against the column with integer or date/datetime data type.</p> <p>Partition upper bound and Partition lower bound: Specify if you want to filter against partition column to retrieve data only between the lower and upper range.</p> <p>During execution, the service replaces <code>?AdfRangePartitionColumnName</code>, <code>?AdfRangePartitionUpbound</code>, and <code>?AdfRangePartitionLowbound</code> with the actual column name and value ranges for each partition, and sends to Azure Database for PostgreSQL.</p> <p>For example, if your partition column "ID" is set with the lower bound as 1 and the upper bound as 80, with parallel copy set as 4, the service retrieves data by 4 partitions. Their IDs are between [1,20], [21, 40], [41, 60], and [61, 80], respectively.</p>

Best practices to load data with partition option:

- Choose distinctive column as partition column (like primary key or unique key) to avoid data skew.
- If the table has built-in partition, use partition option "Physical partitions of table" to get better performance.

Table summary

The following table contains more information about the copy activity in Azure Database for PostgreSQL.

Source information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	< your Azure Database for PostgreSQL connection >	Yes	connection
Connection type	Your source connection type.	Azure Database for PostgreSQL	Yes	/
Use query	The way to read data. Apply Table to read data from the specified table or apply Query to read data using queries.	<ul style="list-style-type: none"> • Table • Query 	Yes	<ul style="list-style-type: none"> • typeProperties (under <code>typeProperties -> source</code>) - schema - table • query
Partition names	The list of physical partitions that needs to be copied. If you use a query to retrieve the source data, hook <code>?AdfTabularPartitionName</code> in the WHERE clause.	< your partition names >	No	partitionNames
Partition column name	The name of the source column in integer or date/datetime type (<code>int, smallint, bigint, date, timestamp without time zone, timestamp with time zone or time without time zone</code>) that will be used by range partitioning for parallel copy. If not specified, the primary key of the table is auto-detected	< your partition column names >	No	partitionColumnName

Name	Description	Value	Required	JSON script property
	and used as the partition column.			
Partition upper bound	The maximum value of the partition column to copy data out. If you use a query to retrieve the source data, hook <code>?AdfRangePartitionUpbound</code> in the WHERE clause.	< your partition upper bound >	No	partitionUpperBound
Partition lower bound	The minimum value of the partition column to copy data out. If you use a query to retrieve the source data, hook <code>?AdfRangePartitionLowbound</code> in the WHERE clause.	< your partition lower bound >	No	partitionLowerBound
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.	<ul style="list-style-type: none"> • Name • Value 	No	additionalColumns: <ul style="list-style-type: none"> • name • value

Destination information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the destination data store.	< your Azure Database for PostgreSQL connection >	Yes	connection
Connection type	Your destination connection type.	Azure Database for PostgreSQL	Yes	/
Table	Your destination data table to write data.	< name of your destination table >	Yes	typeProperties (under <code>typeProperties -> sink</code>): - schema - table

Name	Description	Value	Required	JSON script property
Write method	The method used to write data into Azure Database for PostgreSQL.	<ul style="list-style-type: none"> • Copy command (default) • Bulk insert 	No	writeMethod: <ul style="list-style-type: none"> • CopyCommand • BulkInsert
Pre-copy script	A SQL query for the copy activity to execute before you write data into Azure Database for PostgreSQL in each run. You can use this property to clean up the preloaded data.	< your pre-copy script >	No	preCopyScript
Write batch timeout	The wait time for the batch insert operation to finish before it times out.	timespan (the default is 00:30:00 - 30 minutes)	No	writeBatchTimeout
Write batch size	The number of rows loaded into Azure Database for PostgreSQL per batch.	integer (the default is 1,000,000)	No	writeBatchSize

Next steps

- Azure Database for PostgreSQL connector overview

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Azure Databricks connector overview

Article • 11/15/2023

The Azure Databricks connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to Azure Databricks in Dataflow Gen2, go to Set up your Azure Databricks connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support Azure Databricks in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your Azure Databricks connection

Article • 12/21/2023

This article outlines the steps to create an Azure Databricks connection.

Supported authentication types

The Azure Databricks connector supports the following authentication types for copy and Dataflow Gen2 respectively.

[+] Expand table

Authentication type	Copy	Dataflow Gen2
Username/Password	n/a	✓
Personal Access Token	n/a	✓
Microsoft Entra ID	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Azure Databricks. The following links provide the specific Power Query connector information you need to connect to Azure Databricks in Dataflow Gen2:

- To get started using the Azure Databricks connector in Dataflow Gen2, go to Get data from Data Factory in Microsoft Fabric.
- To connect to the Azure Databricks connector from Power Query, go to Connect to Databricks data from Power Query Online.

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support Azure Databricks in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Azure HDInsight on AKS Trino connector overview

Article • 11/15/2023

The Azure HDInsight on AKS Trino connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to Azure HDInsight on AKS Trino in Dataflow Gen2, go to Set up your Azure HDInsight on AKS Trino connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support Azure HDInsight on AKS Trino in data pipelines.

Feedback

Was this page helpful?



[Provide product feedback ↗](#) | [Ask the community ↗](#)

Set up your Azure HDInsight on AKS Trino connection

Article • 12/06/2023

This article outlines the steps to create an Azure HDInsight on AKS Trino connection.

Supported authentication types

The Access database connector supports the following authentication types for copy and Dataflow Gen2 respectively.

[+] Expand table

Authentication type	Copy	Dataflow Gen2
Organizational account	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Azure HDInsight on AKS Trino. The following links provide the specific Power Query connector information you need to connect to Azure HDInsight on AKS Trino in Dataflow Gen2:

- To get started using the Azure HDInsight on AKS Trino connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- Be sure to install or set up any Azure HDInsight on AKS Trino prerequisites before connecting to the Azure HDInsight on AKS Trino connector.
- To connect to the Azure HDInsight on AKS Trino connector from Power Query, go to [Connect to Azure HDInsight on AKS Trino from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support Azure HDInsight on AKS Trino in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Azure SQL Database connector overview

Article • 11/15/2023

This Azure SQL Database connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to Azure SQL Database in Dataflow Gen2, go to Set up your connection in Dataflow Gen2.

Support in data pipelines

The Azure SQL Database connector supports the following capabilities in data pipelines:

Supported capabilities	Gateway	Authentication
Copy activity (Source/Destination)	None	Basic OAuth2 Service principal
Lookup activity	None	Basic OAuth2 Service principal
GetMetadata activity	None	Basic OAuth2 Service principal
Script activity	None	Basic OAuth2 Service principal
Stored procedure activity	None	Basic OAuth2 Service principal

To learn about how to connect to Azure SQL Database in data pipelines, go to Set up your Azure SQL Database connection.

To learn about the copy activity configuration for Azure SQL Database in data pipelines, go to Configure Azure SQL Database in a copy activity.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your Azure SQL Database connection

Article • 12/21/2023

This article outlines how to set up a connection to Azure SQL Database.

Supported authentication types

The Azure SQL Database connector supports the following authentication types for copy and Dataflow Gen2 respectively.

[+] Expand table

Authentication type	Copy	Dataflow Gen2
Basic	✓	✓
Organizational account	✓	✓
Service Principal	✓	

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to an Azure SQL database. The following links provide the specific Power Query connector information you need to connect to an Azure SQL database in Dataflow Gen2:

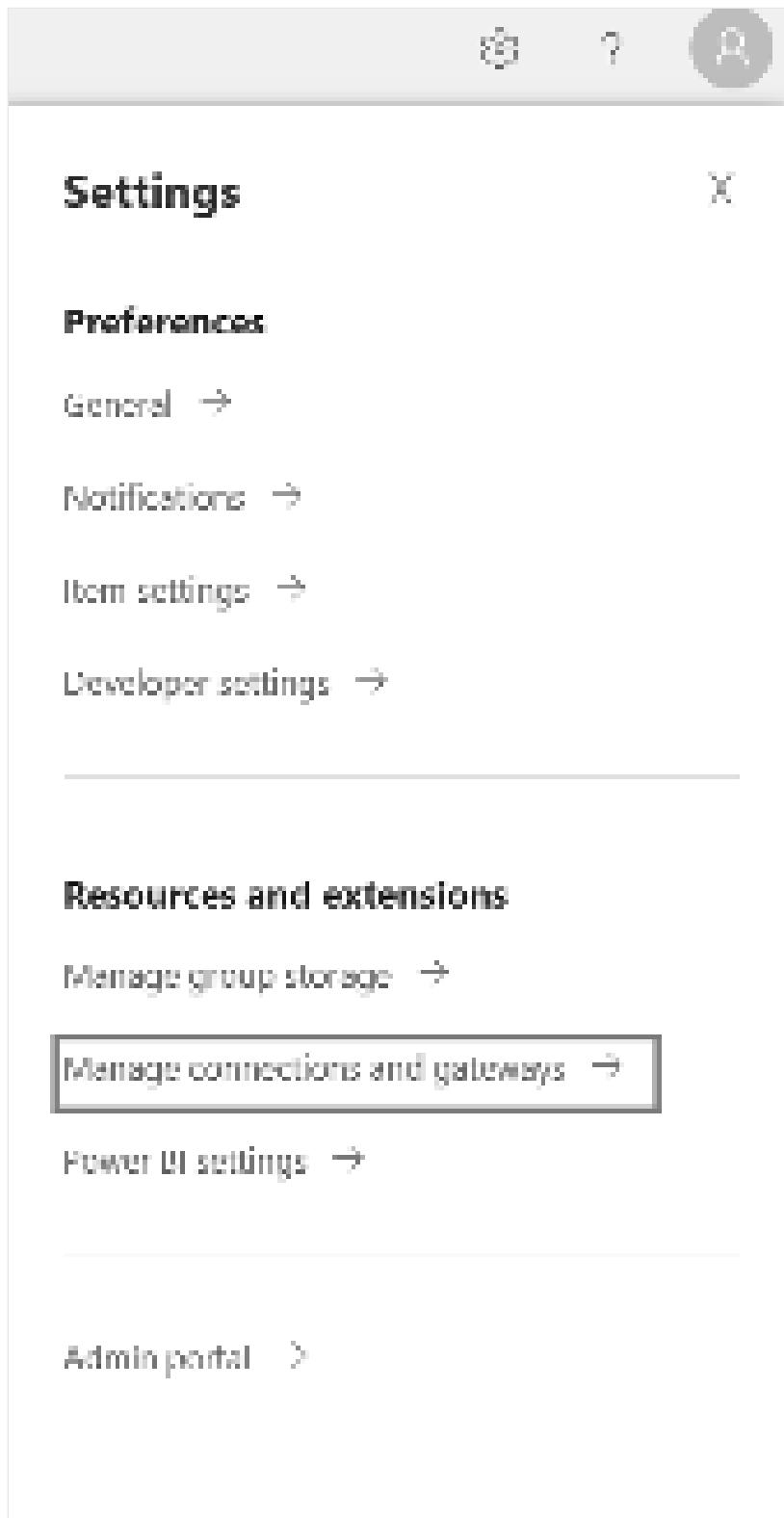
- To get started using the Azure SQL Database connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- Be sure to install or set up any Azure SQL Database prerequisites before connecting to the Azure SQL Database connector.
- To connect to the Azure SQL Database connector from Power Query, go to [Connect to Azure SQL database from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

To create a connection in a data pipeline:

1. From the page header in the Data Factory service, select **Settings** > **Manage connections and gateways**.



2. Select **New** at the top of the ribbon to add a new data source.

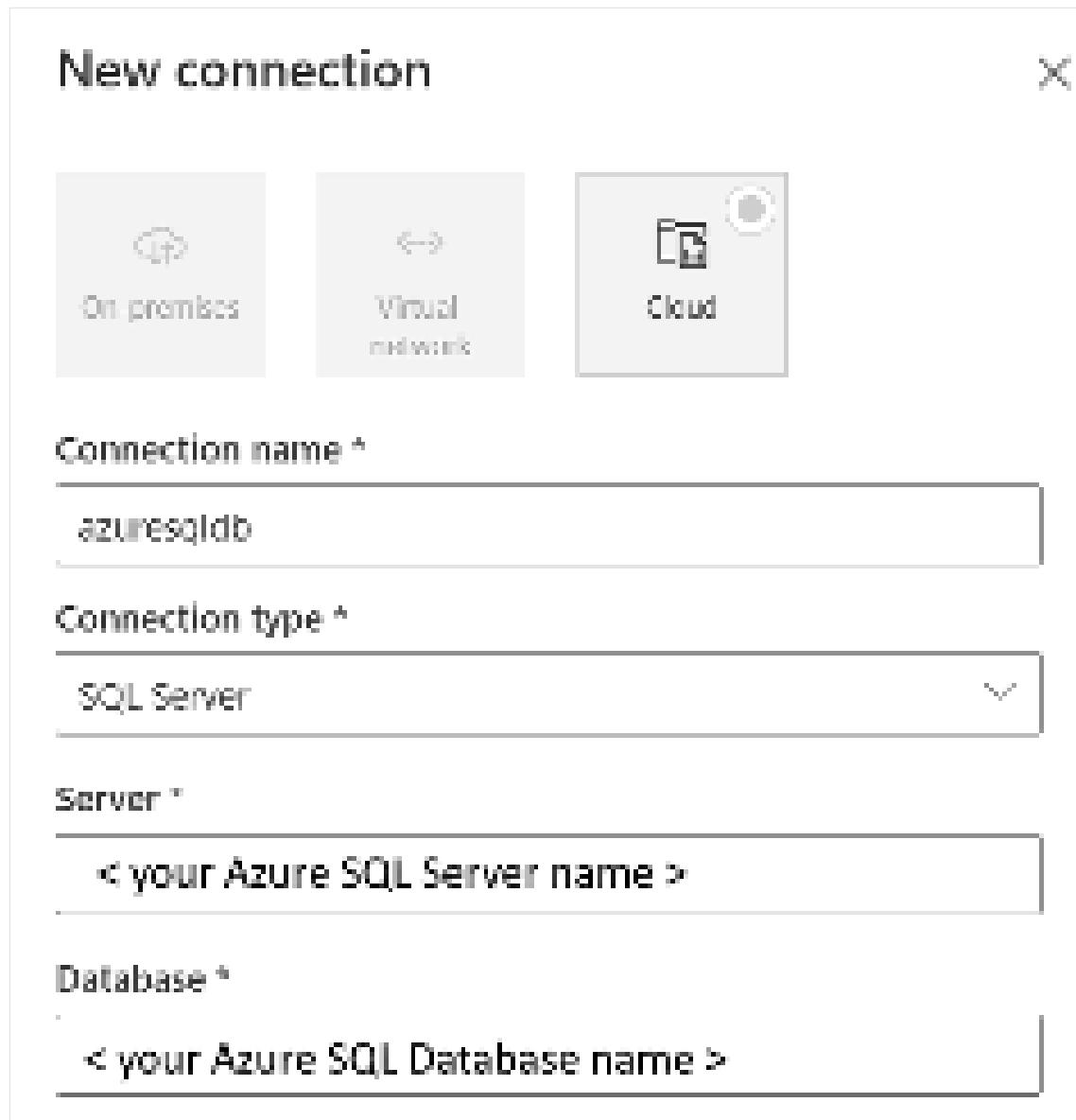


The New connection pane opens on the left side of the page.



Set up your connection

Step 1: Specify the connection name, connection type, account, and domain



In the **New connection** pane, choose **Cloud**, and specify the following fields:

- **Connection name:** Specify a name for your connection.
- **Connection type:** Select **SQL Server**.
- **Server:** Enter your Azure SQL server name. You can find it in the **Overview** page of your Azure SQL server.
- **Database:** Enter your Azure SQL Database name.

Step 2: Select and set your authentication

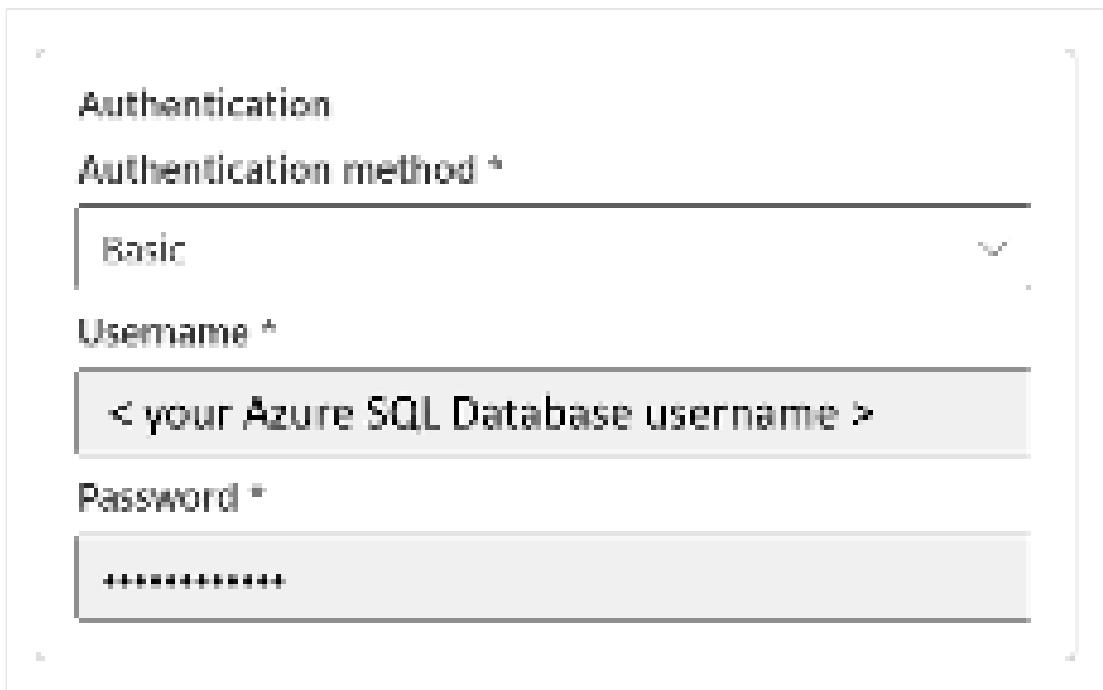
Under **Authentication method**, select your authentication from the drop-down list and complete the related configuration. The Azure SQL Database connector supports the following authentication types.

- Basic
- OAuth2
- Service Principal



Basic authentication

Select **Basic** under **Authentication method**.



- **Username:** Specify the user name of your Azure SQL Database.
- **Password:** Specify the password of your Azure SQL Database.

OAuth2 authentication

Open **Edit credentials**. You'll notice the sign in interface. Enter your account and password to sign in your account. After signing in, go back to the **New connection** page.

Authentication

Authentication method *

▼

[Edit credentials](#)

Service Principal authentication

Authentication

Authentication method *

▼

Tenant Id *

Service principal ID *

Service principal key *

- **Tenant Id:** Specify the tenant information (domain name or tenant ID) under which your application resides. Retrieve it by hovering over the upper-right corner of the Azure portal.
- **Service principal ID:** Specify the application's client ID.
- **Service principal key:** Specify your application's key.

To use service principal authentication, follow these steps:

1. Create a Microsoft Entra application from the Azure portal. Make note of the application name and the following values that define the connection:
 - Tenant ID
 - Application ID
 - Application key

2. Provision a Microsoft Entra administrator for your server on the Azure portal if you haven't already done so. The Microsoft Entra administrator must be a Microsoft Entra user or Microsoft Entra group, but it can't be a service principal. This step is done so that, in the next step, you can use a Microsoft Entra identity to create a contained database user for the service principal.
3. Create contained database users for the service principal. Connect to the database from or to which you want to copy data by using tools like SQL Server Management Studio, with a Microsoft Entra identity that has at least ALTER ANY USER permission. Sign in to your Azure SQL Database through Active Directory authentication and run the following T-SQL:

SQL

```
CREATE USER [your application name] FROM EXTERNAL PROVIDER;
```

4. Grant the service principal needed permissions as you normally do for SQL users or others. Run the following code. For more options, go to ALTER ROLE (Transact-SQL).

SQL

```
ALTER ROLE [role name] ADD MEMBER [your application name];
```

5. Configure an Azure SQL Database connection.

Step 3: Specify the privacy level that you want to apply

In the **General** tab, select the privacy level that you want apply in **Privacy level** drop-down list. Three privacy levels are supported. For more information, go to [privacy levels](#).

Step 4: Create your connection

Select **Create**. Your creation is successfully tested and saved if all the credentials are correct. If not correct, the creation fails with errors.

Data pipeline

Connections Go previous data gateway Go next data gateway

Close this tab | Refresh this page | Help | Feedback | Log in | Logout | About | Contact us | Terms of use

Name	Connection type	Server	Status	Managing connector
My connection	SQL Server	My server	Up to date	

Table summary

The following connector properties in the table are supported in pipeline copy.

Expand table

Name	Description	Required	Property	Copy
Connection name	A name for your connection.	Yes		✓
Connection type	Select a type for your connection. Select SQL Server .	Yes		✓
Server	Azure SQL server name.	Yes		✓
Database	Azure SQL Database name.	Yes		✓
Authentication	Go to Authentication	Yes		Go to Authentication
Privacy Level	The privacy level that you want to apply. Allowed values are Organizational, Privacy, Public	Yes		✓

Authentication

The following properties in the table are the supported authentication types.

Expand table

Name	Description	Required	Property	Copy
Basic				✓
- Username	The user name of your Azure SQL Database.	Yes		
- Password	The password of your Azure SQL Database.	Yes		

Name	Description	Required	Property	Copy
OAuth2				✓
Service Principal				✓
- Tenant ID	The tenant information (domain name or tenant ID).	Yes		
- Service Principal ID	The application's client ID.	Yes		
- Service Principal key	The application's key.	Yes		

Next steps

- Configure Azure SQL Database in a copy activity
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback [↗](#) | Ask the community [↗](#)

Configure Azure SQL Database in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in data pipeline to copy data from and to Azure SQL Database.

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Destination
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

The following properties are supported for Azure SQL Database under the **Source** tab of a copy activity.

The screenshot shows the 'General' tab of a configuration interface. The form fields are as follows:

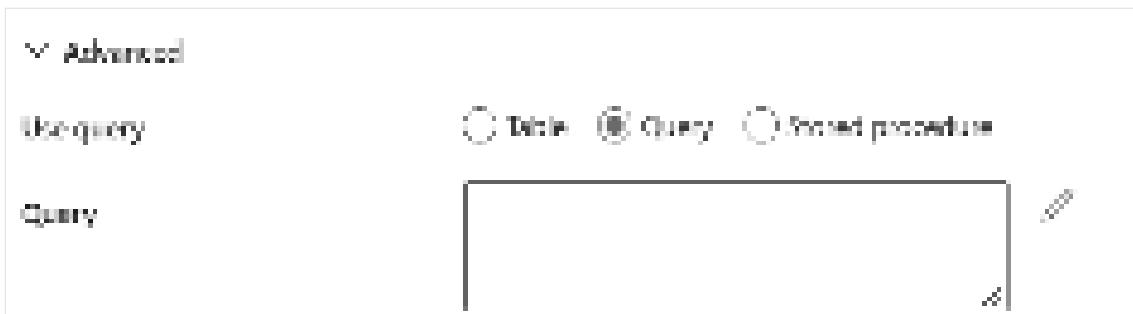
- Data store type:** External
- Connection:** Azure SQL Database
- Table:** MyTable
- Preview data:** Checked
- Use query:** Table (selected)
- Group by (None):** None
- Total or sum:** Total (selected)
- Partitioned:** None
- Note:** Please preview table data before publishing to application.
- Additional columns:** None

The following properties are **required**:

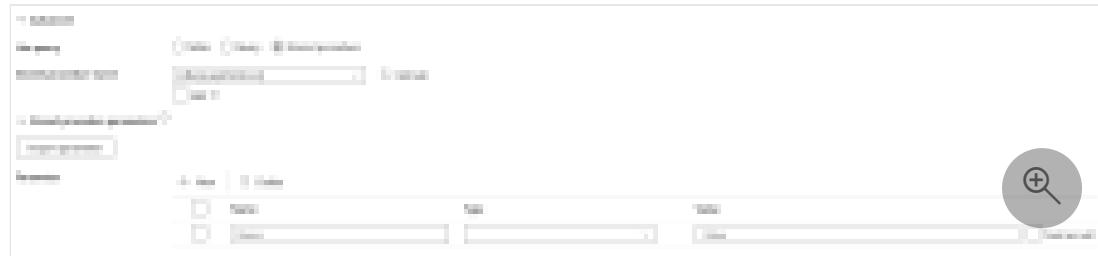
- **Data store type:** Select **External**.
- **Connection:** Select an Azure SQL Database connection from the connection list. If the connection doesn't exist, then create a new Azure SQL Database connection by selecting **New**.
- **Connection type:** Select **Azure SQL Database**.
- **Table:** Select the table in your database from the drop-down list. Or check **Edit** to enter your table name manually.
- **Preview data:** Select **Preview data** to preview the data in your table.

Under **Advanced**, you can specify the following fields:

- **Use query:** You can choose **Table**, **Query**, or **Stored procedure**. The following list describes the configuration of each setting:
 - **Table:** Read data from the table you specified in **Table** if you select this button.
 - **Query:** Specify the custom SQL query to read data. An example is `select * from MyTable`. Or select the pencil icon to edit in code editor.



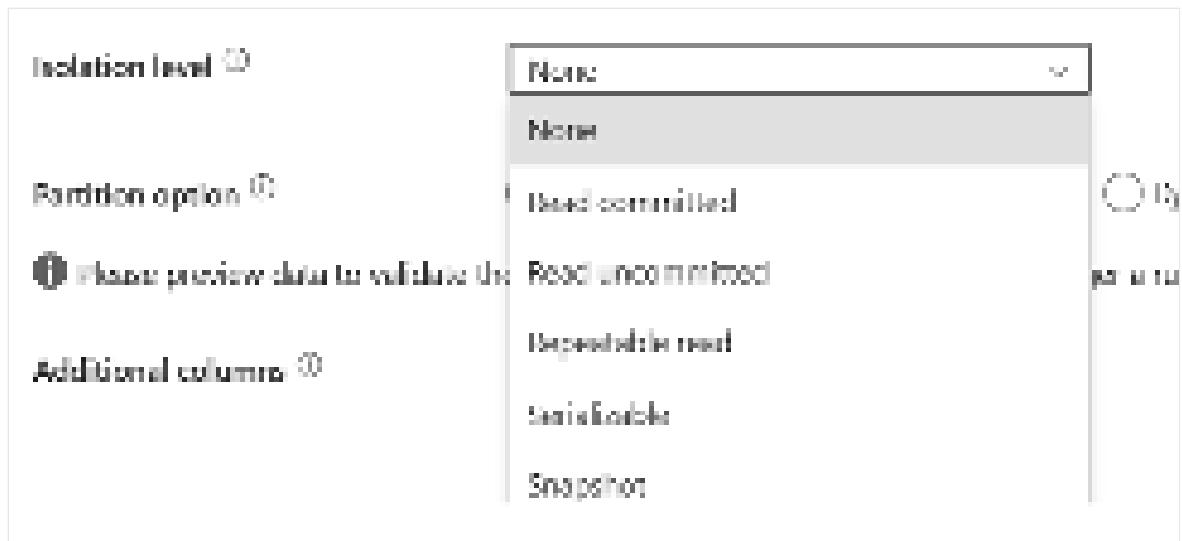
- **Stored procedure:** Use the stored procedure that reads data from the source table. The last SQL statement must be a SELECT statement in the stored procedure.
- **Stored procedure name:** Select the stored procedure or specify the stored procedure name manually when checking the **Edit** box to read data from the source table.
- **Stored procedure parameters:** Specify values for stored procedure parameters. Allowed values are name or value pairs. The names and casing of parameters must match the names and casing of the stored procedure parameters.



- **Query timeout (minutes):** Specify the timeout for query command execution, default is 120 minutes. If a parameter is set for this property, allowed values are timespan, such as "02:00:00" (120 minutes).



- **Isolation level:** Specifies the transaction locking behavior for the SQL source. The allowed values are: **None**, **ReadCommitted**, **ReadUncommitted**, **RepeatableRead**, **Serializable**, or **Snapshot**. If not specified, **None** isolation level is used. Refer to IsolationLevel Enum for more details.



- **Partition option:** Specify the data partitioning options used to load data from Azure SQL Database. Allowed values are: **None** (default), **Physical partitions of table**, and **Dynamic range**. When a partition option is enabled (that is, not **None**), the degree of parallelism to concurrently load data from an Azure SQL Database is controlled by the parallel copy setting on the copy activity.



- **None:** Choose this setting to not use a partition.
- **Physical partitions of table:** When you use a physical partition, the partition column and mechanism are automatically determined based on your physical table definition.
- **Dynamic range:** When you use a query with parallel enabled, the range partition parameter (`?AdfDynamicRangePartitionCondition`) is needed. Sample query:

```
SELECT * FROM <TableName> WHERE ?AdfDynamicRangePartitionCondition.
```

 - **Partition column name:** Specify the name of the source column in **integer or date/datetime** type (`int`, `smallint`, `bigint`, `date`, `smalldatetime`, `datetime`, `datetime2`, or `datetimeoffset`) that's used by range partitioning for parallel copy. If not specified, the index or the primary key of the table is autodetected and used as the partition column.
 - **Partition upper bound:** Specify the maximum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering the rows in table. All rows in the table or query result are partitioned and copied.
 - **Partition lower bound:** Specify the minimum value of the partition column for partition range splitting. This value is used to decide the partition stride,

not for filtering the rows in table. All rows in the table or query result are partitioned and copied.

- **Additional columns:** Add more data columns to store source files' relative path or static value. Expression is supported for the latter. For more information, go to Add additional columns during copy.

Destination

The following properties are supported for Azure SQL Database under the Destination tab of a copy activity.

The screenshot shows the 'Destination' tab of a copy activity configuration. The tab has tabs for General, Source, Destination, Mapping, and Settings. The 'Destination' tab is selected. The configuration includes:

- Data store type:** Set to External (radio button selected).
- Connection:** Set to 'azuresqldatabase' (dropdown menu).
- Connection type:** Set to 'Azure SQL Database' (dropdown menu).
- Table:** Set to 'SalesLT.Customer' (dropdown menu). Below it, there is a checkbox labeled 'Edit'.
- Write behavior:** Set to 'Insert' (radio button selected).
- Bulk insert table lock:** Set to 'Yes' (radio button selected).
- Table option:** Set to 'None' (radio button selected).
- Pre-copy script:** An empty text input field.
- Write batch timeout:** Set to '45.0000'.
- Write batch size:** An empty text input field.
- Max concurrent connections:** An empty text input field.
- Disable performance metric analysis:** A checkbox that is unchecked.

A circular button with a plus sign and a magnifying glass icon is located in the bottom right corner of the form area.

The following properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select an Azure SQL Database connection from the connection list. If the connection doesn't exist, then create a new Azure SQL Database connection by selecting **New**.
- **Connection type:** Select **Azure SQL Database**.
- **Table:** Select the table in your database from the drop-down list. Or check **Edit** to enter your table name manually.

- **Preview data:** Select **Preview data** to preview the data in your table.

Under **Advanced**, you can specify the following fields:

- **Write behavior:** Defines the write behavior when the source is files from a file-based data store. You can choose **Insert**, **Upsert** or **Stored procedure**.



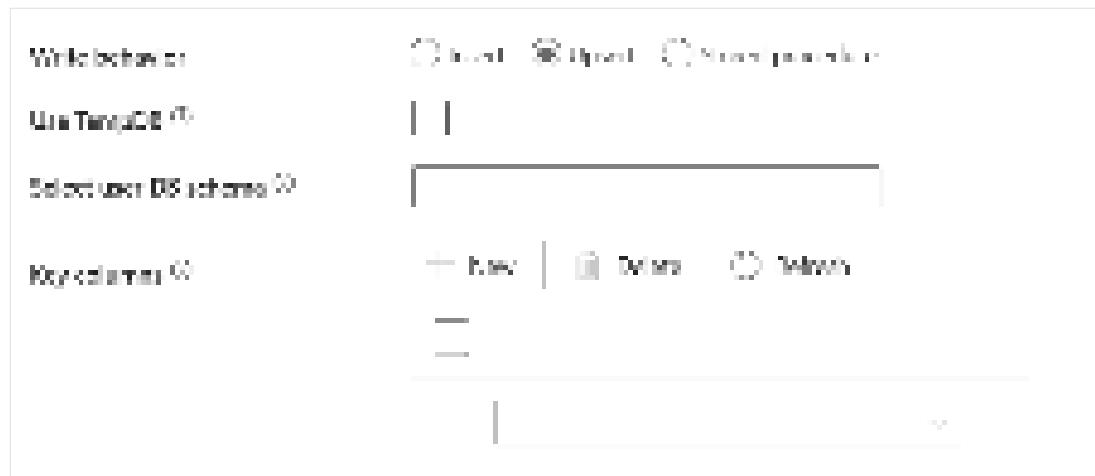
- **Insert:** Choose this option if your source data has inserts.
- **Upsert:** Choose this option if your source data has both inserts and updates.
 - **Use TempDB:** Specify whether to use a global temporary table or physical table as the interim table for upsert. By default, the service uses global temporary table as the interim table and this checkbox is selected.



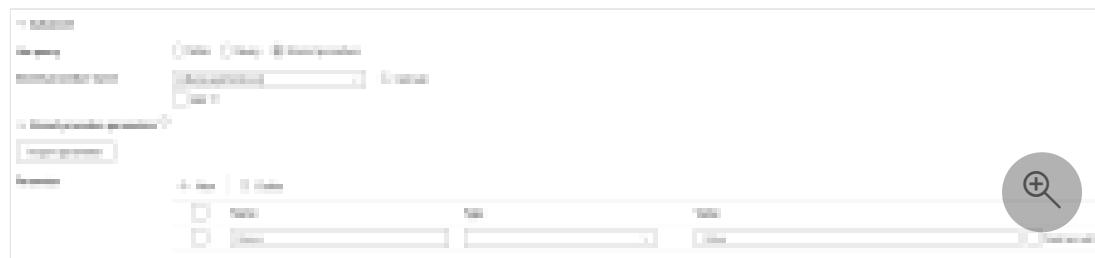
- **Select user DB schema:** When the **Use TempDB** checkbox isn't selected, specify the interim schema for creating an interim table if a physical table is used.

Note

You must have the permission for creating and deleting tables. By default, an interim table will share the same schema as a destination table.



- **Key columns:** Specify the column names for unique row identification. Either a single key or a series of keys can be used. If not specified, the primary key is used.
- **Stored procedure:** Use the stored procedure that defines how to apply source data into a target table. This stored procedure is *invoked per batch*.
- **Stored procedure name:** Select the stored procedure or specify the stored procedure name manually when checking the **Edit** box to read data from the source table.
- **Stored procedure parameters:** Specify values for stored procedure parameters. Allowed values are name or value pairs. The names and casing of parameters must match the names and casing of the stored procedure parameters.



- **Bulk insert table lock:** Choose **Yes** or **No**. Use this setting to improve copy performance during a bulk insert operation on a table with no index from multiple clients. For more information, go to [BULK INSERT \(Transact-SQL\)](#)
- **Table option:** Specifies whether to automatically create the destination table if the table doesn't exist based on the source schema. Choose **None** or **Auto create table**. Auto table creation isn't supported when destination specifies a stored procedure.
- **Pre-copy script:** Specify a script for Copy Activity to execute before writing data into a destination table in each run. You can use this property to clean up the

preloaded data.

- **Write batch timeout:** Specify the wait time for the batch insert operation to finish before it times out. The allowed value is timespan. The default value is "00:30:00" (30 minutes).
- **Write batch size:** Specify the number of rows to insert into the SQL table per batch. The allowed value is integer (number of rows). By default, the service dynamically determines the appropriate batch size based on the row size.
- **Max concurrent connections:** Specify the upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.
- **Disable performance metrics analytics:** This setting is used to collect metrics, such as DTU, DWU, RU, and so on, for copy performance optimization and recommendations. If you're concerned with this behavior, select this checkbox.

Mapping

For **Mapping** tab configuration, go to Configure your mappings under mapping tab.

Settings

For **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

The following tables contain more information about the copy activity in Azure SQL Database.

Source

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	<your connection>	Yes	connection

Name	Description	Value	Required	JSON script property
Connection type	Your connection type. Select Azure SQL Database .	Azure SQL Database	Yes	/
Table	Your source data table.	<name of your destination table>	Yes	schema table
Use query	The custom SQL query to read data.	<ul style="list-style-type: none"> • None • Query • Stored procedure 	No	<ul style="list-style-type: none"> • sqlReaderQuery • sqlReaderStoredProcedureName, storedProcedureParameters
Query timeout	The timeout for query command execution, default is 120 minutes.	timespan	No	queryTimeout
Isolation level	Specifies the transaction locking behavior for the SQL source.	<ul style="list-style-type: none"> • None • ReadCommitted • ReadUncommitted • RepeatableRead • Serializable • Snapshot 	No	isolationLevel
Partition option	The data partitioning options used to load data from Azure SQL Database.	<ul style="list-style-type: none"> • None • Physical partitions of table • Dynamic range 	No	partitionOption: <ul style="list-style-type: none"> • PhysicalPartitionsOfTable • DynamicRange
Additional columns	Add more data columns to store source files' relative path or static value. Expression is	<ul style="list-style-type: none"> • Name • Value 	No	additionalColumns: <ul style="list-style-type: none"> • name • value

Name	Description	Value	Required	JSON script property
		supported for the latter.		

Destination

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the destination data store.	<your connection >	Yes	connection
Connection type	Your connection type. Select Azure SQL Database .	Azure SQL Database	Yes	/
Table	Your destination data table.	<name of your destination table>	Yes	schema table
Write behavior	Defines the write behavior when the source is files from a file-based data store.	<ul style="list-style-type: none"> • Insert • Upsert • Stored procedure 	No	writeBehavior: <ul style="list-style-type: none"> • insert • upsert • sqlWriterStoredProcName, sqlWriterTableType, storedProcedureParameters
Bulk insert table lock	Use this setting to improve copy performance during a bulk insert operation on a table with no index from multiple clients.	Yes or No	No	sqlWriterUseTableLock: true or false
Table option	Specifies whether to automatically create the destination table if it doesn't exist based on the source schema.	<ul style="list-style-type: none"> • None • Auto create 	No	tableOption: <ul style="list-style-type: none"> • autoCreate

Name	Description	Value	Required	JSON script property
Pre-copy script	A script for Copy Activity to execute before writing data into a destination table in each run. You can use this property to clean up the preloaded data.	<pre-copy script> (string)	No	preCopyScript
Write batch timeout	The wait time for the batch insert operation to finish before it times out. The allowed value is timespan. The default value is "00:30:00" (30 minutes).	timespan	No	writeBatchTimeout
Write batch size	The number of rows to insert into the SQL table per batch. By default, the service dynamically determines the appropriate batch size based on the row size.	<number of rows> (integer)	No	writeBatchSize
Max concurrent connections	The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.	<upper limit of concurrent connections> (integer)	No	maxConcurrentConnections
Disable performance metrics analytics	This setting is used to collect metrics, such as DTU, DWU, RU,	select or unselect	No	disableMetricsCollection: true or false

Name	Description	Value	Required	JSON script property
	<p>and so on, for copy performance optimization and recommendations.</p> <p>If you're concerned with this behavior, select this checkbox.</p>			

Next steps

- Set up your Azure SQL Database connection
-

Feedback

Was this page helpful?



Yes



No

Provide product feedback [↗](#) | Ask the community [↗](#)

Azure SQL Database Managed Instance connector overview

Article • 12/08/2023

The Azure SQL Database Managed Instance connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Data pipeline

The Azure SQL Database Managed Instance connector supports the following capabilities in Data pipeline:

 Expand table

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	Basic Organizational account Service principal
Lookup activity	None	Basic Organizational account Service principal
GetMetadata activity	None	Basic Organizational account Service principal
Script activity	None	Basic Organizational account Service principal
Stored procedure activity	None	Basic Organizational account Service principal

To learn more about the copy activity configuration for Azure SQL Database Managed Instance in Data pipeline, go to [Configure in a data pipeline copy activity](#).

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

How to configure Azure SQL Database Managed Instance in copy activity

Article • 12/08/2023

This article outlines how to use the copy activity in data pipeline to copy data from and to Azure SQL Database Managed Instance.

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

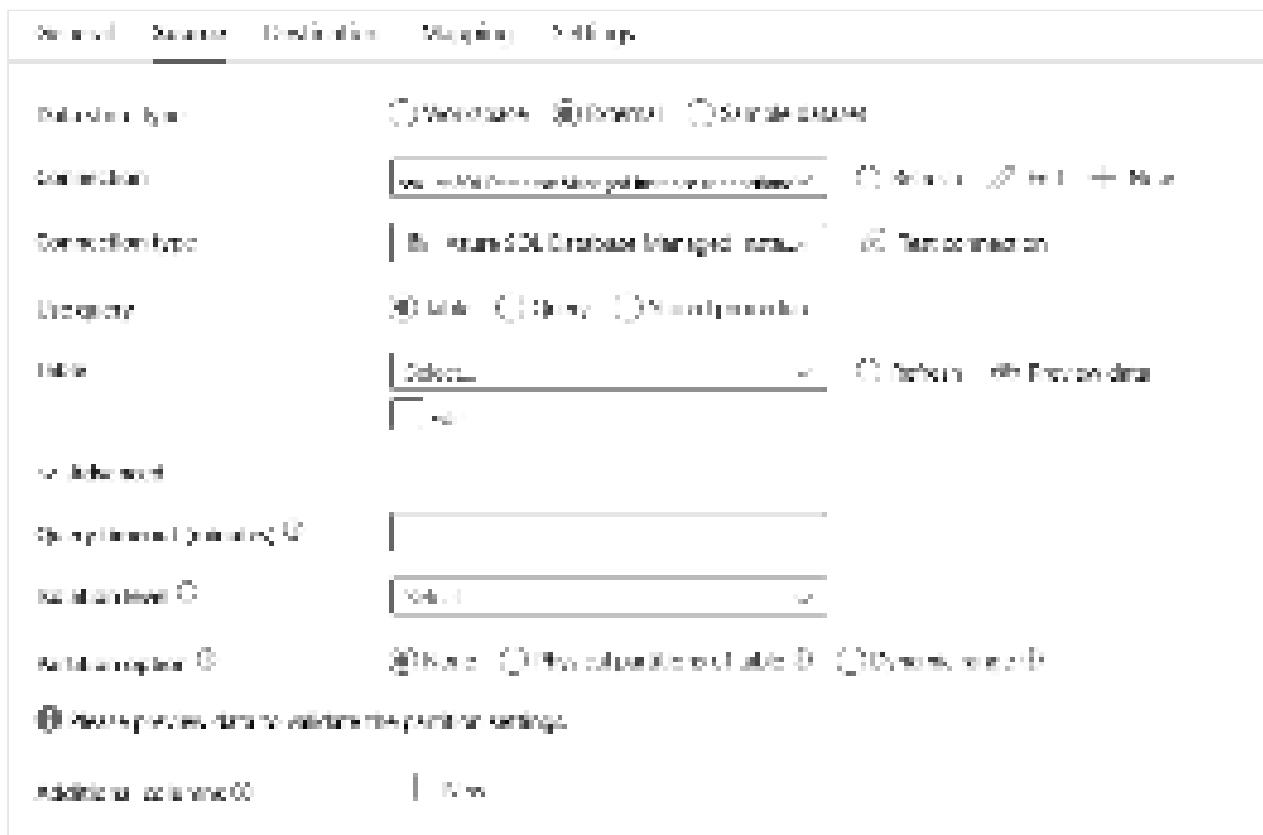
- General
- Source
- Destination
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

The following properties are supported for Azure SQL Database Managed Instance under the **Source** tab of a copy activity.



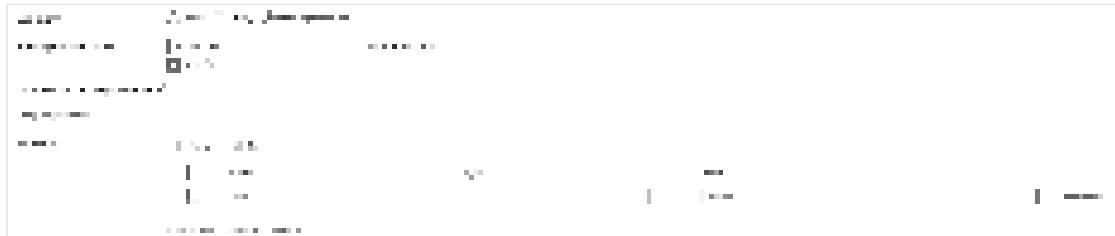
The following properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select an Azure SQL Database Managed Instance connection from the connection list. If the connection doesn't exist, then create a new Azure SQL Database Managed Instance connection by selecting **New**.
- **Connection type:** Select **Azure SQL Database Managed Instance**.
- **Use query:** Specify the way to read data. You can choose **Table**, **Query**, or **Stored procedure**. The following list describes the configuration of each setting:
 - **Table:** Read data from the table specified. Select your source table from the drop-down list or select **Edit** to enter it manually.
 - **Query:** Specify the custom SQL query to read data. An example is `select * from MyTable`. Or select the pencil icon to edit in code editor.



- **Stored procedure:** Use the stored procedure that reads data from the source table. The last SQL statement must be a SELECT statement in the stored procedure.

- **Stored procedure name:** Select the stored procedure or specify the stored procedure name manually when selecting the **Edit** to read data from the source table.
- **Stored procedure parameters:** Specify values for stored procedure parameters. Allowed values are name or value pairs. The names and casing of parameters must match the names and casing of the stored procedure parameters. You can select **Import parameters** to get your stored procedure parameters.



Under **Advanced**, you can specify the following fields:

- **Query timeout (minutes):** Specify the timeout for query command execution, default is 120 minutes. If a parameter is set for this property, allowed values are timespan, such as "02:00:00" (120 minutes).
- **Isolation level:** Specifies the transaction locking behavior for the SQL source. The allowed values are: **Read committed**, **Read uncommitted**, **Repeatable read**, **Serializable**, **Snapshot**. If not specified, the database's default isolation level is used. Refer to **IsolationLevel Enum** for more details.



- **Partition option:** Specify the data partitioning options used to load data from Azure SQL Database Managed Instance. Allowed values are: **None** (default), **Physical partitions of table**, and **Dynamic range**. When a partition option is enabled (that is, not **None**), the degree of parallelism to concurrently load data from Azure SQL Database Managed Instance is controlled by **Degree of copy parallelism** in copy activity settings tab.
 - **None:** Choose this setting to not use a partition.
 - **Physical partitions of table:** When using a physical partition, the partition column and mechanism are automatically determined based on your physical table definition.

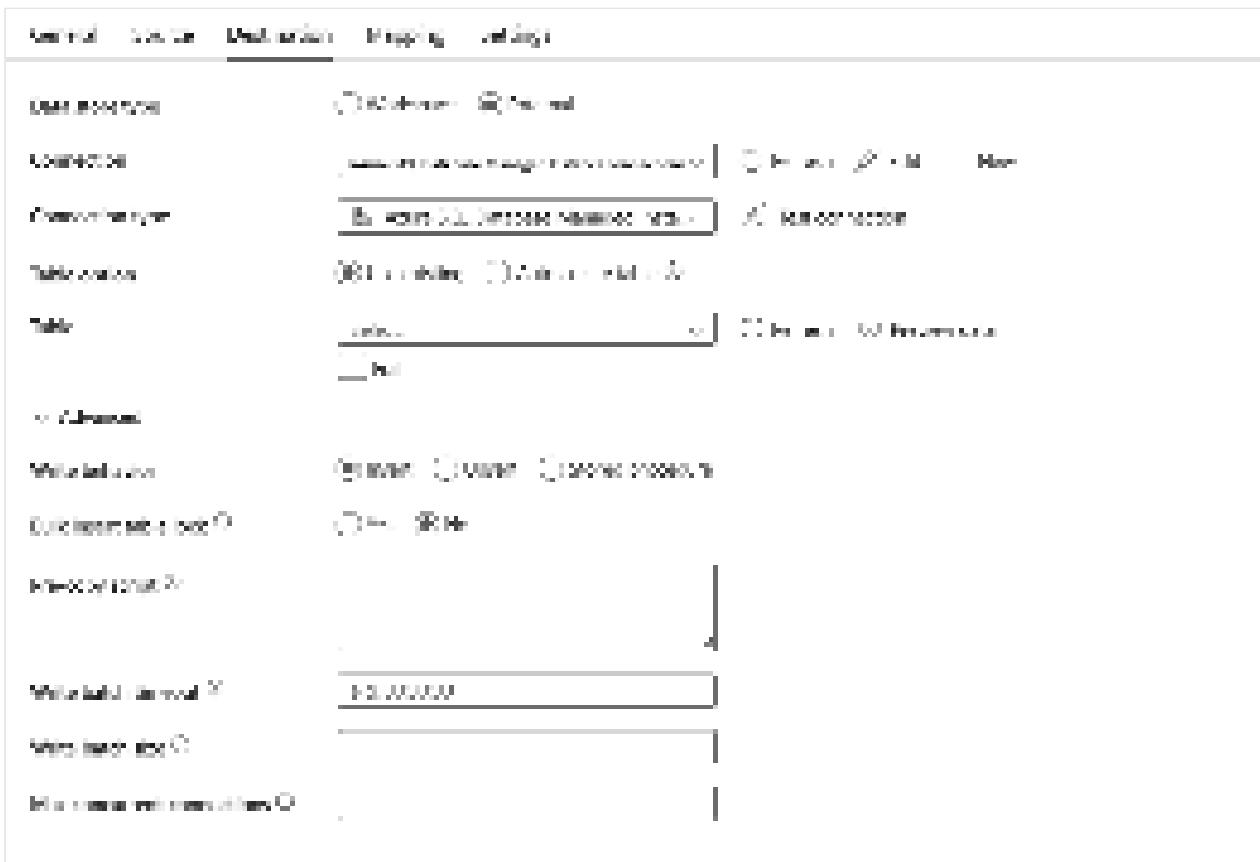
- **Dynamic range:** When using query with parallel enabled, the range partition parameter(`?AdfDynamicRangePartitionCondition`) is needed. Sample query: `SELECT * FROM <TableName> WHERE ?AdfDynamicRangePartitionCondition.`
- **Partition column name:** Specify the name of the source column in `integer` or `date/datetime` type (`int`, `smallint`, `bigint`, `date`, `smalldatetime`, `datetime`, `datetime2`, or `datetimeoffset`) that's used by range partitioning for parallel copy. If not specified, the index or the primary key of the table is auto-detected and used as the partition column.
 If you use a query to retrieve the source data, hook `?` `AdfDynamicRangePartitionCondition` in the WHERE clause. For an example, see the Parallel copy from Azure SQL Database Managed Instance section.
- **Partition upper bound:** Specify the maximum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering the rows in table. All rows in the table or query result will be partitioned and copied. If not specified, copy activity auto detect the value. For an example, see the Parallel copy from Azure SQL Database Managed Instance section.
- **Partition lower bound:** Specify the minimum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering the rows in table. All rows in the table or query result will be partitioned and copied. If not specified, copy activity auto detect the value. For an example, see the Parallel copy from Azure SQL Database Managed Instance section.
- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.

Note the following points:

- If **Query** is specified for source, the copy activity runs this query against the Azure SQL Database Managed Instance source to get the data. You also can specify a stored procedure by specifying **Stored procedure name** and **Stored procedure parameters** if the stored procedure takes parameters.
- When using stored procedure in source to retrieve data, note if your stored procedure is designed as returning different schema when different parameter value is passed in, you might encounter failure or see unexpected result when importing schema from UI or when copying data to SQL database with auto table creation.

Destination

The following properties are supported for Azure SQL Database Managed Instance under the **Destination** tab of a copy activity.



The following properties are required:

- **Data store type:** Select **External**.
- **Connection:** Select an Azure SQL Database Managed Instance connection from the connection list. If the connection doesn't exist, then create a new Azure SQL Database Managed Instance connection by selecting **New**.
- **Connection type:** Select **Azure SQL Database Managed Instance**.
- **Table option:** You can choose **Use existing** to use the table specified. Or choose **Auto create table** to automatically create a destination table if the table doesn't exist in the source schema, and note that this selection is not supported when stored procedure is used as the write behavior.

If you select **Use existing**:

- **Table:** Select the table in your destination database from the drop-down list. Or check **Edit** to enter your table name manually.

If you select: **Auto create table**:

- **Table:** Specify the name for your auto-created destination table.

Under **Advanced**, you can specify the following fields:

- **Write behavior:** Defines the write behavior when the source is files from a file-based data store. You can choose **Insert**, **Upsert** or **Stored procedure**.

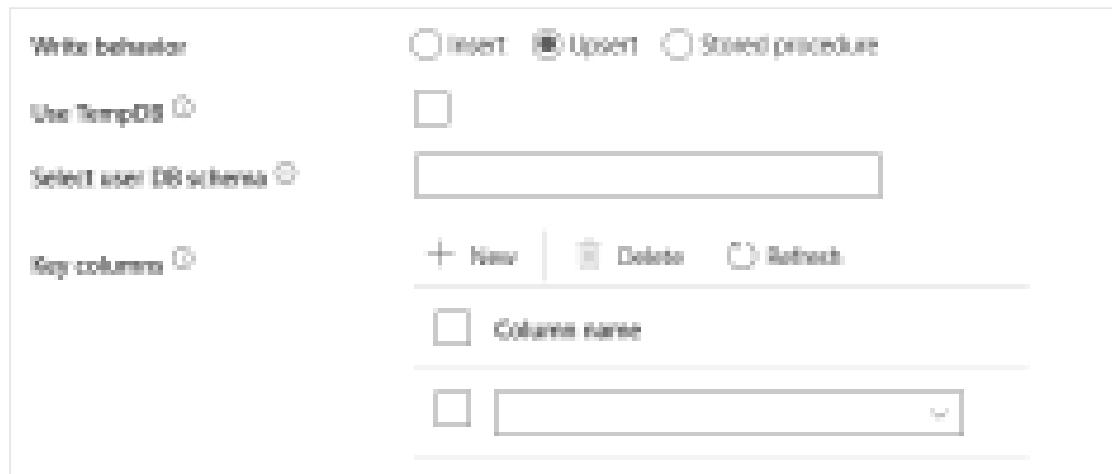
- **Insert:** Choose this option use insert write behavior to load data into Azure SQL Database Managed Instance.
- **Upsert:** Choose this option use upsert write behavior to load data into Azure SQL Database Managed Instance.
- **Use TempDB:** Specify whether to use a global temporary table or physical table as the interim table for upsert. By default, the service uses global temporary table as the interim table and this property is selected.



- **Select user DB schema:** When the **Use TempDB** isn't selected, specify the interim schema for creating an interim table if a physical table is used.

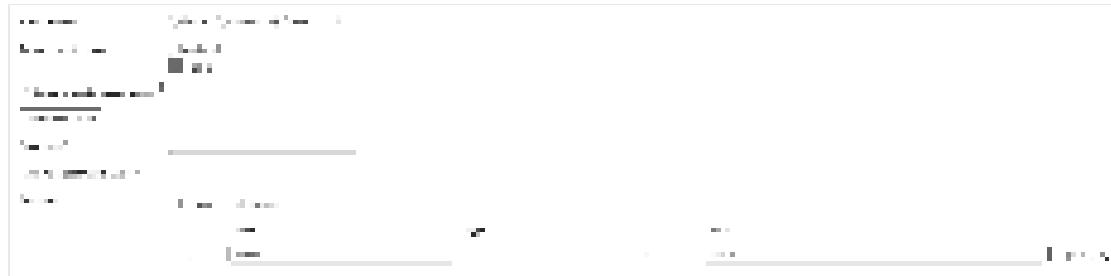
Note

You must have the permission for creating and deleting tables. By default, an interim table will share the same schema as a destination table.



- **Key columns:** Specify the column names for unique row identification. Either a single key or a series of keys can be used. If not specified, the primary key is used.
- **Stored procedure:** Use the stored procedure that defines how to apply source data into a target table. This stored procedure is *invoked per batch*. For operations that run only once and have nothing to do with source data, for example, delete or truncate, use **Pre-copy script** property.

- **Stored procedure name:** Select the stored procedure or specify the stored procedure name manually when checking the **Edit** to read data from the source table.
- **Stored procedure parameters:**
 - **Table type:** Specify the table type name to be used in the stored procedure. The copy activity makes the data being moved available in a temporary table with this table type. Stored procedure code can then merge the data that's being copied with existing data.
 - **Table type parameter name:** Specify the parameter name of the table type specified in the stored procedure.
 - **Parameters:** Specify values for stored procedure parameters. Allowed values are name or value pairs. The names and casing of parameters must match the names and casing of the stored procedure parameters. You can select **Import parameters** to get your stored procedure parameters.



- **Bulk insert table lock:** Choose **Yes** or **No** (default). Use this setting to improve copy performance during a bulk insert operation on a table with no index from multiple clients. You can specify this property when you select **Insert** or **Upsert** as your write behavior. For more information, go to BULK INSERT (Transact-SQL)
- **Pre-copy script:** Specify a script for copy activity to execute before writing data into a destination table in each run. You can use this property to clean up the pre-loaded data.
- **Write batch timeout:** Specify the wait time for the batch insert operation to finish before it times out. The allowed value is timespan. If no value is specified, the timeout defaults to "02:00:00".
- **Write batch size:** Specify the number of rows to insert into the SQL table per batch. The allowed value is integer (number of rows). By default, the service dynamically determines the appropriate batch size based on the row size.
- **Max concurrent connections:** The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.

Mapping

For Mapping tab configuration, go to Configure your mappings under mapping tab.

Settings

For **Settings** tab configuration, go to Configure your other settings under settings tab.

Parallel copy from Azure SQL Database Managed Instance

The Azure SQL Database Managed Instance connector in copy activity provides built-in data partitioning to copy data in parallel. You can find data partitioning options on the **Source** tab of the copy activity.

When you enable partitioned copy, copy activity runs parallel queries against your Azure SQL Database Managed Instance source to load data by partitions. The parallel degree is controlled by the **Degree of copy parallelism** in the copy activity settings tab. For example, if you set **Degree of copy parallelism** to four, the service concurrently generates and runs four queries based on your specified partition option and settings, and each query retrieves a portion of data from your Azure SQL Database Managed Instance.

You are suggested to enable parallel copy with data partitioning especially when you load large amount of data from your Azure SQL Database Managed Instance. The following are suggested configurations for different scenarios. When copying data into file-based data store, it's recommended to write to a folder as multiple files (only specify folder name), in which case the performance is better than writing to a single file.

[+] Expand table

Scenario	Suggested settings
Full load from large table, with physical partitions.	<p>Partition option: Physical partitions of table.</p> <p>During execution, the service automatically detects the physical partitions, and copies data by partitions.</p> <p>To check if your table has physical partition or not, you can refer to this query.</p>
Full load from large table, without physical partitions, while with an integer or datetime column for data partitioning.	<p>Partition options: Dynamic range partition.</p> <p>Partition column (optional): Specify the column used to partition data. If not specified, the index or primary key column is used.</p> <p>Partition upper bound and partition lower bound (optional): Specify if you want to determine the partition stride. This is not for filtering the rows in table, all rows in the table will be partitioned and copied. If not specified, copy activity auto detects the values.</p> <p>For example, if your partition column "ID" has values range from 1 to 100, and you set the lower bound as 20 and the upper bound as 80,</p>

Scenario	Suggested settings
<p>Load a large amount of data by using a custom query, without physical partitions, while with an integer or date/datetime column for data partitioning.</p>	<p>Partition options: Dynamic range partition.</p> <p>Query: <code>SELECT * FROM <TableName> WHERE ?</code></p> <pre>AdfDynamicRangePartitionCondition AND <your_additional_where_clause></pre> <p>Partition column: Specify the column used to partition data.</p> <p>Partition upper bound and partition lower bound (optional): Specify if you want to determine the partition stride. This is not for filtering the rows in table, all rows in the query result will be partitioned and copied. If not specified, copy activity auto detect the value.</p> <p>During execution, the service replaces <code>?AdfRangePartitionColumnName</code> with the actual column name and value ranges for each partition, and sends to Azure SQL Database Managed Instance.</p> <p>For example, if your partition column "ID" has values range from 1 to 100, and you set the lower bound as 20 and the upper bound as 80, with parallel copy as 4, the service retrieves data by 4 partitions- IDs in range <=20, [21, 50], [51, 80], and >=81, respectively.</p> <p>Here are more sample queries for different scenarios:</p> <ol style="list-style-type: none"> 1. Query the whole table: <code>SELECT * FROM <TableName> WHERE ?</code> <pre>AdfDynamicRangePartitionCondition</pre> 2. Query from a table with column selection and additional where-clause filters: <code>SELECT <column_list> FROM <TableName> WHERE ?</code> <pre>AdfDynamicRangePartitionCondition AND <your_additional_where_clause></pre> 3. Query with subqueries: <code>SELECT <column_list> FROM (<your_sub_query>) AS T WHERE ?</code> <pre>AdfDynamicRangePartitionCondition AND <your_additional_where_clause></pre> 4. Query with partition in subquery: <code>SELECT <column_list> FROM (SELECT <your_sub_query_column_list> FROM <TableName> WHERE ?AdfDynamicRangePartitionCondition) AS T</code>

Best practices to load data with partition option:

1. Choose distinctive column as partition column (like primary key or unique key) to avoid data skew.
2. If the table has built-in partition, use partition option **Physical partitions of table** to get better performance.

Sample query to check physical partition

SQL

```
SELECT DISTINCT s.name AS SchemaName, t.name AS TableName, pf.name AS PartitionFunctionName, c.name AS ColumnName, iif(pf.name is null, 'no', 'yes') AS HasPartition
FROM sys.tables AS t
LEFT JOIN sys.objects AS o ON t.object_id = o.object_id
LEFT JOIN sys.schemas AS s ON o.schema_id = s.schema_id
LEFT JOIN sys.indexes AS i ON t.object_id = i.object_id
LEFT JOIN sys.index_columns AS ic ON ic.partition_ordinal > 0 AND ic.index_id = i.index_id AND ic.object_id = t.object_id
LEFT JOIN sys.columns AS c ON c.object_id = ic.object_id AND c.column_id = ic.column_id
LEFT JOIN sys.partition_schemes ps ON i.data_space_id = ps.data_space_id
LEFT JOIN sys.partition_functions pf ON pf.function_id = ps.function_id
WHERE s.name='[your schema]' AND t.name = '[your table name]'
```

If the table has physical partition, you would see "HasPartition" as "yes" like the following.

SchemaName	TableName	PartitionFunctionName	ColumnName	HasPartition
dbo	AdventureworksLT	Range_1	H	yes

Table summary

See the following table for the summary and more information for the Azure SQL Database Managed Instance copy activity.

Source information

 Expand table

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	< your connection >	Yes	connection
Connection type	Your connection type. Select Azure SQL Database Managed Instance.	Azure SQL Database Managed Instance	Yes	/
Use query	The custom SQL query to read data.	<ul style="list-style-type: none">• Table• Query• Stored procedure	Yes	/

Name	Description	Value	Required	JSON script property
Table	Your source data table.	< name of your table>	No	schema table
Query	The custom SQL query to read data.	< your query >	No	sqlReaderQuery
Stored procedure name	This property is the name of the stored procedure that reads data from the source table. The last SQL statement must be a SELECT statement in the stored procedure.	< stored procedure name >	No	sqlReaderStoredProcedureName
Stored procedure parameter	These parameters are for the stored procedure. Allowed values are name or value pairs. The names and casing of parameters must match the names and casing of the stored procedure parameters.	< name or value pairs >	No	storedProcedureParameters
Query timeout	The timeout for query command execution.	timespan (the default is 120 minutes)	No	queryTimeout
Isolation level	Specifies the transaction locking behavior for the SQL source.	<ul style="list-style-type: none"> • Read committed • Read uncommitted • Repeatable read • Serializable • Snapshot 	No	isolationLevel: <ul style="list-style-type: none"> • ReadCommitted • ReadUncommitted • RepeatableRead • Serializable • Snapshot
Partition option	The data partitioning options used to load data from Azure SQL Database Managed Instance.	<ul style="list-style-type: none"> • None (default) • Physical partitions of table • Dynamic range 	No	partitionOption: <ul style="list-style-type: none"> • None (default) • PhysicalPartitionsOfTable • DynamicRange
Partition column name	<p>The name of the source column in integer or date/datetime type (<code>int</code>, <code>smallint</code>, <code>bigint</code>, <code>date</code>, <code>smalldatetime</code>, <code>datetime</code>, <code>datetime2</code>, or <code>datetimeoffset</code>) that's used by range partitioning for parallel copy. If not specified, the index or the primary key of the table is auto-detected and used as the partition column. If you use a</p>	< your partition column names >	No	partitionColumnName

Name	Description	Value	Required	JSON script property
	query to retrieve the source data, hook ? <code>AdfDynamicRangePartitionCondition</code> in the WHERE clause.			
Partition upper bound	The maximum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering the rows in table. All rows in the table or query result will be partitioned and copied. If not specified, copy activity auto detect the value.	< your partition upper bound >	No	partitionUpperBound
Partition lower bound	The minimum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering the rows in table. All rows in the table or query result will be partitioned and copied. If not specified, copy activity auto detect the value.	< your partition lower bound >	No	partitionLowerBound
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.	• Name • Value	No	additionalColumns: • name • value

Destination information

[] [Expand table](#)

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the destination data store.	< your connection >	Yes	connection
Connection type	Your connection type. Select Azure SQL Database Managed Instance	Azure SQL Database Managed Instance	Yes	/

Name	Description	Value	Required	JSON script property
Table option	Specifies whether to automatically create the destination table if it doesn't exist based on the source schema.	<ul style="list-style-type: none"> • Use existing • Auto create table 	Yes	tableOption: <ul style="list-style-type: none"> • autoCreate
Table	Your destination data table.	<name of your table>	Yes	schema table
Write behavior	The write behavior for copy activity to load data into Azure SQL Database Managed Instance database..	<ul style="list-style-type: none"> • Insert • Upsert • Stored procedure 	No	writeBehavior: <ul style="list-style-type: none"> • insert • upsert sqlWriterStoredProcName, sqlWriterTableType, storedProcTableTypeParameterName, storedProcParameters
Use TempDB	Whether to use the global temporary table or physical table as the interim table for upsert.	selected (default) or unselected	No	useTempDB: true (default) or false
Select user DB schema	The interim schema for creating interim table if physical table is used. Note: user need to have the permission for creating and deleting table. By default, interim table will share the same schema as destination table. Apply when you don't select Use TempDB.	selected (default) or unselected	No	interimSchemaName

Name	Description	Value	Required	JSON script property
Key columns	The column names for unique row identification. Either a single key or a series of keys can be used. If not specified, the primary key is used.	< your key column>	No	keys
Stored procedure name	The name of the stored procedure that defines how to apply source data into a target table. This stored procedure is <i>invoked per batch</i> . For operations that run only once and have nothing to do with source data, for example, delete or truncate, use the Pre-copy script property.	< your stored procedure name >	No	sqlWriterStoredProcedureName
Table type	The table type name to be used in the stored procedure. The copy activity makes the data being moved available in a temporary table with this table type. Stored procedure code can then merge the data that's being copied	< your table type name >	No	sqlWriterTableType

Name	Description	Value	Required	JSON script property
	with existing data.			
Table type parameter name	The parameter name of the table type specified in the stored procedure.	< your parameter name of the table type >	No	storedProcedureTableNameParameterName
Parameters	Parameters for the stored procedure. Allowed values are name and value pairs. Names and casing of parameters must match the names and casing of the stored procedure parameters.	< name and value pairs >	No	storedProcedureParameters
Bulk insert table lock	Use this setting to improve copy performance during a bulk insert operation on a table with no index from multiple clients.	Yes or No (default)	No	sqlWriterUseTableLock: true or false (default)
Pre-copy script	A script for copy activity to execute before writing data into a destination table in each run. You can use this property to clean up the pre-loaded data.	< pre-copy script > (string)	No	preCopyScript
Write batch timeout	The wait time for the batch insert operation	timespan (the default is "02:00:00")	No	writeBatchTimeout

Name	Description	Value	Required	JSON script property
Write batch size	The number of rows to insert into the SQL table per batch. By default, the service dynamically determines the appropriate batch size based on the row size.	< number of rows > (integer)	No	writeBatchSize
Max concurrent connections	The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.	< upper limit of concurrent connections > (integer)	No	maxConcurrentConnections

Next steps

- Azure SQL Database Managed Instance overview

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Azure Synapse Analytics connector overview

Article • 11/15/2023

This Azure Synapse Analytics connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to Azure Synapse Analytics in Dataflow Gen2, go to Set up your connection in Dataflow Gen2.

Support in data pipelines

The Azure Synapse analytics connector supports the following capabilities in data pipelines.

Supported capabilities	Gateway	Authentication
Copy activity (Source/Destination)	None	Basic Organizational account Service principal
Lookup activity	None	Basic Organizational account Service principal
GetMetadata activity	None	Basic Organizational account Service principal
Script activity	None	Basic Organizational account Service principal
Stored procedure activity	None	Basic Organizational account Service principal

To learn about the copy activity configuration for Azure Synapse Analytics in data pipelines, go to [Configure Azure Synapse Analytics in a copy activity](#).

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your Azure Synapse Analytics connection

Article • 11/15/2023

This article outlines the steps to create an Azure Synapse Analytics connection.

Supported authentication types

The Azure Synapse Analytics connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Basic	✓	✓
Organizational account	✓	✓
Service principal	✓	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Azure Synapse Analytics. The following links provide the specific Power Query connector information you need to connect to Azure Synapse Analytics in Dataflow Gen2:

- To get started using the Azure Synapse Analytics connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric \(Preview\)](#).
- Be sure to install or set up any Azure Synapse Analytics prerequisites before connecting to the Azure Synapse Analytics connector.
- To connect to the Azure Synapse Analytics connector from Power Query, go to [Connect to Azure Synapse Analytics from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Configure Azure Synapse Analytics in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in data pipeline to copy data from and to Azure Synapse Analytics.

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

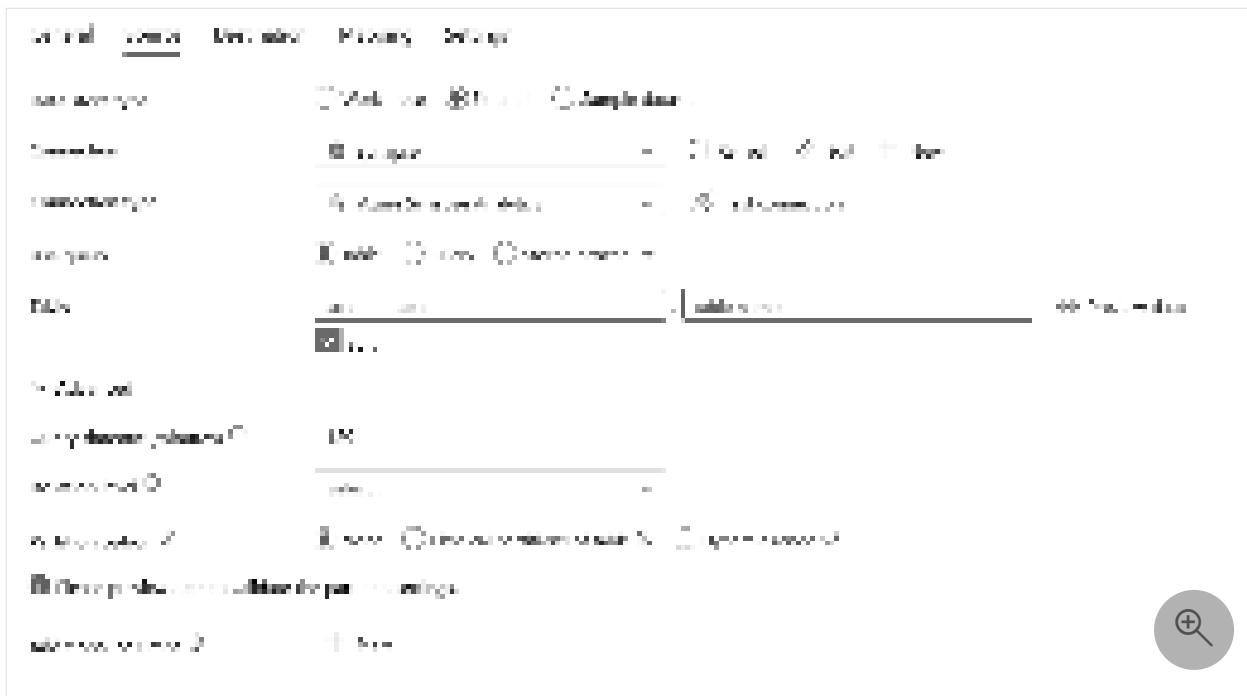
- General
- Source
- Destination
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

The following properties are supported for Azure Synapse Analytics under the **Source** tab of a copy activity.



The following properties are **required**:

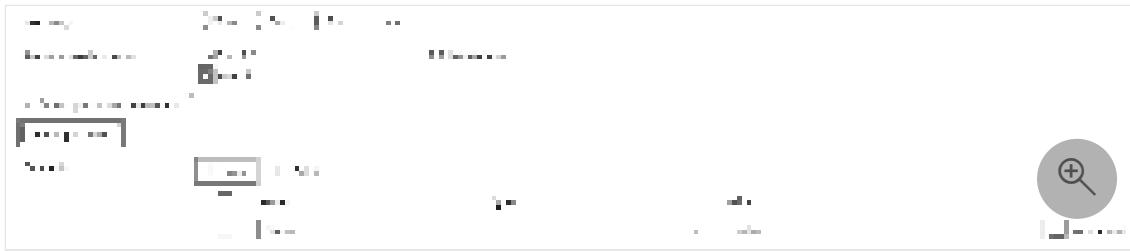
- **Data store type:** Select **External**.
- **Connection:** Select an Azure Synapse Analytics connection from the connection list. If the connection doesn't exist, then create a new Azure Synapse Analytics connection by selecting **New**.
- **Connection type:** Select **Azure Synapse Analytics**.
- **Use query:** You can choose **Table**, **Query**, or **Stored procedure** to read your source data. The following list describes the configuration of each setting:
 - **Table:** Read data from the table you specified in **Table** if you select this button. Select your table from the drop-down list or select **Edit** to enter the schema and table name manually.



- **Query:** Specify the custom SQL query to read data. An example is `select * from MyTable`. Or select the pencil icon to edit in code editor.



- **Stored procedure:** Use the stored procedure that reads data from the source table. The last SQL statement must be a SELECT statement in the stored procedure.



- **Stored procedure name:** Select the stored procedure or specify the stored procedure name manually when select **Edit**.
- **Stored procedure parameters:** Select **Import parameters** to import the parameter in your specified stored procedure, or add parameters for the stored procedure by selecting **+ New**. Allowed values are name or value pairs. Names and casing of parameters must match the names and casing of the stored procedure parameters.

Under **Advanced**, you can specify the following fields:

- **Query timeout (minutes):** Specify the timeout for query command execution, default is 120 minutes. If a parameter is set for this property, allowed values are timespan, such as "02:00:00" (120 minutes).
- **Isolation level:** Specifies the transaction locking behavior for the SQL source. The allowed values are: **None**, **Read committed**, **Read uncommitted**, **Repeatable read**, **Serializable**, or **Snapshot**. If not specified, **None** isolation level is used. Refer to IsolationLevel Enum for more details.



- **Partition option:** Specify the data partitioning options used to load data from Azure Synapse Analytics. Allowed values are: **None** (default), **Physical partitions of table**, and **Dynamic range**. When a partition option is enabled (that is, not **None**),

the degree of parallelism to concurrently load data from an Azure Synapse Analytics is controlled by the parallel copy setting on the copy activity.

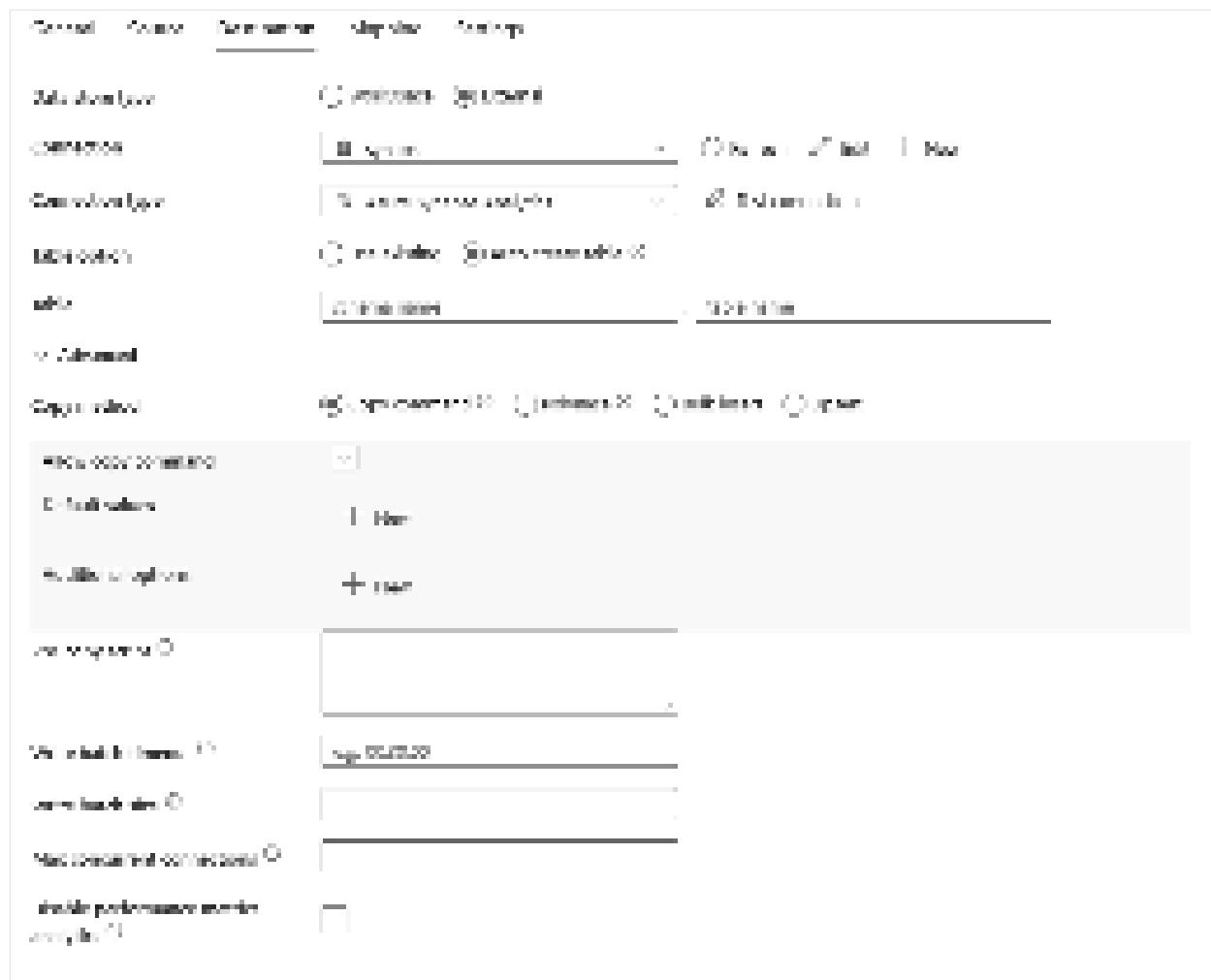
- **None:** Choose this setting to not use a partition.
- **Physical partitions of table:** Choose this setting if you want to use a physical partition. The partition column and mechanism are automatically determined based on your physical table definition.
- **Dynamic range:** Choose this setting if you want to use dynamic range partition. When using query with parallel enabled, the range partition parameter(?) `AdfDynamicRangePartitionCondition`) is needed. Sample query: `SELECT * FROM <TableName> WHERE ?AdfDynamicRangePartitionCondition.`



- **Partition column name:** Specify the name of the source column in **integer or date/datetime type** (`int`, `smallint`, `bigint`, `date`, `smalldatetime`, `datetime`, `datetime2`, or `datetimeoffset`) that's used by range partitioning for parallel copy. If not specified, the index or the primary key of the table is autodetected and used as the partition column.
- **Partition upper bound:** Specify the maximum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering the rows in table. All rows in the table or query result are partitioned and copied.
- **Partition lower bound:** Specify the minimum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering the rows in table. All rows in the table or query result are partitioned and copied.
- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter. For more information, go to [Add additional columns during copy](#).

Destination

The following properties are supported for Azure Synapse Analytics under the Destination tab of a copy activity.



The following properties are required:

- **Data store type:** Select **External**.
- **Connection:** Select an Azure Synapse Analytics connection from the connection list. If the connection doesn't exist, then create a new Azure Synapse Analytics connection by selecting **New**.
- **Connection type:** Select **Azure Synapse Analytics**.
- **Table option:** You can choose **Use existing**, **Auto create table**. The following list describes the configuration of each setting:
 - **Use existing:** Select the table in your database from the drop-down list. Or check **Edit** to enter your schema and table name manually.
 - **Auto create table:** It automatically creates the table (if nonexistent) in source schema.

Under **Advanced**, you can specify the following fields:

- **Copy method** Choose the method that you want to use to copy data. You can choose **Copy command**, **PolyBase**, **Bulk insert** or **Upsert**. The following list

describes the configuration of each setting:

- **Copy command:** Use COPY statement to load data from Azure storage into Azure Synapse Analytics or SQL Pool.



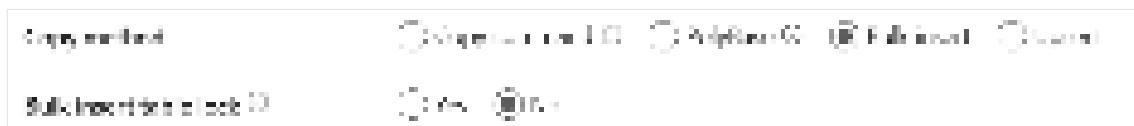
- **Allow copy command:** It is mandatory to be selected when you choose **Copy command**.
- **Default values:** Specify the default values for each target column in Azure Synapse Analytics. The default values in the property overwrite the DEFAULT constraint set in the data warehouse, and identity column cannot have a default value.
- **Additional options:** Additional options that will be passed to an Azure Synapse Analytics COPY statement directly in "With" clause in COPY statement. Quote the value as needed to align with the COPY statement requirements.
- **PolyBase:** PolyBase is a high-throughput mechanism. Use it to load large amounts of data into Azure Synapse Analytics or SQL Pool.



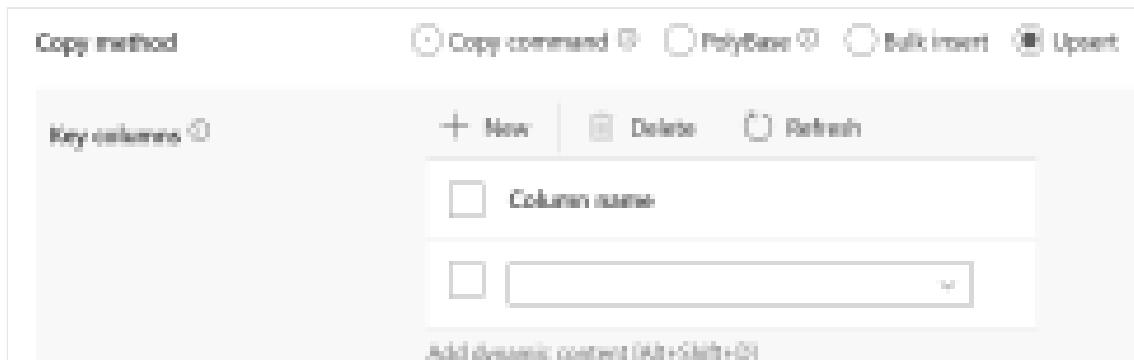
- **Allow PolyBase:** It is mandatory to be selected when you choose **PolyBase**.
- **Reject type:** Specify whether the **rejectValue** option is a literal value or a percentage. Allowed values are **Value** (default) and **Percentage**.
- **Reject value:** Specify the number or percentage of rows that can be rejected before the query fails. Learn more about PolyBase's reject options in the

Arguments section of CREATE EXTERNAL TABLE (Transact-SQL). Allowed values are 0 (default), 1, 2, etc.

- **Reject sample value:** Determines the number of rows to retrieve before PolyBase recalculates the percentage of rejected rows. Allowed values are 1, 2, etc. If you choose **Percentage** as your reject type, this property is required.
- **Use type default:** Specify how to handle missing values in delimited text files when PolyBase retrieves data from the text file. Learn more about this property from the Arguments section in CREATE EXTERNAL FILE FORMAT (Transact-SQL). Allowed values are selected (default) or unselected.
- **Bulk insert:** Use **Bulk insert** to insert data to destination in bulk.



- **Bulk insert table lock:** Use this to improve copy performance during bulk insert operation on table with no index from multiple clients. Learn more from BULK INSERT (Transact-SQL).
- **Upsert:** Specify the group of the settings for write behavior when you want to upsert data to your destination.



- **Key columns:** Choose which column is used to determine if a row from the source matches a row from the destination.
- **Bulk insert table lock:** Use this to improve copy performance during bulk insert operation on table with no index from multiple clients. Learn more from BULK INSERT (Transact-SQL).
- **Pre-copy script:** Specify a script for Copy Activity to execute before writing data into a destination table in each run. You can use this property to clean up the pre-loaded data.
- **Write batch timeout:** Specify the wait time for the batch insert operation to finish before it times out. The allowed value is timespan. The default value is "00:30:00"

(30 minutes).

- **Write batch size:** Specify the number of rows to insert into the SQL table per batch. The allowed value is integer (number of rows). By default, the service dynamically determines the appropriate batch size based on the row size.
- **Max concurrent connections:** Specify the upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.
- **Disable performance metrics analytics:** This setting is used to collect metrics, such as DTU, DWU, RU, and so on, for copy performance optimization and recommendations. If you're concerned with this behavior, select this checkbox. It is unselected by default.

Direct copy by using COPY command

Azure Synapse Analytics COPY command directly supports **Azure Blob Storage** and **Azure Data Lake Storage Gen2** as source data stores. If your source data meets the criteria described in this section, use COPY command to copy directly from the source data store to Azure Synapse Analytics.

1. The source data and format contain the following types and authentication methods:

Supported source data store type	Supported format	Supported source authentication type
Azure Blob Storage	Delimited text Parquet	Anonymous authentication Account key authentication Shared access signature authentication
Azure Data Lake Storage Gen2	Delimited text Parquet	Account key authentication Shared access signature authentication

2. The following Format settings can be set:

- a. For **Parquet**: **Compression type** can be **None**, **snappy**, or **gzip**.
- b. For **DelimitedText**:
 - i. **Row delimiter**: When copying delimited text to Azure Synapse Analytics via direct COPY command, specify the row delimiter explicitly (\r; \n; or \r\n). Only when the row delimiter of the source file is \r\n, the default value (\r, \n, or \r\n) works. Otherwise, enable staging for your scenario.

- ii. **Null value** is left as default or set to **empty string ("")**.
 - iii. **Encoding** is left as default or set to **UTF-8** or **UTF-16**.
 - iv. **Skip line count** is left as default or set to 0.
 - v. **Compression type** can be **None** or **gzip**.
3. If your source is a folder, you must select **Recursively** checkbox.
4. **Start time (UTC)** and **End time (UTC)** in **Filter by last modified**, **Prefix**, **Enable partition discovery**, and **Additional columns** aren't specified.

To learn how to ingest data into your Azure Synapse Analytics using the **COPY** command, see this article.

If your source data store and format isn't originally supported by a **COPY** command, use the **Staged copy** by using the **COPY** command feature instead. It automatically converts the data into a **COPY** command compatible format, then calls a **COPY** command to load data into Azure Synapse Analytics.

Mapping

For **Mapping** tab configuration, go to Configure your mappings under mapping tab.

Settings

For **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

The following tables contain more information about the copy activity in Azure Synapse Analytics.

Source

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	< your connection >	Yes	connection

Name	Description	Value	Required	JSON script property
Connection type	Your source connection type.	Azure Synapse Analytics	Yes	/
Use query	The way to read data.	<ul style="list-style-type: none"> • Table • Query • Stored procedure 	Yes	<ul style="list-style-type: none"> • typeProperties (under <code>typeProperties -> source</code>) - schema - table • <code>sqlReaderQuery</code> • <code>sqlReaderStoredProcedureName</code> <code>storedProcedureParameters</code> - name - value
Query timeout	The timeout for query command execution, default is 120 minutes.	timespan	No	<code>queryTimeout</code>
Isolation level	The transaction locking behavior for the SQL source.	<ul style="list-style-type: none"> • None • Read committed • Read uncommitted • Repeatable read • Serializable • Snapshot 	No	<code>isolationLevel:</code> <ul style="list-style-type: none"> • ReadCommitted • ReadUncommitted • RepeatableRead • Serializable • Snapshot
Partition option	The data partitioning options used to load data from Azure SQL Database.	<ul style="list-style-type: none"> • None • Physical partitions of table • Dynamic range <ul style="list-style-type: none"> - Partition column name - Partition upper bound - Partition lower bound 	No	<code>partitionOption:</code> <ul style="list-style-type: none"> • PhysicalPartitionsOfTable • DynamicRange <code>partitionSettings:</code> <ul style="list-style-type: none"> - <code>partitionColumnName</code> - <code>partitionUpperBound</code> - <code>partitionLowerBound</code>
Additional columns	Add additional data columns to store source files' relative path or static value.	<ul style="list-style-type: none"> • Name • Value 	No	<code>additionalColumns:</code> <ul style="list-style-type: none"> • name • value

Name	Description	Value	Required	JSON script property
	Expression is supported for the latter.			

Destination

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the destination data store.	< your connection >	Yes	connection
Connection type	Your destination connection type.	Azure Synapse Analytics	Yes	/
Table option	Your destination data table option.	<ul style="list-style-type: none"> • Use existing • Auto create table 	Yes	<ul style="list-style-type: none"> • typeProperties (under <code>typeProperties -> sink</code>) - schema - table • tableOption: - autoCreate <p>typeProperties (under <code>typeProperties -> sink</code>)</p> <ul style="list-style-type: none"> - schema - table
Copy method	The method used to copy data.	<ul style="list-style-type: none"> • Copy command • PolyBase • Bulk insert • Upsert 	No	/
When selecting Copy command	Use COPY statement to load data from Azure storage into Azure Synapse Analytics or SQL Pool.	/	No. Apply when using COPY.	allowCopyCommand: true copyCommandSettings
Default values	Specify the default values for each target column in Azure Synapse Analytics.	< default values >	No	defaultValues: <ul style="list-style-type: none"> - columnName - defaultValue

Name	Description	Value	Required	JSON script property
	The default values in the property overwrite the DEFAULT constraint set in the data warehouse, and identity column cannot have a default value.			
Additional options	Additional options that will be passed to an Azure Synapse Analytics COPY statement directly in "With" clause in COPY statement. Quote the value as needed to align with the COPY statement requirements.	< additional options >	No	additionalOptions: - <property name> : <value>
When selecting PolyBase	PolyBase is a high-throughput mechanism. Use it to load large amounts of data into Azure Synapse Analytics or SQL Pool.	/	No. Apply when using PolyBase.	allowPolyBase: true polyBaseSettings
Reject type	The type of the reject value.	• Value • Percentage	No	rejectType: - value - percentage
Reject value	The number or percentage of rows that can be rejected before the query fails.	0 (default), 1, 2, etc.	No	rejectValue
Reject sample value	Determines the number of rows to retrieve before PolyBase recalculates the percentage of rejected rows.	1, 2, etc.	Yes when you specify Percentage as your reject type	rejectSampleValue
Use type default	Specify how to handle missing	selected (default) or	No	useTypeDefault: true (default) or false

Name	Description	Value	Required	JSON script property
	values in delimited text files when PolyBase retrieves data from the text file. Learn more about this property from the Arguments section in CREATE EXTERNAL FILE FORMAT (Transact-SQL).	unselected.		
When selecting Bulk insert	Insert data to destination in bulk.	/	No	writeBehavior: Insert
Bulk insert table lock	Use this to improve copy performance during bulk insert operation on table with no index from multiple clients. Learn more from BULK INSERT (Transact-SQL).	selected or unselected (default)	No	sqlWriterUseTableLock: true or false (default)
When selecting Upsert	Specify the group of the settings for write behavior when you want to upsert data to your destination.	/	No	writeBehavior: Upsert
Key columns	Indicates which column is used to determine if a row from the source matches a row from the destination.	< column name>	No	upsertSettings: - keys: < column name > - interimSchemaName
Bulk insert table lock	Use this to improve copy performance during bulk insert operation on table with no index from multiple clients. Learn more from BULK INSERT (Transact-SQL).	selected or unselected (default)	No	sqlWriterUseTableLock: true or false (default)

Name	Description	Value	Required	JSON script property
Pre-copy script	A script for Copy Activity to execute before writing data into a destination table in each run. You can use this property to clean up the pre-loaded data.	< pre-copy script > (string)	No	preCopyScript
Write batch timeout	The wait time for the batch insert operation to finish before it times out. The allowed value is timespan. The default value is "00:30:00" (30 minutes).	timespan	No	writeBatchTimeout
Write batch size	The number of rows to insert into the SQL table per batch. By default, the service dynamically determines the appropriate batch size based on the row size.	< number of rows > (integer)	No	writeBatchSize
Max concurrent connections	The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.	< upper limit of concurrent connections > (integer)	No	maxConcurrentConnections
Disable performance metrics analytics	This setting is used to collect metrics, such as DTU, DWU, RU, and so on, for copy performance optimization and recommendations. If you're concerned	select or unselect (default)	No	disableMetricsCollection: true or false (default)

Name	Description	Value	Required	JSON script property
	with this behavior, select this checkbox.			

Next steps

Azure Synapse Analytics connector overview

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Azure Table Storage connector overview

Article • 11/15/2023

This Azure Table Storage connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to Azure Table Storage in Dataflow Gen2, go to Set up your connection in Dataflow Gen2.

Support in data pipelines

The Azure Table Storage connector supports the following capabilities in data pipelines:

The Azure Table Storage connector supports the following capabilities in Data pipeline:

Supported capabilities	Gateway	Authentication
Copy activity (source/-)	None	Account key
Lookup activity	None	Account key

To learn more about the copy activity configuration for Azure Table Storage in data pipelines, go to Configure in a data pipeline copy activity.

Feedback

Was this page helpful?



Provide product feedback [↗](#) | Ask the community [↗](#)

Set up your Azure Table Storage connection

Article • 11/15/2023

This article outlines the steps to create an Azure Table Storage connection.

Supported authentication types

The Azure Table Storage connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Account key	√	√

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Azure Table Storage. The following links provide the specific Power Query connector information you need to connect to Azure Table Storage in Dataflow Gen2:

- To get started using the Azure Table Storage connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric \(Preview\)](#).
- To learn where to get a copy of your account key, go to [Copy your account key for Azure Table Storage](#).
- To connect to the Azure Table Storage connector from Power Query, go to [Connect to Azure Table Storage from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback ↗](#) | [Ask the community ↗](#)

Configure Azure Table Storage in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in data pipeline to copy data from and to Azure Table Storage.

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Destination
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

Go to **Source** tab to configure your copy activity source. See the following content for the detailed configuration.



The following three properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select an Azure Table Storage connection from the connection list. If no connection exists, then create a new Azure Table Storage connection by selecting **New**.

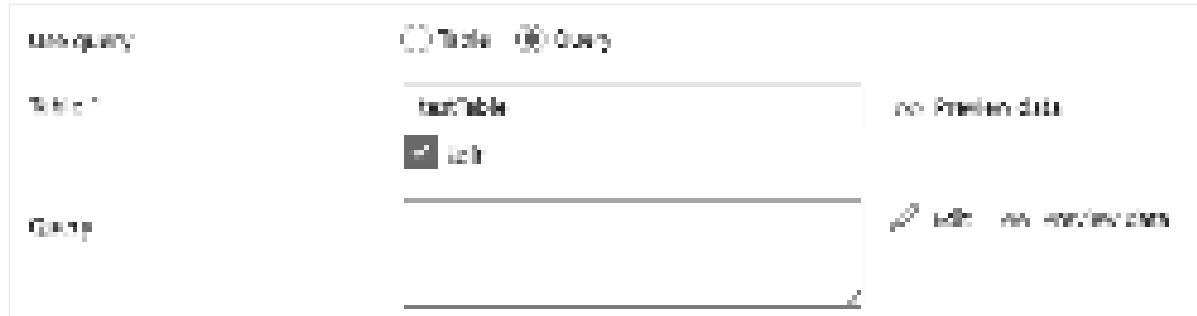
- **Use query:** Specify way to read data. Select **Table** to read data from the specified table or select **Query** to read data using queries.

If you select **Table**:



- **Table:** Specify the name of the table in the Azure Table Storage database instance. Select the table from the drop-down list or enter the name manually by selecting **Edit**.

If you select **Query**:



- **Table:** Specify the name of the table in the Azure Table Storage database instance. Select the table from the drop-down list or enter the name manually by selecting **Edit**.
- **Query:** Specify the custom Table storage query to read data. The source query is a direct map from the `$filter` query option supported by Azure Table Storage, learn more about the syntax from this article.

! Note

Azure Table query operation times out in 30 seconds as enforced by Azure Table service. Learn how to optimize the query from [Design for querying](#) article.

Under **Advanced**, you can specify the following fields:

- **Ignore table not found:** Specifies whether to allow the exception of the table to not exist. It is unselected by default.
- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.

Destination

Go to **Destination** tab to configure your copy activity destination. See the following content for the detailed configuration.



The following three properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select an Azure Table Storage connection from the connection list. If no connection exists, then create a new Azure Table Storage connection by selecting **New**.
- **Table:** Specify the name of the table in the Azure Table Storage database instance. Select the table from the drop-down list or enter the name manually by selecting **Edit**.

Under **Advanced**, you can specify the following fields:

- **Insert type:** Select the mode to insert data into Azure Table. The modes are **Merge** and **Replace**. This property controls whether existing rows in the output table with matching partition and row keys have their values replaced or merged. This setting applies at the row level not the table level. Neither option deletes rows in the output table that don't exist in the input. To learn about how the merge and replace settings work, see [Insert or merge entity](#) and [Insert or replace entity](#).
- **Partition key value selection:** Select from **Specify partition value** or **Use destination column**. Partition key value can be a fixed value or it can take value from a destination column.

If you select **Specify partition value**:

- **Default partition value:** Specify the default partition key value that can be used by the destination.

If you select **Use destination column**:

- **Partition key column:** Select the name of the column whose column values are used as the partition key. If not specified, "AzureTableDefaultPartitionKeyValue" is used as the partition key.

- **Row key value selection:** Select from **Unique identifier** or **Use destination column**. Row key value can be an auto generated unique identifier or it can take value from a destination column.

If you select **Use destination column**:

- **Row key column:** Select the name of the column whose column values are used as the row key. If not specified, use a GUID for each row.
- **Write batch size:** Inserts data into Azure Table when the specified write batch size is hit. Allowed values are integer (number of rows). The default value is 10,000.
- **Write batch timeout:** Inserts data into Azure Table when the specified write batch timeout is hit. The allowed value is timespan.
- **Max concurrent connections:** The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections

Mapping

For **Mapping** tab configuration, see [Configure your mappings under mapping tab](#).

Settings

For **Settings** tab configuration, go to [Configure your other settings under settings tab](#).

Table summary

The following table contains more information about the copy activity in Azure Table Storage.

Source information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	< your Azure Table Storage connection >	Yes	connection
Use query	The way to read data. Apply Table to read data from the specified table or apply Query to read data using queries.	<ul style="list-style-type: none"> • Table • Query 	Yes	/

Name	Description	Value	Required	JSON script property
Table	The name of the table in the Azure Table Storage database instance.	< your table name >	Yes	tableName
Query	Specify the custom Table storage query to read data. The source query is a direct map from the <code>\$filter</code> query option supported by Azure Table Storage, learn more about the syntax from this article.	< your query >	No	azureTableSourceQuery
Ignore table not found	Indicates whether to allow the exception of the table to not exist.	selected or unselected (default)	No	azureTableSourceIgnoreTableNotFound: true or false (default)
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.	<ul style="list-style-type: none"> • Name • Value 	No	additionalColumns: <ul style="list-style-type: none"> • name • value

Destination information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the destination data store.	< your Azure Table Storage connection >	Yes	connection
Table	The name of the table in the Azure Table Storage database instance.	< your table name >	Yes	tableName
Insert type	The mode to insert data into Azure Table. This property controls whether existing rows in the output table with matching partition and row keys have their values replaced or merged.	<ul style="list-style-type: none"> • Merge • Replace 	No	azureTableInsertType: <ul style="list-style-type: none"> • merge • replace
Partition key value	Partition key value can be a fixed value or it can take value from a	<ul style="list-style-type: none"> • Specify partition 	No	/

Name	Description	Value	Required	JSON script property
selection	destination column.	value • Use destination column		
Default partition value	The default partition key value that can be used by the destination	< your default partition value >	No	azureTableDefaultPartitionKeyValue
Partition key column	The name of the column whose values are used as partition keys. If not specified, "AzureTableDefaultPartitionKeyValue" is used as the partition key.	< your partition key column >	No	azureTablePartitionKeyName
Row key value selection	Row key value can be an auto generated unique identifier or it can take value from a destination column.	• Unique identifier • Use destination column	No	/
Row key column	The name of the column whose column values are used as the row key. If not specified, use a GUID for each row.	< your row key column >	No	azureTableRowKeyName
Write batch size	Inserts data into Azure Table when the write batch size is hit.	integer (default is 10,000)	No	writeBatchSize
Write batch timeout	Inserts data into Azure Table when the write batch timeout is hit	timespan	No	writeBatchTimeout
Max concurrent connections	The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.	< your max concurrent connections >	No	maxConcurrentConnections

Next steps

- Azure Table Storage connector overview

Feedback

Was this page helpful?



CloudBluePSA (Beta) connector overview

Article • 11/15/2023

The CloudBluePSA connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to CloudBluePSA in Dataflow Gen2, go to Set up your CloudBluePSA connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support CloudBluePSA in data pipelines.

Feedback

Was this page helpful?



[Provide product feedback ↗](#) | [Ask the community ↗](#)

Set up your CloudBluePSA (Beta) connection

Article • 12/06/2023

This article outlines the steps to create a CloudBluePSA connection.

Supported authentication types

The CloudBluePSA connector supports the following authentication types for copy and Dataflow Gen2 respectively.

[+] Expand table

Authentication type	Copy	Dataflow Gen2
Account key	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to CloudBluePSA. The following links provide the specific Power Query connector information you need to connect to CloudBluePSA in Dataflow Gen2:

- To get started using the CloudBluePSA connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- Be sure to install or set up any CloudBluePSA prerequisites before connecting to the CloudBluePSA connector.
- To connect to the CloudBluePSA connector from Power Query, go to [Connect to CloudBluePSA from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support CloudBluePSA in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Common Data Service (Legacy) connector overview

Article • 11/15/2023

The Common Data Service (Legacy) connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to Common Data Service (Legacy) in Dataflow Gen2, go to Set up your Common Data Service (Legacy) connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support Common Data Service (Legacy) in data pipelines.

Feedback

Was this page helpful?



[Provide product feedback](#) | [Ask the community](#)

Set up your Common Data Service (Legacy) connection

Article • 11/15/2023

This article outlines the steps to create a Common Data Service (Legacy) connection.

Supported authentication types

The Common Data Service (Legacy) connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Organizational account	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Common Data Service (Legacy). The following links provide the specific Power Query connector information you need to connect to Common Data Service (Legacy) in Dataflow Gen2:

- To get started using the Common Data Service (Legacy) connector in Dataflow Gen2, go to Get data from Data Factory in Microsoft Fabric (Preview).
- Be sure to install or set up any Common Data Service (Legacy) prerequisites before connecting to the Common Data Service (Legacy) connector.
- To connect to the Common Data Service (Legacy) connector from Power Query, go to Connect to Common Data Service (Legacy) from Power Query Online.

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support Common Data Service (Legacy) in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Databricks connector overview

Article • 11/15/2023

The Databricks connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to Databricks in Dataflow Gen2, go to Set up your Databricks connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support Databricks in data pipelines.

Feedback

Was this page helpful?



Provide product feedback [↗](#) | Ask the community [↗](#)

Set up your Databricks connection

Article • 12/06/2023

This article outlines the steps to create a Databricks connection.

Supported authentication types

The Databricks connector supports the following authentication types for copy and Dataflow Gen2 respectively.

 Expand table

Authentication type	Copy	Dataflow Gen2
Username/Password	n/a	✓
Personal Access Token	n/a	✓
OAuth (OIDC)	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Databricks data. The following links provide the specific Power Query connector information you need to connect to Databricks data in Dataflow Gen2:

- To get started using the Databricks connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- Be sure to install or set up any Databricks prerequisites before connecting to the Databricks connector.
- To connect to the Databricks connector from Power Query, go to [Connect to Databricks data from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support the Databricks connector in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Dataflows (Power Platform) connector overview

Article • 11/15/2023

The Dataflows connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to Dataflows data in Dataflow Gen2, go to Set up your Dataflows (Power Platform) connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support Dataflows (Power Platform) in data pipelines.

Feedback

Was this page helpful?



[Provide product feedback ↗](#) | [Ask the community ↗](#)

Set up your Dataflows (Power Platform) connection

Article • 11/15/2023

This article outlines the steps to create a Dataflows connection.

Supported authentication types

The Dataflows connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Organizational account	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Dataflows data. The following links provide the specific Power Query connector information you need to connect to Dataflows data in Dataflow Gen2:

- To get started using the Dataflows connector in Dataflow Gen2, go to Get data from Data Factory in Microsoft Fabric (Preview).
- Be sure to install or set up any Dataflows prerequisites before connecting to the Dataflows connector.
- To connect to the Dataflows connector from Power Query, go to Get data from Dataflows in Power Query Online.

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support Dataflows data in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Dataverse connector overview

Article • 11/15/2023

This Dataverse connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to Dataverse in Dataflow Gen2, go to Set up your connection in Dataflow Gen2.

Support in data pipelines

The Dataverse connector supports the following capabilities in data pipelines:

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	Service principal
Lookup activity	None	Service principal

To learn about the copy activity configuration for Dataverse in data pipelines, go to Configure Dataverse in a copy activity.

Feedback

Was this page helpful?



Yes



No

Provide product feedback | Ask the community

Set up your Dataverse connection

Article • 11/15/2023

This article outlines the steps to create a Dataverse connection.

Supported authentication types

The Dataverse connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Organizational account	n/a	✓
Service principal	✓	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Dataverse. The following links provide the specific Power Query connector information you need to connect to Dataverse in Dataflow Gen2:

- To get started using the Dataverse connector in Dataflow Gen2, go to Get data from Data Factory in Microsoft Fabric (Preview).
- Be sure to install or set up any Dataverse prerequisites before connecting to the Dataverse connector.
- To connect to the Dataverse connector from Power Query, go to Connect to Dataverse from Power Query Online.

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Configure Dataverse in a copy activity

Article • 12/21/2023

This article outlines how to use the copy activity in a data pipeline to copy data from and to Dataverse.

Prerequisites

To use this connector with Microsoft Entra service principal authentication, you must set up server-to-server (S2S) authentication in Dataverse. First register the application user (Service Principal) in Microsoft Entra. For more information, see [Create a Microsoft Entra application and service principal that can access resources](#).

During application registration you will need to create that user in Dataverse and grant permissions. Those permissions can either be granted directly or indirectly by adding the application user to a team which has been granted permissions in Dataverse. For more information on how to set up an application user to authenticate with Dataverse, see [Use single-tenant server-to-server authentication](#).

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

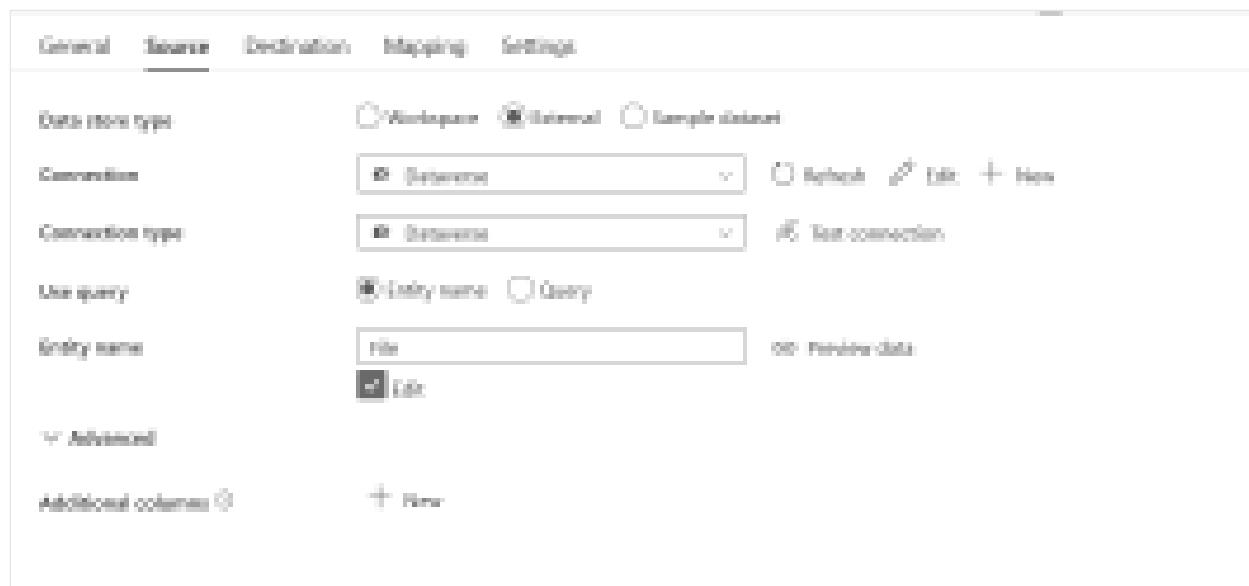
- General
- Source
- Destination
- Mapping
- Settings

General

For General tab configuration, go to General settings.

Source

Go to Source tab to configure your copy activity source. See the following content for the detailed configuration.



The following properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select an existing Dataverse connection from the connection list. If no connection exists, then create a new Dataverse connection by selecting **New**.
- **Connection type:** Select **Dataverse**.
- **Use query:** Specify the way used to read data. You can choose **Entity name** to read data using entity name or **Query** to use query to read data.
 - **Entity name:** Select your entity name from the drop-down list, or select **Edit** to enter it manually. It is the logical name of the entity to retrieve.
 - **Query:** Using FetchXML to read data from Dataverse. FetchXML is a proprietary query language that is used in Dynamics online and on-premises. See the following example. To learn more, see [Build queries with FetchXML](#).

Sample FetchXML query:

```
XML

<fetch>
    <entity name="account">
        <attribute name="accountid" />
        <attribute name="name" />
        <attribute name="marketingonly" />
        <attribute name="modifiedon" />
        <order attribute="modifiedon" descending="false" />
        <filter type="and">
            <condition attribute ="modifiedon" operator="between">
                <value>2017-03-10 18:40:00z</value>
                <value>2017-03-12 20:40:00z</value>
            </condition>
        </filter>
```

```
</entity>  
</fetch>
```

Under **Advanced**, you can specify the following fields:

- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter. For more information, go to [Add additional columns during copy](#).

Destination

Go to **Destination** tab to configure your copy activity destination. See the following content for the detailed configuration.

The screenshot shows the 'Destination' tab selected in a navigation bar. Below it, several configuration fields are displayed:

- Data store type:** A dropdown menu showing 'External' (selected) and 'Dataverse'.
- Connection:** A dropdown menu showing 'Reference' (selected), with options to 'New', 'Edit', and 'Help'.
- Connection type:** A dropdown menu showing 'Dataverse'.
- Entity name:** A dropdown menu showing 'TestFile' (selected), with an 'Edit' button below it.
- Advanced:** A section containing:
 - Write behavior:** Set to 'Append'.
 - Alternate key name:** An empty input field with an 'Edit' button.
 - Ignore null values:** A checked checkbox.
 - Write batch size:** A value of '10'.
 - Max concurrent connections:** An empty input field.

The following properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select an existing Dataverse connection from the connection list. If no connection exists, then create a new Dataverse connection by selecting **New**.
- **Connection type:** Select **Dataverse**.
- **Entity name:** Specify the name of the entity to write data. Select your entity name from the drop-down list, or select **Edit** to enter it manually. This is the logical name of the entity to retrieve.

Under **Advanced**, you can specify the following fields:

- **Write behavior:** The write behavior of the operation. This property is required, and you must select **Upsert**. If you use **Add dynamic content**, specify the value to **Upsert**.
- **Alternate key name:** Specify the alternate key name defined on your entity to upsert records.
- **Ignore null values:** Indicates whether to ignore null values from input data during write operation. It is selected by default.
 - When it is selected: Leave the data in the destination object unchanged when doing upsert/update operation, and insert defined default value when doing insert operation.
 - When it is unselected: Update the data in the destination object to NULL when doing upsert/update operation, and insert NULL value when doing insert operation.
- **Write batch size:** Specify the row count of data written to Dataverse in each batch.
- **Max concurrent connections:** The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.

Mapping

For **Mapping** tab configuration, go to Configure your mappings under mapping tab. If you choose Binary as your file format, mapping will not be supported.

Settings

For the **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

See the following table for the summary and more information for the Dataverse copy activity.

Source information

[] Expand table

Name	Description	Value	Required	JSON script property
Data store	Your data store	External	Yes	/

Name	Description	Value	Required	JSON script property
type	type.			
Connection	Your connection to the source Dataverse.	< your connection >	Yes	connection
Connection type	Your connection type.	Dataverse	Yes	type (under <code>typeProperties -> source -> datasetSettings</code>): CommonDataServiceForAppsEntity
Use query	The way to read data from Dataverse	* Tables * Query	Yes	/
Entity name	The logical name of the entity to retrieve.	< your entity name >	Yes	entityName
Query	Use FetchXML to read data from Dataverse. FetchXML is a proprietary query language that is used in Dynamics online and on-premises. To learn more, see Build queries with FetchXML.	< your query >	Yes	query
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter. For more information, go to Add additional columns during copy.	* Name * Value	No	additionalColumns: * name * value

Destination information

[Expand table](#)

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the destination Dataverse.	< your connection >	Yes	connection
Connection type	Your connection type.	Dataverse	Yes	type (under <code>typeProperties -> sink -> datasetSettings</code>): <code>CommonDataServiceForAppsEntity</code>
Entity name	The logical name of the entity to retrieve.	< your entity >	Yes	entityName
Write behavior	The write behavior of the operation. The value must be Upsert .	Upsert	Yes	writeBehavior: upsert
Alternate key name	The alternate key name defined on your entity to upsert records.	< alternate key name >	No	alternateKeyName
Ignore null values	Indicates whether to ignore null values from input data during write operation. - Selected (true): Leave the data in the destination object unchanged when doing upsert/update operation, and insert defined default value when doing insert operation. - Unselected	selected or unselected (default)	No	ignoreNullValues: true or false (default)

Name	Description	Value	Required	JSON script property
	(false): Update the data in the destination object to NULL when doing upsert/update operation, and insert NULL value when doing insert operation.			
Write batch size	The row count of data written to Dataverse in each batch.	< your write batch size > The default value is 10	No	writeBatchSize
Max concurrent connections	The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.	< max concurrent connections >	No	maxConcurrentConnections

Next steps

Dataverse connector overview

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Data Warehouse connector overview

Article • 11/15/2023

The Data Warehouse connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to a Data Warehouse in Dataflow Gen2, go to Set up your connection in Dataflow Gen2.

Support in data pipeline

The Data Warehouse connector supports the following capabilities in data pipelines:

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	User Auth
Lookup activity	None	User Auth
GetMetadata activity	None	User Auth
Script activity	None	User Auth
Stored Procedure Activity	None	User Auth

To learn more about the copy activity configuration for Data Warehouse in data pipelines, go to Configure in a data pipeline copy activity.

Feedback

Was this page helpful?



Yes



No

Provide product feedback | Ask the community

Set up your Data Warehouse connection

Article • 12/06/2023

This article outlines the steps to create a Data Warehouse connection.

Supported authentication types

The Data Warehouse connector supports the following authentication types for copy and Dataflow Gen2 respectively.

 Expand table

Authentication type	Copy	Dataflow Gen2
Organizational account	✓	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to a Data Warehouse. The following links provide the specific Power Query connector information you need to connect to a Data Warehouse in Dataflow Gen2:

- To get started using the Warehouse connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- Be sure to install or set up any Warehouse prerequisites before connecting to the Warehouse connector.
- To connect to the Warehouse connector from Power Query, go to [Connect to a Warehouse from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Feedback

Was this page helpful?



Yes



No

[Provide product feedback](#) | [Ask the community](#)

Configure Data Warehouse in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in data pipeline to copy data from and to a Data Warehouse.

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Destination
- Mapping
- Settings

General

For the **General** tab configuration, go to General.

Source

The following properties are supported for Data Warehouse as **Source** in a copy activity.

General Source Destination Mapping Set up

Data store type	<input checked="" type="radio"/> Workspace <input type="radio"/> Column <input type="radio"/> Complex value
Workspace data store type	<input checked="" type="radio"/> Data Warehouse <input type="radio"/> Database <input type="radio"/> Table
Data warehouse	<input checked="" type="radio"/> DataWarehouse <input type="radio"/> Database <input type="radio"/> Table
Use query	<input checked="" type="radio"/> Table <input type="radio"/> Query <input type="radio"/> Stored procedure
Table	<input type="text" value=""/>
	<input type="checkbox"/> Edit <input type="checkbox"/> Preview data
SQL statement	
Query timeout (minutes)	<input type="text" value="10"/>
Isolation level	<input type="text" value="None"/>
Partitioning	<input checked="" type="radio"/> None <input type="radio"/> Query range
Important Please provide data to validate the partition settings are correct before you go to run or publish the pipeline.	
Add transformation	<input type="button" value="+ New"/>

The following properties are **required**:

- **Data store type:** Select **Workspace**.
- **Workspace data store type:** Select **Data Warehouse** from the data store type list.
- **Data Warehouse:** Select an existing **Data Warehouse** from the workspace.
- **Use query:** Select **Table**, **Query**, or **Stored procedure**.
 - If you select **Table**, choose an existing table from the table list, or specify a table name manually by selecting the **Edit** box.

Sort by	<input type="radio"/> Name <input type="radio"/> Owner <input type="radio"/> Last modified
Table	<input type="text"/>
	<input type="checkbox"/> Use

- If you select **Query**, use the custom SQL query editor to write a SQL query that retrieves the source data.

User query	<input checked="" type="radio"/> Table <input type="radio"/> Query <input type="radio"/> Stored procedure
Query	<input type="text"/>
	<input type="checkbox"/> Edit <input type="checkbox"/> Preview data

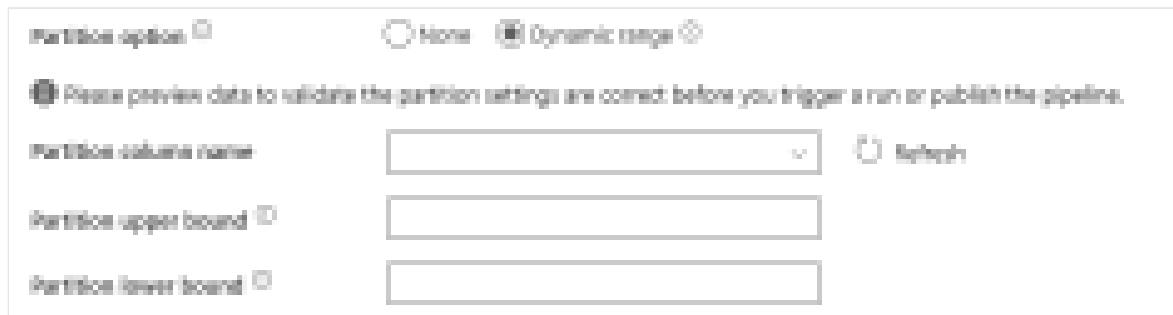
- If you select **Stored procedure**, choose an existing stored procedure from the drop-down list, or specify a stored procedure name as the source by selecting the **Edit** box.



Under **Advanced**, you can specify the following fields:

- **Query timeout (minutes)**: Timeout for query command execution, with a default of 120 minutes. If this property is set, the allowed values are in the format of a timespan, such as "02:00:00" (120 minutes).
- **Isolation level**: Specify the transaction locking behavior for the SQL source.
- **Partition option**: Specify the data partitioning options used to load data from Data Warehouse. You can select **None** or **Dynamic range**.

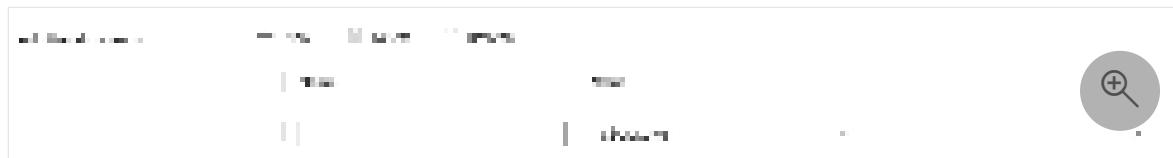
If you select **Dynamic range**, the range partition parameter(`? AdfDynamicRangePartitionCondition`) is needed when using query with parallel enabled. Sample query: `SELECT * FROM <TableName> WHERE ? AdfDynamicRangePartitionCondition`.



- **Partition column name**: Specify the name of the source column in **integer** or **date/datetime type** (`int`, `smallint`, `bigint`, `date`, `smalldatetime`, `datetime`, `datetime2`, or `datetimeoffset`) that is used by range partitioning for parallel copy. If not specified, the index or the primary key of the table is detected automatically and used as the partition column.
- **Partition upper bound**: The maximum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering the rows in table. All rows in the table or query result are partitioned and copied.
- **Partition lower bound**: The minimum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering

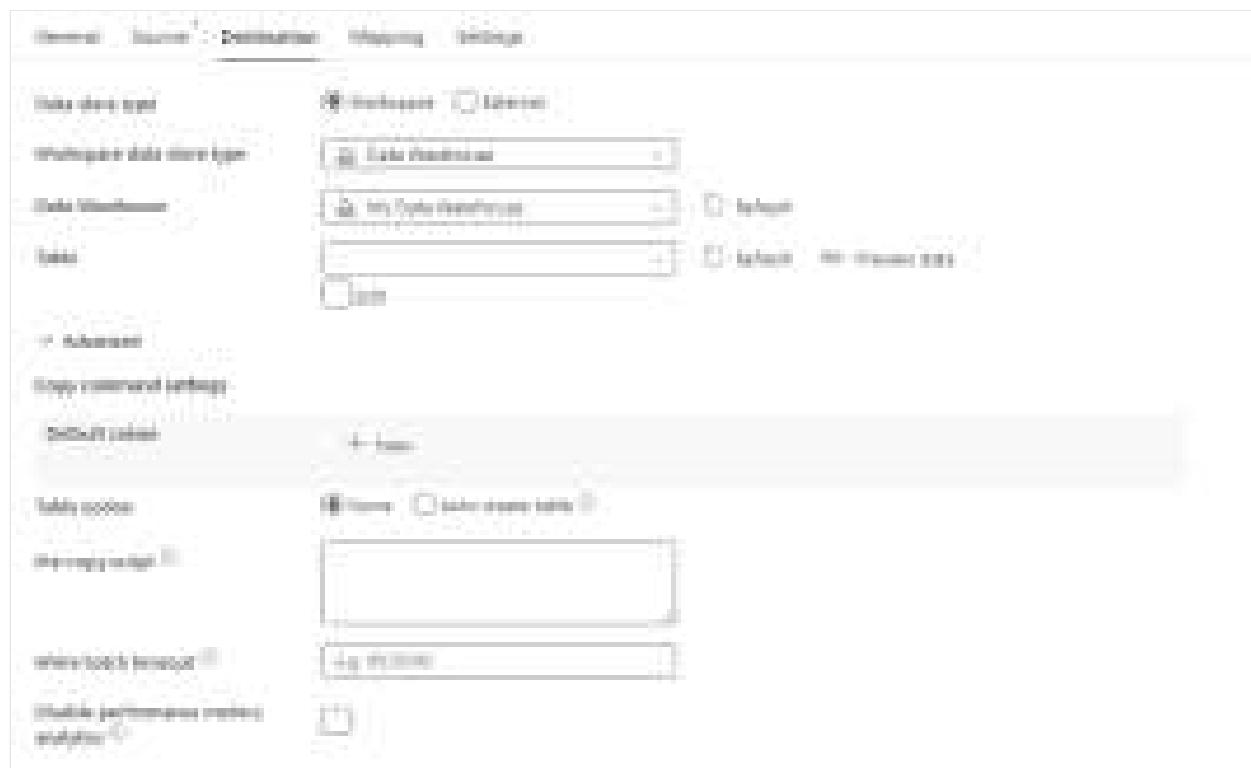
the rows in table. All rows in the table or query result are partitioned and copied.

- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.



Destination

The following properties are supported for Data Warehouse as **Destination** in a copy activity.

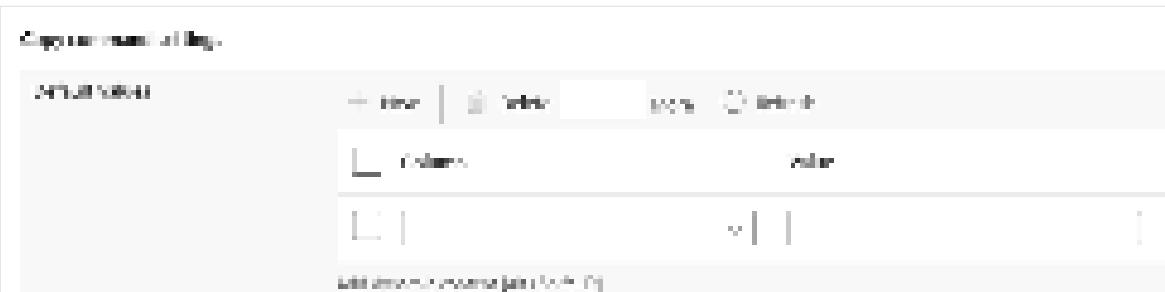


The following properties are **required**:

- **Data store type:** Select **Workspace**.
- **Workspace data store type:** Select **Data Warehouse** from the data store type list.
- **Data Warehouse:** Select an existing **Data Warehouse** from the workspace.
- **Table:** Choose an existing table from the table list or specify a table name as destination.

Under **Advanced**, you can specify the following fields:

- **Copy command settings:** Specify copy command properties.



- **Table options:** Specify whether to automatically create the destination table if none exists based on the source schema. You can select **None** or **Auto create table**.
- **Pre-copy script:** Specify a SQL query to run before writing data into Data Warehouse in each run. Use this property to clean up the preloaded data.
- **Write batch timeout:** The wait time for the batch insert operation to finish before it times out. The allowed values are in the format of a timespan. The default value is "00:30:00" (30 minutes).
- **Disable performance metrics analytics:** The service collects metrics for copy performance optimization and recommendations. If you're concerned with this behavior, turn off this feature.

Direct copy by using COPY command

Data Warehouse COPY command directly supports **Azure Blob Storage** and **Azure Data Lake Storage Gen2** as source data stores. If your source data meets the criteria described in this section, use COPY command to copy directly from the source data store to Data Warehouse.

1. The source data and format contain the following types and authentication methods:

Supported source data store type	Supported format	Supported source authentication type
Azure Blob Storage	Delimited text Parquet	Anonymous authentication Account key authentication Shared access signature authentication
Azure Data Lake Storage Gen2	Delimited text Parquet	Account key authentication Shared access signature authentication

2. The following Format settings can be set:

- a. For Parquet: Compression type can be **None**, **snappy**, or **gzip**.
 - b. For DelimitedText:
 - i. **Row delimiter**: When copying delimited text to Data Warehouse via direct COPY command, specify the row delimiter explicitly (\r; \n; or \r\n). Only when the row delimiter of the source file is \r\n, the default value (\r, \n, or \r\n) works. Otherwise, enable staging for your scenario.
 - ii. **Null value** is left as default or set to empty string ("").
 - iii. **Encoding** is left as default or set to **UTF-8** or **UTF-16**.
 - iv. **Skip line count** is left as default or set to 0.
 - v. **Compression type** can be **None** or **gzip**.
3. If your source is a folder, you must select **Recursively** checkbox.
 4. Start time (UTC) and End time (UTC) in Filter by last modified, Prefix, Enable partition discovery, and Additional columns aren't specified.

To learn how to ingest data into your Data Warehouse using the COPY command, see this article.

If your source data store and format isn't originally supported by a COPY command, use the Staged copy by using the COPY command feature instead. It automatically converts the data into a COPY command compatible format, then calls a COPY command to load data into Data Warehouse.

Mapping

For the **Mapping** tab configuration, go to Mapping.

Settings

For the **Settings** tab configuration, go to Settings.

Table summary

The following tables contain more information about a copy activity in Data Warehouse.

Source information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	Workspace	Yes	/

Name	Description	Value	Required	JSON script property
Workspace data store type	The section to select your workspace data store type.	Data Warehouse	Yes	type
Data Warehouse	The Data Warehouse that you want to use.	<your data warehouse>	Yes	endpoint artifactId
Use query	The way to read data from Data Warehouse.	<ul style="list-style-type: none"> • Tables • Query • Stored procedure 	No	<p>(under <code>typeProperties</code> -> <code>source</code>)</p> <ul style="list-style-type: none"> • <code>typeProperties:</code> schema table • <code>sqlReaderQuery</code> • <code>sqlReaderStoredProcName</code>
Query timeout (minutes)	Timeout for query command execution, with a default of 120 minutes. If this property is set, the allowed values are in the format of a timespan, such as "02:00:00" (120 minutes).	timespan	No	queryTimeout
Isolation level	The transaction locking behavior for source.	<ul style="list-style-type: none"> • None • Snapshot 	No	isolationLevel
Partition option	The data partitioning options used to load data from Data Warehouse.	<ul style="list-style-type: none"> • None • Dynamic range 	No	partitionOption
Partition column name	<p>The name of the source column in integer or date/datetime type</p> <p>(<code>int</code>, <code>smallint</code>, <code>bigint</code>, <code>date</code>, <code>smalldatetime</code>, <code>datetime</code>, <code>datetime2</code>, or <code>datetimeoffset</code>)</p> <p>that is used by range partitioning for parallel copy. If not</p>	<partition column name>	No	partitionColumnName

Name	Description	Value	Required	JSON script property
	specified, the index or the primary key of the table is detected automatically and used as the partition column.			
Partition upper bound	The maximum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering the rows in table. All rows in the table or query result are partitioned and copied.	<partition upper bound>	No	partitionUpperBound
Partition lower bound	The minimum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering the rows in table. All rows in the table or query result are partitioned and copied.	<partition lower bound>	No	partitionLowerBound
Additional columns	Add additional data columns to store source files' relative path or static value.	<ul style="list-style-type: none"> • Name • Value 	No	additionalColumns: <ul style="list-style-type: none"> • name • value

Destination information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	Workspace	Yes	/
Workspace data store	The section to select your workspace data	Data Warehouse	Yes	type

Name	Description	Value	Required	JSON script property
type	store type.			
Data Warehouse	The Data Warehouse that you want to use.	<your data warehouse>	Yes	endpoint artifactId
Table	The destination table to write data.	<name of your destination table>	Yes	schema table
Copy command settings	The copy command property settings. Contains the default value settings.	Default value: • Column • Value	No	copyCommandSettings: defaultValues: • columnName • defaultValue
Table option	Whether to automatically create the destination table if none exists based on the source schema.	• None • Auto create table	No	tableOption: • autoCreate
Pre-copy script	A SQL query to run before writing data into Data Warehouse in each run. Use this property to clean up the preloaded data.	<pre-copy script>	No	preCopyScript
Write batch timeout	The wait time for the batch insert operation to finish before it times out. The allowed values are in the format of a timespan. The default value is "00:30:00" (30 minutes).	timespan	No	writeBatchTimeout
Disable performance metrics analytics	The service collects metrics for copy performance optimization and recommendations, which introduce additional master DB access.	select or unselect	No	disableMetricsCollection: true or false

Next steps

- Data Warehouse connector overview

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Dynamics CRM connector overview

Article • 12/08/2023

This Dynamics CRM connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Data pipeline

The Dynamics CRM connector supports the following capabilities in Data pipeline:

 Expand table

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	Service principal
Lookup activity	None	Service principal

To learn more about the copy activity configuration for Dynamics CRM in Data pipeline, go to Configure in a data pipeline copy activity.

Feedback

Was this page helpful?



Yes



No

Provide product feedback  | Ask the community 

How to configure Dynamics CRM in a copy activity

Article • 12/08/2023

This article outlines how to use the copy activity in a data pipeline to copy data from and to Dynamics CRM.

Prerequisites

To use this connector with Azure AD service-principal authentication, you must set up server-to-server (S2S) authentication in Dynamics CRM. First register the application user (Service Principal) in Microsoft Entra. For more information, see [Create a Microsoft Entra application and service principal that can access resources](#).

During application registration you will need to create that user in Dynamics CRM and grant permissions. Those permissions can either be granted directly or indirectly by adding the application user to a team which has been granted permissions in Dynamics CRM. For more information on how to set up an application user to authenticate with Dynamics CRM, see [Use single-tenant server-to-server authentication](#).

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

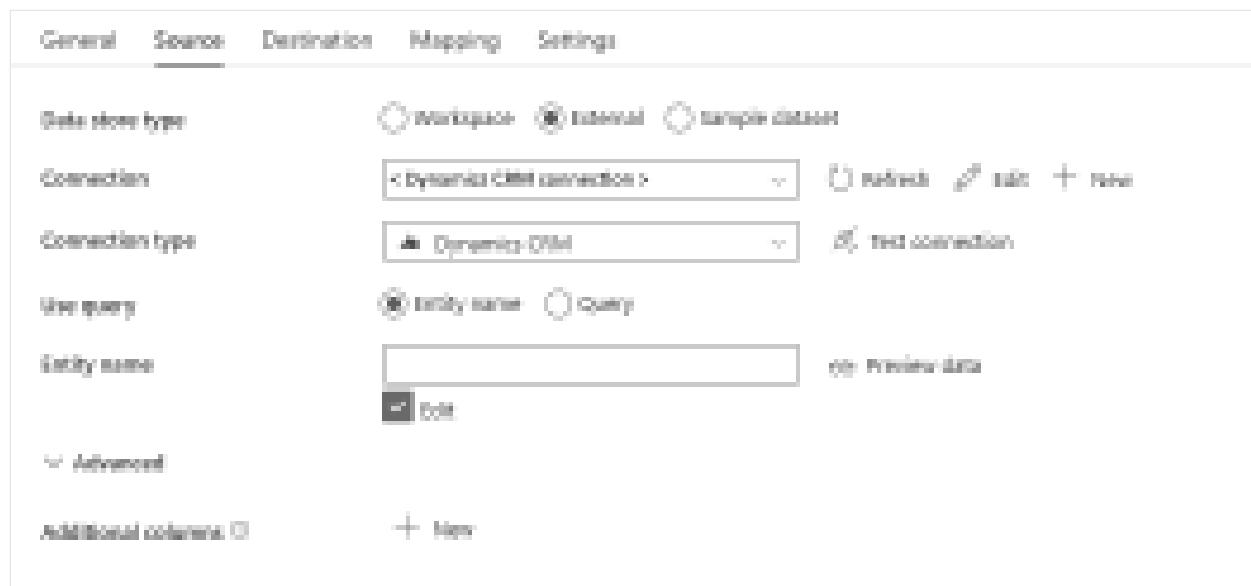
- General
- Source
- Destination
- Mapping
- Settings

General

For **General** tab configuration, go to General settings.

Source

Go to **Source** tab to configure your copy activity source. See the following content for the detailed configuration.



The following properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select an existing Dynamics CRM connection from the connection list. If no connection exists, then create a new Dynamics CRM connection by selecting **New**.
- **Connection type:** Select **Dynamics CRM**.
- **Use query:** Specify the way used to read data. You can choose **Entity name** to read data using entity name or **Query** to use query to read data.
 - **Entity name:** Select your entity name from the drop-down list, or select **Edit** to enter it manually. It is the logical name of the entity to retrieve.
 - **Query:** Using FetchXML to read data from Dynamics CRM. FetchXML is a proprietary query language that is used in Dynamics online and on-premises. See the following example. To learn more, see Build queries with FetchXML.

Sample FetchXML query:

```
XML

<fetch>
    <entity name="account">
        <attribute name="accountid" />
        <attribute name="name" />
        <attribute name="marketingonly" />
        <attribute name="modifiedon" />
        <order attribute="modifiedon" descending="false" />
        <filter type="and">
            <condition attribute ="modifiedon" operator="between">
                <value>2017-03-10 18:40:00z</value>
                <value>2017-03-12 20:40:00z</value>
            </condition>
        </filter>
```

```
</entity>  
</fetch>
```

Under **Advanced**, you can specify the following fields:

- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.

Destination

Go to **Destination** tab to configure your copy activity destination. See the following content for the detailed configuration.

The screenshot shows the 'Destination' tab of a copy activity configuration. The tabs at the top are General, Source, Destination, Mapping, and Settings. The 'Destination' tab is selected. The configuration fields include:

- Data store type:** A radio button group showing 'Workspace' (selected) and 'External'.
- Connection:** A dropdown menu showing '< Dynamics CRM connection >' with options to 'Select', 'Edit', and 'New'. To the right are buttons for 'Refresh', 'Edit', and '+ New'.
- Connection type:** A dropdown menu showing 'Dynamics CRM' with options to 'Select', 'Edit', and 'New'.
- Entity name:** A dropdown menu showing 'Lead' with options to 'Select', 'Edit', and 'Preview data'.
- Advanced:** A section containing the following settings:
 - Write behavior:** A dropdown menu showing 'Insert'.
 - Alternate key name:** A dropdown menu showing 'Lead'.
 - Ignore null values:** A checkbox that is unchecked.
 - Write batch size:** A text input field showing '10'.
 - Max concurrent connections:** A dropdown menu showing '1'.

The following properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select an existing Dynamics CRM connection from the connection list. If no connection exists, then create a new Dynamics CRM connection by selecting **New**.
- **Connection type:** Select **Dynamics CRM**.
- **Entity name:** Specify the name of the entity to write data. Select your entity name from the drop-down list, or select **Edit** to enter it manually. This is the logical name of the entity to retrieve.

Under **Advanced**, you can specify the following fields:

- **Write behavior:** The write behavior of the operation. This property is required, and you must select **Upsert**. If you use **Add dynamic content**, specify the value to **Upsert**.
- **Alternate key name:** Specify the alternate key name defined on your entity to upsert records.
- **Ignore null values:** Indicates whether to ignore null values from input data during write operation. It is unselected by default.
 - When it is selected: Leave the data in the destination object unchanged when doing upsert/update operation, and insert defined default value when doing insert operation.
 - When it is unselected: Update the data in the destination object to NULL when doing upsert/update operation, and insert NULL value when doing insert operation.
- **Write batch size:** Specify the row count of data written to Dynamics CRM in each batch.
- **Max concurrent connections:** The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.

Mapping

For **Mapping** tab configuration, go to Configure your mappings under mapping tab.

Settings

For the **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

See the following table for the summary and more information for the Dynamics CRM copy activity.

Source information

[] Expand table

Name	Description	Value	Required	JSON script property
Data store	Your data store type.	External	Yes	/

Name	Description	Value	Required	JSON script property
type				
Connection	Your connection to the source data store.	< your Dynamics CRM connection >	Yes	connection
Connection type	Your connection type.	Dynamics CRM	Yes	type (under typeProperties -> source -> datasetSettings): DynamicsCrmEntity
Use query	The way to read data from Dynamics CRM	* Tables * Query	Yes	/
Entity name	The logical name of the entity to retrieve.	< your entity name >	Yes	entityName
Query	Use FetchXML to read data from Dynamics CRM. FetchXML is a proprietary query language that is used in Dynamics online and on-premises. To learn more, see Build queries with FetchXML.	< your query >	Yes	query
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.	* Name * Value	No	additionalColumns: * name * value

Destination information

[+] Expand table

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the destination Dynamics CRM.	< your connection >	Yes	connection

Name	Description	Value	Required	JSON script property
Connection type	Your connection type.	Dynamics CRM	Yes	type (under <code>typeProperties -> sink -> datasetSettings</code>): DynamicsCrmEntity
Entity name	The logical name of the entity to retrieve.	< your entity >	Yes	entityName
Write behavior	The write behavior of the operation. The value must be Upsert .	Upsert	Yes	writeBehavior: upsert
Alternate key name	The alternate key name defined on your entity to upsert records.	< alternate key name >	No	alternateKeyName
Ignore null values	Indicates whether to ignore null values from input data during write operation. - Selected (true): Leave the data in the destination object unchanged when doing upsert/update operation, and insert defined default value when doing insert operation. - Unselected (false): Update the data in the destination object to NULL when doing upsert/update operation, and insert NULL value when doing insert operation.	selected or unselected (default)	No	ignoreNullValues: true or false (default)
Write batch size	The row count of data written to Dynamics CRM in each batch.	< your write batch size > The default value is 10	No	writeBatchSize
Max concurrent connections	The upper limit of concurrent connections	< max concurrent	No	maxConcurrentConnections

Name	Description	Value	Required	JSON script property
	established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.	connections >		

Next steps

Dynamics CRM connector overview

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback [↗](#) | Ask the community [↗](#)

FHIR connector overview

Article • 11/15/2023

The Fast Healthcare Interoperability Resources (FHIR® ) connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to FHIR data in Dataflow Gen2, go to Set up your FHIR data connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support FHIR data in data pipelines.

Feedback

Was this page helpful?



Yes



No

Provide product feedback  | Ask the community 

Set up your FHIR data connection

Article • 11/15/2023

This article outlines the steps to create a FHIR data connection.

Supported authentication types

The FHIR connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Anonymous	n/a	✓
Organizational account	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to FHIR data. The following links provide the specific Power Query connector information you need to connect to FHIR data in Dataflow Gen2:

- To get started using the FHIR connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric \(Preview\)](#).
- Be sure to install or set up any FHIR prerequisites before connecting to the FHIR connector.
- To connect to the FHIR connector from Power Query, go to [Connect to a FHIR server from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support FHIR in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Folder connector overview

Article • 11/15/2023

The Folder connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to a folder in Dataflow Gen2, go to Set up your Folder connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support the Folder connector in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your Folder connection

Article • 11/15/2023

This article outlines the steps to create a folder connection.

Supported authentication types

The Folder connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Windows	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to a folder. The following links provide the specific Power Query connector information you need to connect to a folder in Dataflow Gen2:

- To get started using the Folder connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric \(Preview\)](#).
- To connect to a folder from Power Query, go to [Connect to a folder from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support folders in data pipelines.

Feedback

Was this page helpful?



Yes



No

Provide product feedback ↗ | Ask the community ↗

Avro format in Data Factory in Microsoft Fabric

Article • 11/15/2023

This article outlines how to configure Avro format in the data pipeline of Data Factory in Microsoft Fabric.

Supported capabilities

Avro format is supported for the following activities and connectors as a source and destination.

Category	Connector/Activity
Supported connector	Amazon S3
	Azure Blob Storage
	Azure Data Lake Storage Gen1
	Azure Data Lake Storage Gen2
	FTP
	Google Cloud Storage
	HTTP
	SFTP
Supported activity	Copy activity
	Lookup activity
	GetMetadata activity
	Delete activity

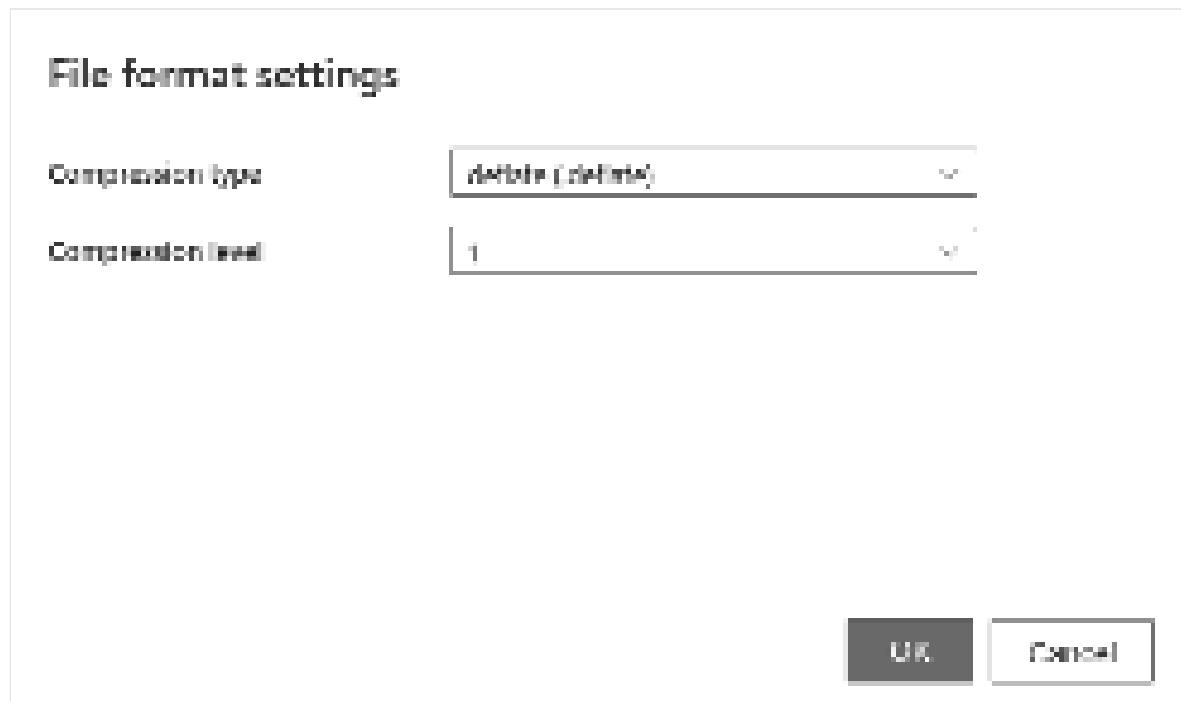
Avro format in copy activity

To configure Avro format, choose your connection in the source or destination of data pipeline copy activity, and then select **Avro** in the drop-down list of **File format**. Select **Settings** for further configuration of this format.



Avro format as source

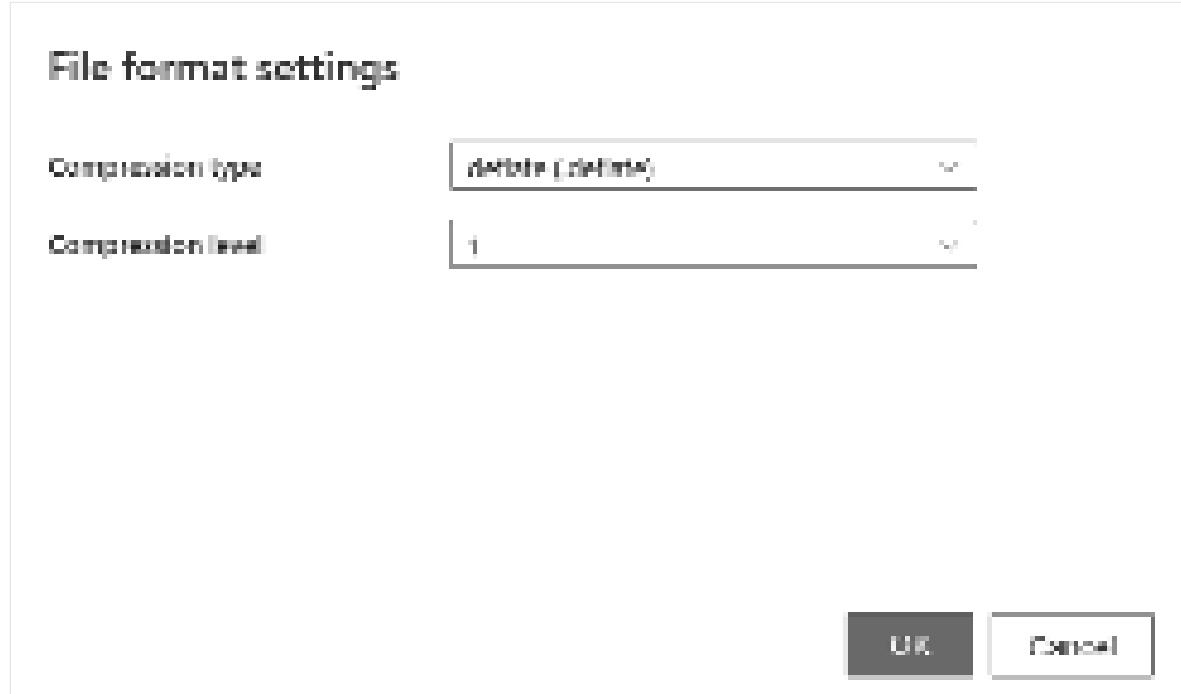
After you select **Settings** in the **File format** section, the following properties are shown in the pop-up **File format settings** dialog box.



- **Compression type:** Choose the compression codec used to read Avro files in the drop-down list. You can choose from **None** or **deflate**. You can also choose **Add dynamic content** to specify the type in the expression builder.
- **Compression level:** Specify the compression ratio. Choose from **1** to **7**. You can also choose **Add dynamic content** to specify the level in the expression builder.

Avro format as destination

After you select **Settings**, the following properties are shown in the pop-up **File format settings** dialog box.



- **Compression type:** Choose the compression codec used to write Avro files in the drop-down list. You can choose from **None** or **deflate**. You can also choose **Add dynamic content** to specify the type in the expression builder.
- **Compression level:** Specify the compression ratio. Choose from **1** to **7**. You can also choose **Add dynamic content** to specify the level in the expression builder.

Under **Advanced** settings in the **Destination** tab, the following Avro format related properties are displayed.

- **Max rows per file:** When writing data into a folder, you can choose to write to multiple files and specify the maximum rows per file.
- **File name prefix:** Applicable when **Max rows per file** is configured. Specify the file name prefix when writing data to multiple files, resulted in this pattern:
`<fileNamePrefix>_00000.<fileExtension>`. If not specified, the file name prefix is auto generated. This property doesn't apply when the source is a file based store or a partition option enabled data store.

Table summary

Avro as source

The following properties are supported in the copy activity **Source** section when using the Avro format.

Name	Description	Value	Required	Avro script property
File format	The file format that you want to use.	Avro	Yes	type (<i>under datasetSettings</i>): Avro
Compression type	The compression codec used to read Avro files.	None deflate	No	avroCompressionCodec: deflate
Compression level	The compression ratio.	1 to 7	No	avroCompressionLevel: 1 to 7

Avro as destination

The following properties are supported in the copy activity **Destination** section when using the Avro format.

Name	Description	Value	Required	Avro script property
File format	The file format that you want to use.	Avro	Yes	type (<i>under datasetSettings</i>): Avro
Compression type	The compression codec used to write Avro files.	None deflate	No	avroCompressionCodec: deflate
Compression level	The compression ratio.	1 to 7	No	avroCompressionLevel: 1 to 7
Max rows per file	When writing data into a folder, you can choose to write to multiple files and specify the maximum rows per file.	< your max rows per file >	No	maxRowsPerFile
File name prefix	Applicable when Max rows per file is configured. Specify the file name prefix when writing data to multiple files, resulted in this pattern: <code><fileNamePrefix>_00000. <fileExtension></code> . If not specified, the file name prefix is	< your file name prefix >	No	fileNamePrefix

Name	Description	Value	Required	Avro script property
	auto generated. This property doesn't apply when the source is a file based store or a partition option enabled data store.			

Next steps

- Connectors overview
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback [↗](#) | Ask the community [↗](#)

Binary format for Data Factory in Microsoft Fabric

Article • 11/15/2023

This article outlines how to configure Binary format in Data Factory.

Supported capabilities

Binary format is supported for the following activities and connectors as source and destination.

Category	Connector/Activity
Supported connector	Amazon S3
	Azure Blob Storage
	Azure Data Lake Storage Gen1
	Azure Data Lake Storage Gen2
	Google Cloud Storage
	HTTP
Supported activity	Copy activity
	GetMetadata activity
	Delete activity

Binary format in copy activity

To configure Binary format, choose your connection in the source or destination of the data pipeline copy activity, and then select **Binary** in the drop-down list of **File format**. Select **Settings** for further configuration of this format.



① Note

When using Binary format in a copy activity, source and destination should both use Binary format.

Binary as source

After you select **Settings** in the **File format** section under the **Source** tab, the following properties are displayed in the pop-up **File format settings** dialog box.



- **Compression type:** The compression codec used to read binary files. You can choose from the **None**, **bzip2**, **gzip**, **deflate**, **ZipDeflate**, **TarGzip**, or **tar** type in the drop-down list.

If you select **ZipDeflate** as the compression type, **Preserve zip file name as folder** is displayed under the **Advanced** settings in the **Source** tab.

- **Preserve zip file name as folder:** Indicates whether to preserve the source zip file name as a folder structure during copy.

- If this box is checked (default), the service writes unzipped files to `<specified file path>/<folder named as source zip file>/`.
- If this box is unchecked, the service writes unzipped files directly to `<specified file path>`. Make sure you don't have duplicated file names in different source zip files to avoid racing or unexpected behavior.

If you select **TarGzip/tar** as the compression type, **Preserve compression file name as folder** is displayed under the **Advanced** settings in the **Source** tab.

- **Preserve compression file name as folder:** Indicates whether to preserve the source compressed file name as a folder structure during copy.
 - If this box is checked (default), the service writes decompressed files to `<specified file path>/<folder named as source compressed file>/`.
 - If this box is unchecked, the service writes decompressed files directly to `<specified file path>`. Make sure you don't have duplicated file names in different source zip files to avoid racing or unexpected behavior.
- **Compression level:** The compression ratio. You can choose from **Optimal** or **Fastest**.
 - **Fastest:** The compression operation should complete as quickly as possible, even if the resulting file isn't optimally compressed.
 - **Optimal:** The compression operation should be optimally compressed, even if the operation takes a longer time to complete. For more information, go to the [Compression Level article](#).

Under **Advanced** settings in the **Source** tab, further Binary format related property are displayed.

- **Delete files after completion:** Indicates whether the binary files are deleted from the source store after successfully moving to the destination store. The file deletion is per file. So when a copy activity fails, some files have already been copied to the destination and deleted from the source, while others still remain on the source store.

Binary as destination

After you select **Settings** in the **File format** section under the **Destination** tab, following properties are displayed in the pop-up **File format settings** dialog box.

File format settings



- **Compression type:** The compression codec used to write binary files. You can choose from the **None**, **bzip2**, **gzip**, **deflate**, **ZipDeflate**, **TarGzip** or **tar** type in the drop-down list.
- **Compression level:** The compression ratio. You can choose from **Optimal** or **Fastest**.
 - **Fastest:** The compression operation should complete as quickly as possible, even if the resulting file isn't optimally compressed.
 - **Optimal:** The compression operation should be optimally compressed, even if the operation takes a longer time to complete. For more information, go to the Compression Level article.

Table summary

Binary as source

The following properties are supported in the copy activity **Source** section when using Binary format.

Name	Description	Value	Required	JSON script property
File format	The file format that you want to use.	Binary	Yes	type (under <code>datasetSettings</code>): Binary
Compression type	The compression codec used to read binary files.	Choose from: None bzip2 gzip deflate ZipDeflate	No	type (under <code>compression</code>): bzip2 gzip deflate ZipDeflate TarGzip tar

Name	Description	Value	Required	JSON script property
		TarGzip tar		
Compression level	The compression ratio. Allowed values are Optimal or Fastest.	Optimal or Fastest	No	level (under <code>compression</code>): Fastest Optimal
Preserve zip file name as folder	Indicates whether to preserve the source zip file name as a folder structure during copy.	Selected or unselect	No	<code>preserveZipFileNameAsFolder</code> (under <code>compressionProperties -> type</code> as <code>ZipDeflateReadSettings</code>)
Preserve compression file name as folder	Indicates whether to preserve the source compressed file name as a folder structure during copy.	Selected or unselect	No	<code>preserveCompressionFileNameAsFolder</code> (under <code>compressionProperties -> type</code> as <code>TarGZipReadSettings</code> or <code>TarReadSettings</code>)
Delete files after completion	Indicates whether the binary files are deleted from the source store after successfully moving to the destination store.	Selected or unselect	No	<code>deleteFilesAfterCompletion:</code> true or false

Binary as destination

The following properties are supported in the copy activity **Destination** section when using Binary format.

Name	Description	Value	Required	JSON script property
File format	The file format that you want to use.	Binary	Yes	type (<i>under datasetSettings</i>): Binary
Compression type	The compression codec used to write binary files.	Choose from: None bzip2 gzip deflate ZipDeflate TarGzip tar	No	type (<i>under compression</i>): bzip2 gzip deflate ZipDeflate TarGzip tar
Compression level	The compression ratio. Allowed values are Optimal or Fastest.	Optimal or Fastest	No	level (<i>under compression</i>): Fastest Optimal

Next steps

- Connectors overview
- Connect to Binary format in dataflows

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Connect to Binary format in dataflows

Article • 11/15/2023

A Binary format connector isn't currently supported in Dataflow Gen2.

Next steps

- How to configure Binary format in Data Factory
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback [↗](#) | Ask the community [↗](#)

Delimited text format in Data Factory in Microsoft Fabric

Article • 11/15/2023

This article outlines how to configure delimited text format in the data pipeline of Data Factory in Microsoft Fabric.

Supported capabilities

Delimited text format is supported for the following activities and connectors as source and destination.

Category	Connector/Activity
Supported connector	Amazon S3
	Azure Blob Storage
	Azure Data Lake Storage Gen1
	Azure Data Lake Storage Gen2
	Google Cloud Storage
	HTTP
Supported activity	Copy activity
	Lookup activity
	GetMetadata activity
	Delete activity

Delimited text format in copy activity

To configure delimited text format, choose your connection in the source or destination of data pipeline copy activity, and then select **DelimitedText** in the drop-down list of **File format**. Select **Settings** for further configuration of this format.



Delimited text format as source

After selecting **Settings** in **File format** section, following properties are shown up in the pop-up **File format settings** dialog box.



- **Compression type:** The compression codec used to read delimited text files. You can choose from **None**, **bzip2**, **gzip**, **deflate**, **ZipDeflate**, **TarGzip** or **tar** type in the drop-down list.

If you select **ZipDeflate** as compression type, **Preserve zip file name as folder** will show up under **Advanced** settings in **Source** tab.

- **Preserve zip file name as folder:** Indicates whether to preserve the source zip file name as folder structure during copy.
 - If this box is checked (default), the service writes unzipped files to `<specified file path>/<folder named as source zip file>/`.
 - If this box is unchecked, the service writes unzipped files directly to `<specified file path>`. Make sure you don't have duplicated file names in different source zip files to avoid racing or unexpected behavior.

If you select **TarGzip/tar** as compression type, **Preserve compression file name as folder** will show up under **Advanced** settings in **Source** tab.

- **Preserve compression file name as folder:** Indicates whether to preserve the source compressed file name as folder structure during copy.
 - If this box is checked (default), the service writes decompressed files to `<specified file path>/<folder named as source compressed file>/`.
 - If this box is unchecked, the service writes decompressed files directly to `<specified file path>`.
Make sure you don't have duplicated file names in different source zip files to avoid racing or unexpected behavior.
- **Compression level:** Specify the compression ratio when you select a compression type. You can choose from **Optimal** or **Fastest**.
 - **Fastest:** The compression operation should complete as quickly as possible, even if the resulting file is not optimally compressed.
 - **Optimal:** The compression operation should be optimally compressed, even if the operation takes a longer time to complete. For more information, see [Compression Level](#) topic.
- **Column delimiter:** The character(s) used to separate columns in a file. The default value is **comma** (,).
- **Row delimiter:** Specify the character used to separate rows in a file. Only one character is allowed. The default value is line feed `\n`.
- **Encoding:** The encoding type used to read/write test files. The default value is **UTF-8**.
- **Escape character:** The single character to escape quotes inside a quoted value. The default value is **backslash** \. When escape character is defined as empty string, the **Quote character** must be set as empty string as well, in which case make sure all column values don't contain delimiters.
- **Quote character:** The single character to quote column values if it contains column delimiter. The default value is **double quotes** ". When **Quote character** is defined as empty string, it means there is no quote char and column value is not quoted, and escape character is used to escape the column delimiter and itself.
- **First row as header:** Specifies whether to treat/make the first row as a header line with names of columns. Allowed values are selected and unselected (default). When first row as header is unselected, note UI data preview and lookup activity output auto generate column names as Prop_{n} (starting from 0), copy activity requires explicit mapping from source to destination and locates columns by ordinal (starting from 1).
- **Null value:** Specifies the string representation of null value. The default value is empty string.

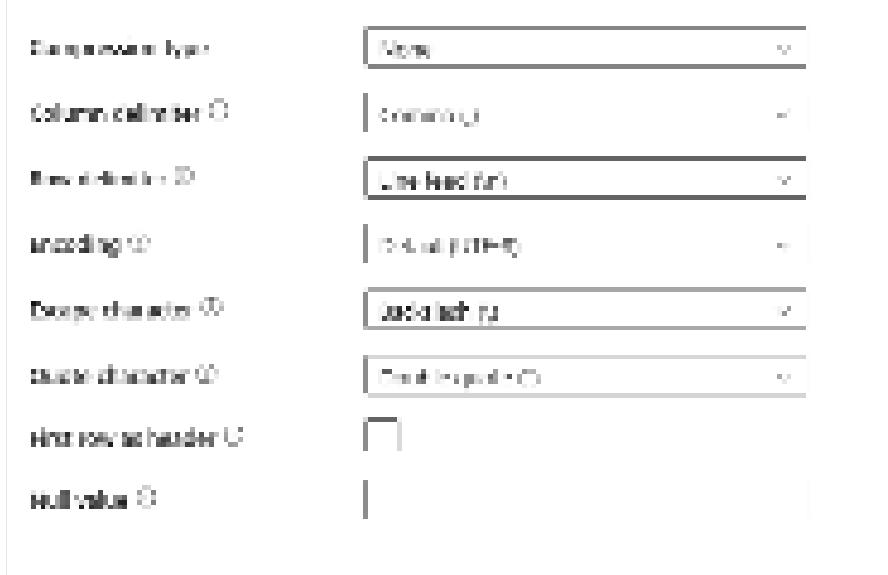
Under **Advanced** settings in **Source** tab, further delimited text format related property are shown up.

- **Skip line count:** Indicates the number of nonempty rows to skip when reading data from input files. If both **Skip line count** and **First row as header** are specified, the lines are skipped first, and then the header information is read from the input file.

Delimited text format as destination

After selecting **Settings** in **File format** section, following properties are shown up in the pop-up **File format settings** dialog box.

File format settings



- **Compression type:** The compression codec used to write delimited text files. You can choose from **None**, **bzip2**, **gzip**, **deflate**, **ZipDeflate**, **TarGzip** or **tar** type in the drop-down list.
- **Compression level:** Specify the compression ratio when you select a compression type. You can choose from **Optimal** or **Fastest**.
 - **Fastest:** The compression operation should complete as quickly as possible, even if the resulting file is not optimally compressed.
 - **Optimal:** The compression operation should be optimally compressed, even if the operation takes a longer time to complete. For more information, see [Compression Level topic](#).
- **Column delimiter:** The character(s) used to separate columns in a file. The default value is comma (,).
- **Row delimiter:** The character used to separate rows in a file. Only one character is allowed. The default value is line feed (\n).
- **Encoding:** The encoding type used to write test files. The default value is **UTF-8**.
- **Escape character:** The single character to escape quotes inside a quoted value. The default value is **backslash** (\). When escape character is defined as empty string, the **Quote character** must be set as empty string as well, in which case make sure all column values don't contain delimiters.
- **Quote character:** The single character to quote column values if it contains column delimiter. The default value is **double quotes** ("). When **Quote character** is defined as empty string, it means there is no quote char and column value is not quoted, and escape character is used to escape the column delimiter and itself.
- **First row as header:** Specifies whether to treat/make the first row as a header line with names of columns. Allowed values are selected and unselected (default). When first row as header is unselected, note UI data preview and lookup activity output auto generate column names as Prop_{n} (starting from 0), copy activity requires explicit mapping from source to destination and locates columns by ordinal (starting from 1).
- **Null value:** Specifies the string representation of null value. The default value is empty string.

Under **Advanced** settings in **Destination** tab, further delimited text format related property are shown up.

- Quote all text:** Enclose all values in quotes.
- File extension:** The file extension used to name the output files, for example, `.csv`, `.txt`.
- Max rows per file:** When writing data into a folder, you can choose to write to multiple files and specify the max rows per file.
- File name prefix:** Applicable when **Max rows per file** is configured. Specify the file name prefix when writing data to multiple files, resulted in this pattern: `<fileNamePrefix>_0000.<fileExtension>`. If not specified, file name prefix will be auto generated. This property does not apply when source is file based store or partition option enabled data store.

Table summary

Delimited text as source

The following properties are supported in the copy activity **Source** section when using delimited text format.

Name	Description	Value	Required	JSON script property
File format	The file format that you want to use.	DelimitedText	Yes	type (under <code>datasetSettings</code>): DelimitedText
Compression type	The compression codec used to read delimited text files.	Choose from: None bzip2 gzip deflate ZipDeflate TarGzip tar	No	type (under <code>compression</code>): bzip2 gzip deflate ZipDeflate TarGzip tar
Preserve zip file name as folder	Indicates whether to preserve the source zip file name as folder structure during copy. Applies when you select ZipDeflate compression.	Selected or unselect	No	preserveZipFileNameAsFolder (under <code>compressionProperties -> type as ZipDeflateReadSettings</code>)
Preserve compression file name as folder	Indicates whether to preserve the source compressed file name as folder structure during copy. Applies when you select TarGzip/tar compression.	Selected or unselect	No	preserveCompressionFileNameAsFolder (under <code>compressionProperties -> type as TarGZipReadSettings or TarReadSettings</code>)
Compression level	The compression ratio. Allowed values are Optimal or Fastest.	Optimal or Fastest	No	level (under <code>compression</code>): Fastest Optimal
Column delimiter	The character(s) used to separate columns in a file.	< the selected column delimiter >	No	columnDelimiter

Name	Description	Value	Required	JSON script property
		comma <code>,</code> (by default)		
Row delimiter	The character used to separate rows in a file.	< the selected row delimiter > <code>\r</code> , <code>\n</code> (by default), or <code>\r\n</code>	No	rowDelimiter
Encoding	The encoding type used to read/write test files.	"UTF-8" (by default), "UTF-8 without BOM", "UTF-16", "UTF-16BE", "UTF-32", "UTF-32BE", "US-ASCII", "UTF-7", "BIG5", "EUC-JP", "EUC-KR", "GB2312", "GB18030", "JOHAB", "SHIFT-JIS", "CP875", "CP866", "IBM00858", "IBM037", "IBM273", "IBM437", "IBM500", "IBM737", "IBM775", "IBM850", "IBM852", "IBM855", "IBM857", "IBM860", "IBM861", "IBM863", "IBM864", "IBM865", "IBM869", "IBM870", "IBM01140", "IBM01141", "IBM01142", "IBM01143", "IBM01144", "IBM01145", "IBM01146", "IBM01147", "IBM01148", "IBM01149", "ISO-2022-JP", "ISO-2022-KR", "ISO-8859-1", "ISO-8859-2", "ISO-8859-3", "ISO-8859-4", "ISO-8859-5", "ISO-8859-6", "ISO-	No	encodingName

Name	Description	Value	Required	JSON script property
		8859-7", "ISO- 8859-8", "ISO- 8859-9", "ISO- 8859-13", "ISO- 8859-15", "WINDOWS-874", "WINDOWS- 1250", "WINDOWS- 1251", "WINDOWS- 1252", "WINDOWS- 1253", "WINDOWS- 1254", "WINDOWS- 1255", "WINDOWS- 1256", "WINDOWS- 1257", "WINDOWS-1258"		
Escape character	The single character to escape quotes inside a quoted value. When escape character is defined as empty string, the Quote character must be set as empty string as well, in which case make sure all column values don't contain delimiters.	< your selected escape character > backslash \ (by default)	No	escapeChar
Quote character	The single character to quote column values if it contains column delimiter. When Quote character is defined as empty string, it means there is no quote char and column value is not quoted, and escape character is used to escape the column delimiter and itself.	< your selected quote character > double quotes "" (by default)	No	quoteChar
First row as header	Specifies whether to treat the first row in the given worksheet/range as a header line with names of columns.	Selected or unselected	No	firstRowAsHeader: true or false (default)

Name	Description	Value	Required	JSON script property
Null value	Specifies the string representation of null value. The default value is empty string.	< the string representation of null value > empty string (by default)	No	nullValue
Skip line count	Indicates the number of non-empty rows to skip when reading data from input files. If both Skip line count and First row as header are specified, the lines are skipped first and then the header information is read from the input file.	< your skip line count >	No	skipLineCount

Delimited text as destination

The following properties are supported in the copy activity **Destination** section when using delimited text format.

Name	Description	Value	Required	JSON script property
File format	The file format that you want to use.	DelimitedText	Yes	type (<i>under datasetSettings</i>): DelimitedText
Compression type	The compression codec used to write delimited text files.	Choose from: None bzip2 gzip deflate ZipDeflate TarGzip tar	No	type (<i>under compression</i>): bzip2 gzip deflate ZipDeflate TarGzip tar
Preserve zip file name as folder	Indicates whether to preserve the source zip file name as folder structure during copy.	Selected or unselect	No	preserveZipFileNameAsFolder (<i>under compressionProperties -> type as ZipDeflateReadSettings</i>)
Preserve compression file name as folder	Indicates whether to preserve the source compressed file name as folder structure during copy.	Selected or unselect	No	preserveCompressionFileNameAsFolder (<i>under compressionProperties -> type as TarGZipReadSettings or TarReadSettings</i>)
Compression level	The compression ratio. Allowed values are Optimal or Fastest.	Optimal or Fastest	No	level (<i>under compression</i>): Fastest Optimal
Column delimiter	The character(s) used to separate columns in a file.	< the selected column delimiter >	No	columnDelimiter

Name	Description	Value	Required	JSON script property
		comma <code>,</code> (by default)		
Row delimiter	The character used to separate rows in a file.	< the selected row delimiter > <code>\r, \n</code> (by default), or <code>r\n</code>	No	rowDelimiter
Encoding	The encoding type used to read/write test files.	"UTF-8" (by default), "UTF-8 without BOM", "UTF-16", "UTF-16BE", "UTF-32", "UTF-32BE", "US-ASCII", "UTF-7", "BIG5", "EUC-JP", "EUC-KR", "GB2312", "GB18030", "JOHAB", "SHIFT-JIS", "CP875", "CP866", "IBM00858", "IBM037", "IBM273", "IBM437", "IBM500", "IBM737", "IBM775", "IBM850", "IBM852", "IBM855", "IBM857", "IBM860", "IBM861", "IBM863", "IBM864", "IBM865", "IBM869", "IBM870", "IBM01140", "IBM01141", "IBM01142", "IBM01143", "IBM01144", "IBM01145", "IBM01146", "IBM01147", "IBM01148", "IBM01149", "ISO-2022-JP", "ISO-2022-KR", "ISO-8859-1", "ISO-8859-2", "ISO-8859-3", "ISO-8859-4", "ISO-8859-5", "ISO-8859-6", "ISO-	No	encodingName

Name	Description	Value	Required	JSON script property
		8859-7", "ISO- 8859-8", "ISO- 8859-9", "ISO- 8859-13", "ISO- 8859-15", "WINDOWS-874", "WINDOWS- 1250", "WINDOWS- 1251", "WINDOWS- 1252", "WINDOWS- 1253", "WINDOWS- 1254", "WINDOWS- 1255", "WINDOWS- 1256", "WINDOWS- 1257", "WINDOWS-1258"		
Escape character	The single character to escape quotes inside a quoted value. When escape character is defined as empty string, the Quote character must be set as empty string as well, in which case make sure all column values don't contain delimiters.	< your selected escape character > backslash \ (by default)	No	escapeChar
Quote character	The single character to quote column values if it contains column delimiter. When Quote character is defined as empty string, it means there is no quote char and column value is not quoted, and escape character is used to escape the column delimiter and itself.	< your selected quote character > double quotes "" (by default)	No	quoteChar
First row as header	Specifies whether to treat the first row in the given worksheet/range as a header line with names of columns.	Selected or unselected	No	firstRowAsHeader: true or false (default)
Quote all text	Enclose all values in quotes.	Selected (default) or unselected	No	quoteAllText: true (default) or false

Name	Description	Value	Required	JSON script property
File extension	The file extension used to name the output files.	< your file extension > .txt (by default)	No	fileExtension
Max rows per file	When writing data into a folder, you can choose to write to multiple files and specify the max rows per file.	< your max rows per file >	No	maxRowsPerFile
File name prefix	Applicable when Max rows per file is configured. Specify the file name prefix when writing data to multiple files, resulted in this pattern: <code><fileNamePrefix>_00000. <fileExtension></code> . If not specified, file name prefix will be auto generated. This property does not apply when source is file based store or partition option enabled data store.	< your file name prefix >	No	fileNamePrefix

Next steps

[Connectors overview](#)

Feedback

Was this page helpful?

 Yes
 No

Provide product feedback [↗](#) | Ask the community [↗](#)

Excel format in Data Factory in Microsoft Fabric

Article • 11/15/2023

This article outlines how to configure Excel format in the data pipeline of Data Factory in Microsoft Fabric.

Supported capabilities

Excel format is supported for the following activities and connectors as source.

Category	Connector/Activity
Supported connector	Amazon S3
	Azure Blob Storage
	Azure Data Lake Storage Gen1
	Azure Data Lake Storage Gen2
	Google Cloud Storage
	HTTP
Supported activity	Copy activity
	Lookup activity
	GetMetadata activity
	Delete activity

Excel format in copy activity

To configure Excel format, choose your connection in the source of data pipeline copy activity, and then select **Excel** in the drop-down list of **File format**. Select **Settings** for further configuration of this format.



Excel as source

After choosing Excel format, following properties are shown up.

- **Worksheet mode:** Select the worksheet mode that you want to use to read Excel data. Choose **Name** or **Index**.
 - **Name:** When you choose **Name**, in **Sheet name** section, select the Excel worksheet name to read data, or select **Edit** to specify the worksheet name manually. If you point to a folder or multiple files, make sure this particular worksheet exists in all those files.



- **Index:** When you choose **Index**, in **Sheet index** section, select the Excel worksheet index to read data, or select **Edit** to specify the worksheet name manually. The data read start from 0. If there is worksheet added or deleted from excel file, the index of existed worksheets will change automatically.



After selecting **Settings** in **File format** section, following properties are shown up in the pop-up **File format settings** dialog box.



- **Compression type:** The compression codec used to read Excel files. You can choose from **None**, **bzip2**, **gzip**, **deflate**, **ZipDeflate**, **TarGzip** or **tar** type in the drop-down list.
- **Compression level:** Specify the compression ratio when you select a compression type. You can choose from **Optimal** or **Fastest**.
 - **Fastest:** The compression operation should complete as quickly as possible, even if the resulting file is not optimally compressed.
 - **Optimal:** The compression operation should be optimally compressed, even if the operation takes a longer time to complete. For more information, see [Compression Level topic](#).
- **Range:** The cell range in the given worksheet to locate the selective data, e.g.:
 - Not specified: reads the whole worksheet as a table from the first non-empty row and column.
 - A3: reads a table starting from the given cell, dynamically detects all the rows below and all the columns to the right.
 - A3:H5: reads this fixed range as a table.
 - A3:A3: reads this single cell.
- **Null value:** Specifies the string representation of null value. The default value is empty string.
- **First row as header:** Specifies whether to treat the first row in the given worksheet/range as a header line with names of columns. Unselected by default.

Table summary

Excel as source

The following properties are supported in the copy activity **Source** section when using Excel format.

Name	Description	Value	Required	JSON script property
File format	The file format that you want to use.	Excel	Yes	type (under <code>datasetSettings</code>): Excel
Worksheet mode	The worksheet mode that you want to use to read Excel data.	- Name - Index	Yes	- sheetName - sheetIndex
Compression type	The compression codec used to read Excel files.	Choose from: None bzip2 gzip deflate ZipDeflate TarGzip tar	No	type (under <code>compression</code>): bzip2 gzip deflate ZipDeflate TarGzip tar
Compression level	The compression ratio. Allowed values are Optimal or Fastest.	Optimal or Fastest	No	level (under <code>compression</code>): Fastest Optimal
Range	The cell range in the given worksheet to locate the selective data.	<your cell range>	No	range
Null value	The string representation of null value.	<your null value> empty string (by default)	No	nullValue
First row as header	Specifies whether to treat the first row in the given worksheet/range as a header line with names of columns.	Selected or unselected	No	firstRowAsHeader: true or false (default)

Next steps

[Connectors overview](#)

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Connect to an Excel workbook in dataflows

Article • 11/15/2023

You can connect to Excel workbooks in Dataflow Gen2 using the Excel connector provided by Data Factory in Microsoft Fabric.

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to an Excel workbook. The following links provide the specific Power Query connector information you need to connect to an Excel workbook in Dataflow Gen2:

- To get started using the Excel connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- Be sure to install or set up any Excel prerequisites before connecting to the Excel connector.
- To connect to the Excel connector from Power Query, go to [Connect to an Excel workbook from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

JSON format in Data Factory in Microsoft Fabric

Article • 11/15/2023

This article outlines how to configure JSON format in the data pipeline of Data Factory in Microsoft Fabric.

Supported capabilities

JSON format is supported for the following activities and connectors as a source and destination.

Category	Connector/Activity
Supported connector	Amazon S3
	Azure Blob Storage
	Azure Data Lake Storage Gen1
	Azure Data Lake Storage Gen2
	FTP
	Google Cloud Storage
	HTTP
	SFTP
Supported activity	Copy activity
	Lookup activity
	GetMetadata activity
	Delete activity

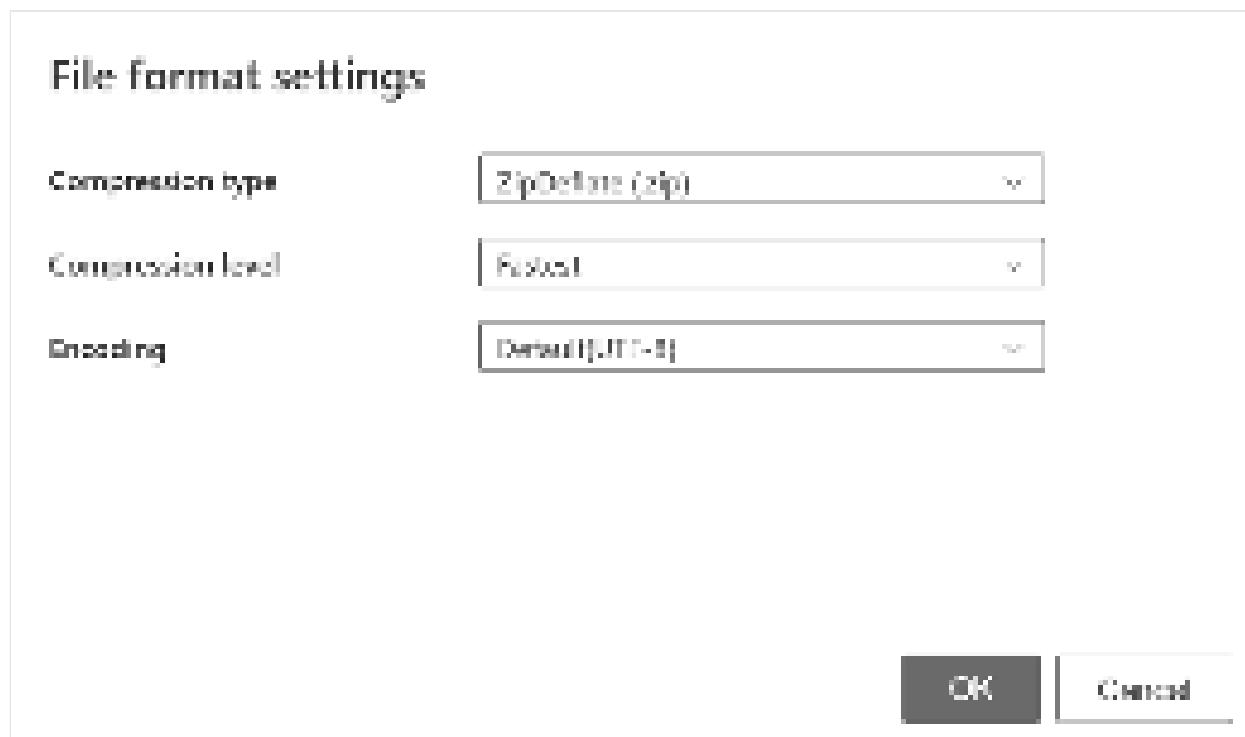
JSON format in copy activity

To configure JSON format, choose your connection in the source or destination of data pipeline copy activity, and then select **JSON** in the drop-down list of **File format**. Select **Settings** for further configuration of this format.



JSON format as source

After you select **Settings** in the **File format** section, the following properties are shown in the pop-up **File format settings** dialog box.



- **Compression type:** Choose the compression codec used to read JSON files in the drop-down list. You can choose from **None**, **bzip2**, **gzip**, **deflate**, **ZipDeflate**, **TarGzip**, or **tar**.

If you select **ZipDeflate** as the compression type, **Preserve zip file name as folder** is displayed under the **Advanced** settings in the **Source** tab.

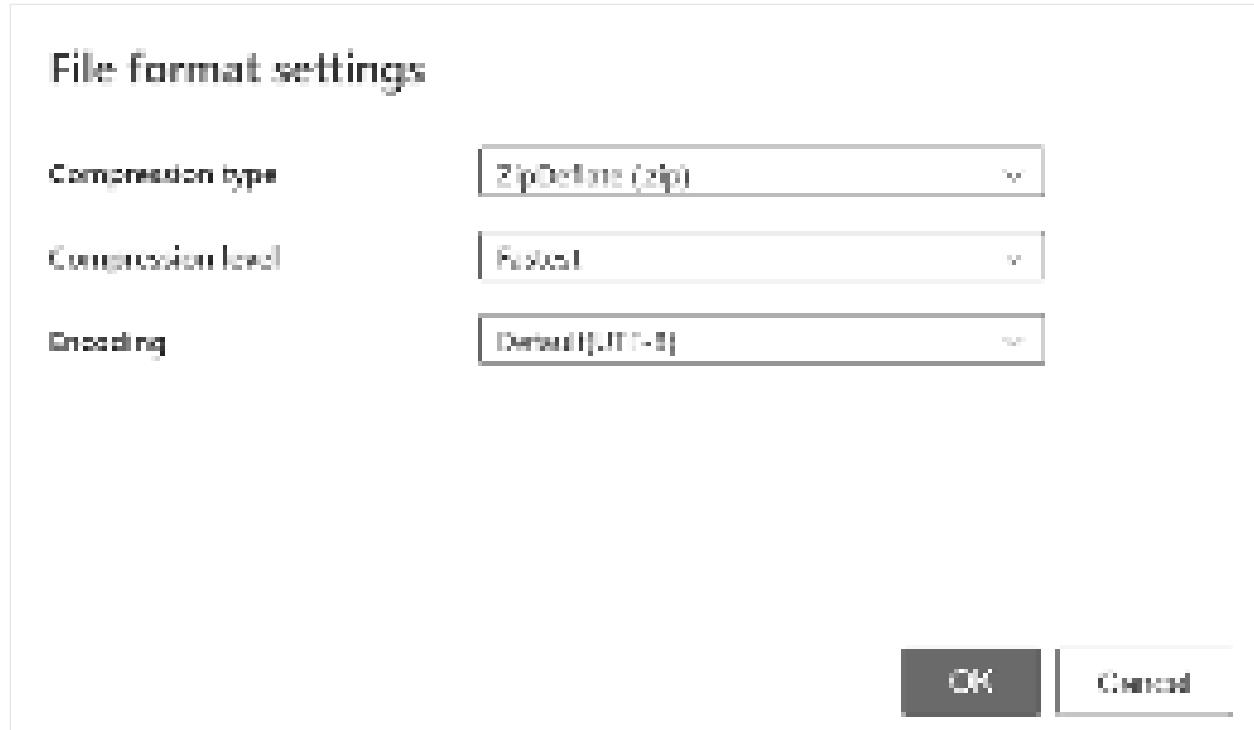
- **Preserve zip file name as folder:** Indicates whether to preserve the source zip file name as a folder structure during copy.
 - If this box is checked (default), the service writes unzipped files to `<specified file path>/<folder named as source zip file>/`.
 - If this box is unchecked, the service writes unzipped files directly to `<specified file path>`. Make sure you don't have duplicated file names in different source zip files to avoid racing or unexpected behavior.

If you select **TarGzip/tar** as the compression type, **Preserve compression file name as folder** is displayed under the **Advanced** settings in the **Source** tab.

- **Preserve compression file name as folder:** Indicates whether to preserve the source compressed file name as a folder structure during copy.
 - If this box is checked (default), the service writes decompressed files to `<specified file path>/<folder named as source compressed file>/`.
 - If this box is unchecked, the service writes decompressed files directly to `<specified file path>`. Make sure you don't have duplicated file names in different source files to avoid racing or unexpected behavior.
- **Compression level:** The compression ratio. You can choose from **Fastest** or **Optimal**.
 - **Fastest:** The compression operation should complete as quickly as possible, even if the resulting file isn't optimally compressed.
 - **Optimal:** The compression operation should be optimally compressed, even if the operation takes a longer time to complete. For more information, go to the Compression Level article.
- **Encoding:** Specify the encoding type used to read test files. Select one type from the drop-down list. The default value is **UTF-8**.

JSON format as destination

After you select **Settings**, the following properties are shown in the pop-up **File format settings** dialog box.



- **Compression type:** Choose the compression codec used to write JSON files in the drop-down list. You can choose from **None**, **bzip2**, **gzip**, **deflate**, **ZipDeflate**, **TarGzip**, or **tar**.
- **Compression level:** The compression ratio. You can choose from **Optimal** or **Fastest**.
 - **Fastest:** The compression operation should complete as quickly as possible, even if the resulting file isn't optimally compressed.
 - **Optimal:** The compression operation should be optimally compressed, even if the operation takes a longer time to complete. For more information, go to the Compression Level article.
- **Encoding:** Specify the encoding type used to write test files. Select one type from the drop-down list. The default value is **UTF-8**.

Under **Advanced** settings in the **Destination** tab, the following JSON format related properties are displayed.

- **File pattern:** Specify the pattern of data stored in each JSON file. Allowed values are: **Set of objects** (JSON Lines) and **Array of objects**. The default value is **Set of objects**. See JSON file patterns section for details about these patterns.

JSON file patterns

When copying data from JSON files, copy activity can automatically detect and parse the following patterns of JSON files. When writing data to JSON files, you can configure the file pattern on copy activity destination.

- **Type I: setOfObjects**

Each file contains single object, JSON lines, or concatenated objects.

- **single object JSON example**

JSON

```
{  
    "time": "2015-04-29T07:12:20.9100000Z",  
    "callingimsi": "466920403025604",  
    "callingnum1": "678948008",  
    "callingnum2": "567834760",  
    "switch1": "China",  
    "switch2": "Germany"  
}
```

- **JSON Lines (default for destination)**

JSON

```
{"time":"2015-04-  
29T07:12:20.9100000Z","callingimsi":"466920403025604","callingnum1":  
"678948008","callingnum2":"567834760","switch1":"China","switch2":"G  
ermany"}  
{ "time": "2015-04-  
29T07:13:21.0220000Z", "callingimsi": "466922202613463", "callingnum1":  
"123436380", "callingnum2": "789037573", "switch1": "US", "switch2": "UK" }  
{ "time": "2015-04-  
29T07:13:21.4370000Z", "callingimsi": "466923101048691", "callingnum1":  
"678901578", "callingnum2": "345626404", "switch1": "Germany", "switch2":  
"UK" }
```

- **concatenated JSON example**

JSON

```
{  
    "time": "2015-04-29T07:12:20.9100000Z",  
    "callingimsi": "466920403025604",  
    "callingnum1": "678948008",  
    "callingnum2": "567834760",  
    "switch1": "China",  
    "switch2": "Germany"  
}  
{
```

```

        "time": "2015-04-29T07:13:21.0220000Z",
        "callingimsi": "466922202613463",
        "callingnum1": "123436380",
        "callingnum2": "789037573",
        "switch1": "US",
        "switch2": "UK"
    }
{
    "time": "2015-04-29T07:13:21.4370000Z",
    "callingimsi": "466923101048691",
    "callingnum1": "678901578",
    "callingnum2": "345626404",
    "switch1": "Germany",
    "switch2": "UK"
}

```

- **Type II: arrayOfObjects**

Each file contains an array of objects.

JSON

```
[
    {
        "time": "2015-04-29T07:12:20.9100000Z",
        "callingimsi": "466920403025604",
        "callingnum1": "678948008",
        "callingnum2": "567834760",
        "switch1": "China",
        "switch2": "Germany"
    },
    {
        "time": "2015-04-29T07:13:21.0220000Z",
        "callingimsi": "466922202613463",
        "callingnum1": "123436380",
        "callingnum2": "789037573",
        "switch1": "US",
        "switch2": "UK"
    },
    {
        "time": "2015-04-29T07:13:21.4370000Z",
        "callingimsi": "466923101048691",
        "callingnum1": "678901578",
        "callingnum2": "345626404",
        "switch1": "Germany",
        "switch2": "UK"
    }
]
```

Table summary

JSON as source

The following properties are supported in the copy activity **Source** section when using the JSON format.

Name	Description	Value	Required	JSON script property
File format	The file format that you want to use.	JSON	Yes	type (<i>under datasetSettings</i>): Json
Compression type	The compression codec used to read JSON files.	Choose from: None bzip2 gzip deflate ZipDeflate TarGzip tar	No	type (<i>under compression</i>): bzip2 gzip deflate ZipDeflate TarGzip tar
Compression level	The compression ratio.	Fastest Optimal	No	level (<i>under compression</i>): Fastest Optimal
Encoding	The encoding type used to read test files.	"UTF-8" (by default), "UTF-8 without BOM", "UTF-16LE", "UTF-16BE", "UTF-32LE", "UTF-32BE", "US-ASCII", "UTF-7", "BIG5", "EUC-JP", "EUC-KR", "GB2312", "GB18030", "JOHAB", "SHIFT-JIS", "CP875", "CP866", "IBM00858", "IBM037", "IBM273", "IBM437", "IBM500", "IBM737",	No	encodingName

Name	Description	Value	Required	JSON script property
		"IBM775", "IBM850", "IBM852", "IBM855", "IBM857", "IBM860", "IBM861", "IBM863", "IBM864", "IBM865", "IBM869", "IBM870", "IBM01140", "IBM01141", "IBM01142", "IBM01143", "IBM01144", "IBM01145", "IBM01146", "IBM01147", "IBM01148", "IBM01149", "ISO-2022- JP", "ISO- 2022-KR", "ISO-8859-1", "ISO-8859-2", "ISO-8859-3", "ISO-8859-4", "ISO-8859-5", "ISO-8859-6", "ISO-8859-7", "ISO-8859-8", "ISO-8859-9", "ISO-8859- 13", "ISO- 8859-15", "WINDOWS- 874", "WINDOWS- 1250", "WINDOWS- 1251", "WINDOWS- 1252", "WINDOWS- 1253", "WINDOWS-		

Name	Description	Value	Required	JSON script property
		1254", "WINDOWS- 1255", "WINDOWS- 1256", "WINDOWS- 1257", "WINDOWS- 1258"		
Preserve zip file name as folder	Indicates whether to preserve the source zip file name as a folder structure during copy.	Selected (default) or unselect	No	preserveZipFileNameAsFolder (under <code>compressionProperties->type as ZipDeflateReadSettings</code>): true (default) or false
Preserve compression file name as folder	Indicates whether to preserve the source compressed file name as a folder structure during copy.	Selected (default) or unselect	No	preserveCompressionFileNameAsFolder (under <code>compressionProperties->type as TarGZipReadSettings or TarReadSettings</code>): true (default) or false

JSON as destination

The following properties are supported in the copy activity **Destination** section when using the JSON format.

Name	Description	Value	Required	JSON script property
File format	The file format that you want to use.	JSON	Yes	type (under <code>datasetSettings</code>): Json
Compression type	The compression codec used to write JSON files.	Choose from: None bzip2 gzip deflate	No	type (under <code>compression</code>): bzip2 gzip

Name	Description	Value	Required	JSON property
		Deflate TarGzip tar		ZipDeflate TarGzip tar
Compression level	The compression ratio.	Fastest Optimal	No	level (<i>under compression</i>): Fastest Optimal
Encoding	The encoding type used to write test files.	"UTF-8" (by default), "UTF-8 without BOM", "UTF-16LE", "UTF-16BE", "UTF-32LE", "UTF-32BE", "US-ASCII", "UTF-7", "BIG5", "EUC-JP", "EUC-KR", "GB2312", "GB18030", "JOHAB", "SHIFT-JIS", "CP875", "CP866", "IBM00858", "IBM037", "IBM273", "IBM437", "IBM500", "IBM737", "IBM775", "IBM850", "IBM852", "IBM855", "IBM857", "IBM860", "IBM861", "IBM863", "IBM864", "IBM865", "IBM869", "IBM870", "IBM01140", "IBM01141", "IBM01142", "IBM01143", "IBM01144", "IBM01145", "IBM01146", "IBM01147", "IBM01148", "IBM01149", "ISO-2022-JP", "ISO-2022-KR", "ISO-8859-1", "ISO-8859-2", "ISO-8859-3", "ISO-8859-4", "ISO-8859-5", "ISO-8859-6", "ISO-8859-7", "ISO-8859-8", "ISO-8859-9", "ISO-8859-13", "ISO-8859-15", "WINDOWS-874", "WINDOWS-1250", "WINDOWS-1251", "WINDOWS-1252", "WINDOWS-1253", "WINDOWS-1254", "WINDOWS-1255", "WINDOWS-1256", "WINDOWS-1257", "WINDOWS-1258"	No	encodingName
File pattern	Indicate the pattern of data stored in each JSON file.	Set of objects Array of objects	No	filePattern: setOfObjects arrayOfObjects

Next steps

- Connectors overview
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

ORC format in Data Factory in Microsoft Fabric

Article • 11/15/2023

This article outlines how to configure ORC format in the data pipeline of Data Factory in Microsoft Fabric.

Supported capabilities

ORC format is supported for the following activities and connectors as a source and destination.

Category	Connector/Activity
Supported connector	Amazon S3
	Azure Blob Storage
	Azure Data Lake Storage Gen1
	Azure Data Lake Storage Gen2
	FTP
	Google Cloud Storage
	HTTP
	SFTP
Supported activity	Copy activity
	Lookup activity
	GetMetadata activity
	Delete data activity

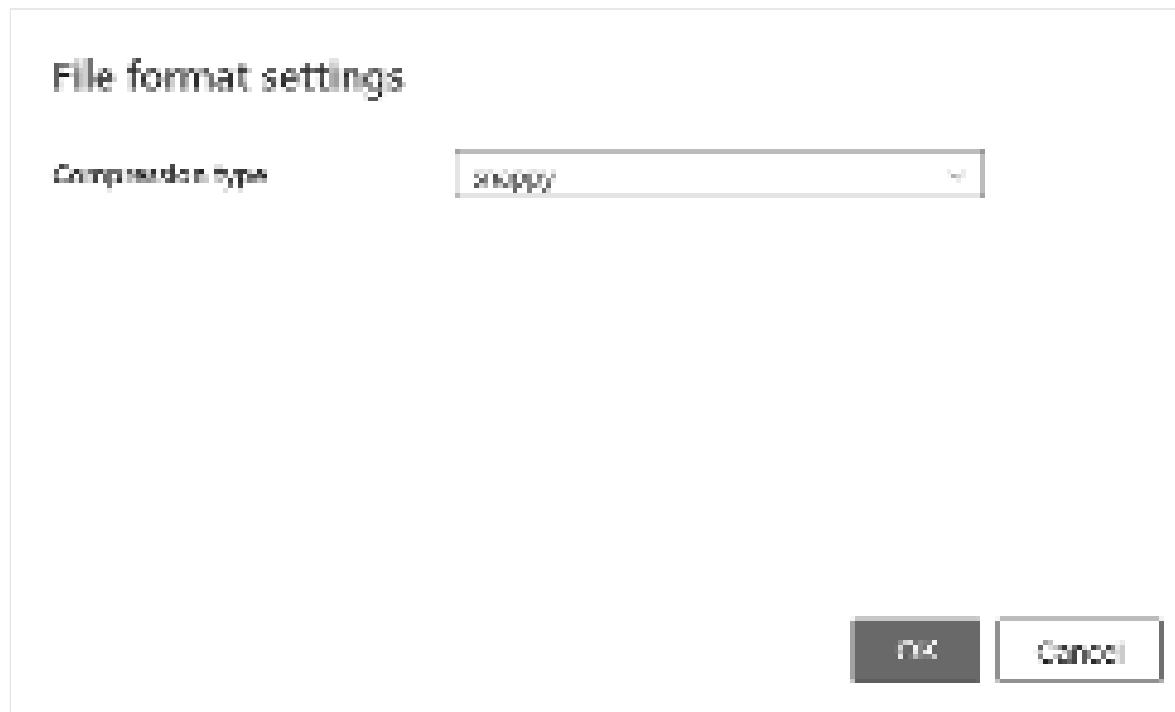
ORC format in copy activity

To configure ORC format, choose your connection in the source or destination of data pipeline copy activity, and then select **ORC** in the drop-down list of **File format**. Select **Settings** for further configuration of this format.



ORC format as source

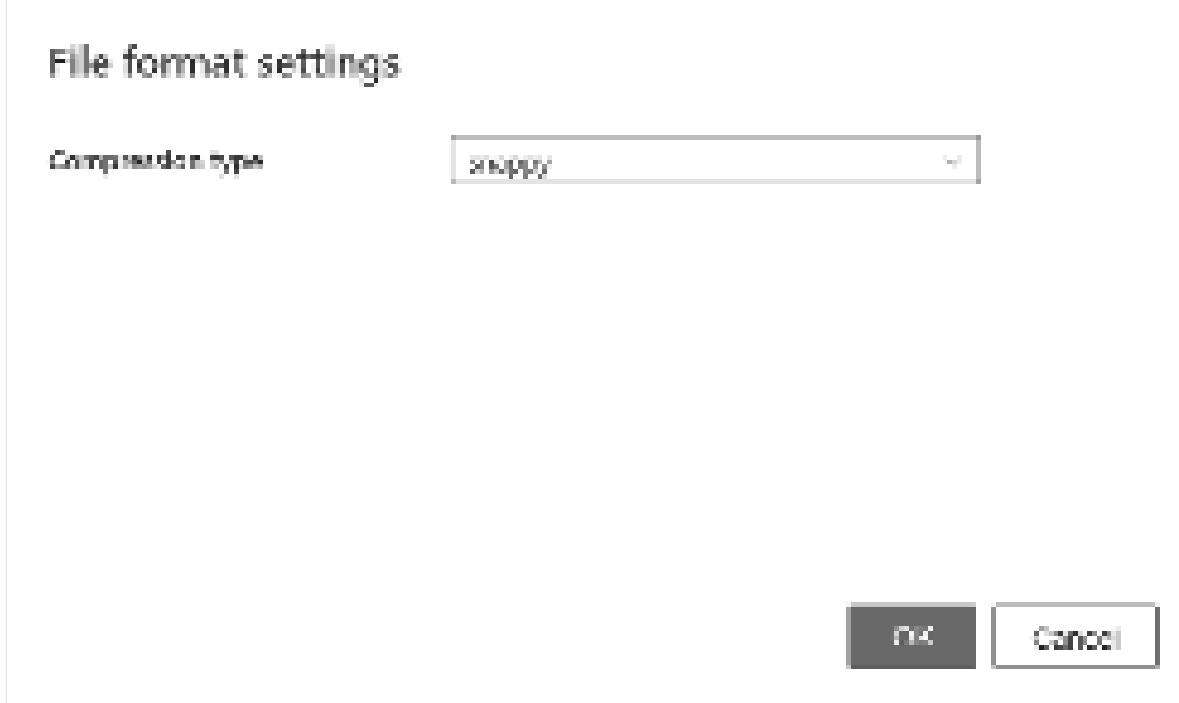
After you select **Settings** in the **File format** section, the following properties are shown in the pop-up **File format settings** dialog box.



- **Compression type:** Choose the compression codec used to read ORC files in the drop-down list. You can choose from **None**, **zlib** or **snappy**.

ORC format as destination

After you select **Settings**, the following properties are shown in the pop-up **File format settings** dialog box.



- **Compression type:** Choose the compression codec used to write ORC files in the drop-down list. You can choose from **None**, **zlib** or **snappy**.

Under **Advanced** settings in the **Destination** tab, the following ORC format related properties are displayed.

- **Max rows per file:** When writing data into a folder, you can choose to write to multiple files and specify the maximum rows per file. Specify the maximum rows that you want to write per file.
- **File name prefix:** Applicable when **Max rows per file** is configured. Specify the file name prefix when writing data to multiple files, resulted in this pattern:
`<fileNamePrefix>_00000.<fileExtension>`. If not specified, the file name prefix is auto generated. This property doesn't apply when the source is a file based store or a partition option enabled data store.

Table summary

ORC as source

The following properties are supported in the copy activity **Source** section when using ORC format.

Name	Description	Value	Required	JSON script property
File format	The file format that you want to use.	ORC	Yes	type (<i>under datasetSettings</i>):

Name	Description	Value	Required	JSON script property
		Orc		
Compression type	The compression codec used to read ORC files.	None zlib snappy	No	orcCompressionCodec: none zlib snappy

ORC as destination

The following properties are supported in the copy activity **Destination** section when using the ORC format.

Name	Description	Value	Required	JSON script property
File format	The file format that you want to use.	ORC	Yes	type (<i>under datasetSettings</i>): Orc
Compression type	The compression codec used to write ORC files.	None zlib snappy	No	orcCompressionCodec: none zlib snappy
Max rows per file	When writing data into a folder, you can choose to write to multiple files and specify the maximum rows per file. Specify the maximum rows that you want to write per file.	<your max rows per file>	No	maxRowsPerFile
File name prefix	Applicable when Max rows per file is configured. Specify the file name prefix when writing data to multiple files, resulted in this pattern: <fileNamePrefix>_0000.<fileExtension>. If not specified, the file name prefix is auto generated. This property doesn't apply when the source is a file based store or a partition option enabled data store.	<your file name prefix>	No	fileNamePrefix

Next steps

- Connectors overview

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Parquet format in Data Factory in Microsoft Fabric

Article • 11/15/2023

This article outlines how to configure Parquet format in the data pipeline of Data Factory in Microsoft Fabric.

Supported capabilities

Parquet format is supported for the following activities and connectors as a source and destination.

Category	Connector/Activity
Supported connector	Amazon S3
	Azure Blob Storage
	Azure Data Lake Storage Gen1
	Azure Data Lake Storage Gen2
	Google Cloud Storage
	HTTP
Supported activity	Copy activity
	Lookup activity
	GetMetadata activity
	Delete activity

Parquet format in copy activity

To configure Parquet format, choose your connection in the source or destination of data pipeline copy activity, and then select **Parquet** in the drop-down list of **File format**. Select **Settings** for further configuration of this format.



Parquet format as source

After you select **Settings** in the **File format** section, the following properties are shown in the pop-up **File format settings** dialog box.



- **Compression type:** Choose the compression codec used to read Parquet files in the drop-down list. You can choose from **None**, **gzip (.gz)**, **snappy**, **Izo**, **Brotli (.br)**, **Zstandard**, **Iz4**, **Iz4frame**, **bzip2 (.bz2)**, or **Iz4hadoop**.

Parquet format as destination

After you select **Settings**, the following properties are shown in the pop-up **File format settings** dialog box.



- **Compression type:** Choose the compression codec used to write Parquet files in the drop-down list. You can choose from **None**, **gzip (.gz)**, **snappy**, **Izo**, **Brotli (.br)**, **Zstandard**, **Iz4**, **Iz4frame**, **bzip2 (.bz2)**, or **Iz4hadoop**.
- **Use V-Order:** Enable a write time optimization to the parquet file format. For more information, see Delta Lake table optimization and V-Order. It is enabled by default.

Under **Advanced** settings in the **Destination** tab, the following Parquet format related properties are displayed.

- **Max rows per file:** When writing data into a folder, you can choose to write to multiple files and specify the maximum rows per file. Specify the maximum rows that you want to write per file.
- **File name prefix:** Applicable when **Max rows per file** is configured. Specify the file name prefix when writing data to multiple files, resulted in this pattern:
`<fileNamePrefix>_0000.<fileExtension>`. If not specified, the file name prefix is auto generated. This property doesn't apply when the source is a file based store or a partition option enabled data store.

Table summary

Parquet as source

The following properties are supported in the copy activity **Source** section when using the Parquet format.

Name	Description	Value	Required	JSON script property
File format	The file format that you want to use.	Parquet	Yes	type (<i>under datasetSettings</i>): Parquet

Name	Description	Value	Required	JSON script property
Compression type	The compression codec used to read Parquet files.	Choose from: None gzip (.gz) snappy lzo Brotli (.br) Zstandard lz4 lz4frame bzip2 (.bz2) lz4hadoop	No	compressionCodec:

Parquet as destination

The following properties are supported in the copy activity **Destination** section when using the Parquet format.

Name	Description	Value	Required	JSON script property
File format	The file format that you want to use.	Parquet	Yes	type (<i>under datasetSettings</i>): Parquet
Use V-Order	A write time optimization to the parquet file format.	selected or unselected	No	enableVertiParquet
Compression type	The compression codec used to write Parquet files.	Choose from: None gzip (.gz) snappy lzo Brotli (.br) Zstandard lz4 lz4frame bzip2 (.bz2) lz4hadoop	No	compressionCodec:
Max rows per file	When writing data into a folder, you can choose to write to multiple files and specify the	<your max rows per file>	No	maxRowsPerFile

Name	Description	Value	Required	JSON script property
	maximum rows per file. Specify the maximum rows that you want to write per file.			
File name prefix	Applicable when Max rows per file is configured. Specify the file name prefix when writing data to multiple files, resulted in this pattern: <code><fileNamePrefix>_00000.<fileExtension></code> . If not specified, the file name prefix is auto generated. This property doesn't apply when the source is a file based store or a partition option enabled data store.	<your file name prefix>	No	fileNamePrefix

Next steps

- Connect to Parquet files in dataflows
- Connectors overview

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Connect to Parquet files in dataflows

Article • 11/15/2023

You can connect to Parquet files in Dataflow Gen2 using the Parquet connector provided by Data Factory in Microsoft Fabric.

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Parquet files. The following links provide the specific Power Query connector information you need to connect to Parquet files in Dataflow Gen2:

- To get started using the Parquet connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- To connect to the Parquet connector from Power Query, go to [Connect to a Parquet file from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Feedback

Was this page helpful?



[Provide product feedback](#) | [Ask the community](#)

XML format in Data Factory in Microsoft Fabric

Article • 11/15/2023

This article outlines how to configure XML format in the data pipeline of Data Factory in Microsoft Fabric.

Supported capabilities

XML format is supported for the following activities and connectors as source.

Category	Connector/Activity
Supported connector	Amazon S3
	Azure Blob Storage
	Azure Data Lake Storage Gen1
	Azure Data Lake Storage Gen2
	FTP
	Google Cloud Storage
	HTTP
	SFTP
Supported activity	Copy activity
	Lookup activity
	GetMetadata activity
	Delete activity

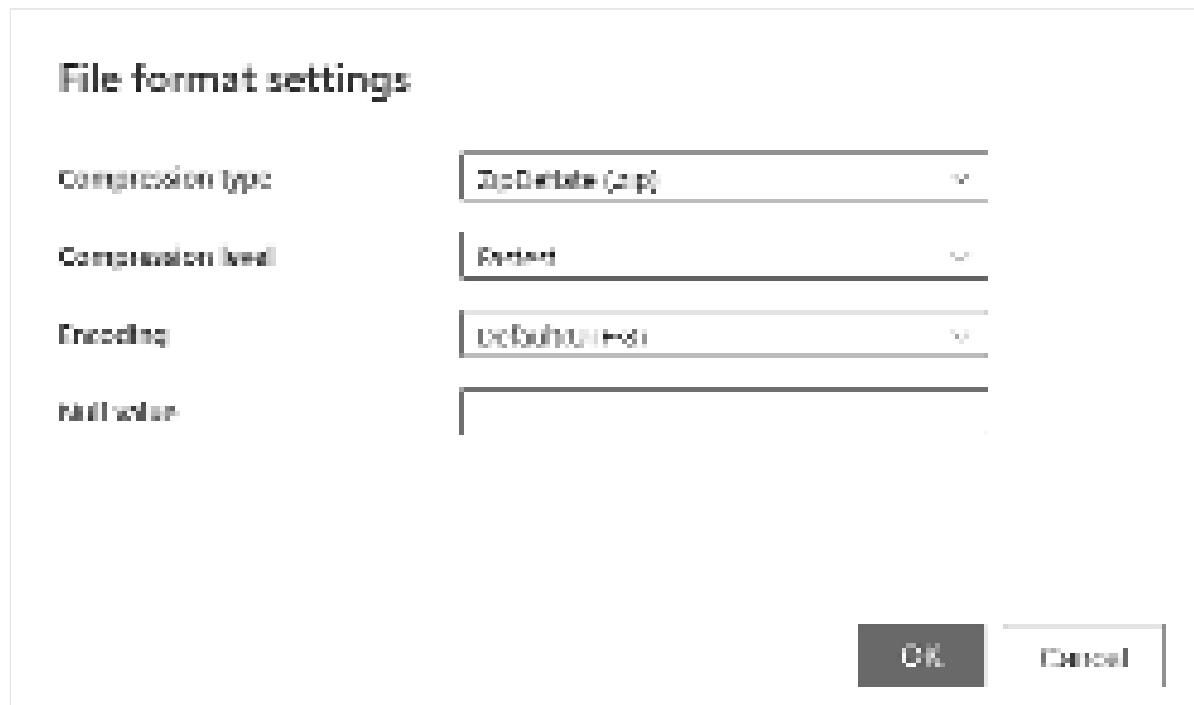
XML format in copy activity

To configure XML format, choose your connection in the source of data pipeline copy activity, and then select **XML** in the drop-down list of **File format**. Select **Settings** for further configuration of this format.



XML as source

After you select **Settings** in the **File format** section, the following properties are shown in the pop-up **File format settings** dialog box.



- **Compression type:** The compression codec used to read XML files. You can choose from **None**, **bzip2**, **gzip**, **deflate**, **ZipDeflate**, **TarGZip** or **tar** type in the drop-down list.

If you select **ZipDeflate** as the compression type, **Preserve zip file name as folder** is displayed under the **Advanced** settings in the **Source** tab.

- **Preserve zip file name as folder:** Indicates whether to preserve the source zip file name as a folder structure during copy.

- If this box is checked (default), the service writes unzipped files to `<specified file path>/<folder named as source zip file>/`.
- If this box is unchecked, the service writes unzipped files directly to `<specified file path>`. Make sure you don't have duplicated file names in different source zip files to avoid racing or unexpected behavior.

If you select **TarGZip/tar** as the compression type, **Preserve compression file name as folder** is displayed under the **Advanced** settings in the **Source** tab.

- **Preserve compression file name as folder:** Indicates whether to preserve the source compressed file name as a folder structure during copy.
 - If this box is checked (default), the service writes decompressed files to `<specified file path>/<folder named as source compressed file>/`.
 - If this box is unchecked, the service writes decompressed files directly to `<specified file path>`. Make sure you don't have duplicated file names in different source files to avoid racing or unexpected behavior.
- **Compression level:** Specify the compression ratio when you select a compression type. You can choose from **Fastest** or **Optimal**.
 - **Fastest:** The compression operation should complete as quickly as possible, even if the resulting file is not optimally compressed.
 - **Optimal:** The compression operation should be optimally compressed, even if the operation takes a longer time to complete. For more information, see Compression Level topic.
- **Encoding:** Specify the encoding type used to write test files. Select one type from the drop-down list. The default value is **UTF-8**.
- **Null value:** Specifies the string representation of null value. The default value is empty string.

Under **Advanced** settings in the **Source** tab, the following XML format related properties are displayed.

- **Validation mode:** Specifies whether to validate the XML schema. Select one mode from the drop-down list.
 - **None:** Select this to not use validation mode.
 - **xsd:** Select this to validate the XML schema using XSD.
 - **dtd:** Select this to validate the XML schema using DTD.

Namespace	Prefix
XML	xml
XML	xml

- Namespaces:** Specify whether to enable namespace when parsing the XML files. It is selected by default.
- Namespace prefix pairs:** If the Namespaces is enabled, selecting + New and specify the **URL** and **Prefix**. You can add more pairs by selecting + New. Namespace URI to prefix mapping is used to name fields when parsing the XML file. If an XML file has namespace and namespace is enabled, by default, the field name is the same as it is in the XML document. If there is an item defined for the namespace URI in this map, the field name is `prefix:fieldName`.

URL	Prefix
http://www.w3.org/2001/XMLSchema	xml
http://www.w3.org/2001/XMLSchema	xml

- Detect data type:** Specify whether to detect integer, double, and Boolean data types. It is selected by default.

Table summary

XML as source

The following properties are supported in the copy activity **Source** section when using XML format.

Name	Description	Value	Required	JSON script property
File format	The file format that you want to use.	XML	Yes	type (<i>under datasetSettings</i>): Xml
Compression type	The compression codec used to read XML files.	None bzip2 gzip deflate ZipDeflate	No	type (<i>under compression</i>): bzip2 gzip deflate

Name	Description	Value	Required	JSON script property
		TarGZip tar		ZipDeflate TarGZip tar
Compression level	The compression ratio.	Fastest Optimal	No	level (<i>under compression</i>): Fastest Optimal
Encoding	The encoding type used to read test files.	"UTF-8" (by default), "UTF-8 without BOM", "UTF-16LE", "UTF-16BE", "UTF-32LE", "UTF-32BE", "US-ASCII", "UTF-7", "BIG5", "EUC-JP", "EUC-KR", "GB2312", "GB18030", "JOHAB", "SHIFT-JIS", "CP875", "CP866", "IBM00858", "IBM037", "IBM273", "IBM437", "IBM500", "IBM737", "IBM775", "IBM850", "IBM852", "IBM855", "IBM857", "IBM860", "IBM861", "IBM863", "IBM864", "IBM865", "IBM869", "IBM870", "IBM01140", "IBM01141", "IBM01142", "IBM01143",	No	encodingName

Name	Description	Value	Required	JSON script property
		"IBM01144", "IBM01145", "IBM01146", "IBM01147", "IBM01148", "IBM01149", "ISO-2022- JP", "ISO- 2022-KR", "ISO-8859-1", "ISO-8859-2", "ISO-8859-3", "ISO-8859-4", "ISO-8859-5", "ISO-8859-6", "ISO-8859-7", "ISO-8859-8", "ISO-8859-9", "ISO-8859- 13", "ISO- 8859-15", "WINDOWS- 874", "WINDOWS- 1250", "WINDOWS- 1251", "WINDOWS- 1252", "WINDOWS- 1253", "WINDOWS- 1254", "WINDOWS- 1255", "WINDOWS- 1256", "WINDOWS- 1257", "WINDOWS- 1258"		
Preserve zip file name as folder	Indicates whether to preserve the source zip file name as a folder	Selected (default) or unselect	No	preserveZipFileNameAsFolder <i>(under compressionProperties -> type as ZipDeflateReadSettings):</i> true (default) or false

Name	Description	Value	Required	JSON script property
	structure during copy.			
Preserve compression file name as folder	Indicates whether to preserve the source compressed file name as a folder structure during copy.	Selected (default) or unselect	No	preserveCompressionFileNameAsFolder (under <code>compressionProperties -> type</code> as <code>TarGZipReadSettings</code> or <code>TarReadSettings</code>): true (default) or false
Null value	The string representation of null value.	<your null value> empty string (by default)	No	nullValue
Validation mode	Whether to validate the XML schema.	None xsd dtd	No	validationMode: xsd dtd
Namespaces	Whether to enable namespace when parsing the XML files.	Selected (default) or unselected	No	namespaces: true (default) or false
Namespace prefix pairs	Namespace URI to prefix mapping, which is used to name fields when parsing the XML file. If an XML file has namespace and namespace is enabled, by default, the field name is the same as it is in the XML document. If there is an item defined for the namespace URI in this map, the field name is <code>prefix:fieldName</code> .	< url >:< prefix >	No	namespacePrefixes: < url >:< prefix >

Name	Description	Value	Required	JSON script property
Detect data type	Whether to detect integer, double, and Boolean data types.	Selected (default) or unselected	No	detectDataType: true (default) or false

Next steps

[Connectors overview](#)

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#) | [Ask the community](#)

FTP connector overview

Article • 11/15/2023

The FTP connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

Data Factory in Microsoft Fabric doesn't currently support FTP in Dataflow Gen2.

Support in data pipelines

The FTP connector supports the following capabilities in data pipelines:

Supported capabilities	Gateway	Authentication
Copy activity (source/-)	None	Anonymous Basic
Lookup activity	None	Anonymous Basic
GetMetadata activity	None	Anonymous Basic
Delete activity	None	Anonymous Basic

To learn more about the copy activity configuration for FTP in data pipelines, go to [Configure in a data pipeline copy activity](#).

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Configure FTP in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in data pipeline to copy data from FTP.

Supported format

FTP supports the following file formats. Refer to each article for format-based settings.

- Avro format
- Binary format
- Delimited text format
- Excel format
- JSON format
- ORC format
- Parquet format
- XML format

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

Go to **Source** tab to configure your copy activity source. See the following content for the detailed configuration.



The following three properties are required:

- **Data store type:** Select **External**.
- **Connection:** Select an FTP connection from the connection list. If no connection exists, then create a new FTP connection by selecting **New**.
- **File path type:** Select from **File path**, **Wildcard file path** and **List of files** based on the way that you want to read files.
 - **File path:** If you choose this type, specify your source file path. You can select **Browse** to select your source files or enter your file path manually.
 - **Wildcard file path:** If you choose this type, specify the **Wildcard paths** to filter your source folders or files.

Allowed wildcards are ***** (matches zero or more characters) and **?** (matches zero or single character). Use **** to escape if your folder name has a wildcard or this escape character inside. For more examples, go to [Folder and file filter examples](#).



Wildcard folder path: Specify the folder path with wildcard characters to filter source folders.

Wildcard file name: Specify the file name with wildcard characters under the given folderPath/wildcard folder path to filter source files.

- **List of files:** If you select this type, specify the **Folder path** and **Path to file list** to indicates to copy a given file set. Point to a text file that includes a list of files you want to copy, one file per line. For more examples, go to File list examples.

Folder path: Specify the path to your source folder. It is required.

Path to file list: Specify the path of the text file that includes a list of files you want to copy.



- **File format:** Select the file format applied from the drop-down list. Select **Settings** to configure the file format. For settings of different file formats, refer to articles in Supported format for detailed information.

Under **Advanced**, you can specify the following fields:

- **Filter by last modified:** Files are filtered based on the last modified dates. This property doesn't apply when you configure your file path type as List of files.
 - **Start time (UTC):** The files are selected if their last modified time is greater than or equal to the configured time.
 - **End time (UTC):** The files are selected if their last modified time is less than the configured time.

When **Start time (UTC)** has datetime value but **End time (UTC)** is NULL, it means the files whose last modified attribute is greater than or equal with the datetime value will be selected. When **End time (UTC)** has datetime value but **Start time (UTC)** is NULL, it means the files whose last modified attribute is less than the datetime value will be selected. The properties can be NULL, which means no file attribute filter will be applied to the data.

- **Disable chunking:** The chunking is designed to optimize the performance and happens underneath. This option allows you to disable chunking within each file. When copying data from FTP, the service tries to get the file length first, then divide the file into multiple parts and read them in parallel. Specify whether your FTP server supports getting file length or seeking to read from a certain offset. It is unselected by default.
- **Enable partition discovery:** Specify whether to parse the partitions from the file path and add them as additional source columns. It is unselected by default and not supported when you use binary file format.

- **Partition root path:** When partition discovery is enabled, specify the absolute root path in order to read partitioned folders as data columns. If it is not specified, by default,
 - When you use file path or list of files on source, partition root path is the path that you configured.
 - When you use wildcard folder filter, partition root path is the sub-path before the first wildcard.

For example, assuming you configure the path as

`root/folder/year=2020/month=08/day=27`:

- If you specify partition root path as `root/folder/year=2020`, copy activity will generate two more columns month and day with value "08" and "27" respectively, in addition to the columns inside the files.
- If partition root path is not specified, no extra column will be generated.



- **Use binary transfer:** Specify whether to use the binary transfer mode. Select it to use binary mode (default) or unselect it to use ASCII.
- **Max concurrent connections:** This property indicates the upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.
- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter. For more information, go to Add additional columns during copy.

Mapping

For **Mapping** tab configuration, see Configure your mappings under mapping tab. If you choose Binary as your file format, mapping will not be supported.

Settings

For **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

The following table contains more information about the copy activity in FTP.

Source

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your FTP connection to the source data store.	< your FTP connection >	Yes	connection
File path type	The file path type used to get source data.	<ul style="list-style-type: none"> • File path • Wildcard file path • List of files 	Yes	/
File path	The path to the source file.	< file path>	Yes	fileName folderpath
Wildcard paths	The wildcard path to the source file.	< your wildcard file path >	Yes for Wildcard file name	wildcardFolderPath wildcardFileName
Folder path	The path to your source folder.	< common home folder path>	Yes	folderPath
Path to file list	Indicates to copy a given file set. Point to a text file that includes a list of files you want to copy, one file per line.	< file list path >	No	fileListPath
File format	The file format for your source data. For the information of different file formats, refer to articles in Supported format for detailed information.	/	Yes	/
Filter by last modified	The files with last modified time in the range [Start time, End time) will be filtered for further processing. The time will be applied to UTC time zone in the	datetime	No	modifiedDatetimeStart modifiedDatetimeEnd

Name	Description	Value	Required	JSON script property
	<p>format of <code>yyyy-mm-ddThh:mm:ss.ffffz</code>. These properties can be skipped which means no file attribute filter will be applied. This property doesn't apply when you configure your file path type as List of files.</p>			
Disable chunking	<p>The chunking is designed to optimize the performance and happens underneath. This option allows you to disable chunking within each file. When copying data from FTP, the service tries to get the file length first, then divide the file into multiple parts and read them in parallel. Specify whether your FTP server supports getting file length or seeking to read from a certain offset.</p>	selected or unselected (default)	No	disableChunking: true or false (default)
Enable partition discovery	<p>Indicates whether to parse the partitions from the file path and add them as additional source columns.</p>	selected or unselected (default)	No	enablePartitionDiscovery: true or false (default)
Partition root path	<p>The absolute partition root path in order to read partitioned folders as data columns. Specify it when partition discovery is enabled.</p>	< partition root path >	No	partitionRootPath
Use binary transfer	<p>Indicates whether to use the binary transfer mode. The values are true for binary mode (default), and false for ASCII.</p>	selected (default) or unselected	No	useBinaryTransfer: true (default) or false

Name	Description	Value	Required	JSON script property
Max concurrent connections	The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.	< upper limit of concurrent connections > (integer)	No	maxConcurrentConnections
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter. For more information, go to Add additional columns during copy	• Name • Value	No	additionalColumns: • name • value

Next steps

- FTP connector overview
-

Feedback

Was this page helpful?



Provide product feedback [↗](#) | Ask the community [↗](#)

Google Analytics connector overview

Article • 11/15/2023

The Google Analytics connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to Google Analytics in Dataflow Gen2, go to Set up your Google Analytics connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support the Google Analytics connector in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your Google Analytics connection

Article • 11/15/2023

This article outlines the steps to create a Google Analytics connection.

Supported authentication types

The Google Analytics connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Organizational account	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Google Analytics data. The following links provide the specific Power Query connector information you need to connect to Google Analytics data in Dataflow Gen2:

- To get started using the Google Analytics connector in Dataflow Gen2, go to Get data from Data Factory in Microsoft Fabric (Preview).
- Be sure to install or set up any Google Analytics prerequisites before connecting to the Google Analytics connector.
- To connect to the Google Analytics connector from Power Query, go to Connect to Google Analytics data from Power Query Online.

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support Google Analytics data in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Google BigQuery connector overview

Article • 11/15/2023

The Google BigQuery connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to Google BigQuery data in Dataflow Gen2, go to Set up your Google BigQuery connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support the Google BigQuery connector in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your Google BigQuery connection

Article • 11/15/2023

This article outlines the steps to create a Google BigQuery connection.

Supported authentication types

The Google BigQuery connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Service Account Login	n/a	✓
Organizational account	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Google BigQuery data. The following links provide the specific Power Query connector information you need to connect to Google BigQuery data in Dataflow Gen2:

- To get started using the Google BigQuery connector in Dataflow Gen2, go to Get data from Data Factory in Microsoft Fabric (Preview).
- Be sure to install or set up any Google BigQuery prerequisites before connecting to the Google BigQuery connector.
- To connect to the Google BigQuery connector from Power Query, go to Connect to Google BigQuery data from Power Query Online.

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support Google BigQuery data in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Google Cloud Storage connector Overview

Article • 11/15/2023

This Google Cloud Storage connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

Data Factory in Microsoft Fabric doesn't currently support Google Cloud Storage in Dataflow Gen2.

Support in data pipelines

The Google Cloud Storage connector supports the following capabilities in data pipelines:

Supported capabilities	Gateway	Authentication
Copy activity (source/-)	None	Basic
Lookup activity	None	Basic
GetMetadata activity	None	Basic
Delete activity	None	Basic

To learn about how to connect to Google Cloud Storage data in data pipelines, go to Set up your Google Cloud Storage connection.

To learn about the copy activity configuration for Google Cloud Storage in data pipelines, go to Configure Google Cloud Storage in a copy activity.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your Google Cloud Storage connection

Article • 11/15/2023

This article outlines the steps to create a Google Cloud Storage connection.

Supported authentication types

This Google Cloud Storage connector supports the following authentication types for copy activity.

Authentication type	Copy	Dataflow Gen2
Basic	✓	n/a

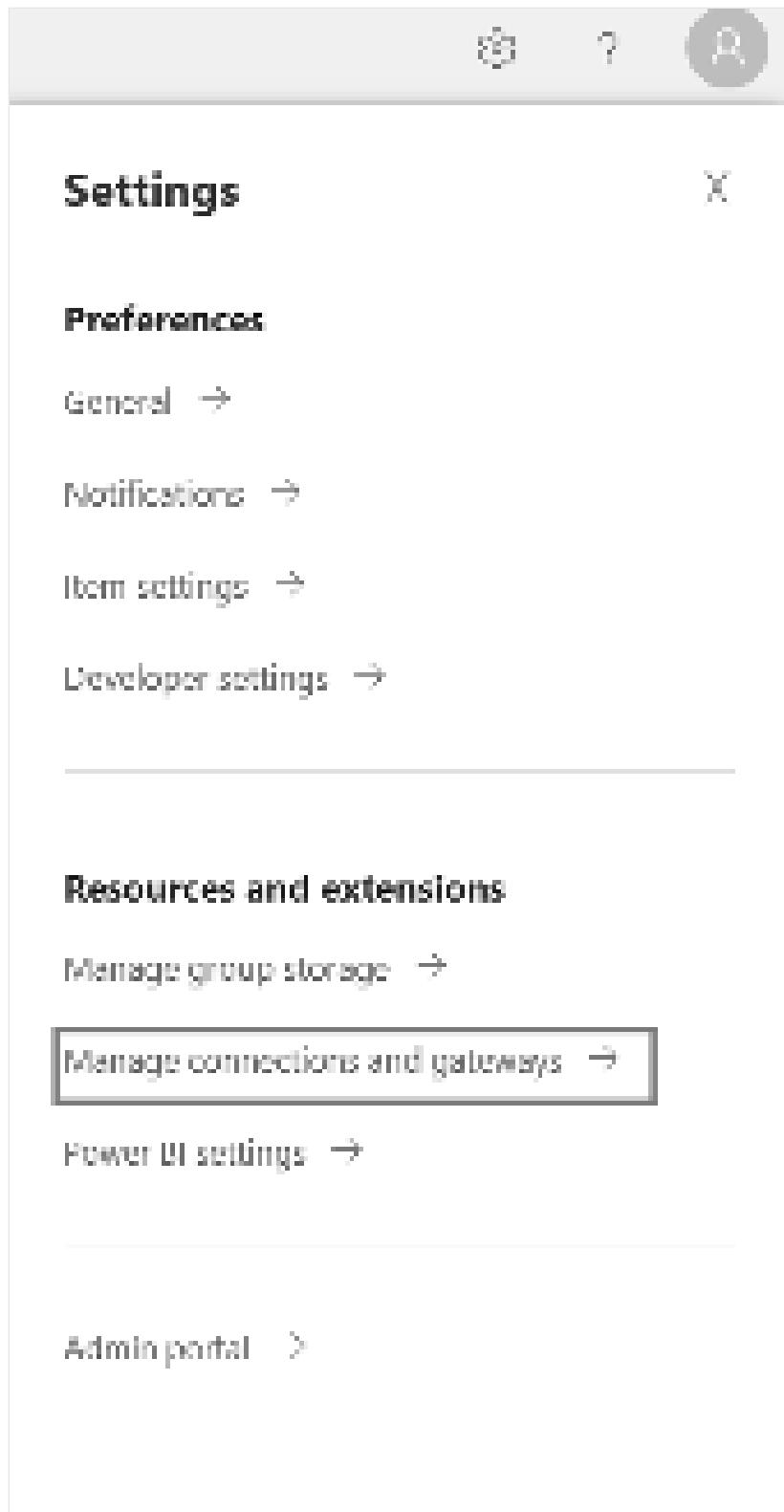
Set up your connection in Dataflow Gen2

The Google Cloud Storage connector isn't currently supported in Dataflow Gen2.

Set up your connection in a data pipeline

To create a connection in a data pipeline:

1. From the page header in the Data Factory service, select **Settings**  > **Manage connections and gateways**



2. Select **New** at the top of the ribbon to add a new data source.

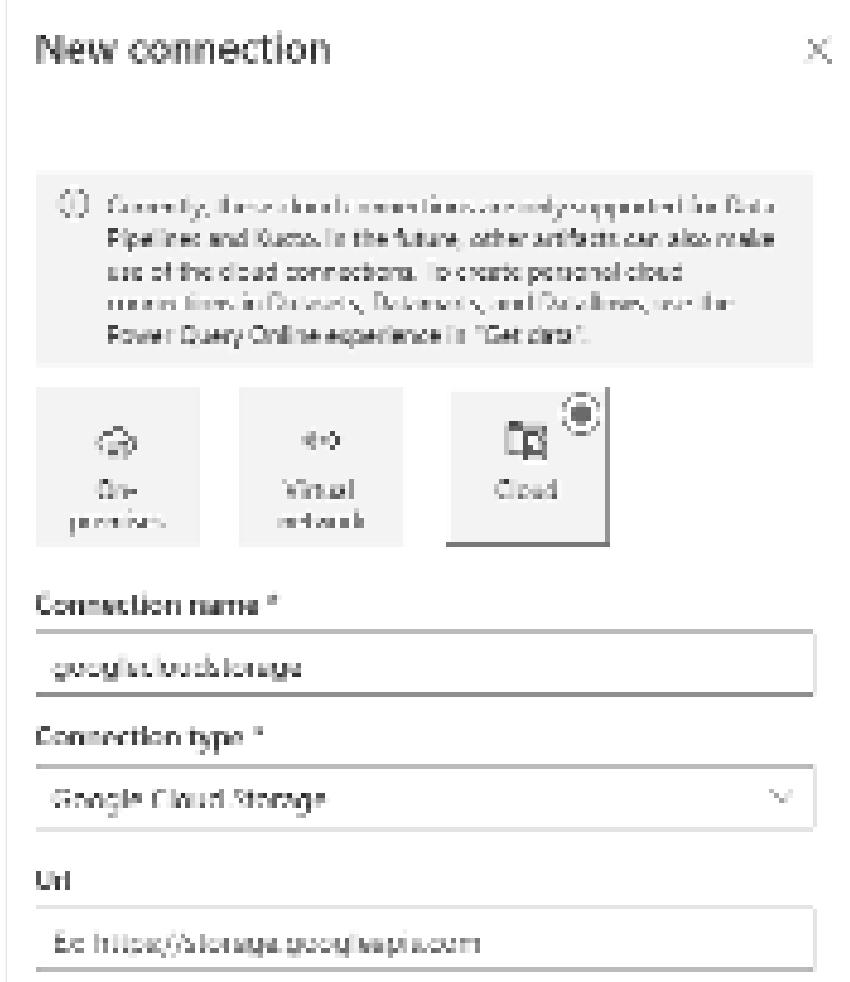


The New connection pane now appears on the left side of the page.



Setup connection

Step 1: Specify the new connection name, type, and URL



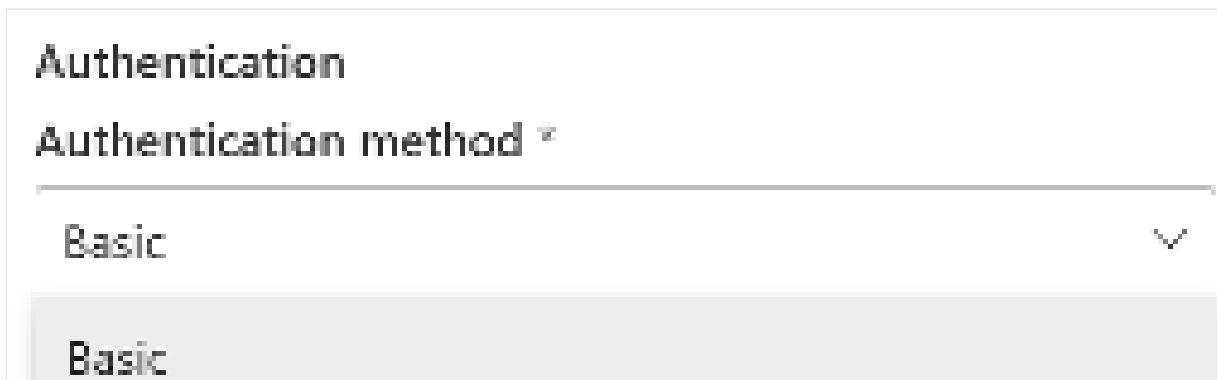
In the **New connection** pane, choose **Cloud**, and specify the following fields:

- **Connection name:** Specify a name for your connection.
- **Connection type:** Select **Google Cloud Storage** for your connection type.
- **Url:** Specify the custom GCS endpoint as `https://storage.googleapis.com`.

Step 2: Select and set your authentication

Under **Authentication method**, select your authentication from the drop-down list and complete the related configuration. This Google Cloud Storage connector supports the following authentication types.

- Basic



Basic authentication

- **Username:** ID of the secret access key. To find the access key and secret, go to Prerequisites.
 - **Password:** The secret access key itself.

Authentication

Authentication method *

Basic 

Username *

< Your username >

Password *

< Your password >

Skip test connection

Step 3: Specify the privacy level that you want to apply

In the **General** tab, select the privacy level that you want apply in the **Privacy level** drop-down list. Three privacy levels are supported. For more information, go to [privacy levels](#).

Step 4: Create your connection

Select **Create**. Your creation is successfully tested and saved if all the credentials are correct. If not correct, the creation fails with errors.

Digitized by srujanika@gmail.com

Table summary

The following table contains connector properties that are supported in pipeline copy.

Name	Description	Required	Property	Copy
Connection name	A name for your connection.	Yes		✓
Connection type	Select a type for your connection. Select Google Cloud Storage .	Yes		✓
Url	The base Url to the Google Cloud Storage service.	Yes		✓
Authentication	Go to Authentication	Yes		Go to Authentication

Authentication

The following properties in the table are the supported authentication types.

Name	Description	Required	Property	Copy
Basic				✓
- Username	ID of the secret access key. To find the access key and secret, go to Prerequisites.	Yes		
- Password	The secret access key itself.	Yes		

Next steps

- Configure Google Cloud Storage in copy activity

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Configure Google Cloud Storage in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in data pipeline to copy data from and to Google Cloud Storage.

Prerequisites

The following setup is required on your Google Cloud Storage account:

1. Enable interoperability for your Google Cloud Storage account.
2. Set the default project that contains the data you want to copy from the target GCS bucket.
3. Create a service account and define the right levels of permissions by using Cloud IAM on GCP.
4. Generate the access keys for this service account.



Required permissions

To copy data from Google Cloud Storage, make sure you've been granted the following permissions for object operations: `storage.objects.get` and `storage.objects.list`.

In addition, `storage.buckets.list` permission is required for operations like testing connection and browsing from root.

For the full list of Google Cloud Storage roles and associated permissions, go to IAM roles for Cloud Storage [↗](#) on the Google Cloud site.

Supported format

Google Cloud Storage supports the following file formats. Refer to each article for format-based settings.

- Avro format
- Binary format
- Delimited text format
- Excel format
- JSON format
- ORC format
- Parquet format
- XML format

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Mapping
- Settings

General

For **General** tab configuration, go to General.

Source

The following properties are supported for Google Cloud Storage under the **Source** tab of a copy activity.



The following properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select a **Google Cloud Storage** connection from the connection list. If no connection exists, then create a new Google Cloud Storage connection by selecting **New**.
- **File path:** Select **Browse** to choose the file that you want to copy, or fill in the path manually.
- **File settings:** Select **File settings** to configure the file format. For settings of different file formats, refer to Supported format for detailed information.

Under **Advanced**, you can specify the following fields:

- **File path type:** You can choose **File path**, **Prefix**, **Wildcard file path**, or **List of files** as your file path type. The configuration of each of these settings is:
 - **File path:** If you choose this type, the data can be copied from the given bucket or folder/file path specified in **File path**.
 - **Prefix:** Prefix for the GCS key name under the given bucket configured to filter source GCS files. GCS keys whose names start with `given_bucket/this_prefix` are selected. It utilizes GCS's service-side filter, which provides better performance than a wildcard filter.



- **Wildcard file path:** Specify the folder or file path with wildcard characters under your given bucket to filter your source folders or files.

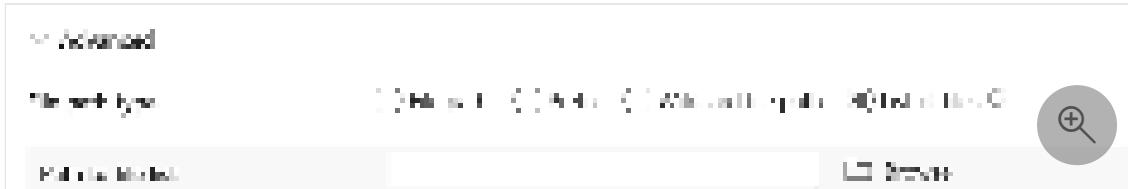
Allowed wildcards are: * (matches zero or more characters) and ? (matches zero or single character). Use ^ to escape if your folder name has wildcard or this escape character inside. For more examples, go to [Folder and file filter examples](#).

- Wildcard folder path: Specify the folder path with wildcard characters under the given bucket to filter source folders.



- Wildcard file name: Specify the file name with wildcard characters under the given bucket and folder path (or wildcard folder path) to filter source files.
- List of files:** Indicates to copy a given file set. Point to a text file that includes a list of files you want to copy, one file per line, which is the relative path to the path configured in **File path**.

When you're using this option, don't specify a file name. For more examples, go to [File list examples](#).



- Recursively:** Indicates whether the data is read recursively from the subfolders or only from the specified folder. Note that when this checkbox is selected, and the destination is a file-based store, an empty folder or subfolder isn't copied or created at the destination.
- Delete files after completion:** Indicates whether the binary files are deleted from the source store after successfully moving to the destination store. The file deletion is per file, so when a copy activity fails, you'll note that some files have already been copied to the destination and deleted from the source, while others are still remaining on source store. This property is only valid in the binary files copy scenario.
- Max concurrent connection:** The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.

Mapping

For **Mapping** tab configuration, see [Configure your mappings under mapping tab](#). If you choose Binary as your file format, mapping will not be supported.

Settings

For the **Settings** tab configuration, go to [Configure your other settings under settings tab](#).

Table summary

The following tables contain more information about the copy activity in Google Cloud Storage.

Source information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	<your connection>	Yes	connection
File path	If you choose this type, the data can be copied from the given bucket or folder/file path specified in File path .	Yes		container fileName
File path type	The file path type that you want to use.	<ul style="list-style-type: none">• File path• Prefix• Wildcard folder path• List of files	No	<ul style="list-style-type: none">• prefix• wildcardFolderPath, wildcardFileName• path to file list
Recursively	Indicates whether the data is read recursively from the subfolders or only from the specified folder. Note that when this checkbox is selected, and the destination is a file-based store, an empty	Selected or unselect	No	recursive

Name	Description	Value	Required	JSON script property
	folder or subfolder isn't copied or created at the destination.			
Delete files after completion	Indicates whether the binary files will be deleted from the source store after successfully moving to the destination store. The file deletion is per file, so when copy activity fails, you'll note some files have already been copied to the destination and deleted from the source, while others are still remaining on the source store. This property is only valid in binary files copy scenario.	Selected or unselect	No	deleteFilesAfterCompletion
Max concurrent connection	The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.	<max concurrent connections>	No	maxConcurrentConnections

Next steps

- Set up your Google Cloud Storage connection

Feedback

Was this page helpful?

 Yes

 No

Hive LLAP connector overview

Article • 11/15/2023

The Hive LLAP connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to Hive LLAP data in Dataflow Gen2, go to Set up your Hive LLAP connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support Hive LLAP data in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your Hive LLAP connection

Article • 12/06/2023

This article outlines the steps to create a Hive LLAP connection.

Supported authentication types

The Hive LLAP connector supports the following authentication types for copy and Dataflow Gen2 respectively.

 Expand table

Authentication type	Copy	Dataflow Gen2
Windows	n/a	√
Basic	n/a	√

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Hive LLAP data. The following links provide the specific Power Query connector information you need to connect to Hive LLAP data in Dataflow Gen2:

- To get started using the Hive LLAP connector in Dataflow Gen2, go to Get data from Data Factory in Microsoft Fabric.
- Be sure to install or set up any Hive LLAP prerequisites before connecting to the Hive LLAP connector.
- To connect to the Hive LLAP connector from Power Query, go to Connect to Hive LLAP data from Power Query Online.

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support Hive LLAP data in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

HTTP connector overview

Article • 11/15/2023

The HTTP connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

Data Factory in Microsoft Fabric doesn't currently support the HTTP connector in Dataflow Gen2.

Support in data pipelines

The Google Cloud Storage connector supports the following capabilities in data pipelines:

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	Basic
Lookup activity	None	Basic

To learn about how to connect to HTTP data in data pipelines, go to Set up your HTTP connection.

To learn about the copy activity configuration for HTTP in data pipelines, go to Configure HTTP in a copy activity.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your HTTP connection

Article • 11/15/2023

This article outlines the steps to create HTTP connection.

Supported authentication types

The HTTP connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Anonymous	✓	n/a
Basic	✓	n/a

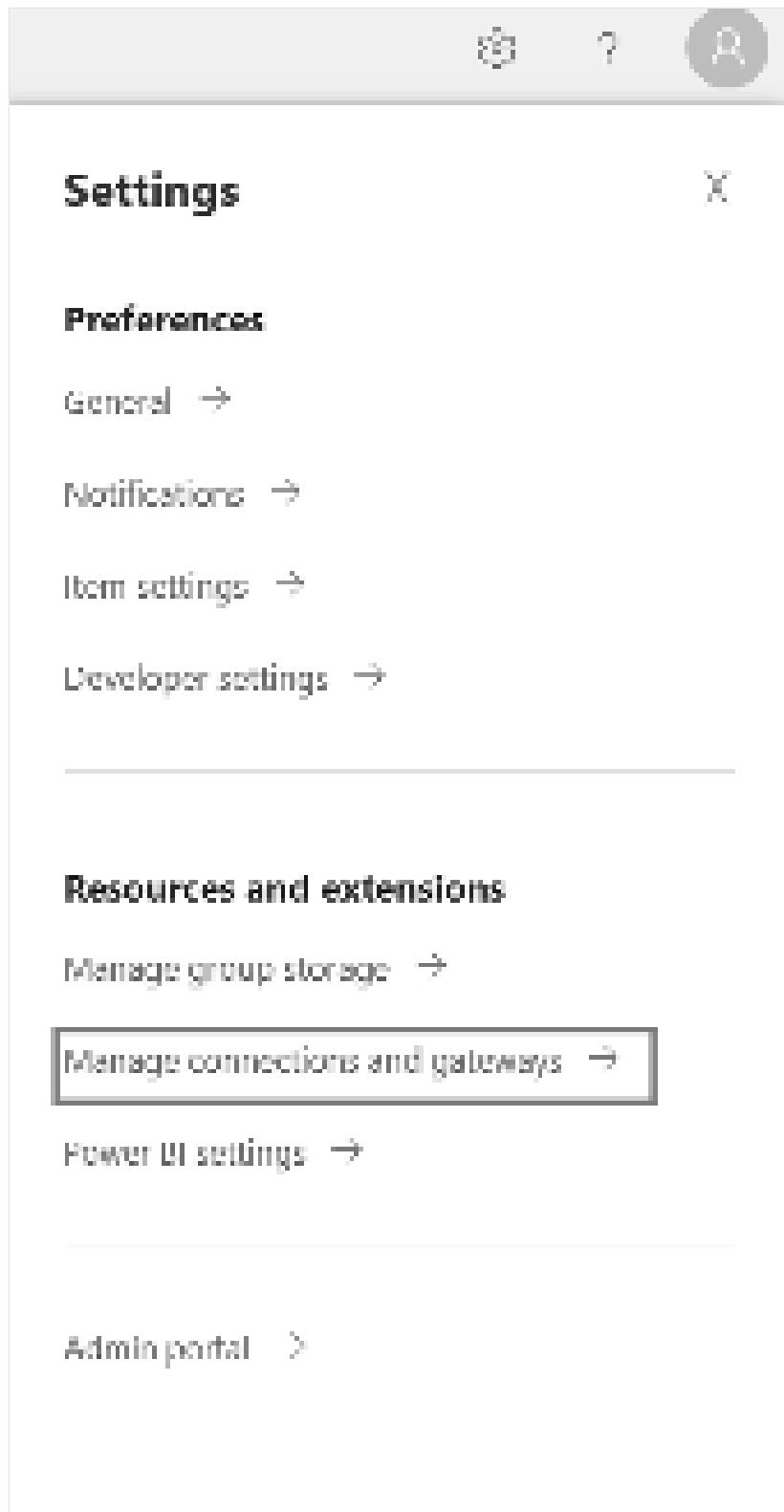
Set up your connection in Dataflow Gen2

The HTTP connector isn't currently supported in Dataflow Gen2.

Set up your connection in a data pipeline

To create a connection in a data pipeline:

1. From the page header in the Data Factory service, select **Settings**  > **Manage connections and gateways**.



2. Select **New** at the top of the ribbon to add a new data source.



The New connection pane opens on the left side of the page.



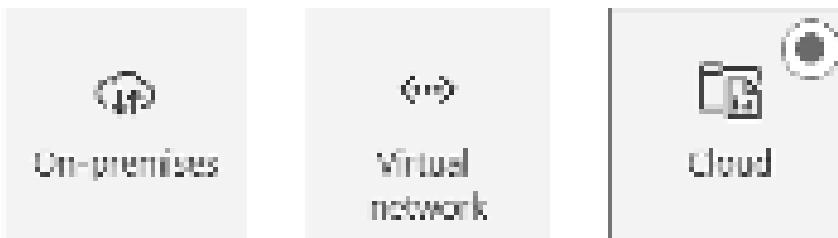
Setup connection

Step 1: Specify the new connection name, type, and URL

New connection

X

ⓘ Currently, these cloud connections are only supported for Data Pipelines and Kusto. In the future, other artifacts can also make use of the cloud connections. To create personal cloud connections in Datasets, Dataflows, and Dataflows, use the Power Query Online experience in "Get data".



Connection name *

generichhttp

Connection type *

Web

URL *

<The base URL to the web server >

In the **New connection** pane, choose **Cloud**, and then specify the following fields:

- **Connection name:** Specify a name for your connection.
- **Connection type:** Select Web for your connection type.
- **URL:** The base URL to the web server.

Step 2: Select and set your authentication

Under **Authentication method**, select your authentication from the drop-down list and complete the related configuration. This HTTP connector supports the following authentication types:

- Basic

Authentication

Authentication method *

Anonymous

Anonymous

Basic

OAuth2

Service Principal

Basic authentication

- **Username:** The user name to use to access the HTTP endpoint.
- **Password:** The password for specified username.

Authentication

Authentication method *

Basic

Username

<Your username>

Password

<Your password>

Skip test connection

Step 3: Specify the privacy level that you want to apply

In the **General** tab, select the privacy level that you want apply in the **Privacy level** drop-down list. Three privacy levels are supported. For more information, go to [privacy levels](#).

Step 4: Create your connection

Select **Create**. Your creation is successfully tested and saved if all the credentials are correct. If not correct, the creation fails with errors.



Table summary

The following connector properties in the table are supported in pipeline copy:

Name	Description	Required	Property	Copy
Connection name	A name for your connection.	Yes		✓/
Connection type	Select Web for your connection type.	Yes		✓/
URL	The base URL to the HTTP server.	Yes		✓/
Authentication	Go to Authentication	Yes		Go to Authentication
Privacy Level	The privacy level that you want to apply. Allowed values are Organizational, Privacy, Public	Yes		✓/

Authentication

The following properties in the table are the supported authentication types.

Name	Description	Required	Property	Copy
Basic				✓
- Username	The user name to use to access the HTTP endpoint.	Yes		
- Password	The password for specified username.		Yes	

Next steps

- Configure HTTP in a copy activity
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback [↗](#) | Ask the community [↗](#)

Configure HTTP in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in data pipeline to copy data from and to HTTP.

Supported format

HTTP supports the following file formats. Refer to each article for format-based settings.

- Avro format
- Binary format
- Delimited text format
- Excel format
- JSON format
- ORC format
- Parquet format
- XML format

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Mapping
- Settings

ⓘ Note

Destination isn't supported in the HTTP connector.

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

The following properties are supported for HTTP under the **Source** tab of a copy activity.

The screenshot shows the 'Source' tab of the Azure Data Factory copy activity configuration. The tabs at the top are 'General', 'Source', 'Destination', 'Mapping', and 'Settings'. The 'Source' tab is selected. Under 'Connection type', 'HTTP' is selected. The 'Connection' dropdown shows 'My connection' (selected), 'New...', 'File system', and 'Azure blob storage'. The 'Relative URL' field contains '/[relative URL specified]'. The 'Request method' dropdown shows 'Get' (selected). The 'Additional headers' section is empty. Below these are sections for 'Headers' (empty), 'Request body' (empty), 'Response headers' (empty), 'Response body' (empty), and 'Miscellaneous' (empty). The 'Additional columns' section shows 'HTTP' as the column type and a 'Add dynamic column' button with a search icon.

The following three properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select an HTTP connection from the connection list. If no connection exists, then create a new HTTP connection by selecting **New**.
- **Connection type:** Select **HTTP**.
- **Relative URL:** A relative URL to the resource that contains the data. When this property isn't specified, only the URL that's specified in the connection definition is used. The HTTP connector copies data from the combined URL: `/[relative URL specified]`.
- **File settings:** Select **File settings** to configure the file format. For settings of different file formats, refer to articles in Supported format for detailed information.

Under **Advanced**, you can specify the following fields:

- **Request method:** The HTTP method. Allowed values are **Get** (default) and **Post**.
- **Additional headers:** Additional HTTP request headers.

- **Request body:** The request body for the HTTP request.
- **Request timeout:** The timeout (the timespan value) for the HTTP request to get a response. This value is the timeout to get a response, not the timeout to read response data. The default value is 00:01:40.
- **Max concurrent connections:** The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.
- **Skip line count:** The number of non-empty rows to skip when reading data from input files.
- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.



Mapping

For **Mapping** tab configuration, see Configure your mappings under mapping tab. If you choose Binary as your file format, mapping will not be supported.

Settings

For **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

The following table contains more information about the copy activity in HTTP.

Source

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	<your connection>	Yes	connection

Name	Description	Value	Required	JSON script property
Connection type	The connection of your source data.	<connection of your source>	Yes	/
Relative URL	A relative URL to the resource that contains the data. When this property isn't specified, only the URL that's specified in the connection definition is used. The HTTP connector copies data from the combined URL: /[relative URL specified].	<your relative url>	No	relativeUrl
Request method	The HTTP method. Allowed values are Get (default) and Post .	•GET •POST	No	requestMethod
Additional headers	Additional HTTP request headers.	<your additional headers>	No	additionalHeaders
Request body	The request body for the HTTP request.	<body for POST HTTP request>	No	requestBody
Request timeout	The timeout (the timespan value) for the HTTP request to get a response. This value is the timeout to get a response, not the timeout to read response data. The default value is 00:01:40.	timespan	No	requestTimeout
Max concurrent connections	The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit	<max concurrent connections>	No	maxConcurrentConnections

Name	Description	Value	Required	JSON script property
	concurrent connections.			
Skip line count	The number of non-empty rows to skip when reading data from input files.	<your skip line count>	No	skipLineCount
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.	<ul style="list-style-type: none"> • Name • Value 	No	additionalColumns: <ul style="list-style-type: none"> • name • value

Next steps

- Set up your HTTP connection
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

IBM Db2 database connector overview

Article • 11/15/2023

The IBM Db2 database connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to an IBM Db2 database in Dataflow Gen2, go to Set up your IBM Db2 database connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support IBM Db2 databases in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your IBM Db2 database connection

Article • 11/15/2023

This article outlines the steps to create an IBM Db2 database connection.

Supported authentication types

The IBM Db2 database connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Windows	n/a	√
Basic	n/a	√

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to an IBM Db2 database. The following links provide the specific Power Query connector information you need to connect to an IBM Db2 database in Dataflow Gen2:

- To get started using the IBM Db2 database connector in Dataflow Gen2, go to Get data from Data Factory in Microsoft Fabric (Preview).
- Be sure to install or set up any IBM Db2 database prerequisites before connecting to the IBM Db2 database connector.
- To connect to the IBM Db2 database connector from Power Query, go to Connect to an IBM Db2 database from Power Query Online.

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support IBM Db2 databases in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Impala connector overview

Article • 11/15/2023

The Impala connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to an Impala database in Dataflow Gen2, go to Set up your Impala database connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support an Impala database in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your Impala database connection

Article • 11/15/2023

This article outlines the steps to create an Impala database connection.

Supported authentication types

The Impala connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Anonymous	n/a	✓
Windows	n/a	✓
Database	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to an Impala database. The following links provide the specific Power Query connector information you need to connect to an Impala database in Dataflow Gen2:

- To get started using the Impala connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric \(Preview\)](#).
- To connect to the Impala connector from Power Query, go to [Connect to an Impala database from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support an Impala database in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

KQL Database connector overview

Article • 11/15/2023

This KQL Database connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to a KQL database in Dataflow Gen2, go to Set up your connection in Dataflow Gen2.

Support in data pipelines

The KQL Database connector supports the following capabilities in data pipelines.

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	User Auth
Lookup activity	None	User Auth

To learn about the copy activity configuration for KQL Database in data pipelines, go to Configure KQL Database in a copy activity.

Feedback

Was this page helpful?



Yes



No

Provide product feedback | Ask the community

Set up your KQL Database connection

Article • 11/15/2023

This article outlines the steps to create a KQL Database connection.

Supported authentication types

The KQL Database connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Organizational account	✓	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to a KQL database. The following links provide the specific Power Query connector information you need to connect to KQL database in Dataflow Gen2:

- To get started using the KQL Database connector in Dataflow Gen2, go to Get data from Data Factory in Microsoft Fabric (Preview).
- Be sure to install or set up any KQL Database prerequisites before connecting to the KQL Database connector.
- To connect to the KQL Database connector from Power Query, go to Connect to a KQL database from Power Query Online.

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Feedback

Was this page helpful?



Yes



No

Provide product feedback ↗ | Ask the community ↗

Configure KQL Database in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in a data pipeline to copy data from and to KQL Database.

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Destination
- Mapping
- Settings

General

For **General** tab configuration, go to General.

Source

The following properties are supported for KQL Database under the **Source** tab of a copy activity.

General Source ¹ Destination Mapping Settings

Data store type	<input checked="" type="radio"/> Workspace	<input type="radio"/> External	<input type="radio"/> Configuration
Workspace data store type	<input checked="" type="radio"/> KQL Database		
KQL Database	<input checked="" type="radio"/> KQLDatabase	<input type="radio"/> Default	
Use query	<input checked="" type="radio"/> Table	<input type="radio"/> Query	
Table	Name: <input type="text" value="New"/> <input type="button" value="Edit"/> <input type="button" value="Delete"/> <input type="button" value="Advanced"/>		
Advanced			
Query timeout	00:10:00		
No transaction	<input type="checkbox"/>		
Additional columns	<input type="button" value="+ New"/> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 5px;"> Select query column... [More details >] </div>		
<input type="button" value=""/>			

The following properties are **required**:

- **Data store type:** Select **Workspace**.
- **Workspace data store type:** Select **KQL Database** from the data store type list.
- **KQL Database:** Select an existing KQL Database from the workspace.
- **Use query:** Select **Table or Query**.
 - **Table:** Select a table from the drop-down list or select **Edit** to manually enter it to read data..

Source	<input checked="" type="radio"/> Use query	<input type="radio"/> Use table
Table	<input type="text" value="New"/> <input type="button" value="Edit"/> <input type="button" value="Delete"/> <input type="button" value="Advanced"/>	

- **Query:** Specify the read-only request given in a KQL format. Use the custom KQL query as a reference.

Use query	<input checked="" type="radio"/> Table	<input type="radio"/> Query
Query	<input type="text" value="SELECT [table1].[column1]"/> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 5px;"> Select query column... [More details >] </div>	

Under **Advanced**, you can specify the following fields:

- **Query timeout:** Specify the wait time before the query request times out. Default value is 10 minutes (00:10:00). Allowed max value is 1 hour (01:00:00).
- **No truncation:** Indicates whether to truncate the returned result set. By default result is truncated after 500,000 records or 64 MB. Truncation is strongly recommended for a proper behavior of the activity.
- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter. To learn more, go to Add additional columns during copy.



Destination

The following properties are supported for KQL Database under the **Destination** tab of a copy activity.

The screenshot shows the 'Destination' tab of a copy activity configuration. The tab navigation bar includes General, Source, Destination (selected), Mapping, and Settings. The 'Destination' section contains the following fields:

- Data store type:** Radio buttons for Workspace (selected) and External.
- Workspace data store type:** A dropdown menu showing 'KQL Database' (selected).
- KQL Database:** A dropdown menu showing 'KQLDatabase' (selected).
- Table:** A dropdown menu showing 'None' (selected) with an 'Edit' link below it.
- Advanced:** A collapsed section containing 'ingestion mapping name' and 'Additional properties'.
- New:** A link at the bottom left of the destination section.

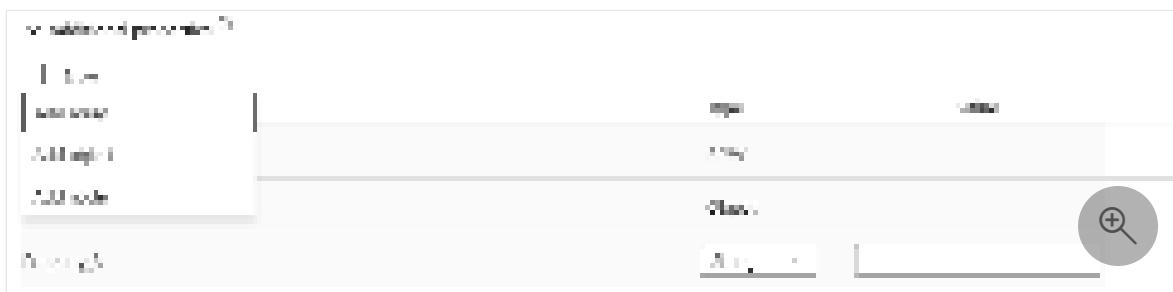
The following properties are **required**:

- **Data store type:** Select **Workspace**.
- **Workspace data store type:** Select **KQL Database** from the data store type list.
- **KQL Database:** Select an existing KQL Database from the workspace.

- **Table:** Select a table from the drop-down list or select **Edit** to manually enter it to write data.

Under **Advanced**, you can specify the following fields:

- **Ingestion mapping name:** The name of a mapping that was pre-created and assigned to a KQL Database destination table in advance.
- **Additional properties:** A property bag that can be used for specifying any of the ingestion properties that aren't being set already by the KQL Database destination. Specifically, it can be useful for specifying ingestion tags. To learn more, go to Azure Data Explorer data ingestion.



Mapping

For **Mapping** tab configuration, go to Configure your mappings under mapping tab.

Settings

For **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

The following tables contain more information about a copy activity in a KQL Database.

Source information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	Workspace	Yes	/
Workspace data store type	Select KQL Database from the data store type list.	KQL Database	Yes	/

Name	Description	Value	Required	JSON script property
KQL Database	Select an existing KQL Database from the workspace.	<your KQL Database>	Yes	/
Use query	Select Table or Query.	<ul style="list-style-type: none"> • Table • Query 	No	table query
Query timeout	Specify the wait time before the query request times out. Default value is 10 minutes (00:10:00). Allowed maximum value is 1 hour (01:00:00).	timespan	No	queryTimeout
No truncation	Indicates whether to truncate the returned result set. By default, the result is truncated after 500,000 records or 64 MB. Truncation is strongly recommended for a proper behavior of the activity.	select or unselect	No	noTruncation: true or false
Additional columns	Add additional data columns to the store source files' relative path or static value. Expression is supported for the latter. Learn more from Add additional columns during copy.	<ul style="list-style-type: none"> • Name • Value 	No	additionalColumns: <ul style="list-style-type: none"> • name • value

Destination information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	Workspace	Yes	/
Workspace data store type	Select KQL Database from the data store type list.	KQL Database	Yes	/
KQL Database	Select an existing KQL Database from the workspace.	<your KQL Database>	Yes	/
Table	Your destination data table to write data.	<your table name>	Yes	table

Name	Description	Value	Required	JSON script property
Ingestion mapping name	The name of a mapping that was pre-created and assigned to KQL Database destination table in advance.	<your ingestion mapping name>	Yes	ingestionMappingName
Additional properties	A property bag that can be used for specifying any of the ingestion properties that aren't being set already by the KQL Database destination. Specifically, it can be useful for specifying ingestion tags. Learn more from Azure Data Explorer data ingestion.	<ul style="list-style-type: none"> • Name • Type • Value 	Yes	additionalProperties

Next steps

- KQL Database connector overview
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Lakehouse connector overview

Article • 11/15/2023

The Lakehouse connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to a Lakehouse in Dataflow Gen2, go to Set up your connection in Dataflow Gen2.

Support in data pipelines

The Lakehouse connector supports the following capabilities in data pipelines:

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	User Auth
Delete activity	None	User Auth

To learn about the copy activity configuration for a Lakehouse in data pipelines, go to Configure Lakehouse in a copy activity.

Feedback

Was this page helpful?



Yes



No

Provide product feedback | Ask the community

Set up your Lakehouse connection

Article • 11/15/2023

You can connect to a Lakehouse data lake in Dataflow Gen2 using the Lakehouse connector provided by Data Factory in Microsoft Fabric.

Supported authentication types

The Azure Blob Storage connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Organizational account	✓	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to a Lakehouse. The following links provide the specific Power Query connector information you need to connect to a Lakehouse in Dataflow Gen2:

- To get started using the Lakehouse connector in Dataflow Gen2, go to Get data from Data Factory in Microsoft Fabric.
- Be sure to install or set up any Lakehouse prerequisites before connecting to the Lakehouse connector.
- To connect to the Lakehouse connector from Power Query, go to Connect to a Lakehouse from Power Query Online.

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Next steps

- Configure Lakehouse in a copy activity

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Configure Lakehouse in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in a data pipeline to copy data from and to the Fabric Lakehouse.

Supported format

Lakehouse supports the following file formats. Refer to each article for format-based settings.

- Avro format
- Binary format
- Delimited text format
- Excel format
- JSON format
- ORC format
- Parquet format
- XML format

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Destination
- Mapping
- Settings

General

For the **General** tab configuration, go to General.

Source

The following properties are supported for Lakehouse under the **Source** tab of a copy activity.



The following properties are **required**:

- **Data store type:** Select **Workspace**.
- **Workspace data store type:** Select **Lakehouse** from the data store type list.
- **Lakehouse:** Select an existing Lakehouse from the workspace. If none exists, then create a new Lakehouse by selecting **New**. If you use **Add dynamic content** to specify your Lakehouse, add a parameter and specify the Lakehouse object ID as the parameter value. To get your Lakehouse object ID, open your Lakehouse in your workspace, and the ID is after `/lakehouses/` in your URL.



- **Root folder:** Select **Tables** or **Files**, which indicates the virtual view of the managed or unmanaged area in your lake. For more information, refer to Lakehouse introduction.
 - If you select **Tables**:
 - **Table name:** Choose an existing table from the table list or specify a table name as the source.
 - Under **Advanced**, you can specify the following fields:
 - **Timestamp:** Specify to query an older snapshot by timestamp.
 - **Version:** Specify to query an older snapshot by version.
 - **Additional columns:** Add additional data columns to the store source files' relative path or static value. Expression is supported for the latter.

- If you select **Files**:
- **File path type**: You can choose **File path**, **Wildcard file path**, or **List of files** as your file path type. The following list describes the configuration of each setting:



- **File path**: Select **Browse** to choose the file that you want to copy, or fill in the path manually.
- **Wildcard file path**: Specify the folder or file path with wildcard characters under your given Lakehouse unmanaged area (under Files) to filter your source folders or files. Allowed wildcards are: `*` (matches zero or more characters) and `?` (matches zero or single character). Use `\` to escape if your folder or file name has wildcard or this escape character inside.
- **Wildcard folder path**: The path to the folder under the given container. If you want to use a wildcard to filter the folder, skip this setting and specify that information in the activity source settings.
- **Wildcard file name**: The file name under the given Lakehouse unmanaged area (under Files) and folder path.



- **List of files**: Indicates to copy a given file set.
 - **Folder path**: Points to a folder that includes files you want to copy.
 - **Path to file list**: Points to a text file that includes a list of files you want to copy, one file per line, which is the relative path to the file path configured.



- **Recursively**: Indicates whether the data is read recursively from the subfolders or only from the specified folder. If enabled, all files in the input folder and its subfolders are processed recursively. This property doesn't apply when you configure your file path type as **List of files**.
- **File format**: Select your file format from the drop-down list. Select the **Settings** button to configure the file format. For settings of different file

formats, refer to articles in Supported format for detailed information.

- Under **Advanced**, you can specify the following fields:
 - **Filter by last modified**: Files are filtered based on the last modified dates. This property doesn't apply when you configure your file path type as **List of files**.
 - **Start time**: The files are selected if their last modified time is greater than or equal to the configured time.
 - **End time**: The files are selected if their last modified time is less than the configured time.
 - **Enable partition discovery**: For files that are partitioned, specify whether to parse the partitions from the file path and add them as extra source columns.
 - **Partition root path**: When partition discovery is enabled, specify the absolute root path in order to read partitioned folders as data columns.
 - **Max concurrent connections**: Indicates the upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.

Destination

The following properties are supported for Lakehouse under the **Destination** tab of a copy activity.

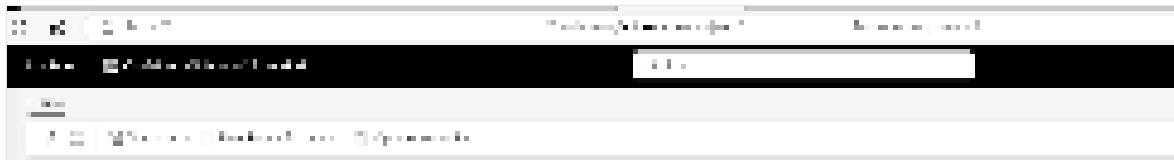
The screenshot shows the 'Destination' tab of a copy activity configuration. The tab has several sections:

- Data store type:** A radio button group where 'Workspace' is selected, and 'External' is unselected.
- Lakehouse data store type:** A dropdown menu showing 'Lakehouse'.
- Lakehouse:** A dropdown menu showing 'Lakehouse'.
- Root folder:** A radio button group where 'Tables' is selected, and 'Files' is unselected.
- Table name:** An input field containing 'DB1'.
- Advanced:** A section with the following controls:
 - Max rows per file:** An input field.
 - Table action:** A radio button group where 'Append' is selected, and 'Overwrite' is unselected.
 - Max concurrent connections:** An input field.
- Search:** A circular icon with a magnifying glass and a plus sign.

The following properties are **required**:

- **Data store type**: Select **Workspace**.

- **Workspace data store type:** Select **Lakehouse** from the data store type list.
- **Lakehouse:** Select an existing Lakehouse from the workspace. If none exists, then create a new Lakehouse by selecting **New**. If you use **Add dynamic content** to specify your Lakehouse, add a parameter and specify the Lakehouse object ID as the parameter value. To get your Lakehouse object ID, open your Lakehouse in your workspace, and the ID is after `/lakehouses/` in your URL.



- **Root folder:** Select **Tables** or **Files**, which indicates the virtual view of the managed or unmanaged area in your lake. For more information, refer to Lakehouse introduction.

- If you select **Tables**:

- **Table name:** Choose an existing table from the table list or specify a table name as the destination.



- Under **Advanced**, you can specify the following fields:
 - **Max rows per file:** Specify the maximum rows per file when writing data into Lakehouse.
 - **Table actions:** Specify the operation against the selected table.
 - **Append:** Append new values to existing table.
 - **Overwrite:** Overwrite the existing data and schema in the table using the new values. If this operation is selected, you can enable partition on your target table:
 - **Enable Partition:** This selection allows you to create partitions in a folder structure based on one or multiple columns. Each distinct column value (pair) is a new partition. For example, "year=2000/month=01/file". This selection supports insert-only mode and requires an empty directory in the destination.
 - **Partition column name:** Select from the destination columns in schemas mapping. Supported data types are string, integer, boolean, and datetime. Format respects type conversion settings under the **Mapping** tab.

- **Max concurrent connections:** The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.
- If you select **Files:**

- **File path:** Select **Browse** to choose the file that you want to copy, or fill in the path manually.



- **File format:** Select your file format from the drop-down list. Select **Settings** to configure the file format. For settings of different file formats, refer to articles in Supported format for detailed information.
- Under **Advanced**, you can specify the following fields:
 - **Copy behavior:** Defines the copy behavior when the source is files from a file-based data store. You can choose **Add Dynamic content**, **None**, **Flatten hierarchy**, or **Preserve hierarchy** as your copy behavior. The configuration of each setting is:
 - **Add dynamic content:** To specify an expression for a property value, select **Add dynamic content**. This field opens the expression builder where you can build expressions from supported system variables, activity output, functions, and user-specified variables or parameters. For more information about the expression language, go to Expressions and functions.
 - **None:** Choose this selection to not use any copy behavior.
 - **Flatten hierarchy:** All files from the source folder are in the first level of the destination folder. The destination files have autogenerated names.
 - **Preserve hierarchy:** Preserves the file hierarchy in the target folder. The relative path of a source file to the source folder is identical to the relative path of a target file to the target folder.



- **Max concurrent connections:** The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.
- **Block size (MB):** Specify the block size in MB when writing data to Lakehouse. Allowed value is between 4 MB and 100 MB.
- **Metadata:** Set custom metadata when copying to the destination data store. Each object under the `metadata` array represents an extra column. The `name` defines the metadata key name, and the `value` indicates the data value of that key. If preserve attributes feature is used, the specified metadata will union/overwrite with the source file metadata. The allowed data values are:
 - `$$LASTMODIFIED`: a reserved variable indicates to store the source files' last modified time. Apply to a file-based source with binary format only.
 - Expression
 - Static value



Mapping

For the **Mapping** tab configuration, if you don't apply Lakehouse table as your destination data store, go to Mapping.

If you apply Lakehouse table as your destination data store, except the configuration in Mapping, you can edit the type for your destination columns. After selecting **Import schemas**, you can specify the column type in your destination.

For example, the type for *PersonID* column in source is int, and you can change it to string type when mapping to destination column.



① Note

Editing the destination type currently is not supported when your source is decimal type.

If you choose Binary as your file format, mapping isn't supported.

Settings

For the **Settings** tab configuration, go to [Settings](#).

Table summary

The following tables contain more information about a copy activity in Lakehouse.

Source information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	Workspace	Yes	/
Workspace data store type	The section to select your workspace data store type.	Lakehouse	Yes	type
Lakehouse	The Lakehouse that you use as source.	<your Lakehouse>	Yes	workspaceld artifactId
Root folder	The type of the root folder.	* Tables * Files	No	rootFolder: Table or Files

Name	Description	Value	Required	JSON script property
Table name	The name of the table to read data.	<table name>	Yes when you select Tables in Root folder	table (under <code>typeProperties</code> -> <code>source</code> -> <code>typeProperties</code>)
Timestamp	The timestamp to query an older snapshot.	<timestamp>	No	timestampAsOf
Version	The version to query an older snapshot.	<version>	No	versionAsOf
Additional columns	Additional data columns to store source files' relative path or static value. Expression is supported for the latter.	* Name * Value	No	additionalColumns: * name * value
File path type	The type of the file path that you use.	* File path * Wildcard file path * List of files	Yes	/
File path	Copy from the path to a folder/file under source data store. Apply when choosing File path in File path type .	<file path>	Yes when choosing File path	* filePath * fileName
Wildcard paths	The folder path with wildcard characters under the source data store configured to filter source folders. Apply when choosing Wildcard file path in File path type .	<wildcard paths>	Yes when choosing Wildcard file path	* wildcardFolderPath * wildcardFileName
Folder path	Points to a folder that includes files you want to copy. Apply when choosing List of files in File path type .	<folder path>	No	folderPath

Name	Description	Value	Required	JSON script property
Path to file list	Indicates to copy a given file set. Point to a text file that includes a list of files you want to copy, one file per line, which is the relative path to the path configured. Apply when choosing List of files in File path type .	<path to file list>	No	fileListPath
Recursively	Process all files in the input folder and its subfolders recursively or just the ones in the selected folder. This setting is disabled when a single file is selected.	select or unselect	No	recursive: true or false
File format	The format of the file that you use.	<file format>	Yes	type (under <code>formatSettings</code>): DelimitedTextReadSettings
Filter by last modified	The files with last modified time in the range [Start time, End time) will be filtered for further processing. The time is applied to UTC time zone in the format of <code>yyyy-mm-ddThh:mm:ss.ffffZ</code> . This property can be skipped which means no file attribute filter is applied. This property doesn't apply when you configure your file path type as List of files .	* Start time * End time	No	modifiedDatetimeStart modifiedDatetimeEnd
Enable partition	Whether to parse the partitions from the	Selected or unselected	No	enablePartitionDiscovery: true or false (default)

Name	Description	Value	Required	JSON script property
Discovery	file path and add them as extra source columns.			
Partition root path	The absolute partition root path to read partitioned folders as data columns.	<your partition root path>	No	partitionRootPath
Max concurrent connections	The upper limit of concurrent connections established to the data store during the activity run. A value is needed only when you want to limit concurrent connections.	<max concurrent connections>	No	maxConcurrentConnections

Destination information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	Workspace	Yes	/
Workspace data store type	The section to select your workspace data store type.	Lakehouse	Yes	type
Lakehouse	The Lakehouse that you use as destination.	<your Lakehouse>	Yes	workspaceId artifactId
Root folder	The type of the root folder.	* Tables * Files	Yes	rootFolder: Table or Files
Table name	The name of the table to which you want to write data.	<your table name>	Yes when you select Tables in Root folder	table (under <code>typeProperties</code> -> <code>sink</code> -> <code>typeProperties</code>)
Max rows per file	When writing data into a folder, you can choose to write to multiple	<max rows per file>	No	maxRowsPerFile

Name	Description	Value	Required	JSON script property
	files and specify the max rows per file.			
Table action	Append new values to an existing table or overwrite the existing data and schema in the table using the new values.	* Append * Overwrite	No	tableActionOption: Append or Overwrite
Max concurrent connections	The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.	<max concurrent connections>	No	maxConcurrentConnections
File path	Write data to the path to a folder/file under destination data store.	<file path>	No	* filePath * fileName
File format	The format of the file that you use.	<file format>	Yes	type (under <code>formatSettings</code>): DelimitedTextWriteSettings
Copy behavior	The copy behavior defined when the source is files from a file-based data store.	* Add dynamic content * None * Flatten hierarchy * Preserve hierarchy	No	copyBehavior: * FlattenHierarchy * PreserveHierarchy
Block size (MB)	The block size in MB used to write data to Lakehouse. Allowed value is	<block size>	No	blockSizeInMB

Name	Description	Value	Required	JSON script property
	between 4 MB and 100 MB.			
Metadata	The custom metadata set when copying to a destination.	* \$\$LASTMODIFIED * Expression * Static value	No	metadata

Next steps

- Lakehouse connector overview
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Microsoft 365 connector overview

Article • 12/06/2023

The Microsoft 365 connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

Data Factory in Microsoft Fabric doesn't currently support the Microsoft 365 connector in Dataflow Gen2.

Support in data pipelines

The Microsoft 365 connector supports the following capabilities in data pipelines:

[+] Expand table

Supported capabilities	Gateway	Authentication
Copy activity (source/-)	None	Service principal

To learn about how to connect to Microsoft 365 in data pipelines, go to Set up your Microsoft 365 connection.

To learn about the copy activity configuration for Microsoft 365 in data pipelines, go to Configure Microsoft 365 in a copy activity.

Feedback

Was this page helpful?

 Yes  No

Provide product feedback  | Ask the community 

How to create a Microsoft 365 connection

Article • 12/21/2023

This article outlines the steps to create a Microsoft 365 connection.

Supported authentication types

This Microsoft 365 connector supports the following authentication types for copy and Dataflow Gen2 respectively.

[] Expand table

Authentication type	Copy	Dataflow Gen2
Service principal	✓	n/a

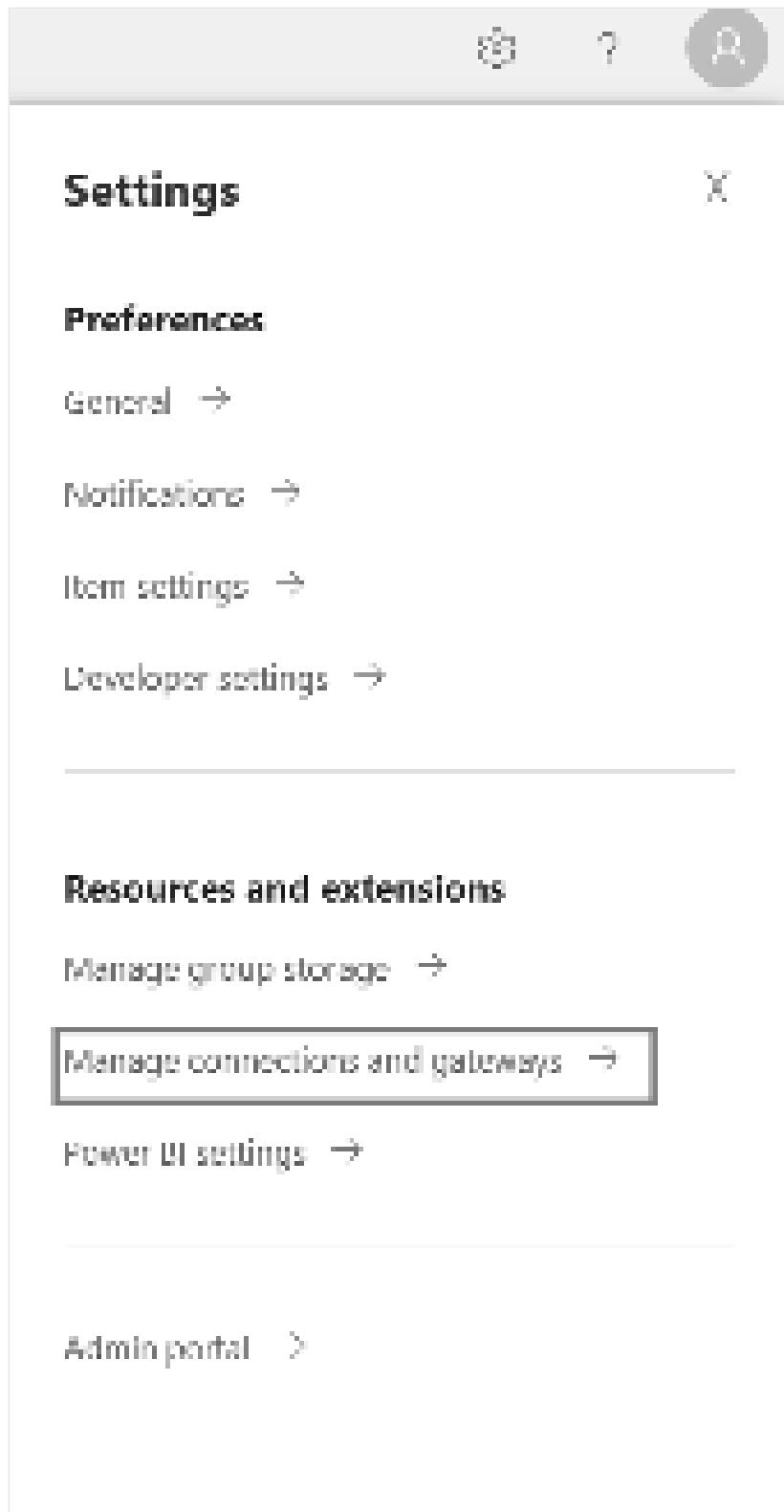
Set up your connection in Dataflow Gen2

The Microsoft 365 connector isn't currently supported in Dataflow Gen2.

Set up your connection in a data pipeline

To create a connection in a data pipeline:

1. From the page header in Data Integration service, select **Settings**  > **Manage connections and gateways**.



2. Select **New** at the top of the ribbon to add a new data source.



The New connection pane is displayed on the left side of the page.

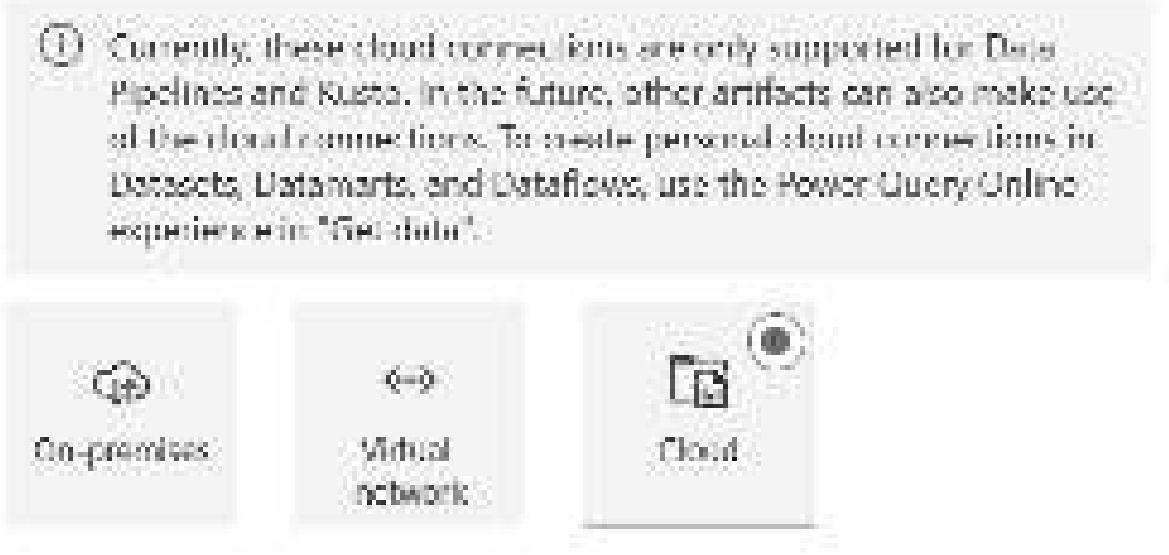


Setup connection

Step 1: Specify the new connection name, type

New connection

Currently, these cloud connections are only supported for Data Pipelines and Kusto. In the future, other artifacts can also make use of the cloud connections. To create personal client connections in Datasets, Datamarks, and Dataflows, use the Power Query Online experience with "Get Data".



Connection name:

Connection type:

In the **New connection** pane, choose **Cloud**, and then specify the following fields:

- **Connection name:** Specify a name for your connection.
- **Connection type:** Select **Microsoft 365** for your connection type.

Step 2: Select and set your authentication

Under **Authentication method**, select your authentication from the drop-down list and complete the related configuration. This Microsoft 365 connector supports the following authentication types.

Service Principal

Authentication

Authentication method *

OAuth2

OAuth2

Service Principal

Service Principal authentication

Authentication

Authentication method *

Service Principal

Tenant Id *

< your tenant ID >

Service principal ID *

< your application's client ID >

Service principal key *

- **Tenant Id:** Your service principal tenant ID. Specify the tenant information under which your Microsoft Entra web application resides.
- **Service principal ID:** Specify the application's client ID.
- **Service principal key:** Specify the application's key.

Step 3: Specify the privacy level that you want to apply

In the **General** tab, select the privacy level that you want apply in the **Privacy level** drop-down list. Three privacy levels are supported. For more information, go to [privacy levels](#).

Step 4: Create your connection

Select **Create**. Your creation is successfully tested and saved if all the credentials are correct. If not correct, the creation fails with errors.



Table summary

The connector properties in the following table are supported in pipeline copy:

[Expand table](#)

Name	Description	Required	Property	Copy
Connection name	A name for your connection.	Yes		✓
Connection type	Select Microsoft-365 for your connection type.	Yes		✓
Authentication	Go to Authentication	Yes		Go to Authentication

Authentication

The following table contains the supported authentication type properties.

Name	Description	Required	Property	Copy
Service Principal				✓
- Tenant ID	Your service principal tenant ID. Specify the tenant information under which your Microsoft Entra web application resides.	Yes		
- Service Principal ID	Specify the application's client ID.	Yes		
- Service Principal key	Specify the application's key.	Yes		

Next steps

- How to configure Microsoft 365 in a copy activity
-

Feedback

Was this page helpful?

Provide product feedback [↗](#) | Ask the community [↗](#)

Configure Microsoft 365 (Preview) in a copy activity

Article • 11/15/2023

Microsoft Fabric Data pipelines integrate with Microsoft Graph data connect, allowing you to bring the rich organizational data in your Microsoft 365 tenant into Fabric and Azure in a scalable way and build analytics applications and extract insights based on these valuable data assets. Integration with Privileged Access Management provides secured access control for the valuable curated data in Microsoft 365. Please refer to this link for an overview of Microsoft Graph data connect.

This article outlines how to use the copy activity in a data pipeline to copy data from Microsoft 365 (Preview). For now, within a single copy activity, you can ingest data from Microsoft 365 into Microsoft Fabric Lakehouse Table, Azure Blob Storage, Azure Data Lake Storage Gen1, and Azure Data Lake Storage Gen2. The supported data format is Avro, Delimited text, JSON, ORC and Parquet format.

Prerequisites

To copy data from Microsoft 365, you need to complete the following prerequisite steps:

- Your Microsoft 365 tenant admin must complete on-boarding actions as described here.
- Create and configure an Azure AD web application in Azure Active Directory. For instructions, go to [Create an Azure AD application](#).
- Make note of the following values, which you use to define the connection for Microsoft 365: Tenant ID. For instructions, go to [Get tenant ID](#).
- Application ID and Application key. For instructions, go to [Get application ID and authentication key](#). Add the user identity who will be making the data access request as the owner of the Azure AD web application (from the Azure AD web application > **Settings > Owners > Add owner**).
- The user identity must be in the Microsoft 365 organization you're getting data from and must not be a Guest user.

Approving new data access requests

If you're requesting data for this context for the first time (a combination of which data table is being accessed, which destination account is the data being loaded into, and which user identity is making the data access request), the copy activity status is displayed as **In Progress**. Only when you select the **Details** link under **Actions** will the status be displayed as **RequestingConsent**. A member of the data access approver group needs to approve the request in the Privileged Access Management before the data extraction can proceed.

Refer to the frequently asked questions on how the approver can approve the data access request. Refer to the data connect integration with PAM article for an explanation of the overall integration with Privileged Access Management, including how to set up the data access approver group.

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Mapping
- Settings

General

For the **General** tab configuration, go to General.

Source

The following properties are supported for Microsoft 365 under the **Source** tab of a copy activity.



The following properties are **required**:

- **Data store type:** Select External.
- **Connection:** Select a Microsoft 365 connection from the connection list. If no connection exists, then create a new Microsoft 365 connection by selecting New.

- **Table:** Name of the table to extract from Microsoft 365. You can preview the sample data by selecting **Preview sample data**.

Under **Advanced**, you can specify the following fields:

- **Scope:** You can select **All users or groups in the Microsoft 365 tenant** or **Select groups from the Microsoft 365 tenant**

If you select **All users or groups in the Microsoft 365 tenant**, the scope filter is displayed.



- **Scope filter:** You can use a predicate expression that's applied on the entire tenant to filter the specific rows to extract from Microsoft 365. The predicate format should match the query format of Microsoft Graph APIs, for example

```
https://graph.microsoft.com/v1.0/users?$filter=Department eq 'Finance'.
```

If you select **Select groups from the Microsoft 365 tenant**, you can select **Add user groups** to select groups from the Microsoft 365 tenant. Use this property to select up to 10 user groups for whom the data is retrieved. If no groups are specified, then data is returned for the entire organization.



- **Date filter:** Specify the name of the DateTime filter column. Use this property to limit the time range for which Microsoft 365 data is extracted. If your dataset has one or more DateTime columns, you need to specify a column here. Refer to Filtering for a list of datasets that require this DateTime filter.

Specify the **Start time (UTC)** and **End time (UTC)** to filter on when you select a DateTime filter column.



Mapping

For the **Mapping** tab configuration, go to Configure your mappings under mapping tab.

Settings

For the **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

The following tables contain more information about the copy activity in Microsoft 365.

Source information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	<your connection>	Yes	connection
Table	Name of the table to extract from Microsoft 365.	<table>	Yes	table
Scope	When the user group isn't specified, you can use a predicate expression that's applied on the entire tenant to filter the specific rows to extract from Microsoft 365. The predicate format should match the query format of Microsoft Graph APIs, for example <code>https://graph.microsoft.com/v1.0/users?</code> <code>\$filter=Department eq 'Finance'.</code>	<your scope>	Yes	scope
Scope filter	When the <code>allowedGroups</code> property isn't specified, you can use a predicate expression that's applied on the entire tenant to filter the specific rows to extract from Microsoft 365. The predicate format should match the query format of Microsoft Graph APIs, for example <code>https://graph.microsoft.com/v1.0/users?</code> <code>\$filter=Department eq 'Finance'.</code>	<scope filter>	No	userScopeFilterUri
Group ID	Group selection predicate. Use this property to select up to 10 user groups for whom the data is retrieved. If no groups are specified, then data is returned for the entire organization.	<group id>	No	allowedGroups

Name	Description	Value	Required	JSON script property
Date filter (Column name)	Name of the DateTime filter column. Use this property to limit the time range for which Microsoft 365 data is extracted.	<your DateTime filter column>	Yes if data has one or more DateTime columns.	dateFilterColumn
Start time (UTC)	Start DateTime value to filter on.	<start time>	Yes if dateFilterColumn is specified	startTime
End time (UTC)	End DateTime value to filter on.	<end time>	Yes if dateFilterColumn is specified	endTime

Next steps

- How to create a Microsoft 365 connection
-

Feedback

Was this page helpful?

 Yes
 No

Provide product feedback  | Ask the community 

Microsoft Exchange Online connector overview

Article • 11/15/2023

The Microsoft Exchange Online connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to Microsoft Exchange Online in Dataflow Gen2, go to Set up your Microsoft Exchange Online connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support Microsoft Exchange Online in data pipelines.

Feedback

Was this page helpful?



[Provide product feedback](#) | [Ask the community](#)

Set up your Microsoft Exchange Online connection

Article • 12/06/2023

This article outlines the steps to create a Microsoft Exchange Online connection.

Supported authentication types

The Microsoft Exchange Online connector supports the following authentication types for copy and Dataflow Gen2 respectively.

[] Expand table

Authentication type	Copy	Dataflow Gen2
Organizational account	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Microsoft Exchange Online. The following links provide the specific Power Query connector information you need to connect to Microsoft Exchange Online in Dataflow Gen2:

- To get started using the Microsoft Exchange Online connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- To connect to the Microsoft Exchange Online connector from Power Query, go to [Connect to Microsoft Exchange Online from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support Microsoft Exchange Online in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

MongoDB connector overview

Article • 11/15/2023

The MongoDB connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support for Dataflow Gen2

Data Factory in Microsoft Fabric doesn't currently support the MongoDB connector in Dataflow Gen2.

Support in data pipelines

The MongoDB connector supports the following capabilities in data pipelines:

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	Basic

To learn more about the copy activity configuration for MongoDB in data pipelines, go to [Configure in a data pipeline copy activity](#).

Feedback

Was this page helpful?



Yes



No

[Provide product feedback](#) | [Ask the community](#)

Configure MongoDB in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in data pipelines to copy data from and to MongoDB.

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Destination
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

Go to **Source** tab to configure your copy activity source. See the following content for the detailed configuration.



The following properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select a MongoDB connection from the connection list. If no connection exists, then create a new MongoDB connection by selecting **New**.
- **Database:** Select your database from the drop-down list.
- **Collection name:** Specify the name of the collection in MongoDB database. You can select the collection from the drop-down list or select **Edit** to enter it manually.

Under **Advanced**, you can specify the following fields:

- **Filter:** Specifies selection filter using query operators. To return all documents in a collection, omit this parameter or pass an empty document ({}).
- **Cursor methods:** Select **+ New** to specify the way that the underlying query is executed. The ways to execute query are:
 - **project:** Specifies the fields to return in the documents for projection. To return all fields in the matching documents, omit this parameter.
 - **sort:** Specifies the order in which the query returns matching documents. Go to cursor.sort() ↗ for more information.
 - **limit:** Specifies the maximum number of documents the server returns. Go to cursor.limit() ↗ for more information.
 - **skip:** Specifies the number of documents to skip and from where MongoDB begins to return results. Go to cursor.skip() ↗ for more information.
- **Batch size:** Specifies the number of documents to return in each batch of the response from MongoDB instance. In most cases, modifying the batch size will not affect the user or the application.

- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.

Destination

Go to **Destination** tab to configure your copy activity destination. See the following content for the detailed configuration.



The following properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select a MongoDB connection from the connection list. If no connection exists, then create a new MongoDB connection by selecting **New**.
- **Database:** Select your database from the drop-down list.
- **Collection name:** Specify the name of the collection in MongoDB database. You can select the collection from the drop-down list or select **Edit** to enter it manually.

Under **Advanced**, you can specify the following fields:

- **Write behavior:** Describes how to write data to MongoDB. Allowed values: **Insert** and **Upsert**.

The behavior of **Upsert** is to replace the document if a document with the same `_id` already exists; otherwise, insert the document.

! Note

The service automatically generates an `_id` for a document if an `_id` isn't specified either in the original document or by column mapping. This means

that you must ensure that, for **Upsert** to work as expected, your document has an ID.

- **Write batch timeout:** Specify the wait time for the batch insert operation to finish before it times out. The allowed value is timespan.
- **Write batch size:** This property controls the size of documents to write in each batch. You can try increasing the value to improve performance and decreasing the value if your document size being large.

Mapping

For **Mapping** tab configuration, see [Configure your mappings under mapping tab](#). Mapping is not supported when both source and destination are hierarchical data.

Settings

For **Settings** tab configuration, go to [Configure your other settings under settings tab](#).

Table summary

The following table contains more information about the copy activity in MongoDB.

Source information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	< your MongoDB connection >	Yes	connection
Database	Your database that you use as source.	< your database >	Yes	database
Collection name	Name of the collection in MongoDB database.	< your collection >	Yes	collection
Filter	The selection filter using query operators. To return all documents in a collection,	< your selection filter >	No	filter

Name	Description	Value	Required	JSON script property
Cursor methods	The way that the underlying query is executed.	<ul style="list-style-type: none"> • project • sort • limit • skip 	No	cursorMethods: <ul style="list-style-type: none"> • project • sort • limit • skip
Batch size	The number of documents to return in each batch of the response from MongoDB instance.	< your write batch size > (the default is 100)	No	batchSize
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.	<ul style="list-style-type: none"> • Name • Value 	No	additionalColumns: <ul style="list-style-type: none"> • name • value

Destination information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the destination data store.	< your MongoDB connection >	Yes	connection
Database	Your database that you use as destination.	< your database >	Yes	database
Collection name	Name of the collection in MongoDB database.	< your collection >	Yes	collection
Write behavior	Describes how to write data to MongoDB. Allowed values: Insert and Upsert .	<ul style="list-style-type: none"> • Insert (default) • Upsert <p>The behavior of Upsert is to replace the document if a document with the same <code>_id</code> already exists; otherwise, insert the document.</p> <p>Note: The service automatically generates an <code>_id</code> for a</p>	No	writeBehavior: <ul style="list-style-type: none"> • insert • upsert

Name	Description	Value	Required	JSON script property
	document if an <code>_id</code> isn't specified either in the original document or by column mapping. This means that you must ensure that, for Upsert to work as expected, your document has an ID.			
Write batch timeout	The wait time for the batch insert operation to finish before it times out.	timespan (the default is 00:30:00 - 30 minutes)	No	writeBatchTimeout
Write batch size	Controls the size of documents to write in each batch. You can try increasing this value to improve performance and decreasing the value if your document size being large.	< your write batch size >	No	writeBatchSize

Next steps

- MongoDB connector overview
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

MongoDB Atlas connector overview

Article • 11/15/2023

The MongoDB Atlas connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support for Dataflow Gen2

Data Factory in Microsoft Fabric doesn't currently support the MongoDB Atlas connector in Dataflow Gen2. Use the MongoDB Atlas SQL connector instead.

Support in data pipelines

The MongoDB Atlas connector supports the following capabilities in data pipelines:

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	Basic

To learn more about the copy activity configuration for MongoDB Atlas in data pipelines, go to [Configure in a data pipeline copy activity](#).

Feedback

Was this page helpful?

 Yes  No

[Provide product feedback](#) | [Ask the community](#)

Configure MongoDB Atlas in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in data pipeline to copy data from and to MongoDB Atlas.

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Destination
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

Go to **Source** tab to configure your copy activity source. See the following content for the detailed configuration.



The following properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select a MongoDB Atlas connection from the connection list. If no connection exists, then create a new MongoDB Atlas connection by selecting **New**.
- **Database:** Select your database from the drop-down list.
- **Collection name:** Specify the name of the collection in MongoDB Atlas database. You can select the collection from the drop-down list or select **Edit** to enter it manually.

Under **Advanced**, you can specify the following fields:

- **Filter:** Specifies selection filter using query operators. To return all documents in a collection, omit this parameter or pass an empty document ({}).
- **Cursor methods:** Select **+ New** to specify the way that the underlying query is executed. The ways to execute query are:
 - **project:** Specifies the fields to return in the documents for projection. To return all fields in the matching documents, omit this parameter.
 - **sort:** Specifies the order in which the query returns matching documents. Refer to cursor.sort() ↴ .
 - **limit:** Specifies the maximum number of documents the server returns. Refer to cursor.limit() ↴ .
 - **skip:** Specifies the number of documents to skip and from where MongoDB Atlas begins to return results. Refer to cursor.skip() ↴ .
- **Batch size:** Specifies the number of documents to return in each batch of the response from MongoDB Atlas instance. In most cases, modifying the batch size

will not affect the user or the application.

- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.

Destination

Go to **Destination** tab to configure your copy activity destination. See the following content for the detailed configuration.



The following properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select a MongoDB Atlas connection from the connection list. If no connection exists, then create a new MongoDB Atlas connection by selecting **New**.
- **Database:** Select your database from the drop-down list.
- **Collection name:** Specify the name of the collection in MongoDB Atlas database. You can select the collection from the drop-down list or select **Edit** to enter it manually.

Under **Advanced**, you can specify the following fields:

- **Write behavior:** Describes how to write data to MongoDB Atlas. Allowed values: **Insert** and **Upsert**.

The behavior of **Upsert** is to replace the document if a document with the same `_id` already exists; otherwise, insert the document.

Note

The service automatically generates an `_id` for a document if an `_id` isn't specified either in the original document or by column mapping. This means that you must ensure that, for **Upsert** to work as expected, your document has an ID.

- **Write batch timeout:** Specify the wait time for the batch insert operation to finish before it times out. The allowed value is timespan.
- **Write batch size:** This property controls the size of documents to write in each batch. You can try increasing the value to improve performance and decreasing the value if your document size being large.

Mapping

For **Mapping** tab configuration, see Configure your mappings under mapping tab. Mapping is not supported when both source and destination are hierarchical data.

Settings

For **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

The following table contains more information about the copy activity in MongoDB Atlas.

Source information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	< your MongoDB Atlas connection >	Yes	connection
Database	Your database that you use as source.	< your database >	Yes	database

Name	Description	Value	Required	JSON script property
Collection name	Name of the collection in MongoDB Atlas database.	< your collection >	Yes	collection
Filter	The selection filter using query operators. To return all documents in a collection, omit this parameter or pass an empty document ({}).	< your selection filter >	No	filter
Cursor methods	The way that the underlying query is executed.	<ul style="list-style-type: none"> • project • sort • limit • skip 	No	cursorMethods: <ul style="list-style-type: none"> • project • sort • limit • skip
Batch size	The number of documents to return in each batch of the response from MongoDB Atlas instance.	< your write batch size > (the default is 100)	No	batchSize
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.	<ul style="list-style-type: none"> • Name • Value 	No	additionalColumns: <ul style="list-style-type: none"> • name • value

Destination information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the destination data store.	< your MongoDB Atlas connection >	Yes	connection
Database	Your database that you use as destination.	< your database >	Yes	database
Collection name	Name of the collection in MongoDB Atlas database.	< your collection >	Yes	collection

Name	Description	Value	Required	JSON script property
Write behavior	<p>Describes how to write data to MongoDB Atlas. Allowed values: Insert and Upsert.</p> <p>The behavior of Upsert is to replace the document if a document with the same <code>_id</code> already exists; otherwise, insert the document.</p> <p>Note: The service automatically generates an <code>_id</code> for a document if an <code>_id</code> isn't specified either in the original document or by column mapping. This means that you must ensure that, for Upsert to work as expected, your document has an ID.</p>	<ul style="list-style-type: none"> • Insert (default) • Upsert 	No	writeBehavior: <ul style="list-style-type: none"> • <code>insert</code> • <code>upsert</code>
Write batch timeout	The wait time for the batch insert operation to finish before it times out.	timespan (the default is 00:30:00 - 30 minutes)	No	<code>writeBatchTimeout</code>
Write batch size	Controls the size of documents to write in each batch. You can try increasing this value to improve performance and decreasing the value if your document size being large.	< your write batch size >	No	<code>writeBatchSize</code>

Next steps

- MongoDB Atlas connector overview

Feedback

Was this page helpful?



MongoDB Atlas SQL connector overview

Article • 11/15/2023

The MongoDB Atlas SQL connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to a MongoDB Atlas SQL database in Dataflow Gen2, go to [Set up your MongoDB Atlas SQL \(Beta\) connection](#).

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support a MongoDB Atlas SQL database in data pipelines.

Feedback

Was this page helpful?



[Provide product feedback](#) | [Ask the community](#)

Set up your MongoDB Atlas SQL (Beta) connection

Article • 12/06/2023

This article outlines the steps to create a MongoDB Atlas SQL connection.

Supported authentication types

The MongoDB Atlas SQL connector supports the following authentication types for copy and Dataflow Gen2 respectively.

[+] Expand table

Authentication type	Copy	Dataflow Gen2
Basic (Username/Password)	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to a MongoDB Atlas SQL database. The following links provide the specific Power Query connector information you need to connect to a MongoDB Atlas SQL database in Dataflow Gen2:

- To get started using the MongoDB Atlas SQL connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- Be sure to install or set up any MongoDB Atlas SQL prerequisites before connecting to the MongoDB Atlas SQL connector.
- To connect to the MongoDB Atlas SQL connector from Power Query, go to [Connect to MongoDB Atlas federated database using Atlas SQL interface from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support a MongoDB Atlas SQL database in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

MySQL database connector overview

Article • 11/15/2023

The MySQL database connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to a MySQL database in Dataflow Gen2, go to Set up your MySQL database connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support a MySQL database in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your MySQL database connection

Article • 11/15/2023

This article outlines the steps to create a MySQL database connection.

Supported authentication types

The MySQL database connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Basic (Username/Password)	n/a	✓
Windows	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to a MySQL database. The following links provide the specific Power Query connector information you need to connect to a MySQL database in Dataflow Gen2:

- To get started using the MySQL database connector in Dataflow Gen2, go to Get data from Data Factory in Microsoft Fabric (Preview).
- Be sure to install or set up any MySQL database prerequisites before connecting to the MySQL database connector.
- To connect to the MySQL database connector from Power Query, go to Connect to MySQL database from Power Query Online.

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support a MySQL database in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

OData connector overview

Article • 11/15/2023

This OData connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to OData in Dataflow Gen2, go to Set up your connection in Dataflow Gen2.

Support in data pipelines

The OData connector supports the following capabilities in data pipelines:

Supported capabilities	Gateway	Authentication
Copy activity (source/-)	None	Anonymous Basic
Lookup activity	None	Anonymous Basic

To learn about how to connect to OData in data pipelines, go to Set up your OData connection.

To learn about the copy activity configuration for Azure Blob Storage in data pipelines, go to Configure OData in a copy activity.

Feedback

Was this page helpful?



Yes



No

Provide product feedback [↗](#) | Ask the community [↗](#)

Set up your OData connection

Article • 11/15/2023

This article outlines the steps to create an OData connection.

Supported authentication types

This OData connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Anonymous	✓	✓
Basic	✓	✓
Organizational account	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to OData. The following links provide the specific Power Query connector information you need to connect to OData in Dataflow Gen2:

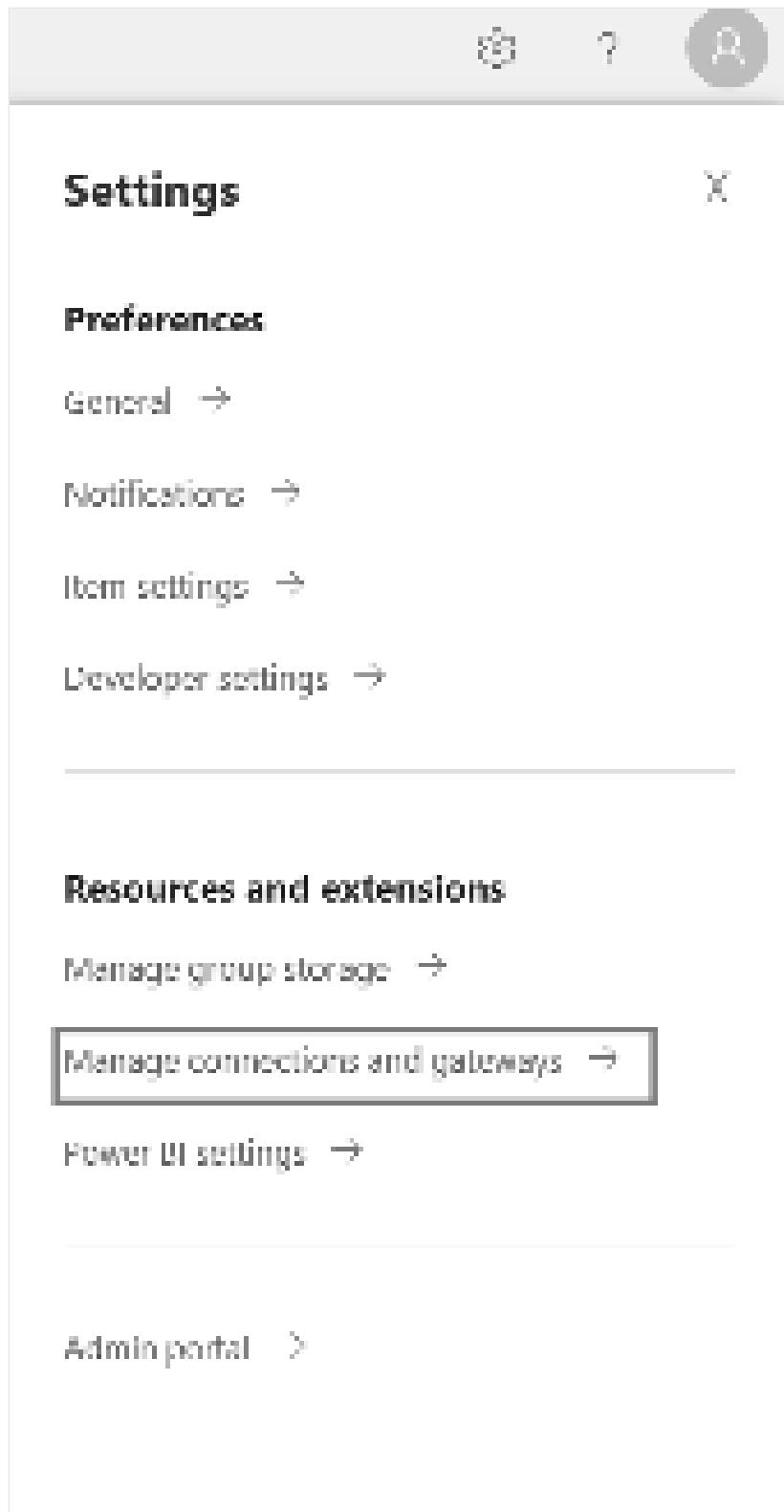
- To get started using the OData connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- To load data from the OData connector from Power Query, go to [Load data from an OData Feed in Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

To create a connection in a data pipeline:

1. From the page header in the Data Factory service, select **Settings**  > **Manage connections and gateways**.



2. Select **New** at the top of the ribbon to add a new data source.



The New connection pane shows up on the left side of the page.



Setup connection

Step 1: Specify the new connection name, type, and URL

New connection



ⓘ Currently, these cloud connections are only supported for Data Pipelines and Kusto. In the future, other artifacts can also make use of the cloud connections. To create personal cloud connections in Datasets, Databricks, and Dataflows, use the Power Query Online experience in "Get data".



Connection name *

Connection type *

URL *

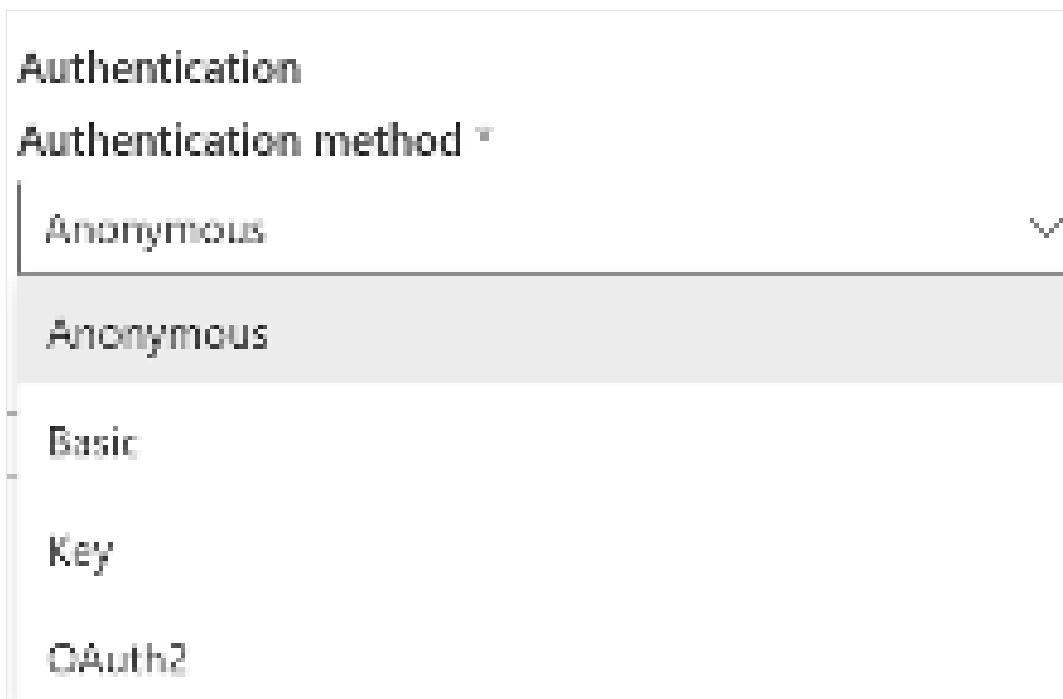
In the New connection pane, choose **Cloud**, and specify the following fields:

- **Connection name:** Specify a name for your connection.
- **Connection type:** Select **OData** for your connection type.
- **URL:** Enter the root URL of the **OData** service.

Step 2: Select and set your authentication

Under **Authentication method**, select your authentication from the drop-down list and complete the related configuration. This OData connector supports the following authentication types:

- Anonymous
- Basic



Anonymous authentication

Under Authentication method, select **Anonymous**.



Basic authentication

- **Username:** The user name to use to access the OData endpoint.
- **Password:** The password for the specified username.

Authentication

Authentication method *

Basic



Username *

< Your username >

Password *

< Your password >



Skip test connection

Step 3: Specify the privacy level that you want to apply

In the **General** tab, select the privacy level that you want apply in the **Privacy level** drop-down list. Three privacy levels are supported. For more information, go to [privacy levels](#).

Step 4: Create your connection

Select **Create**. Your creation is successfully tested and saved if all the credentials are correct. If not correct, the creation fails with errors.

The screenshot shows the 'Create connection' dialog box with the 'General' tab selected. The 'Description' field contains the placeholder 'Enter a descriptive name for this connection'. Below it, there's a note: 'Select the privacy level that you want to apply in the Privacy level drop-down list. Three privacy levels are supported.' At the bottom, there are tabs for 'General', 'Advanced', 'Script', and 'Reference URLs', with 'General' being the active tab. A large 'Create' button is visible at the bottom right.

Table summary

The connector properties in the following table are supported in pipeline copy:

Name	Description	Required	Property	Copy
Connection name	A name for your connection.	Yes		✓
Connection type	Select OData for your connection type.	Yes		✓
URL	The base URL to the OData server.	Yes		✓
Authentication	Go to Authentication	Yes	Go to Authentication	
Privacy Level	The privacy level that you want to apply. Allowed values are Organizational, Privacy, Public	Yes		✓

Authentication

The properties in the following table are the supported authentication type.

Name	Description	Required	Property	Copy
Anonymous				✓
Basic				✓
- Username	The user name to use to access the OData endpoint.	Yes		
- Password	The password for the specified username.	Yes		

Next steps

- Configure OData in a copy activity

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Configure OData in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in a data pipeline to copy data from and to OData.

Supported configuration

For the configuration of each tab under the copy activity, go to the following sections respectively.

- General
- Source
- Mapping
- Settings

General

For **General** tab configuration, go to General.

Source

The following properties are supported for OData under the **Source** tab of a copy activity.



The following three properties are **required**:

- **Data store type**: Select **External**.

- **Connection:** Select an OData connection from the connection list. If no connection exists, then create a new OData connection by selecting **New**.
- **Path:** Select the path to the OData resource. Or you can select **Edit** to enter the path manually.

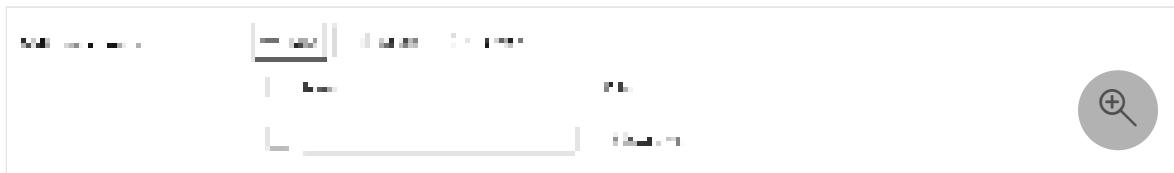
Under **Advanced**, you can specify the following fields:

- **Use query:** You can choose **Path** or **Query** as your use query. The following list describes the configuration of each setting.
 - **Path:** Read data from the specified path if you select this button.
 - **Query:** OData query options for filtering data. Example:
"\$select=Name,Description&\$top=5".

Note

The OData connector copies data from the combined URL: [URL specified in the connection]/[path specified]?[query specified in copy activity source]. For more information, go to [OData URL components](#).

- **Request timeout:** Timeout for the HTTP request to get a response. Format is in timespan. This value is the timeout to get a response, not the timeout to read response data. The default value is 00:05:00.
- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.



Mapping

For the **Mapping** tab configuration, go to Configure your mappings under mapping tab.

Settings

For the **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

The following table contains more information about the copy activity in OData.

Source

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	<your connection>	Yes	connection
Path	The path to the OData resource.	<the path to the OData resource>	Yes	path
Use query	You can choose Path or Query as your use query.	<ul style="list-style-type: none">• Path• Query	No	query
Request timeout	Timeout for the HTTP request to get a response. Format is in timespan. This value is the timeout to get a response, not the timeout to read response data. The default value is 00:05:00.	timespan	No	requestTimeout
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.	- Name - Value	No	additionalColumns: - name - value

Next steps

- Set up your OData connection

Feedback

Was this page helpful?



Yes



No

Odbc connector overview

Article • 11/15/2023

The open database connectivity (Odbc) connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to Odbc in Dataflow Gen2, go to Set up your Odbc connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support Odbc in data pipelines.

Feedback

Was this page helpful?



Provide product feedback [↗](#) | Ask the community [↗](#)

Set up your Odbc connection

Article • 12/06/2023

This article outlines the steps to create an Odbc connection.

Supported authentication types

The Odbc connector supports the following authentication types for copy and Dataflow Gen2 respectively.

 Expand table

Authentication type	Copy	Dataflow Gen2
Anonymous	n/a	✓
Basic (Username/Password)	n/a	✓
Windows	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to an Odbc data source. The following links provide the specific Power Query connector information you need to connect to an Odbc data source in Dataflow Gen2:

- To get started using the Odbc connector in Dataflow Gen2, go to Get data from Data Factory in Microsoft Fabric.
- Be sure to install or set up any Odbc prerequisites before connecting to the Odbc connector.
- To connect to the Odbc data source from Power Query, go to Connect to an ODBC data source from Power Query Online.

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support an Odbc data source in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Palantir Foundry connector overview

Article • 11/15/2023

The Palantir Foundry connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to Palantir Foundry in Dataflow Gen2, go to Set up your Palantir Foundry connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support Palantir Foundry in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback [↗](#) | Ask the community [↗](#)

Set up your Palantir Foundry connection

Article • 12/06/2023

This article outlines the steps to create a Palantir Foundry connection.

Supported authentication types

The Palantir Foundry connector supports the following authentication types for copy and Dataflow Gen2 respectively.

 Expand table

Authentication type	Copy	Dataflow Gen2
Foundry Token	n/a	✓
Foundry OAuth	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Palantir Foundry. The following links provide the specific Power Query connector information you need to connect to Palantir Foundry in Dataflow Gen2:

- To get started using the Palantir Foundry connector in Dataflow Gen2, go to Get data from Data Factory in Microsoft Fabric.
- Be sure to install or set up any Palantir Foundry prerequisites before connecting to the Palantir Foundry connector.
- To connect to the Palantir Foundry connector from Power Query, go to Connect to Palantir Foundry from Power Query Online.

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support Palantir Foundry in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

PostgreSQL database connector overview

Article • 11/15/2023

The PostgreSQL database connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to a PostgreSQL database in Dataflow Gen2, go to [Set up your PostgreSQL database connection](#).

Support in Data pipeline

The PostgreSQL database connector supports the following capabilities in Data pipeline:

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	Basic
Lookup activity	None	Basic

To learn more about the copy activity configuration for PostgreSQL database in Data pipeline, go to [Configure in a data pipeline copy activity](#).

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback ↗](#) | [Ask the community ↗](#)

Set up your PostgreSQL database connection

Article • 12/06/2023

This article outlines the steps to create a PostgreSQL database connection.

Supported authentication types

The PostgreSQL database connector supports the following authentication types for copy and Dataflow Gen2 respectively.

[Expand table

Authentication type	Copy	Dataflow Gen2
Basic (Username/Password)	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to a PostgreSQL database. The following links provide the specific Power Query connector information you need to connect to a PostgreSQL database in Dataflow Gen2:

- To get started using the PostgreSQL database connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- Be sure to install or set up any PostgreSQL database prerequisites before connecting to the PostgreSQL database connector.
- To connect to the PostgreSQL database connector from Power Query, go to [Connect to a PostgreSQL database from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Feedback

Was this page helpful?

Yes

No

[Provide product feedback](#) | [Ask the community](#)

Configure PostgreSQL in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in data pipelines to copy data from PostgreSQL.

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

Go to **Source** tab to configure your copy activity source. See the following content for the detailed configuration.



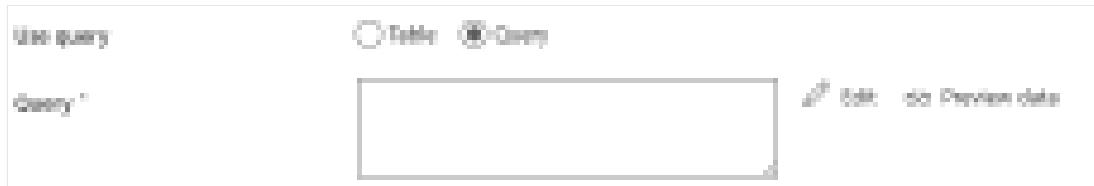
The following three properties are **required**:

- **Data store type**: Select **External**.
- **Connection**: Select a PostgreSQL connection from the connection list. If no connection exists, then create a new PostgreSQL connection by selecting **New**.

- **Connection type:** Select PostgreSQL.
- **Use query:** Select from Table or Query.
 - If you select **Table**:
 - **Table:** Specify the name of the table to read data. Select the table from the drop-down list or select **Edit** to manually enter it.



- If you select **Query**:
 - **Query:** Specify the custom SQL query to read data.



Note

Schema and table names are case-sensitive. Enclose them in "" (double quotes) in the query.

Under **Advanced**, you can specify the following fields:

- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.

Mapping

For **Mapping** tab configuration, see Configure your mappings under mapping tab.

Settings

For **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

The following table contains more information about the copy activity in PostgreSQL.

Source information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	< your PostgreSQL connection >	Yes	connection
Connection type	Your source connection type.	PostgreSQL	Yes	/
Use query	The way to read data. Apply Table to read data from the specified table or apply Query to read data using SQL queries.	<ul style="list-style-type: none">• Table• Query	Yes	<ul style="list-style-type: none">• typeProperties (under <code>typeProperties</code> -> <code>source</code>)<ul style="list-style-type: none">- schema- table• query
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.	<ul style="list-style-type: none">• Name• Value	No	additionalColumns: <ul style="list-style-type: none">• name• value

Next steps

- PostgreSQL connector overview

Feedback

Was this page helpful?



Yes



No

Provide product feedback ↗ | Ask the community ↗

REST connector overview

Article • 11/15/2023

The REST connector is supported in Data Factory in Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

Data Factory in Microsoft Fabric doesn't currently support the REST connector in Dataflow Gen2. To get REST data in Dataflow Gen2, use the Web API connector instead.

Support in data pipelines

The REST connector supports the following capabilities in data pipelines:

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	Anonymous

To learn about how to connect to REST data in data pipelines, go to Set up your REST connection.

To learn about the copy activity configuration for REST in data pipelines, go to Configure REST in a copy activity.

Feedback

Was this page helpful?



Yes



No

Provide product feedback | Ask the community

How to create REST connection

Article • 11/15/2023

This article outlines the steps to create REST connection.

Supported authentication types

This REST connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2 (Web API)
Anonymous	✓	✓
Basic		✓
Organizational account		✓
Windows	✓	

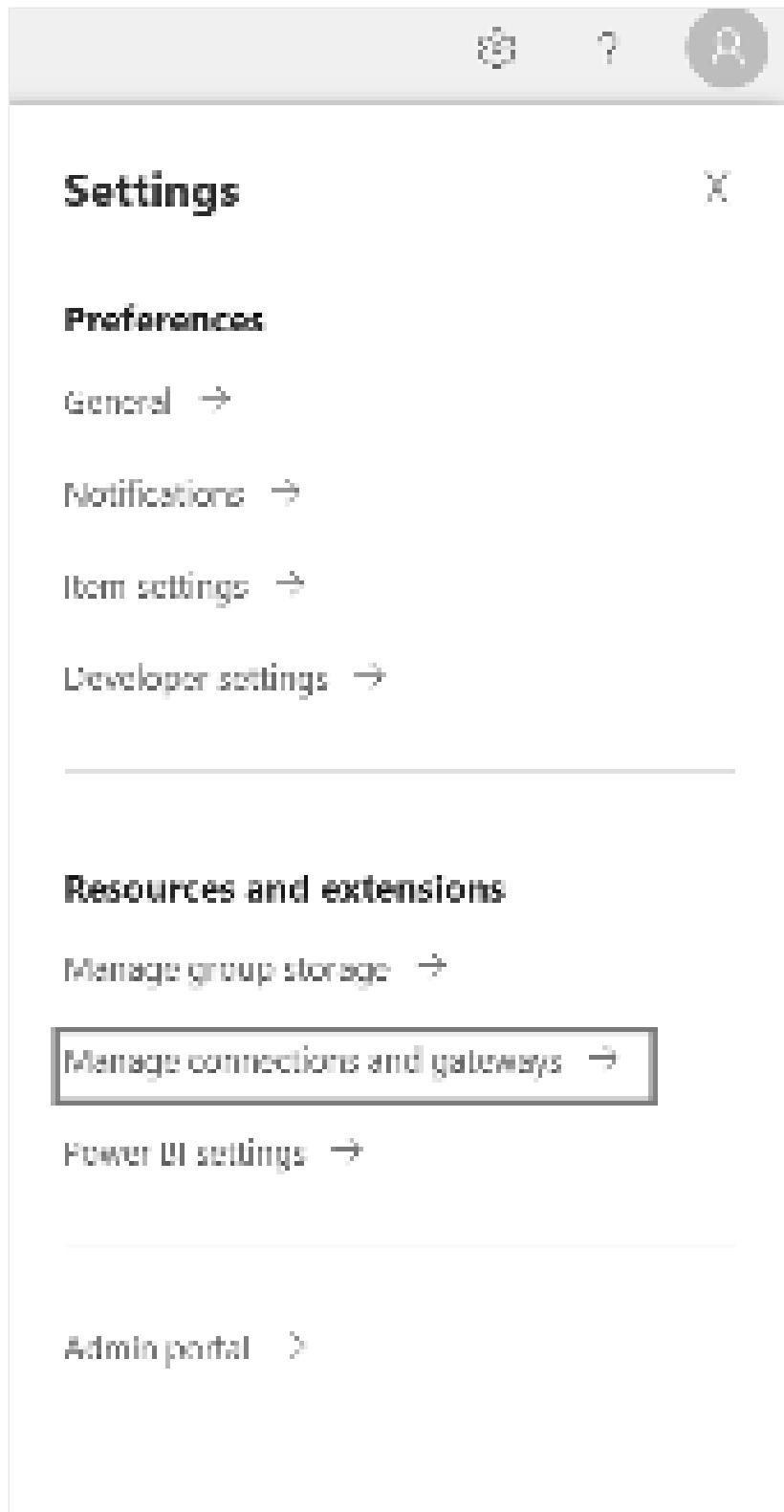
Set up your connection in Dataflow Gen2

The Microsoft 365 connector isn't currently supported in Dataflow Gen2. To connect to REST data in Dataflow Gen2, use the Web API connector.

Set up your connection in a data pipeline

To create a connection in a data pipeline:

1. From the page header in Data Factory service, select **Settings**  > **Manage connections and gateways**.



2. Select **New** at the top of the ribbon to add a new data source.



The New connection pane appears on the left side of the page.



Setup connection

Step 1: Specify the new connection name, type, and URL

New connection



ⓘ Currently, these cloud connections are only supported for Data Pipelines and Kusto. In the future, other artifacts can also make use of the cloud connections. To create personal cloud connections in Datasets, Dataloads, and Dataflows, use the Power Query Online experience in "Get data".



Connection name *

genericrest

Connection type *

Web

URL *

< The base URL to the web server >

In the **New connection** pane, choose **Cloud**, and then specify the following fields:

- **Connection name:** Specify a name for your connection.
- **Connection type:** Select **Web** for your connection type.
- **URL:** The base URL to the web server.

Step 2: Select and set your authentication

Under **Authentication method**, select your authentication from the drop-down list and complete the related configuration. The REST connector supports the following authentication types:

- Anonymous

Authentication

Authentication method *

The screenshot shows a dropdown menu with the following options:

- Anonymous
- Basic
- OAuth2
- Service Principal

The 'Anonymous' option is highlighted with a dark grey background.

Anonymous authentication

Select **Anonymous** under **Authentication method**.

Authentication

Authentication method *

The screenshot shows a dropdown menu with the following options:

- Anonymous
- Basic
- OAuth2
- Service Principal

The 'Anonymous' option is selected and highlighted with a dark grey background. Below the dropdown is a button labeled "Skip test connection".

Step 3: Specify the privacy level that you want to apply

In the **General** tab, select the privacy level that you want to apply in the **Privacy level** drop-down list. Three privacy levels are supported. For more information, go to [privacy levels](#).

Step 4: Create your connection

Select **Create**. Your creation is successfully tested and saved if all the credentials are correct. If not correct, the creation fails with errors.



Table summary

The following table contains connector properties that are supported in pipeline copy.

Name	Description	Required	Property	Copy
Connection name	A name for your connection.	Yes		✓
Connection type	Select Web for your connection type.	Yes		✓
URL	The base URL to the REST server.	Yes		✓
Authentication	Go to Authentication	Yes	See Authentication	
Privacy Level	The privacy level that you want to apply. Allowed values are Organizational, Privacy, Public	Yes		✓

Authentication

The following the following table contains the properties for supported authentication types.

Name	Description	Required	Property	Copy
Anonymous				✓

Next steps

- How to configure REST in copy activity

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Configure REST in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in data pipeline to copy data from and to a REST endpoint.

Specifically, this generic REST connector supports:

- Copying data from a REST endpoint by using the **GET** or **POST** methods and copying data to a REST endpoint by using the **POST**, **PUT** or **PATCH** methods.
- **Pagination** in the REST APIs.
- For REST as a source, copy the REST JSON response as-is or parse it by using schema mapping. Only the response payload in **JSON** is supported.

Supported configuration

For the configuration of each tab under a copy activity, go to the following sections respectively.

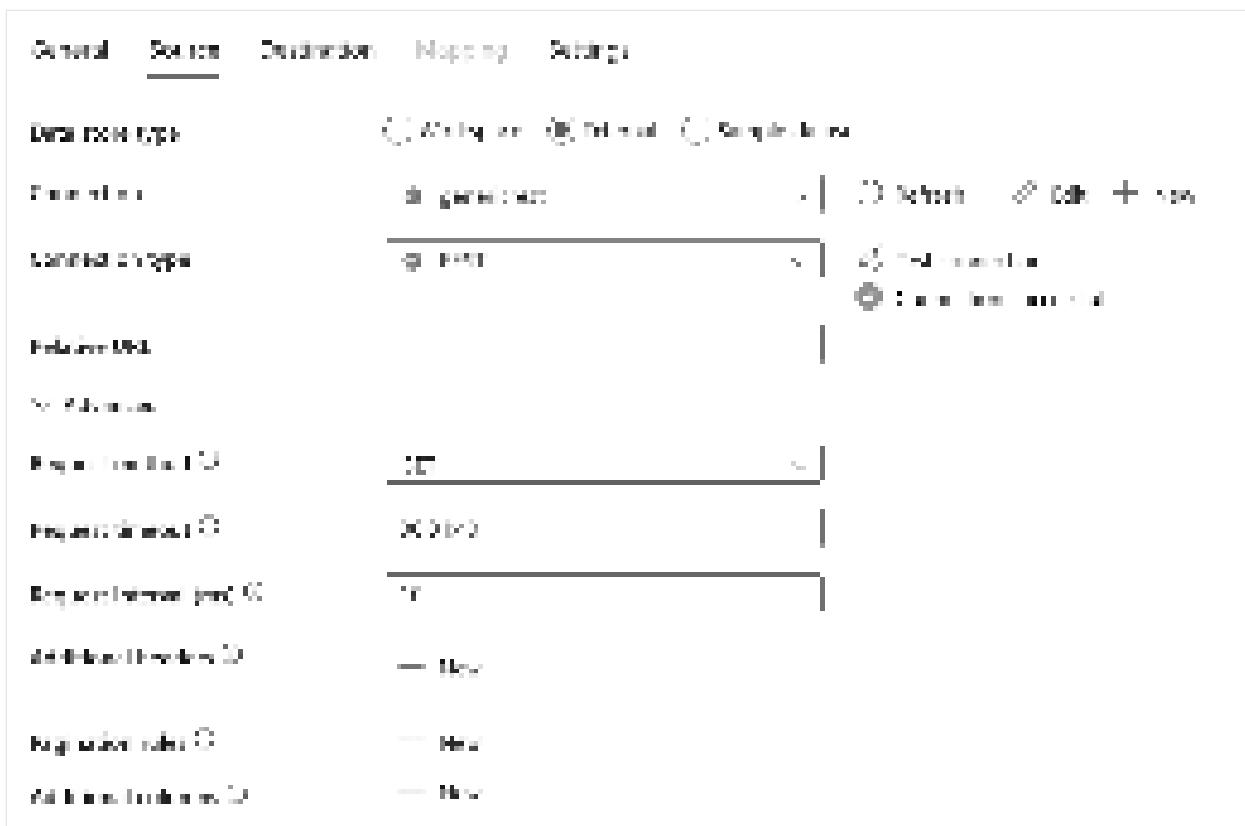
- General
- Source
- Destination
- Mapping
- Settings

General

For **General** tab configuration, go to General.

Source

The following properties are supported for REST under the **Source** tab of a copy activity.



The following first three properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select an **REST** connection from the connection list. If no connection exists, then create a new **REST** connection by selecting **New**.
- **Connection type:** Select **REST**.
- **Relative URL:** A relative URL to the resource that contains the data. When this property isn't specified, only the URL that's specified in the connection definition is used. The HTTP connector copies data from the combined URL: `[URL specified in connection]/[relative URL specified]`.

Under **Advanced**, you can specify the following fields:

- **Request method:** The HTTP method. Allowed values are **GET** (default) and **POST**.



- **Request timeout:** The timeout (the timespan value) for the HTTP request to get a response. This value is the timeout to get a response, not the timeout to read response data. The default value is 00:01:40.

- **Request interval (ms):** The interval time between different requests for multiple pages in milliseconds. Request interval value should be a number between [10, 60000].
- **Additional headers:** Additional HTTP request headers.

Additional headers	
Name	Value
Content-Type	text/html

- **Pagination rules:** The pagination rules to compose next page requests. Refer to the pagination support section for details.

Pagination rules	
Index	Value
0	value
1	value

- **Additional columns:** Add more data columns to store source files' relative path or static value. Expression is supported for the latter.

Additional columns	
Index	Relative path
0	relativePath
1	relativePath

Destination

The following properties are supported for REST under the **Destination** tab of a copy activity.

System	Source	<u>Destination</u>	Mappings	Actions
Data store type	<input checked="" type="checkbox"/> WorkSpace	<input checked="" type="checkbox"/> External		
Connection	<input checked="" type="checkbox"/> REST API	<input type="button" value="New"/>	<input type="button" value="Remove"/>	<input type="button" value="Edit"/>
Connection type	<input checked="" type="checkbox"/> REST	<input type="button" value="Edit connection"/>		
Relative URL				
Advanced				
Request method	POST			
Request timeout	00:01:40			
Write batch size	10000			
HTTP Compressions type	None			
Failure tolerance	100%			

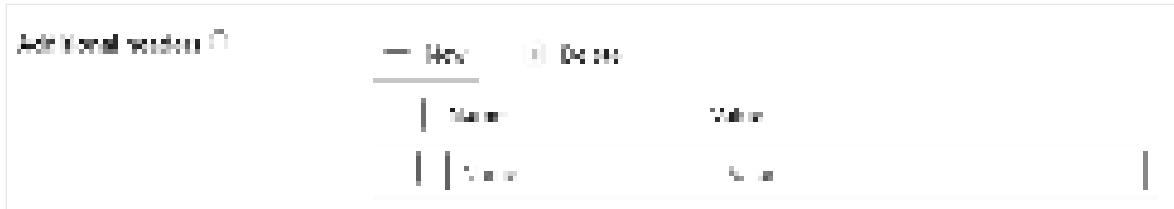
The following first three properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select a **REST** connection from the connection list. If no connection exists, then create a new REST connection by selecting **New**.
- **Connection type:** Select **REST**.
- **Relative URL:** A relative URL to the resource that contains the data. When this property isn't specified, only the URL that's specified in the connection definition is used. The HTTP connector copies data from the combined URL: `[URL specified in connection]/[relative URL specified]`.

Under **Advanced**, you can specify the following fields:

- **Request method:** The HTTP method. Allowed values are **POST** (default), **PUT**, and **PATCH**.
- **Request timeout:** The timeout (the timespan value) for the HTTP request to get a response. This value is the timeout to get a response, not the timeout to write the data. The default value is `00:01:40`.
- **Request interval (ms):** The interval time between different requests for multiple pages in milliseconds. Request interval value should be a number between [10, 60000].
- **Write batch size:** Number of records to write to the REST destination per batch. The default value is 10000.

- **Http Compression type:** HTTP compression type to use while sending data with Optimal Compression Level. Allowed values are **None** and **GZip**.
- **Additional headers:** Additional HTTP request headers.



Mapping

For the **Mapping** tab configuration, go to Configure your mappings under mapping tab. Mapping is not supported when both source and destination are hierarchical data.

Settings

For the **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

The following tables contain more information about the copy activity in REST.

Source information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	<ul style="list-style-type: none"> • Workspace • External • Sample dataset 	Yes	type
Connection	Your connection to the source data store.	<your connection>	Yes	connection
Connection type	Your connection type. Select REST.	REST	Yes	type: RestResource
Relative URL	A relative URL to the resource that contains the data. When this property isn't specified, only the URL that's specified in	<your relative url>	No	relativeUrl

Name	Description	Value	Required	JSON script property
	the connection definition is used. The HTTP connector copies data from the combined URL: [URL specified in connection]/[relative URL specified].			
Request method	The HTTP method. Allowed values are GET (default) and POST .	<ul style="list-style-type: none"> • GET • POST 	No	requestMethod
Request timeout	The timeout (the timespan value) for the HTTP request to get a response. This value is the timeout to get a response, not the timeout to read response data. The default value is 00:01:40.	timespan	No	httpRequestTimeout
Request interval (ms)	The interval time between different requests for multiple pages in milliseconds. Request interval value should be a number between [10, 60000].	[10, 60000]	No	requestInterval
Additional headers	Additional HTTP request headers.	<your additional headers>	No	additionalHeaders
Pagination rules	The pagination rules to compose next page requests. Refer to the pagination support section for details.	Go to pagination	No	paginationRules
Additional columns	Add more data columns to the store source files' relative path or static value. Expression is supported for the latter.	<ul style="list-style-type: none"> • Name • Value 	No	additionalColumns: <ul style="list-style-type: none"> • name • value

Destination information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	<ul style="list-style-type: none"> • Workspace • External 	Yes	type

Name	Description	Value	Required	JSON script property
Connection	Your connection to the source data store.	<your connection>	Yes	connection
Connection type	Your connection type. Select REST.	REST	Yes	type: RestResource
Relative URL	A relative URL to the resource that contains the data. When this property isn't specified, only the URL that's specified in the connection definition is used. The HTTP connector copies data from the combined URL: <code>[url specified in connection]/[relative URL specified]</code> .	<your relative url>	No	relativeUrl
Request method	The HTTP method. Allowed values are POST (default), PUT, and PATCH.	<ul style="list-style-type: none"> • POST • GET • PATCH 	No	requestMethod
Request timeout	The timeout (the timespan value) for the HTTP request to get a response. This value is the timeout to get a response, not the timeout to read response data. The default value is 00:01:40.	timespan	No	httpRequestTimeout
Request interval (ms)	The interval time between different requests for multiple pages in milliseconds. Request interval value should be a number between [10, 60000].	[10, 60000]	No	requestInterval
Write batch size	Number of records to write to the REST destination per batch. The default value is 10000.	<number of rows> (integer)	No	writeBatchSize
Http Compression type	HTTP compression type to use while sending data with Optimal Compression Level. Allowed values are None and GZip .	<ul style="list-style-type: none"> • None • GZip 	No	httpCompressionType

Name	Description	Value	Required	JSON script property
Additional headers	Additional HTTP request headers.	<your additional headers>	No	additionalHeaders

Next steps

- How to create a REST connection
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Salesforce objects connector overview

Article • 11/15/2023

The Salesforce objects connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to Salesforce objects in Dataflow Gen2, go to Set up your Salesforce objects connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support Salesforce objects in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your Salesforce objects connection

Article • 12/06/2023

This article outlines the steps to create a Salesforce objects connection.

Supported authentication types

The Salesforce objects connector supports the following authentication types for copy and Dataflow Gen2 respectively.

[+] Expand table

Authentication type	Copy	Dataflow Gen2
Organizational account	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Salesforce objects. The following links provide the specific Power Query connector information you need to connect to Salesforce objects in Dataflow Gen2:

- To get started using the Salesforce objects connector in Dataflow Gen2, go to Get data from Data Factory in Microsoft Fabric.
- Be sure to install or set up any Salesforce objects prerequisites before connecting to the Salesforce objects connector.
- To connect to the Salesforce objects connector from Power Query, go to Connect to Salesforce objects from Power Query Online.

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support Salesforce objects in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Salesforce reports connector overview

Article • 11/15/2023

The Salesforce reports connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to Salesforce reports in Dataflow Gen2, go to Set up your Salesforce reports connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support Salesforce reports in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your Salesforce reports connection

Article • 12/06/2023

This article outlines the steps to create a Salesforce reports connection.

Supported authentication types

The Salesforce reports connector supports the following authentication types for copy and Dataflow Gen2 respectively.

[+] Expand table

Authentication type	Copy	Dataflow Gen2
Organizational account	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to Salesforce reports. The following links provide the specific Power Query connector information you need to connect to Salesforce reports in Dataflow Gen2:

- To get started using the Salesforce reports connector in Dataflow Gen2, go to Get data from Data Factory in Microsoft Fabric.
- Be sure to install or set up any Salesforce reports prerequisites before connecting to the Salesforce reports connector.
- To connect to the Salesforce reports connector from Power Query, go to Connect to Salesforce reports from Power Query Online.

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support Salesforce reports in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

SAP BW Application Server connector overview

Article • 11/15/2023

The SAP Business Warehouse (BW) Application Server connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to an SAP BW Application Server in Dataflow Gen2, go to [Set up your SAP BW Application Server connection](#).

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support an SAP BW Application Server connection in data pipelines.

Feedback

Was this page helpful?



[Provide product feedback](#) | [Ask the community](#)

Set up your SAP BW Application Server connection

Article • 12/06/2023

This article outlines the steps to create an SAP BW Application Server connection.

Supported authentication types

The SAP BW Application Server connector supports the following authentication types for copy and Dataflow Gen2 respectively.

[+] Expand table

Authentication type	Copy	Dataflow Gen2
Basic (Username/Password)	n/a	✓
Windows	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to an SAP BW Application Server. The following links provide the specific Power Query connector information you need to connect to an SAP BW Application Server in Dataflow Gen2:

- To get started using the SAP BW Application Server connector in Dataflow Gen2, go to Get data from Data Factory in Microsoft Fabric.
- Be sure to install or set up any SAP BW Application Server prerequisites before connecting to the SAP BW Application Server connector.
- To connect to the SAP BW Application Server connector from Power Query, go to Connect to an SAP BW Application Server from Power Query Online.

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support an SAP BW Application Server in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

SAP BW Message Server connector overview

Article • 11/15/2023

The SAP Business Warehouse (BW) Message Server connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to an SAP BW Message Server in Dataflow Gen2, go to [Set up your SAP BW Message Server connection](#).

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support an SAP BW Message Server connection in data pipelines.

Feedback

Was this page helpful?



[Provide product feedback](#) | [Ask the community](#)

Set up your SAP BW Message Server connection

Article • 12/06/2023

This article outlines the steps to create an SAP BW Message Server connection.

Supported authentication types

The SAP BW Message Server connector supports the following authentication types for copy and Dataflow Gen2 respectively.

[+] Expand table

Authentication type	Copy	Dataflow Gen2
Basic (Username/Password)	n/a	✓
Windows	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to an SAP BW Message Server. The following links provide the specific Power Query connector information you need to connect to an SAP BW Message Server in Dataflow Gen2:

- To get started using the SAP BW Message Server connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- Be sure to install or set up any SAP BW Message Server prerequisites before connecting to the SAP BW Message Server connector.
- To connect to the SAP BW Message Server connector from Power Query, go to [Connect to an SAP BW Message Server from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support an SAP BW Message Server in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

SAP HANA connector overview

Article • 11/15/2023

The SAP HANA database connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to an SAP HANA database in Dataflow Gen2, go to [Set up your SAP HANA database connection](#).

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support an SAP HANA database connection in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#) | [Ask the community](#)

Set up your SAP HANA database connection

Article • 12/06/2023

This article outlines the steps to create an SAP HANA database connection.

Supported authentication types

The SAP HANA database connector supports the following authentication types for copy and Dataflow Gen2 respectively.

[+] Expand table

Authentication type	Copy	Dataflow Gen2
Basic (Username/Password)	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to an SAP HANA database. The following links provide the specific Power Query connector information you need to connect to an SAP HANA database in Dataflow Gen2:

- To get started using the SAP HANA database connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- Be sure to install or set up any SAP HANA database prerequisites before connecting to the SAP HANA database connector.
- To connect to the SAP HANA database connector from Power Query, go to [Connect to an SAP HANA database from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support an SAP HANA database in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

SFTP connector overview

Article • 11/15/2023

The SFTP connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

Data Factory in Microsoft Fabric doesn't currently support the SFTP connector in Dataflow Gen2.

Support in data pipelines

The SFTP connector supports the following capabilities in data pipelines:

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	Basic
Lookup activity	None	Basic
GetMetadata activity	None	Basic
Delete activity	None	Basic

To learn about the copy activity configuration for SFTP in data pipelines, go to [Configure SFTP in a copy activity](#).

Feedback

Was this page helpful?

👍 Yes👎 No

Provide product feedback [↗](#) | Ask the community [↗](#)

Configure SFTP in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in data pipeline to copy data from SFTP.

Supported format

SFTP supports the following file formats. Refer to each article for format-based settings.

- Avro format
- Binary format
- Delimited text format
- Excel format
- JSON format
- ORC format
- Parquet format
- XML format

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Destination
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

Go to **Source** tab to configure your copy activity source. See the following content for the detailed configuration.



The following three properties are required:

- **Data store type:** Select **External**.
- **Connection:** Select an SFTP connection from the connection list. If no connection exists, then create a new SFTP connection by selecting **New**.
- **File path type:** Select from **File path**, **Wildcard file path** and **List of files** based on the way that you want to read files.
 - **File path:** If you choose this type, specify your source file path. You can select **Browse** to select your source files or enter your file path manually.
 - **Wildcard file path:** If you choose this type, specify the **Wildcard paths** to filter your source folders or files.

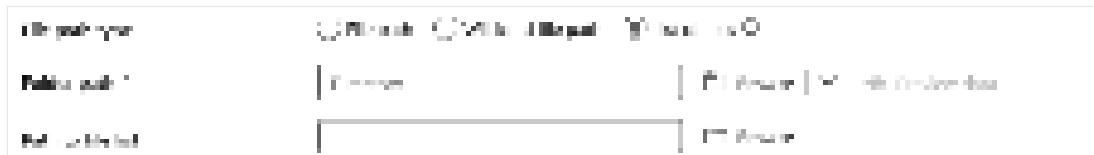
Allowed wildcards are ***** (matches zero or more characters) and **?** (matches zero or single character). Use **^** to escape if your folder name has a wildcard or this escape character inside. For more examples, go to Folder and file filter examples.



Wildcard folder path: Specify the folder path with wildcard characters to filter source folders.

Wildcard file name: Specify the file name with wildcard characters under the given folder path/wildcard folder path to filter source files.

- **List of files:** If you select this type, specify the **Folder path** and **Path to file list** to indicates to copy a given file set. Point to a text file that includes a list of files you want to copy, one file per line. For more examples, go to File list examples.
- **Folder path:** Specify the path to your source folder. It is required.
- **Path to file list:** Specify the path of the text file that includes a list of files you want to copy.



- **File format:** Select the file format applied from the drop-down list. Select **Settings** to configure the file format. For settings of different file formats, refer to articles in Supported format for detailed information.

Under **Advanced**, you can specify the following fields:

- **Filter by last modified:** Files are filtered based on the last modified dates that you specified. This property doesn't apply when you configure your file path type as **List of files**.
 - **Start time (UTC):** The files are selected if their last modified time is greater than or equal to the configured time.
 - **End time (UTC):** The files are selected if their last modified time is less than the configured time.

When **Start time (UTC)** has datetime value but **End time (UTC)** is NULL, it means the files whose last modified attribute is greater than or equal with the datetime value will be selected. When **End time (UTC)** has datetime value but **Start time (UTC)** is NULL, it means the files whose last modified attribute is less than the datetime value will be selected. The properties can be NULL, which means no file attribute filter will be applied to the data.

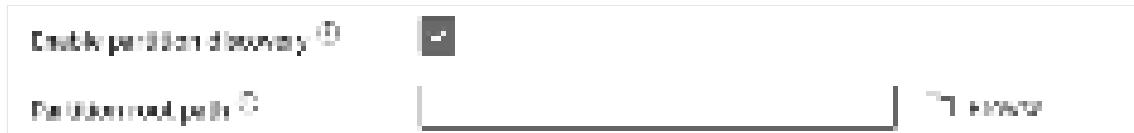
- **Disable chunking:** The chunking is designed to optimize the performance and happens underneath. This option allows you to disable chunking within each file. When copying data from SFTP, the service tries to get the file length first, then divide the file into multiple parts and read them in parallel. Specify whether your SFTP server supports getting file length or seeking to read from a certain offset. It is unselected by default.
- **Enable partition discovery:** Specify whether to parse the partitions from the file path and add them as additional source columns. It is unselected by default and not supported when you use binary file format.

- **Partition root path:** When partition discovery is enabled, specify the absolute root path in order to read partitioned folders as data columns.
If it is not specified, by default,
 - When you use file path or list of files on source, partition root path is the path that you configured.
 - When you use wildcard folder filter, partition root path is the sub-path before the first wildcard.

For example, assuming you configure the path as

`root/folder/year=2020/month=08/day=27`:

- If you specify partition root path as `root/folder/year=2020`, copy activity will generate two more columns month and day with value "08" and "27" respectively, in addition to the columns inside the files.
- If partition root path is not specified, no extra column will be generated.



- **Max concurrent connections:** This property indicates the upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.
- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter. For more information, go to Add additional columns during copy.

Destination

Go to **Destination** tab to configure your copy activity destination. See the following content for the detailed configuration.



- **Data store type:** Select **External**.
- **Connection:** Select an SFTP connection from the connection list. If no connection exists, then create a new SFTP connection by selecting **New**.
- **File path:** Specify the file path to write your data to. You can select **Browse** to select your source files or enter your file path manually.
- **File format:** Select the file format applied from the drop-down list. Select **Settings** to configure the file format. For settings of different file formats, refer to articles in Supported format for detailed information.

Under **Advanced**, you can specify the following fields:

- **Copy behavior:** Specify the copy behavior when the source is files from a file-based data store. Choose from the following properties.
 - **Flatten hierarchy:** All files from the source folder are in the first level of the target folder. The target files have autogenerated names.
 - **Merge files:** Merges all files from the source folder to one file. If the file name is specified, the merged file name is the specified name. Otherwise, it's an auto-generated file name.
 - **Preserve hierarchy (default):** Preserves the file hierarchy in the target folder. The relative path of the source file to the source folder is identical to the relative path of the target file to the target folder.
 - **Add dynamic content:** Select this to specify your copy behavior using dynamic content.
- **Max concurrent connections:** The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.
- **Operation timeout (minutes):** Specify the timeout for writing each chunk to SFTP server. The default value is 60 minutes.

- **Upload with temp file:** Specify whether to upload to temporary files and rename them, or directly write to the target folder or file location. By default, it is selected and the service first writes to temporary files and then renames them when the upload is finished.

This sequence helps to (1) avoid conflicts that might result in a corrupted file if you have other processes writing to the same file, and (2) ensure that the original version of the file exists during the transfer. If your SFTP server doesn't support a rename operation, disable this option and make sure that you don't have a concurrent write to the target file.

💡 Tip

If you receive the error "UserErrorSftpPathNotFound," "UserErrorSftpPermissionDenied," or "SftpOperationFail" when you're writing data into SFTP, and the SFTP user you use does have the proper permissions, check to see whether your SFTP server support file rename operation is working. If it isn't, disable the **Upload with temp file** option and try again.

Mapping

For **Mapping** tab configuration, see Configure your mappings under mapping tab. If you choose Binary as your file format, mapping will not be supported.

Settings

For **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

The following table contains more information about the copy activity in SFTP.

Source

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your SFTP connection to the source data store.	< your SFTP connection	Yes	connection

Name	Description	Value	Required	JSON script property
		>		
File path type	The file path type used to get source data.	<ul style="list-style-type: none"> • File path • Wildcard file path • List of files 	Yes	/
File path	The path to the source file.	< file path>	Yes	fileName folderPath
Wildcard paths	The wildcard path to the source file.	< your wildcard file path >	Yes for Wildcard file name	wildcardFolderPath wildcardFileName
Folder path	The path to your source folder.	< your folder path>	Yes	folderPath
Path to file list	Indicates to copy a given file set. Point to a text file that includes a list of files you want to copy, one file per line.	< file list path >	No	fileListPath
File format	The file format for your source data. For the information of different file formats, refer to articles in Supported format for detailed information.	/	Yes	/
Filter by last modified	The files with last modified time in the range [Start time, End time) will be filtered for further processing. The time will be applied to UTC time zone in the format of <code>yyyy-mm-ddThh:mm:ss.ffffZ</code> . These properties can be skipped which means no file attribute filter will be applied. This property doesn't apply when you configure your file path type as List of files.	datetime	No	modifiedDatetimeStart modifiedDatetimeEnd

Name	Description	Value	Required	JSON script property
Disable chunking	The chunking is designed to optimize the performance and happens underneath. This option allows you to disable chunking within each file. When copying data from SFTP, the service tries to get the file length first, then divide the file into multiple parts and read them in parallel. Specify whether your SFTP server supports getting file length or seeking to read from a certain offset.	selected or unselected (default)	No	disableChunking: true or false (default)
Enable partition discovery	Indicates whether to parse the partitions from the file path and add them as additional source columns.	selected or unselected (default)	No	enablePartitionDiscovery: true or false (default)
Partition root path	The absolute partition root path in order to read partitioned folders as data columns. Specify it when partition discovery is enabled.	< partition root path >	No	partitionRootPath
Max concurrent connections	The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.	< upper limit of concurrent connections > (integer)	No	maxConcurrentConnections
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter. For more information, go to Add	<ul style="list-style-type: none"> • Name • Value 	No	additionalColumns: <ul style="list-style-type: none"> • name • value

Name	Description	Value	Required	JSON script property
	additional columns during copy			

Destination

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source SFTP.	< your connection >	Yes	connection
File path	The file path of your destination data.	File path of the destination	Yes	folderPath fileName
File format	The file format for your source data. For the information of different file formats, refer to articles in Supported format for detailed information.	/	Yes	/
Copy behavior	Defines the copy behavior when the source is files from a file-based data store.	<ul style="list-style-type: none"> • Flatten hierarchy • Merge files • Preserve hierarchy 	No	copyBehavior: - FlattenHierarchy - MergeFiles - PreserveHierarchy
Max concurrent connections	The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.	< max concurrent connections >	No	maxConcurrentConnections
Operation timeout (minutes)	The timeout for writing each chunk to SFTP server.	< your operation timeout > The default value is 60	No	operationTimeout

Name	Description	Value	Required	JSON script property
Upload with temp file	Indicates whether to upload temporary file(s) and rename. Disable this option if your SFTP server doesn't support rename operation.	selected (default) or unselected	No	useTempFileRename: true (default) or false

Next steps

- SFTP connector overview
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback [↗](#) | Ask the community [↗](#)

SharePoint folder connector overview

Article • 11/15/2023

The SharePoint folder connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to a SharePoint folder in Dataflow Gen2, go to Set up your SharePoint folder connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support a SharePoint folder in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your SharePoint folder connection

Article • 12/06/2023

This article outlines the steps to create a SharePoint folder connection.

Supported authentication types

The SharePoint folder connector supports the following authentication types for copy and Dataflow Gen2 respectively.

[+] Expand table

Authentication type	Copy	Dataflow Gen2
Anonymous	n/a	✓
Windows	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to a SharePoint folder. The following links provide the specific Power Query connector information you need to connect to a SharePoint folder in Dataflow Gen2:

- To get started using the SharePoint folder connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- To determine the URL to use to access your SharePoint folder, go to [Determine the site URL](#).
- To connect to the SharePoint folder connector from Power Query, go to [Connect to a SharePoint folder from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support a SharePoint folder in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

SharePoint list connector overview

Article • 11/15/2023

The SharePoint list connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to a SharePoint folder in Dataflow Gen2, go to Set up your SharePoint list connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support a SharePoint list in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your SharePoint list connection

Article • 12/06/2023

This article outlines the steps to create a SharePoint list connection.

Supported authentication types

The SharePoint list connector supports the following authentication types for copy and Dataflow Gen2 respectively.

 Expand table

Authentication type	Copy	Dataflow Gen2
Anonymous	n/a	✓
Windows	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to a SharePoint list. The following links provide the specific Power Query connector information you need to connect to a SharePoint list in Dataflow Gen2:

- To get started using the SharePoint list connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- To determine the URL to use to access your SharePoint list, go to [Determine the site URL](#).
- To connect to the SharePoint list connector from Power Query, go to [Connect to a SharePoint list from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support a SharePoint list in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

SharePoint Online List connector overview

Article • 11/15/2023

The SharePoint Online List connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to a SharePoint Online List in Dataflow Gen2, go to Set up your connection in Dataflow Gen2.

Support in data pipelines

The SharePoint Online List connector supports the following capabilities in data pipelines:

Supported capabilities	Gateway	Authentication
Copy activity (source/-)	None	Service principal
Lookup activity	None	Service principal

To learn about how to connect to a SharePoint Online List in data pipelines, go to Set up your SharePoint Online List connection.

To learn about the copy activity configuration for a SharePoint Online List in data pipelines, go to Configure SharePoint Online List in a copy activity.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your SharePoint Online List connection

Article • 11/15/2023

This article outlines the steps to create a SharePoint Online List connection.

Supported authentication types

The SharePoint Online List connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Organizational account	n/a	✓
Service Principal	✓	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to a SharePoint Online List. The following links provide the specific Power Query connector information you need to connect to a SharePoint Online List in Dataflow Gen2:

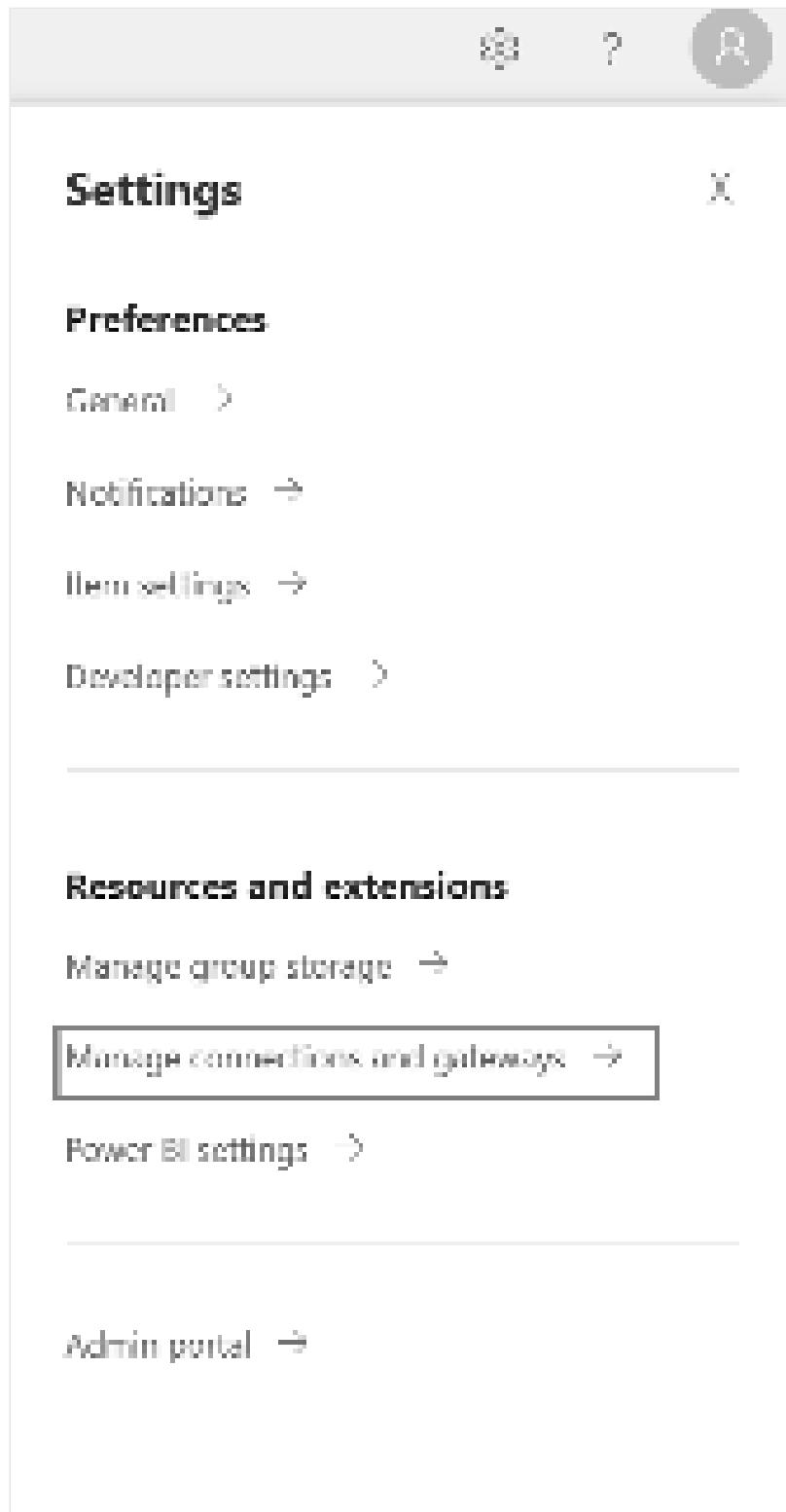
- To get started using the SharePoint Online list connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- To determine the URL to use to access your SharePoint Online list, go to [Determine the site URL](#).
- To connect to the Azure Blobs connector from Power Query, go to [Connect to a SharePoint Online list from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

To create a connection in a data pipeline:

1. From the page header in Data Factory service, select **Settings**  > **Manage connections and gateways**.



2. Select **New** at the top of the ribbon to add a new data source.

[New](#)

[Get Help](#)

Data (preview)

[Connections](#)

[On-premises data gateway](#)

[Virtual network data gateway](#)



The **New connection** pane appears on the left side of the page.



Setup connection

Step 1: Specify the new connection name, type, and URL

New connection



Currently, these cloud connections are only supported for Data Pipelines and Kusto. In the future, other artifacts can also make use of the cloud connections. To create personal cloud connections in Datasets, Databricks, and Dataflows, use the Power Query Online experience in "Get data".



Connection name *

SharePointOnlineList

Connection type *

SharePoint

SharePoint site URL *

< Your URL >

In the New connection pane, choose **Cloud**, and specify the following fields:

- **Connection name:** Specify a name for your connection.
- **Connection type:** Select **SharePoint** for your connection type.
- **URL:** The SharePoint Online site URL, for example
`https://contoso.sharepoint.com/sites/siteName`.

Step 2: Select and set your authentication

Under **Authentication method**, select your authentication from the drop-down list and complete the related configuration. The SharePoint Online List connector supports the following authentication types.

- Service Principal

Authentication

Authentication method *

Anonymous 

Anonymous

OAuth2

Service Principal

Service Principal authentication

Select **Service Principal** under **Authentication method**, and fill in the required properties.

Authentication

Authentication method *

Service Principal 

Tenant ID *

< Your Tenant ID >

Service principal ID *

< Your Service Principal ID >

Service principal key *

< Your Service Principal key >

- **Tenant Id:** The tenant ID under which your application resides.
- **Service principal ID:** The Application (client) ID of the application registered in Azure Active Directory. Refer to Prerequisites for more details, including the permission settings.
- **Service principal key:** The application's key.

Step 3: Specify the privacy level that you want to apply

In the General tab, select the privacy level that you want to apply in the Privacy level drop-down list. Three privacy levels are supported. For more information, go to [privacy levels](#).

Step 4: Create your connection

Select **Create**. Your creation is successfully tested and saved if all the credentials are correct. If not correct, the creation fails with errors.

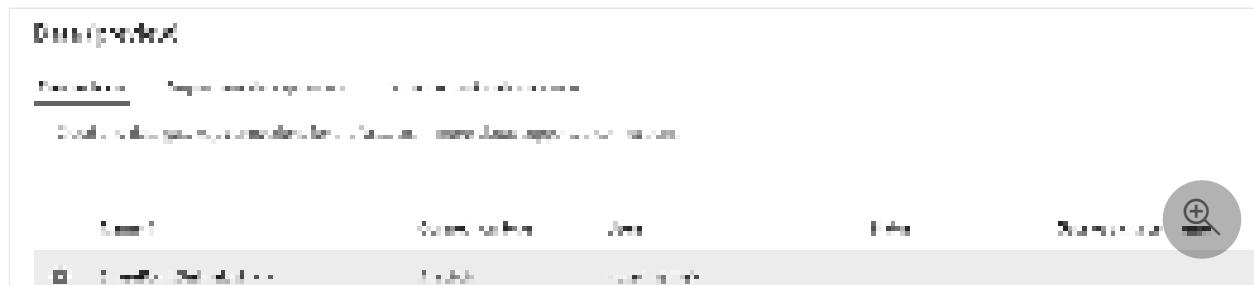


Table summary

The following table contains the connector properties supported in a pipeline copy.

Name	Description	Required	Property	Copy
Connection name	A name for your connection.	Yes		✓
Connection type	Select SharePoint for your connection type. If no connection exists, then create a new connection by selecting New .	Yes		✓
URL	The SharePoint Online site URL, for example <code>https://contoso.sharepoint.com/sites/siteName</code> .	Yes		✓
Authentication	Go to Authentication	Yes		Go to Authentication
Privacy Level	The privacy level that you want to apply. Allowed values are Organizational , Privacy , Public	Yes		✓

Authentication

The following table contains the supported authentication type properties.

Name	Description	Required	Property	Copy
Service Principal				✓
- Tenant ID	The tenant ID under which your application resides.	Yes		
- Service Principal ID	The Application (client) ID of the application registered in Azure Active Directory. Refer to Prerequisites for more details including the permission settings.	Yes		
- Service Principal key	The application's key.	Yes		

Next steps

- Configure SharePoint Online List in a copy activity
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Configure SharePoint Online List in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in a data pipeline to copy data from SharePoint Online List.

Prerequisites

1. The SharePoint List Online connector uses service principal authentication to connect to SharePoint. Follow these steps to set it up:
 2. Register an application with the Microsoft Identity platform. To learn how, go to Quickstart: Register an application with the Microsoft identity platform. Make note of these values, which you use to define the connection:
 - Application ID
 - Application key
 - Tenant ID
 3. Use the following steps to grant SharePoint Online site permission to your registered application. To grant permission, you need a site admin role.
 - a. Open a SharePoint Online site link, for example
`https://[your_site_url]/_layouts/15/appinv.aspx` (replace the site URL).
 - b. Search the application ID you registered, fill the empty fields, and then select **Create**.
 - App Domain: `contoso.com`
 - Redirect URL: `https://www.contoso.com`
 - Permission Request XML:

XML

```
<AppPermissionRequests AllowAppOnlyPolicy="true">
  <AppPermissionRequest
    Scope="http://sharepoint/content/sitecollection/web"
    Right="Read"/>
</AppPermissionRequests>
```

App Id

<application GUID>

Lookup

Title

spodemo

App Domain

localhost.com

Example: "www.contoso.com"

Redirect URL:

https://www.localhost.com

Example: "https://www.contoso.com/default.aspx"

Permission Request XML:

```
<AppPermissionRequests  
AllowAppOnlyPolicy="true">  
    <AppPermissionRequest  
        Scope="http://sharepoint/content/sitecollection/we  
b" Right="Read"/>  
</AppPermissionRequests>
```

① Note

In the context of configuring the SharePoint connector, the **App Domain** and **Redirect URL** refer to the SharePoint app that you've registered in Azure Active Directory (Azure AD) to allow access to your SharePoint data. The **App Domain** is the domain where your SharePoint site is hosted. For example, if your SharePoint site is located at <https://contoso.sharepoint.com>, then the **App Domain** would be contoso.sharepoint.com. The **Redirect URL** is the URL that the SharePoint app redirects to after the user has authenticated and granted permissions.

to the app. This URL should be a page on your SharePoint site that the app has permission to access. For example, you could use the URL of a page that displays a list of files in a library, or a page that displays the contents of a document.

- c. Select **Trust It** for this app.

Supported configuration

For the configuration of each tab under a copy activity, go to the following sections respectively.

- General
- Source
- Mapping
- Settings

General

For **General** tab configuration, go to General.

Source

The following properties are supported for SharePoint Online List under the **Source** tab of a copy activity.

The screenshot shows the 'Source' tab configuration for a SharePoint Online List. The tabs at the top are General, Source, Destination, Mapping, and Settings. The 'Source' tab is selected. The configuration includes:

- Data store type:** External (radio button selected).
- Connection:** SharePointOnlineList (dropdown menu).
- Uri query:** (radio button selected) and (Query) (radio button selected).
- List name:** Hong (dropdown menu).
- Advanced:** Refresh timeout (dropdown menu set to 30000).
- Additional columns:** + New (button) and a search icon.

The following properties are **required**:

- **Data store type:** Select **External**.

- **Connection:** Select a SharePoint Online List connection from the connection list. If no connection exists, then create a new HTTP connection by selecting **New**.
- **Use query:** Choose either **List name** or **Query** as your use query. The configuration of each setting is:
 - **List name:** The name of the SharePoint Online list.



- **Query:** The OData query to filter the data in SharePoint Online list. For example, `"$top=1"`.



Under **Advanced**, you can specify the following fields:

- **Request timeout:** The wait time to get a response from SharePoint Online. Default value is 5 minutes (00:05:00).
- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.



Mapping

For **Mapping** tab configuration, go to Configure your mappings under mapping tab.

Settings

For **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

The following table contains more information about a copy activity in SharePoint Online List.

Source

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	<your connection>	Yes	connection
Use query	You can choose List name or Query as your use query.	-List name -Query	No	type
List name	The name of the SharePoint Online list.	<your connection>	Yes	listName
Request timeout	The wait time to get a response from SharePoint Online. Default value is 5 minutes (00:05:00).	timespan	No	requestTimeout
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.	• Name • Value	No	additionalColumns: • name • value

Next steps

- Set up your SharePoint Online List connection

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Snowflake connector overview

Article • 11/15/2023

The Snowflake connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

To learn about how to connect to a Snowflake database in Dataflow Gen2, go to Set up your connection in Dataflow Gen2.

Support in data pipelines

The Snowflake connector supports the following capabilities in data pipelines.

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	Snowflake
Lookup activity	None	Snowflake
Script activity	None	Snowflake

To learn about the copy activity configuration for Snowflake in data pipelines, go to Configure Snowflake in a copy activity.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your Snowflake database connection

Article • 11/15/2023

This article outlines the steps to create a Snowflake database connection.

Supported authentication types

The Snowflake database connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Snowflake	√	√
Microsoft account	n/a	√

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to a Snowflake database. The following links provide the specific Power Query connector information you need to connect to Snowflake database in Dataflow Gen2:

- To get started using the Snowflake database connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric \(Preview\)](#).
- To connect to the Snowflake database connector from Power Query, go to [Connect to a Snowflake database from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback ↗](#) | [Ask the community ↗](#)

Configure Snowflake in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in data pipeline to copy data from and to Snowflake.

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Destination
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

The following properties are supported for Snowflake under the **Source** tab of a copy activity.



The following properties are required:

- **Data store type:** Select **External**.
- **Connection:** Select a Snowflake connection from the connection list. If the connection doesn't exist, then create a new Snowflake connection by selecting **New**.
- **Database:** The default database to use once connected. It should be an existing database for which the specified role has privileges.
- **Use query:** You can choose either **Table** or **Query** as your use query. The following list describes the configuration of each setting.
 - **Table:** Select the table in your database from the drop-down list. Or check **Edit** to enter your table name manually.
 - **Query:** Specify the SQL query to read data from Snowflake. If the names of the schema, table and columns contain lower case, quote the object identifier in query e.g. `select * from "schema"."myTable"`.

Under **Advanced**, you can specify the following fields:

- **Additional Snowflake copy options:** Specify additional Snowflake copy options which will be used in Snowflake COPY statement to load data. Additional copy options are provided as a dictionary of key-value pairs. Examples: MAX_FILE_SIZE, OVERWRITE. For more information, see [Snowflake Copy Options](#).

The screenshot shows a user interface for configuring 'Additional Snowflake copy options'. At the top, there is a header with a minus sign and the word 'Advanced'. Below it is another header with a minus sign and 'Additional Snowflake copy options'. Underneath these are two buttons: '+ New' and 'Delete'. A table follows, with columns 'Property name' and 'Value'. There is one row in the table containing a single cell with a checkbox icon. At the bottom of the table is a button labeled 'Add dynamic column (Alt+Shift+D)'.

Property name	Value
<input type="checkbox"/>	

- **Additional Snowflake format options:** Specify additional Snowflake format options, which will be used in Snowflake COPY statement to load data. Additional file format options provided to the COPY command are provided as a dictionary of key-value pairs. Examples: DATE_FORMAT, TIME_FORMAT, TIMESTAMP_FORMAT. For more information, see [Snowflake Format Type Options](#).



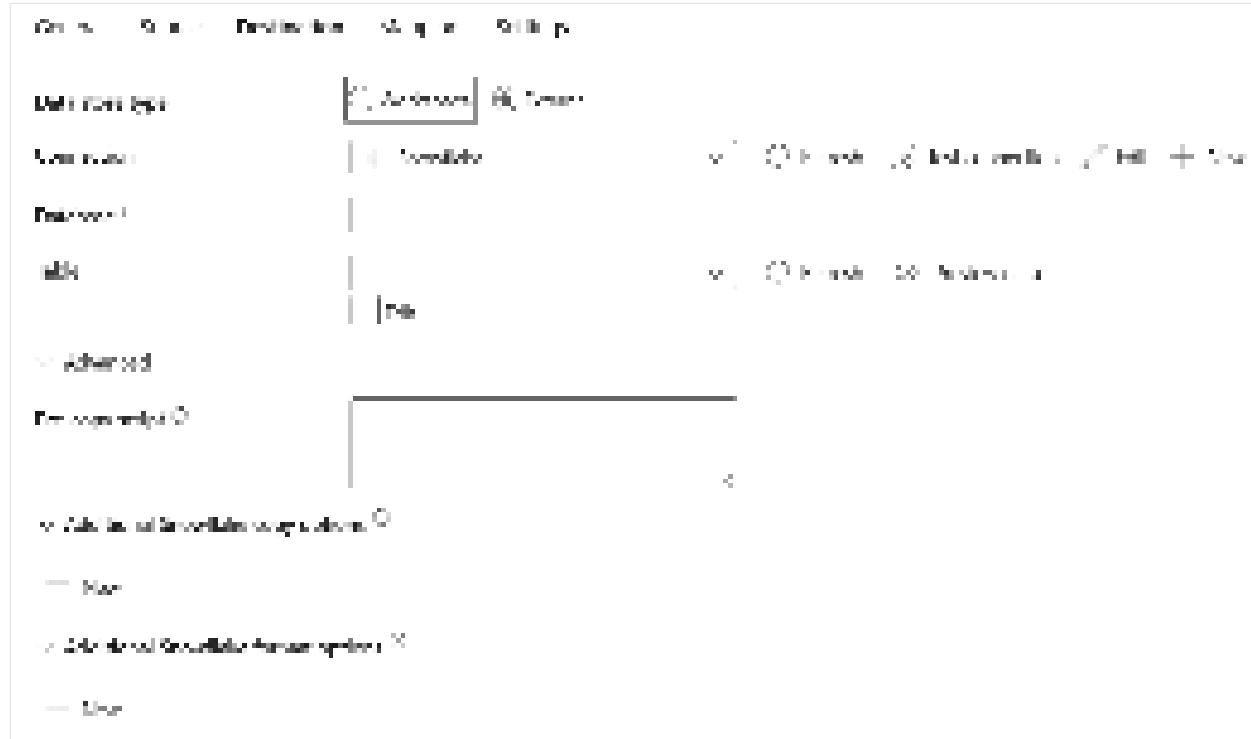
Direct copy from Snowflake

If your destination data store and format meet the criteria described in this section, you can use the Copy activity to directly copy from Snowflake to destination. The service checks the settings and fails the Copy activity run if the following criteria is not met:

- The **destination connection** is **Azure Blob storage** with **shared access signature** authentication. If you want to directly copy data to Azure Data Lake Storage Gen2 in the following supported format, you can create an Azure Blob connection with SAS authentication against your ADLS Gen2 account.
- The **destination data format** is of **Parquet**, **DelimitedText**, or **JSON** with the following configurations:
 - For **Parquet** format, the compression codec is **None**, **Snappy**, or **Lzo**.
 - For **DelimitedText** format:
 - Row delimiter is **\r\n**, or any single character.
 - Compression type can be **None**, **gzip**, **bzip2**, or **deflate**.
 - Encoding is left as default or set to **UTF-8**.
 - Quote character is **Double quote**, **Single quote**, or **No quote character**.
 - For **JSON** format, direct copy only supports the case that source Snowflake table or query result only has single column and the data type of this column is **VARIANT**, **OBJECT**, or **ARRAY**.
 - Compression type can be **None**, **gzip**, **bzip2**, or **deflate**.
 - Encoding is left as default or set to **UTF-8**.
 - File pattern in copy activity destination is left as default or set to **Set of objects**.
- In copy activity source, **Additional columns** is not specified.
- Column mapping is not specified.

Destination

The following properties are supported for Snowflake under the Destination tab of a copy activity.

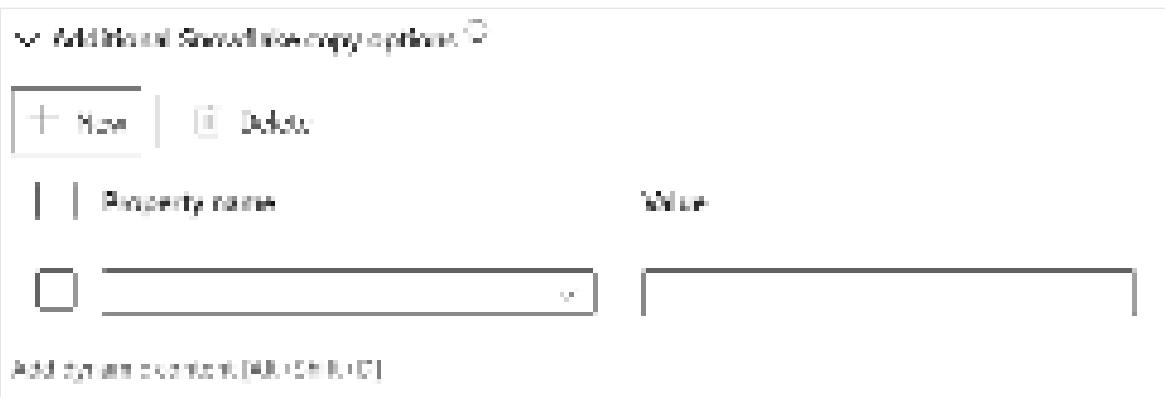


The following properties are **required**:

- **Data store type:** Select **External**.
- **Connection:** Select a Snowflake connection from the connection list. If the connection doesn't exist, then create a new Snowflake connection by selecting **New**.
- **Database:** The default database to use once connected. It should be an existing database for which the specified role has privileges.
- **Table:** Select the table in your database from the drop-down list. Or check **Edit** to enter your table name manually.

Under **Advanced**, you can specify the following fields:

- **Pre-copy script:** Specify a script for Copy Activity to execute before writing data into destination table in each run. You can use this property to clean up the pre-loaded data.
- **Additional Snowflake copy options:** Specify additional Snowflake copy options, which will be used in Snowflake COPY statement to load data. Additional copy options are provided as a dictionary of key-value pairs. Examples: ON_ERROR, FORCE, LOAD_UNCERTAIN_FILES. For more information, see [Snowflake Copy Options](#).



- **Additional Snowflake format options:** Specify additional Snowflake format options, which will be used in Snowflake COPY statement to load data. Additional file format options provided to the COPY command are provided as a dictionary of key-value pairs. Examples: DATE_FORMAT, TIME_FORMAT, TIMESTAMP_FORMAT. For more information, see [Snowflake Format Type Options](#).



Direct copy to Snowflake

If your source data store and format meet the criteria described in this section, you can use the Copy activity to directly copy from source to Snowflake. The service checks the settings and fails the Copy activity run if the following criteria is not met:

- The **source connection** is **Azure Blob storage** with **shared access signature** authentication. If you want to directly copy data from Azure Data Lake Storage Gen2 in the following supported format, you can create an Azure Blob connection with SAS authentication against your ADLS Gen2 account.
- The **source data format** is **Parquet**, **DelimitedText**, or **JSON** with the following configurations:
 - For **Parquet** format, the compression codec is **None**, or **Snappy**.
 - For **DelimitedText** format:
 - **Row delimiter** is `\r\n`, or any single character. If row delimiter is not `\r\n`, **First row as header** is unselected, and **Skip line count** is not specified.
 - **Compression type** can be **None**, **gzip**, **bzip2**, or **deflate**.

- **Encoding** is left as default or set to "UTF-8", "UTF-16", "UTF-16BE", "UTF-32", "UTF-32BE", "BIG5", "EUC-JP", "EUC-KR", "GB18030", "ISO-2022-JP", "ISO-2022-KR", "ISO-8859-1", "ISO-8859-2", "ISO-8859-5", "ISO-8859-6", "ISO-8859-7", "ISO-8859-8", "ISO-8859-9", "WINDOWS-1250", "WINDOWS-1251", "WINDOWS-1252", "WINDOWS-1253", "WINDOWS-1254", "WINDOWS-1255".
- **Quote character** is **Double quote**, **Single quote**, or **No quote character**.
- For **JSON** format, direct copy only supports the case that destination Snowflake table only has single column and the data type of this column is **VARIANT**, **OBJECT**, or **ARRAY**.
 - **Compression type** can be **None**, **gzip**, **bzip2**, or **deflate**.
 - **Encoding** is left as default or set to **UTF-8**.
 - Column mapping is not specified.
- In the Copy activity source:
 - **Additional columns** is not specified.
 - If your source is a folder, **Recursively** is selected.
 - **Prefix**, **Start time (UTC)** and **End time (UTC)** in **Filter by last modified** and **Enable partition discovery** are not specified.

Mapping

For **Mapping** tab configuration, go to Configure your mappings under mapping tab.

Settings

For **Settings** tab configuration, go to Configure your other settings under settings tab.

Table summary

The following tables contain more information about the copy activity in Snowflake.

Source

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	< your connection	Yes	connection

Name	Description	Value	Required	JSON script property
		>		
Database	Your database that you use as source.	< your database >	Yes	database
Use query	The way to read data from Snowflake.	<ul style="list-style-type: none"> • Table • Query 	No	<ul style="list-style-type: none"> • table • query
Table	The name of the table to read data.	< name of your source table>	Yes	schema table
Query	The SQL query to read data from Snowflake.	< name of your source query>	Yes	query
Additional Snowflake copy options	Additional copy options, provided as a dictionary of key-value pairs. Examples: MAX_FILE_SIZE, OVERWRITE. For more information, see Snowflake Copy Options .	<ul style="list-style-type: none"> • Name • Value 	No	additionalCopyOptions
Additional Snowflake format options	Additional file format options that are provided to COPY command as a dictionary of key-value pairs. Examples: DATE_FORMAT, TIME_FORMAT, TIMESTAMP_FORMAT. For more information, see Snowflake Format Type Options .	<ul style="list-style-type: none"> • Name • Value 	No	additionalFormatOptions

Destination

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the destination data store.	< your connection >	Yes	connection
Database	Your database that you use as destination.	< your database>	Yes	/

Name	Description	Value	Required	JSON script property
Table	Your destination data table.	< name of your destination table>	Yes	<ul style="list-style-type: none"> • schema • table
Pre-copy script	A SQL query for the Copy activity to run before writing data into Snowflake in each run. Use this property to clean up the preloaded data.	< your pre-copy script>	NO	preCopyScript
Additional Snowflake copy options	Additional copy options, provided as a dictionary of key-value pairs. Examples: ON_ERROR, FORCE, LOAD_UNCERTAIN_FILES. For more information, see Snowflake Copy Options ↗ .	<ul style="list-style-type: none"> • Name • Value 	No	additionalCopyOptions
Additional Snowflake format options	Additional file format options provided to the COPY command, provided as a dictionary of key-value pairs. Examples: DATE_FORMAT, TIME_FORMAT, TIMESTAMP_FORMAT. For more information, see Snowflake Format Type Options ↗ .	<ul style="list-style-type: none"> • Name • Value 	No	additionalFormatOptions

Next steps

- Snowflake connector overview

Feedback

Was this page helpful?

Provide product feedback [↗](#) | Ask the community [↗](#)

SQL Server database connector overview

Article • 11/15/2023

The SQL Server database connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to a SQL Server database in Dataflow Gen2, go to [Set up your SQL Server database connection](#).

Support in Data pipeline

The SQL Server database connector supports the following capabilities in Data pipeline:

Supported capabilities	Gateway	Authentication
Copy activity (source/destination)	None	Basic
Lookup activity	None	Basic
GetMetadata activity	None	Basic
Script activity	None	Basic
Stored procedure activity	None	Basic

To learn more about the copy activity configuration for SQL Server database in Data pipeline, go to [Configure in a data pipeline copy activity](#).

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#) | [Ask the community](#)

Set up your SQL Server database connection

Article • 12/06/2023

This article outlines the steps to create a SQL Server database connection.

Supported authentication types

The SQL Server database connector supports the following authentication types for copy and Dataflow Gen2 respectively.

[+] Expand table

Authentication type	Copy	Dataflow Gen2
Basic (Username/Password)	n/a	✓
Organizational account	n/a	✓
Windows	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to a SQL Server database. The following links provide the specific Power Query connector information you need to connect to a SQL Server database in Dataflow Gen2:

- To get started using the SQL Server database connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric](#).
- Be sure to install or set up any SQL Server database prerequisites before connecting to the SQL Server database connector.
- To connect to the SQL Server database connector from Power Query, go to [Connect to SQL Servers database from Power Query Online](#).

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Configure SQL server in a copy activity

Article • 11/15/2023

This article outlines how to use the copy activity in data pipeline to copy data from and to SQL server.

Supported configuration

For the configuration of each tab under copy activity, go to the following sections respectively.

- General
- Source
- Destination
- Mapping
- Settings

General

Refer to the **General** settings guidance to configure the **General** settings tab.

Source

The following properties are supported for SQL server under the **Source** tab of a copy activity.



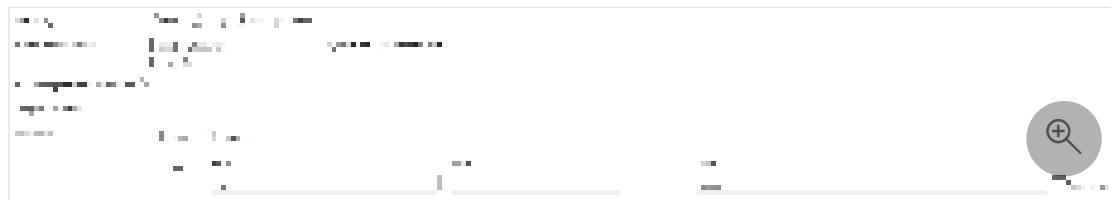
The following properties are **required**:

- **Data store type:** Select **External**.

- **Connection:** Select a SQL server connection from the connection list. If the connection doesn't exist, then create a new SQL server connection by selecting **New**.
- **Connection type:** Select **SQL server**.
- **Use query:** Specify the way to read data. You can choose **Table**, **Query**, or **Stored procedure**. The following list describes the configuration of each setting:
 - **Table:** Read data from the table specified. Select your source table from the drop-down list or select **Edit** to enter it manually.
 - **Query:** Specify the custom SQL query to read data. An example is `select * from MyTable`. Or select the pencil icon to edit in code editor.

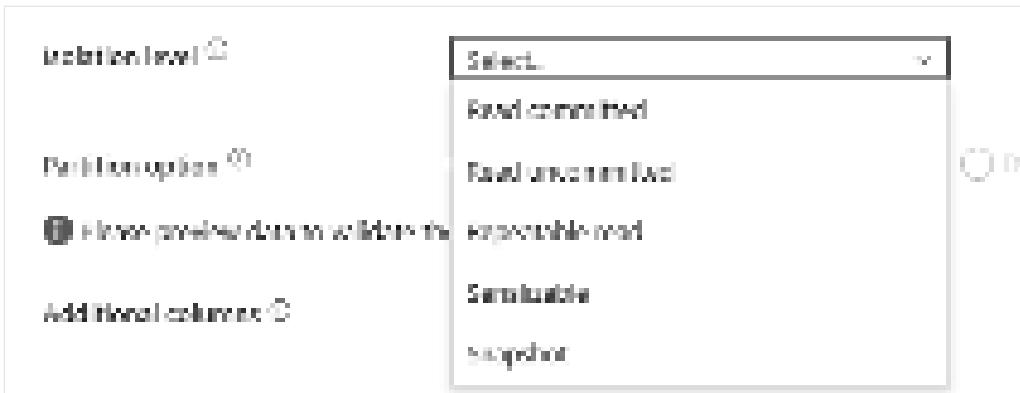


- **Stored procedure:** Use the stored procedure that reads data from the source table. The last SQL statement must be a SELECT statement in the stored procedure.
- **Stored procedure name:** Select the stored procedure or specify the stored procedure name manually when selecting the **Edit** to read data from the source table.
- **Stored procedure parameters:** Specify values for stored procedure parameters. Allowed values are name or value pairs. The names and casing of parameters must match the names and casing of the stored procedure parameters. You can select **Import parameters** to get your stored procedure parameters.



Under **Advanced**, you can specify the following fields:

- **Query timeout (minutes):** Specify the timeout for query command execution, default is 120 minutes. If a parameter is set for this property, allowed values are timespan, such as "02:00:00" (120 minutes).
- **Isolation level:** Specifies the transaction locking behavior for the SQL source. The allowed values are: **None**, **Read committed**, **Read uncommitted**, **Repeatable read**, **Serializable**, or **Snapshot**. If not specified, the database's default isolation level is used. Refer to **IsolationLevel** Enum for more details.



- **Partition option:** Specify the data partitioning options used to load data from SQL server. Allowed values are: **None** (default), **Physical partitions of table**, and **Dynamic range**. When a partition option is enabled (that is, not **None**), the degree of parallelism to concurrently load data from an SQL server is controlled by **Degree of copy parallelism** in copy activity settings tab.
 - **None:** Choose this setting to not use a partition.
 - **Physical partitions of table:** When using a physical partition, the partition column and mechanism are automatically determined based on your physical table definition.
 - **Dynamic range:** When using query with parallel enabled, the range partition parameter(`?AdfDynamicRangePartitionCondition`) is needed. Sample query: `SELECT * FROM <TableName> WHERE ?AdfDynamicRangePartitionCondition.`
 - **Partition column name:** Specify the name of the source column in **integer or date/datetime** type (`int`, `smallint`, `bigint`, `date`, `smalldatetime`, `datetime`, `datetime2`, or `datetimeoffset`) that's used by range partitioning for parallel copy. If not specified, the index or the primary key of the table is auto-detected and used as the partition column.
If you use a query to retrieve the source data, hook `?AdfDynamicRangePartitionCondition` in the WHERE clause. For an example, see the Parallel copy from SQL database section.
 - **Partition upper bound:** Specify the maximum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering the rows in table. All rows in the table or query result will be partitioned and copied. If not specified, copy activity auto detect the value. For an example, see the Parallel copy from SQL database section.
 - **Partition lower bound:** Specify the minimum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering the rows in table. All rows in the table or query result will be partitioned and copied. If not specified, copy activity auto detect the value. For an example, see the Parallel copy from SQL database section.

- **Additional columns:** Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.

Note the following points:

- If **Query** is specified for source, the copy activity runs this query against the SQL Server source to get the data. You also can specify a stored procedure by specifying **Stored procedure name** and **Stored procedure parameters** if the stored procedure takes parameters.
- When using stored procedure in source to retrieve data, note if your stored procedure is designed as returning different schema when different parameter value is passed in, you might encounter failure or see unexpected result when importing schema from UI or when copying data to SQL database with auto table creation.

Destination

The following properties are supported for SQL server under the **Destination** tab of a copy activity.



The following properties are required:

- **Data store type:** Select **External**.
- **Connection:** Select an SQL server connection from the connection list. If the connection doesn't exist, then create a new SQL server connection by selecting **New**.
- **Connection type:** Select **SQL server**.

- **Table option:** You can choose **Use existing** to use the table specified. Or choose **Auto create table** to automatically create a destination table if the table doesn't exist in the source schema, and note that this selection is not supported when stored procedure is used as the write behavior.

If you select **Use existing**:

- **Table:** Select the table in your destination database from the drop-down list. Or check **Edit** to enter your table name manually.

If you select: **Auto create table**:

- **Table:** Specify the name for your auto-created destination table.

Under **Advanced**, you can specify the following fields:

- **Write behavior:** Defines the write behavior when the source is files from a file-based data store. You can choose **Insert**, **Upsert** or **Stored procedure**.
 - **Insert:** Choose this option use insert write behavior to load data into SQL server.
 - **Upsert:** Choose this option use upsert write behavior to load data into SQL server.
 - **Use TempDB:** Specify whether to use a global temporary table or physical table as the interim table for upsert. By default, the service uses global temporary table as the interim table and this property is selected.



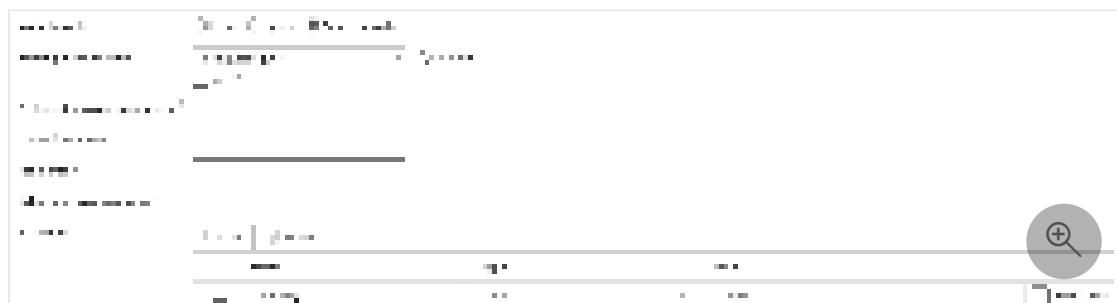
- **Select user DB schema:** When the **Use TempDB** isn't selected, specify the interim schema for creating an interim table if a physical table is used.

Note

You must have the permission for creating and deleting tables. By default, an interim table will share the same schema as a destination table.



- **Key columns:** Specify the column names for unique row identification. Either a single key or a series of keys can be used. If not specified, the primary key is used.
- **Stored procedure:** Use the stored procedure that defines how to apply source data into a target table. This stored procedure is *invoked per batch*. For operations that run only once and have nothing to do with source data, for example, delete or truncate, use **Pre-copy script** property.
- **Stored procedure name:** Select the stored procedure or specify the stored procedure name manually when checking the **Edit** to read data from the source table.
- **Stored procedure parameters:**
 - **Table type:** Specify the table type name to be used in the stored procedure. The copy activity makes the data being moved available in a temporary table with this table type. Stored procedure code can then merge the data that's being copied with existing data.
 - **Table type parameter name:** Specify the parameter name of the table type specified in the stored procedure.
 - **Parameters:** Specify values for stored procedure parameters. Allowed values are name or value pairs. The names and casing of parameters must match the names and casing of the stored procedure parameters. You can select **Import parameters** to get your stored procedure parameters.



- **Bulk insert table lock:** Choose **Yes** or **No** (default). Use this setting to improve copy performance during a bulk insert operation on a table with no index from multiple

clients. You can specify this property when you select **Insert** or **Upsert** as your write behavior. For more information, go to [BULK INSERT \(Transact-SQL\)](#)

- **Pre-copy script:** Specify a script for copy activity to execute before writing data into a destination table in each run. You can use this property to clean up the pre-loaded data.
- **Write batch timeout:** Specify the wait time for the batch insert operation to finish before it times out. The allowed value is timespan. If no value is specified, the timeout defaults to "02:00:00".
- **Write batch size:** Specify the number of rows to insert into the SQL table per batch. The allowed value is integer (number of rows). By default, the service dynamically determines the appropriate batch size based on the row size.
- **Max concurrent connections:** The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.

Mapping

For **Mapping** tab configuration, go to Configure your mappings under mapping tab.

Settings

For **Settings** tab configuration, go to Configure your other settings under settings tab.

Parallel copy from SQL database

The SQL Server connector in copy activity provides built-in data partitioning to copy data in parallel. You can find data partitioning options on the **Source** tab of the copy activity.

When you enable partitioned copy, copy activity runs parallel queries against your SQL Server source to load data by partitions. The parallel degree is controlled by the **Degree of copy parallelism** in the copy activity settings tab. For example, if you set **Degree of copy parallelism** to four, the service concurrently generates and runs four queries based on your specified partition option and settings, and each query retrieves a portion of data from your SQL Server.

You are suggested to enable parallel copy with data partitioning especially when you load large amount of data from your SQL Server. The following are suggested configurations for different scenarios. When copying data into file-based data store, it's recommended to write to a folder as multiple files (only specify folder name), in which case the performance is better than writing to a single file.

Scenario	Suggested settings
Full load from large table, with physical partitions.	<p>Partition option: Physical partitions of table.</p> <p>During execution, the service automatically detects the physical partitions, and copies data by partitions.</p> <p>To check if your table has physical partition or not, you can refer to this query.</p>
Full load from large table, without physical partitions, while with an integer or datetime column for data partitioning.	<p>Partition options: Dynamic range partition.</p> <p>Partition column (optional): Specify the column used to partition data. If not specified, the primary key column is used.</p> <p>Partition upper bound and partition lower bound (optional): Specify if you want to determine the partition stride. This is not for filtering the rows in table, all rows in the table will be partitioned and copied. If not specified, copy activity auto detects the values and it can take long time depending on MIN and MAX values. It is recommended to provide upper bound and lower bound.</p> <p>For example, if your partition column "ID" has values range from 1 to 100, and you set the lower bound as 20 and the upper bound as 80, with parallel copy as 4, the service retrieves data by 4 partitions - IDs in range <=20, [21, 50], [51, 80], and >=81, respectively.</p>
Load a large amount of data by using a custom query, without physical partitions, while with an integer or date/datetime column for data partitioning.	<p>Partition options: Dynamic range partition.</p> <p>Query: <code>SELECT * FROM <TableName> WHERE ? AdfDynamicRangePartitionCondition AND <your_additional_where_clause></code>.</p> <p>Partition column: Specify the column used to partition data.</p> <p>Partition upper bound and partition lower bound (optional): Specify if you want to determine the partition stride. This is not for filtering the rows in table, all rows in the query result will be partitioned and copied. If not specified, copy activity auto detect the value.</p> <p>During execution, the service replaces <code>?AdfRangePartitionColumnName</code> with the actual column name and value ranges for each partition, and sends to SQL Server.</p> <p>For example, if your partition column "ID" has values range from 1 to 100, and you set the lower bound as 20 and the upper bound as 80, with parallel copy as 4, the service retrieves data by 4 partitions- IDs in range <=20, [21, 50], [51, 80], and >=81, respectively.</p> <p>Here are more sample queries for different scenarios:</p> <ol style="list-style-type: none"> 1. Query the whole table: <code>SELECT * FROM <TableName> WHERE ?AdfDynamicRangePartitionCondition</code> 2. Query from a table with column selection and additional where-clause filters: <code>SELECT <column_list> FROM <TableName> WHERE ? AdfDynamicRangePartitionCondition AND <your_additional_where_clause></code> 3. Query with subqueries: <code>SELECT <column_list> FROM (<your_sub_query>) AS T WHERE ? AdfDynamicRangePartitionCondition AND <your_additional_where_clause></code>

Scenario	Suggested settings
	<p>4. Query with partition in subquery:</p> <pre>SELECT <column_list> FROM (SELECT <your_sub_query_column_list> FROM <TableName> WHERE ?AdfDynamicRangePartitionCondition) AS T</pre>

Best practices to load data with partition option:

1. Choose distinctive column as partition column (like primary key or unique key) to avoid data skew.
2. If the table has built-in partition, use partition option **Physical partitions of table** to get better performance.

Sample query to check physical partition

SQL
<pre>SELECT DISTINCT s.name AS SchemaName, t.name AS TableName, pf.name AS PartitionFunctionName, c.name AS ColumnName, iif(pf.name is null, 'no', 'yes') AS HasPartition FROM sys.tables AS t LEFT JOIN sys.objects AS o ON t.object_id = o.object_id LEFT JOIN sys.schemas AS s ON o.schema_id = s.schema_id LEFT JOIN sys.indexes AS i ON t.object_id = i.object_id LEFT JOIN sys.index_columns AS ic ON ic.partition_ordinal > 0 AND ic.index_id = i.index_id AND ic.object_id = t.object_id LEFT JOIN sys.columns AS c ON c.object_id = ic.object_id AND c.column_id = ic.column_id LEFT JOIN sys.partition_schemes ps ON i.data_space_id = ps.data_space_id LEFT JOIN sys.partition_functions pf ON pf.function_id = ps.function_id WHERE s.name='[your schema]' AND t.name = '[your table name]'</pre>

If the table has physical partition, you would see "HasPartition" as "yes" like the following.

SchemaName	TableName	PartitionFunctionName	ColumnName	HasPartition
dbo	Adventure	RangePF1	H	yes

Table summary

See the following table for the summary and more information for the SQL server copy activity.

Source information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/
Connection	Your connection to the source data store.	< your connection >	Yes	connection
Connection type	Your connection type. Select SQL server.	SQL server	Yes	/
Use query	The custom SQL query to read data.	<ul style="list-style-type: none"> • Table • Query • Stored procedure 	No	/
Table	Your source data table.	< name of your table>	No	schema table
Query	The custom SQL query to read data.	< your query >	No	sqlReaderQuery
Stored procedure name	This property is the name of the stored procedure that reads data from the source table. The last SQL statement must be a SELECT statement in the stored procedure.	< stored procedure name >	No	sqlReaderStoredProcedureName
Stored procedure parameter	These parameters are for the stored procedure. Allowed values are name or value pairs. The names and casing of parameters must match the names and casing of the stored procedure parameters.	< name or value pairs >	No	storedProcedureParameters
Query timeout	The timeout for query command execution.	timespan (the default is 120 minutes)	No	queryTimeout
Isolation level	Specifies the transaction locking behavior for the SQL source.	<ul style="list-style-type: none"> • Read committed • Read uncommitted • Repeatable read • Serializable • Snapshot 	No	isolationLevel: <ul style="list-style-type: none"> • ReadCommitted • ReadUncommitted • RepeatableRead • Serializable • Snapshot
Partition option	The data partitioning options used to load data from SQL server.	<ul style="list-style-type: none"> • None (default) • Physical partitions of table 	No	partitionOption: <ul style="list-style-type: none"> • None (default) • PhysicalPartitionsOfTable • DynamicRange

Name	Description	Value	Required	JSON script property
		• Dynamic range		
Partition column name	The name of the source column in integer or date/datetime type (<code>int</code> , <code>smallint</code> , <code>bigint</code> , <code>date</code> , <code>smalldatetime</code> , <code>datetime</code> , <code>datetime2</code> , or <code>datetimeoffset</code>) that's used by range partitioning for parallel copy. If not specified, the index or the primary key of the table is auto-detected and used as the partition column. If you use a query to retrieve the source data, hook ? <code>AdfDynamicRangePartitionCondition</code> in the WHERE clause.	< your partition column names >	No	partitionColumnName
Partition upper bound	The maximum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering the rows in table. All rows in the table or query result will be partitioned and copied. If not specified, copy activity auto detect the value.	< your partition upper bound >	No	partitionUpperBound
Partition lower bound	The minimum value of the partition column for partition range splitting. This value is used to decide the partition stride, not for filtering the rows in table. All rows in the table or query result will be partitioned and copied. If not specified, copy activity auto detect the value.	< your partition lower bound >	No	partitionLowerBound
Additional columns	Add additional data columns to store source files' relative path or static value. Expression is supported for the latter.	• Name • Value	No	additionalColumns: • name • value

Destination information

Name	Description	Value	Required	JSON script property
Data store type	Your data store type.	External	Yes	/

Name	Description	Value	Required	JSON script property
Connection	Your connection to the destination data store.	< your connection >	Yes	connection
Connection type	Your connection type. Select SQL server.	SQL server	Yes	/
Table option	Specifies whether to automatically create the destination table if it doesn't exist based on the source schema.	<ul style="list-style-type: none"> • Use existing • Auto create table 	No	<tableoption: <ul=""> • autoCreate </tableoption:>
Table	Your destination data table.	<name of your table>	Yes	schema table
Write behavior	The write behavior for copy activity to load data into SQL server database..	<ul style="list-style-type: none"> • Insert • Upsert • Stored procedure 	No	writeBehavior: <ul style="list-style-type: none"> • insert • upsert sqlWriterStoredProcName, sqlWriterTableType, storedProcTableTypeParameterName, storedProcParameters
Use TempDB	Whether to use the a global temporary table or physical table as the interim table for upsert.	selected (default) or unselected	No	useTempDB: true (default) or false
Select user DB schema	The interim schema for creating interim table if physical table is used. Note: user need to have the permission for creating and deleting table. By default, interim table will share the same schema as	selected (default) or unselected	No	interimSchemaName

Name	Description	Value	Required	JSON script property
	destination table. Apply when you don't select Use TempDB .			
Key columns	The column names for unique row identification. Either a single key or a series of keys can be used. If not specified, the primary key is used.	< your key column>	No	keys
Stored procedure name	The name of the stored procedure that defines how to apply source data into a target table. This stored procedure is <i>invoked per batch</i> . For operations that run only once and have nothing to do with source data, for example, delete or truncate, use the Pre-copy script property.	< your stored procedure name >	No	sqlWriterStoredProcedureName
Table type	The table type name to be used in the stored procedure. The copy activity makes the data being moved available in a temporary table with this table	< your table type name >	No	sqlWriterTableType

Name	Description	Value	Required	JSON script property
	type. Stored procedure code can then merge the data that's being copied with existing data.			
Table type parameter name	The parameter name of the table type specified in the stored procedure.	< your parameter name of the table type >	No	storedProcedureTableTypeParameterName
Parameters	Parameters for the stored procedure. Allowed values are name and value pairs. Names and casing of parameters must match the names and casing of the stored procedure parameters.	< name and value pairs >	No	storedProcedureParameters
Bulk insert table lock	Use this setting to improve copy performance during a bulk insert operation on a table with no index from multiple clients.	Yes or No (default)	No	sqlWriterUseTableLock: true or false (default)
Pre-copy script	A script for copy activity to execute before writing data into a destination table in each run. You can use this property to clean up the	< pre-copy script > (string)	No	preCopyScript

Name	Description	Value	Required	JSON script property
	pre-loaded data.			
Write batch timeout	The wait time for the batch insert operation to finish before it times out.	timespan (the default is "02:00:00")	No	writeBatchTimeout
Write batch size	The number of rows to insert into the SQL table per batch. By default, the service dynamically determines the appropriate batch size based on the row size.	< number of rows > (integer)	No	writeBatchSize
Max concurrent connections	The upper limit of concurrent connections established to the data store during the activity run. Specify a value only when you want to limit concurrent connections.	< upper limit of concurrent connections > (integer)	No	maxConcurrentConnections

Next steps

- SQL server overview

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Teradata database connector overview

Article • 11/15/2023

The Teradata database connector is supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to a Teradata database in Dataflow Gen2, go to Set up your Teradata database connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support a Teradata database in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your Teradata database connection

Article • 11/15/2023

This article outlines the steps to create a Teradata database connection.

Supported authentication types

The Teradata database connector supports the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Basic (Username/Password)	n/a	✓
Windows	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to a Teradata database. The following links provide the specific Power Query connector information you need to connect to a Teradata database in Dataflow Gen2:

- To get started using the Teradata database connector in Dataflow Gen2, go to Get data from Data Factory in Microsoft Fabric (Preview).
- Be sure to install or set up any Teradata database prerequisites before connecting to the Teradata database connector.
- To connect to the Teradata database connector from Power Query, go to Connect to a Teradata database from Power Query Online.

In some cases, the Power Query connector article might include advanced options, troubleshooting, known issues and limitations, and other information that could also prove useful.

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support a Teradata database in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Web connector overview

Article • 11/15/2023

Both the Web API and Web page connectors are supported in Data Factory for Microsoft Fabric with the following capabilities.

Support in Dataflow Gen2

For information on how to connect to the Web API or a Web page in Dataflow Gen2, go to Set up your Web connection.

Support in data pipelines

Data Factory in Microsoft Fabric doesn't currently support a Web API or Web page connector in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Set up your Web connection

Article • 11/15/2023

This article outlines the steps to create either a Web API or Web page connection.

Supported authentication types

Both the Web API and Web page connectors support the following authentication types for copy and Dataflow Gen2 respectively.

Authentication type	Copy	Dataflow Gen2
Anonymous	n/a	✓
Basic (Username/Password)	n/a	✓
Organizational account	n/a	✓
Windows	n/a	✓

Set up your connection in Dataflow Gen2

Data Factory in Microsoft Fabric uses Power Query connectors to connect Dataflow Gen2 to a website. The following links provide the specific Power Query connector information you need to connect to a website in Dataflow Gen2:

- To get started using the Web API or Web page connector in Dataflow Gen2, go to [Get data from Data Factory in Microsoft Fabric \(Preview\)](#).
- Be sure to install or set up any Web connector prerequisites before connecting to the Web API or Web page connector.
- To connect to the Web API or Web page connector from Power Query, go to [Load Web data using Power Query Online](#).

The following Power Query Web connector articles supply more useful information when you use the Web API or Web page connectors:

- [Get webpage data by providing examples](#)
- [Troubleshooting the Web connector](#)

Set up your connection in a data pipeline

Data Factory in Microsoft Fabric doesn't currently support the Web API or Web page connector in data pipelines.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Activity overview

Article • 11/15/2023

This article helps you understand activities in Microsoft Fabric and use them to construct end-to-end data-driven workflows for your data movement and data processing scenarios.

Overview

A Microsoft Fabric Workspace can have one or more pipelines. A pipeline is a logical grouping of activities that together perform a task. For example, a pipeline could contain a set of activities that ingest and clean log data, and then kick off a mapping data flow to analyze the log data. The pipeline allows you to manage the activities as a set instead of each one individually. You deploy and schedule the pipeline instead of the activities independently.

The activities in a pipeline define actions to perform on your data. For example, you can use a copy activity to copy data from SQL Server to an Azure Blob Storage. Then, use a Dataflow activity or a Notebook activity to process and transform data from the blob storage to an Azure Synapse Analytics pool on top of which business intelligence reporting solutions are built.

Microsoft Fabric has three types of activities: data movement activities, data transformation activities, and control activities.

Data movement activities

Copy activity in Microsoft Fabric copies data from a source data store to a sink data store. Fabric supports the data stores listed in the Connector overview article. Data from any source can be written to any sink.

For more information, see [How to copy data using the copy activity](#).

Data transformation activities

Microsoft Fabric supports the following transformation activities that can be added either individually or chained with another activity.

For more information, see the [data transformation activities article](#).

Data transformation activity	Compute environment
Copy data	Compute manager by Microsoft Fabric
Dataflow Gen2	Compute manager by Microsoft Fabric
Delete data	Compute manager by Microsoft Fabric
Fabric Notebook	Apache Spark clusters managed by Microsoft Fabric
Fabric Spark job definition (coming soon)	Apache Spark clusters managed by Microsoft Fabric
Stored Procedure	Azure SQL, Azure Synapse Analytics, or SQL Server
SQL script	Azure SQL, Azure Synapse Analytics, or SQL Server

Control flow activities

The following control flow activities are supported:

Control activity	Description
Append variable	Add a value to an existing array variable.
Azure Batch activity	Runs an Azure Batch script.
Azure Machine Learning activity	Runs an Azure Machine Learning job.
Deactivate activity	Deactivates another activity.
Filter	Apply a filter expression to an input array.
ForEach	ForEach Activity defines a repeating control flow in your pipeline. This activity is used to iterate over a collection and executes specified activities in a loop. The loop implementation of this activity is similar to the Foreach looping structure in programming languages.
Functions activity	Executes an Azure Function.
Get metadata	GetMetadata activity can be used to retrieve metadata of any data in a Data Factory or Synapse pipeline.

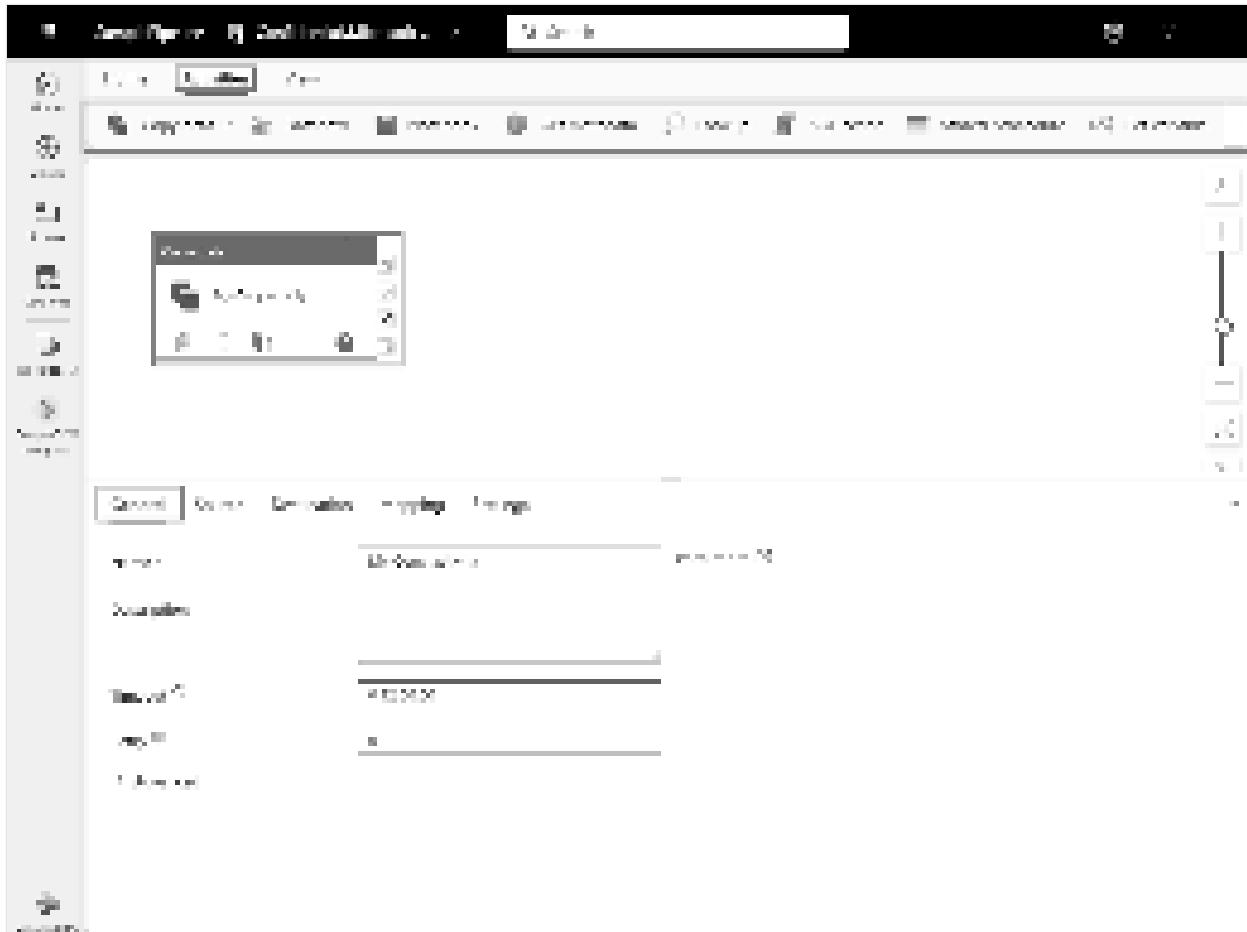
Control activity	Description
If condition	The If Condition can be used to branch based on condition that evaluates to true or false. The If Condition activity provides the same functionality that an if statement provides in programming languages. It evaluates a set of activities when the condition evaluates to <code>true</code> and another set of activities when the condition evaluates to <code>false</code> .
Invoke pipeline	Execute Pipeline activity allows a Data Factory or Synapse pipeline to invoke another pipeline.
KQL activity	Executes a KQL script against a Kusto instance.
Lookup Activity	Lookup Activity can be used to read or look up a record/ table name/ value from any external source. This output can further be referenced by succeeding activities.
Set Variable	Set the value of an existing variable.
Switch activity	Implements a switch expression that allows multiple subsequent activities for each potential result of the expression.
Teams activity	Posts a message in a Teams channel or group chat.
Until activity	Implements Do-Until loop that is similar to Do-Until looping structure in programming languages. It executes a set of activities in a loop until the condition associated with the activity evaluates to true. You can specify a timeout value for the until activity.
Wait activity	When you use a Wait activity in a pipeline, the pipeline waits for the specified time before continuing with execution of subsequent activities.
Web activity	Web Activity can be used to call a custom REST endpoint from a pipeline.
Webhook activity	Using the webhook activity, call an endpoint, and pass a callback URL. The pipeline run waits for the callback to be invoked before proceeding to the next activity.

Adding activities to a pipeline with the Microsoft Fabric UI

Use these steps to add and configure activities in a Microsoft Fabric pipeline:

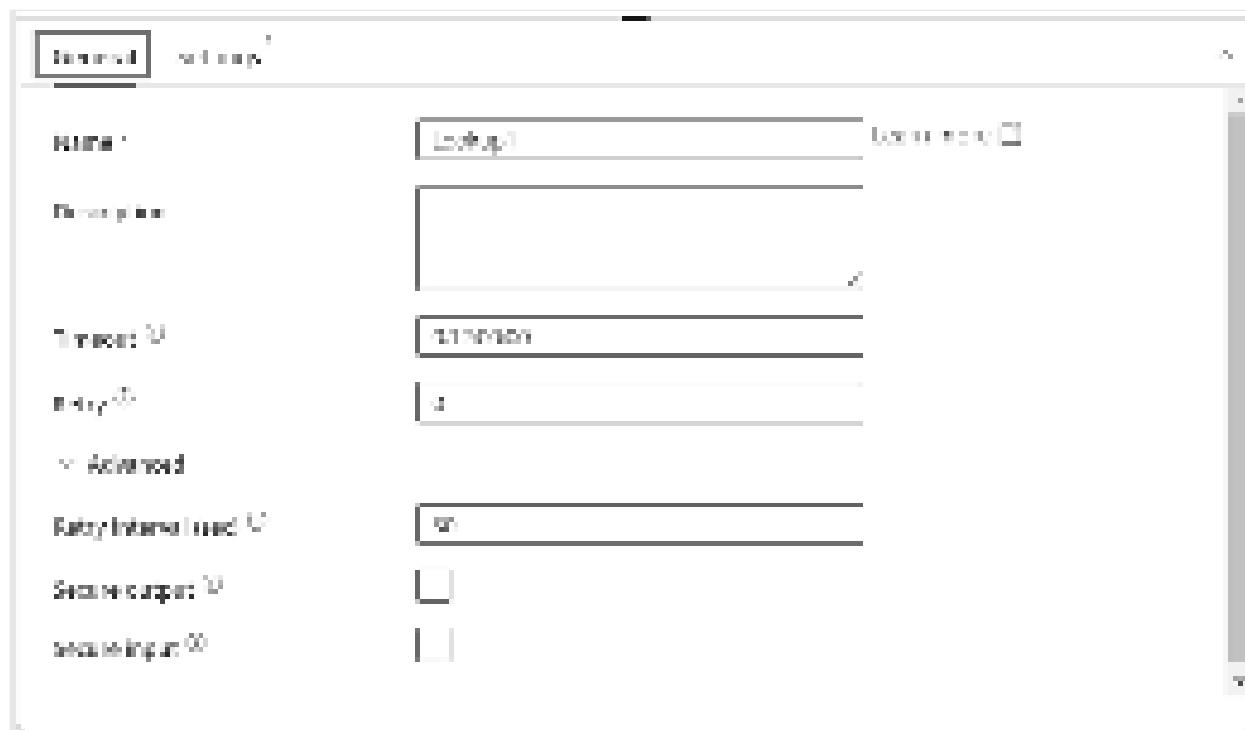
1. Create a new pipeline in your workspace.
2. On the Activities tab for the pipeline, browse the activities displayed, scrolling to the right if necessary to see all activities. Select an activity to add it to the pipeline editor.

3. When you add an activity and select it in the pipeline editor canvas, its **General** settings will appear in the properties pane below the canvas.
4. Each activity also contains custom properties specific to its configuration on other tabs in the properties pane.



General settings

When you add a new activity to a pipeline and select it, you'll see its properties panes in the area at the bottom of the screen. These properties panes include **General**, **Settings**, and sometimes other panes as well.



The general settings will always include **Name** and **Description** fields for every activity. Some activities also include the following:

Setting	Description
Timeout	The maximum amount of time an activity can run. The default is 12 hours, and the maximum amount of time allowed is seven days. The format for the timeout is in D.HH:MM:SS.
Retry	Maximum number of retry attempts.
(Advanced properties) Retry interval (sec)	The number of seconds between each retry attempt.
(Advanced properties) Secure output	When checked, output from the activity isn't captured in logging.
(Advanced properties) Secure input	The number of seconds between each retry attempt.

Next steps

Create your first pipeline

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Use the Append Variable activity in Fabric

Article • 11/15/2023

Use the Append Variable activity to add a value to an existing array variable defined in a Fabric pipeline.

Prerequisites

To get started, you must complete the following prerequisites:

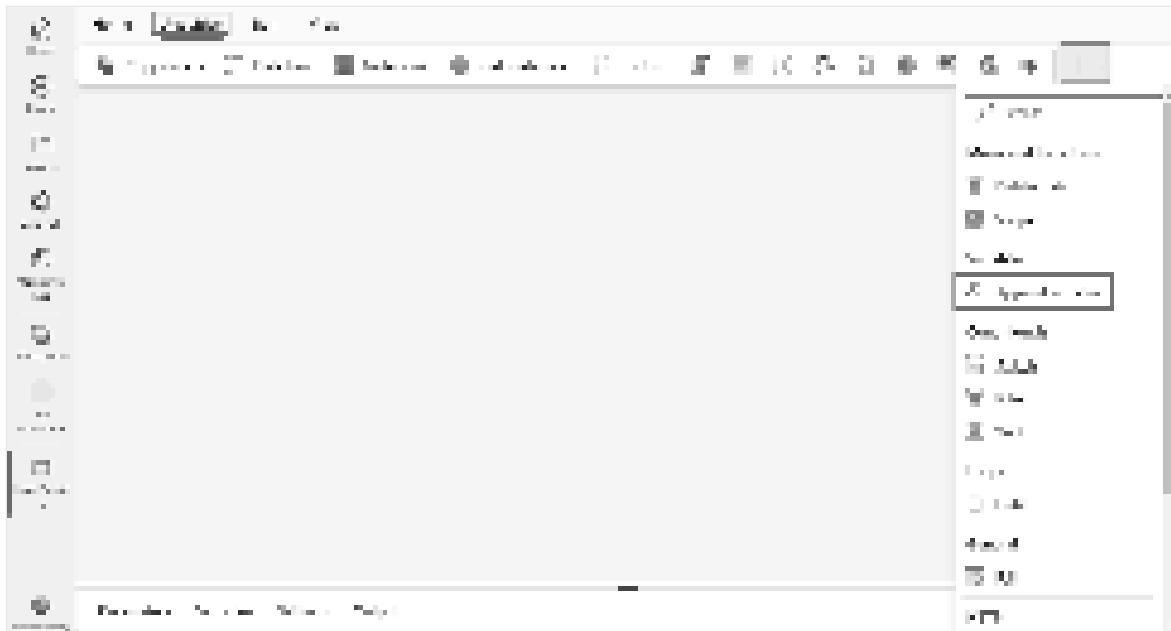
- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add an Append variable activity to a pipeline with UI

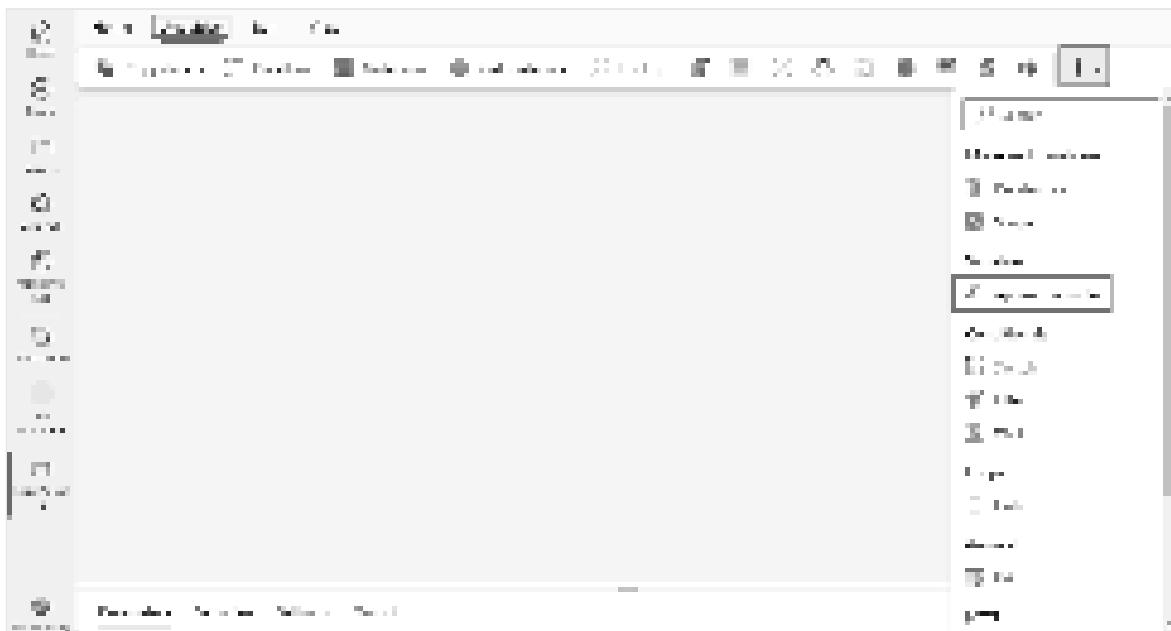
To use an Append variable activity in a pipeline, complete the following steps:

Creating the activity

1. Create a new pipeline in your workspace.
2. Before you use the Append variable activity, you'll need an array type variable in your pipeline. Select the **Variables** tab from the pipeline settings, and then select **+ New**, to add a variable of Array type.



3. Now that you have an array type variable in your pipeline, search for **Append variable** in the pipeline **Activities** pane, and select it to add it to the pipeline canvas. You may need to expand the list of available activities using the dropdown + button at the far right of the toolbar.



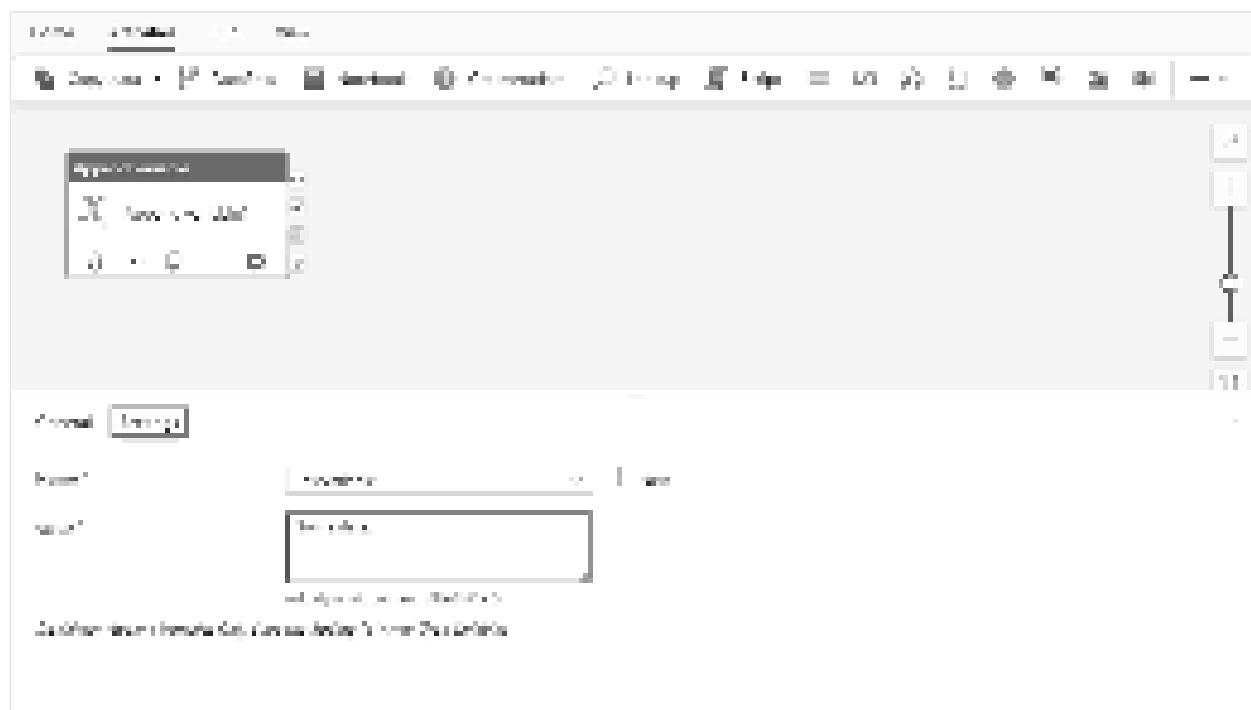
4. Select the new activity on the canvas if it isn't already selected.



Refer to the **General** settings guidance to configure the **General** settings tab.

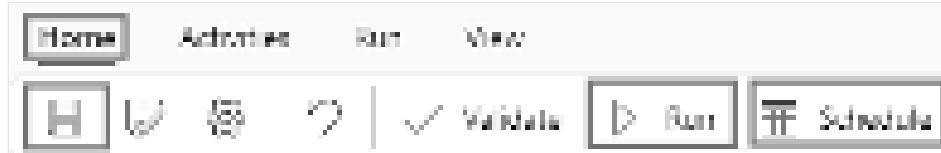
Append variable settings

Select the **Settings** tab, then use the **Name** dropdown to select your previously created array type variable. You can also use the **+ New** button to create a new variable in the pipeline directly if you didn't create one previously. Provide data to be appended to the array variable. You can use dynamic expressions here as well as directly enter data, of which the type is always string.



Save and run or schedule the pipeline

Although Append variable is typically used in conjunction with other activities, it can be run directly as is. To run the simple pipeline here, switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

Provide product feedback [↗](#) | Ask the community [↗](#)

Use the Azure Batch activity to run a command on an Azure Batch instance

Article • 11/15/2023

The Azure Batch activity in Data Factory for Microsoft Fabric allows you to run a command against an Azure Batch instance.

Prerequisites

To get started, you must complete the following prerequisites:

- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add an Azure Batch activity to a pipeline with UI

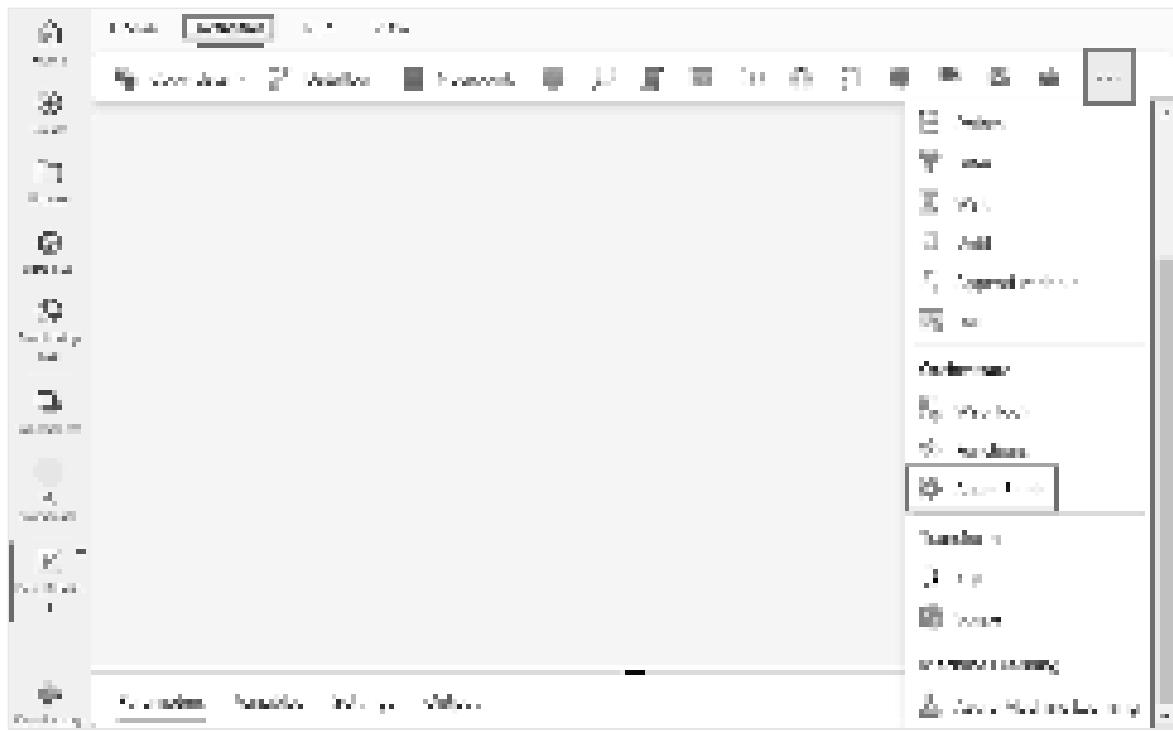
To use an Azure Batch activity in a pipeline, complete the following steps:

Create the activity

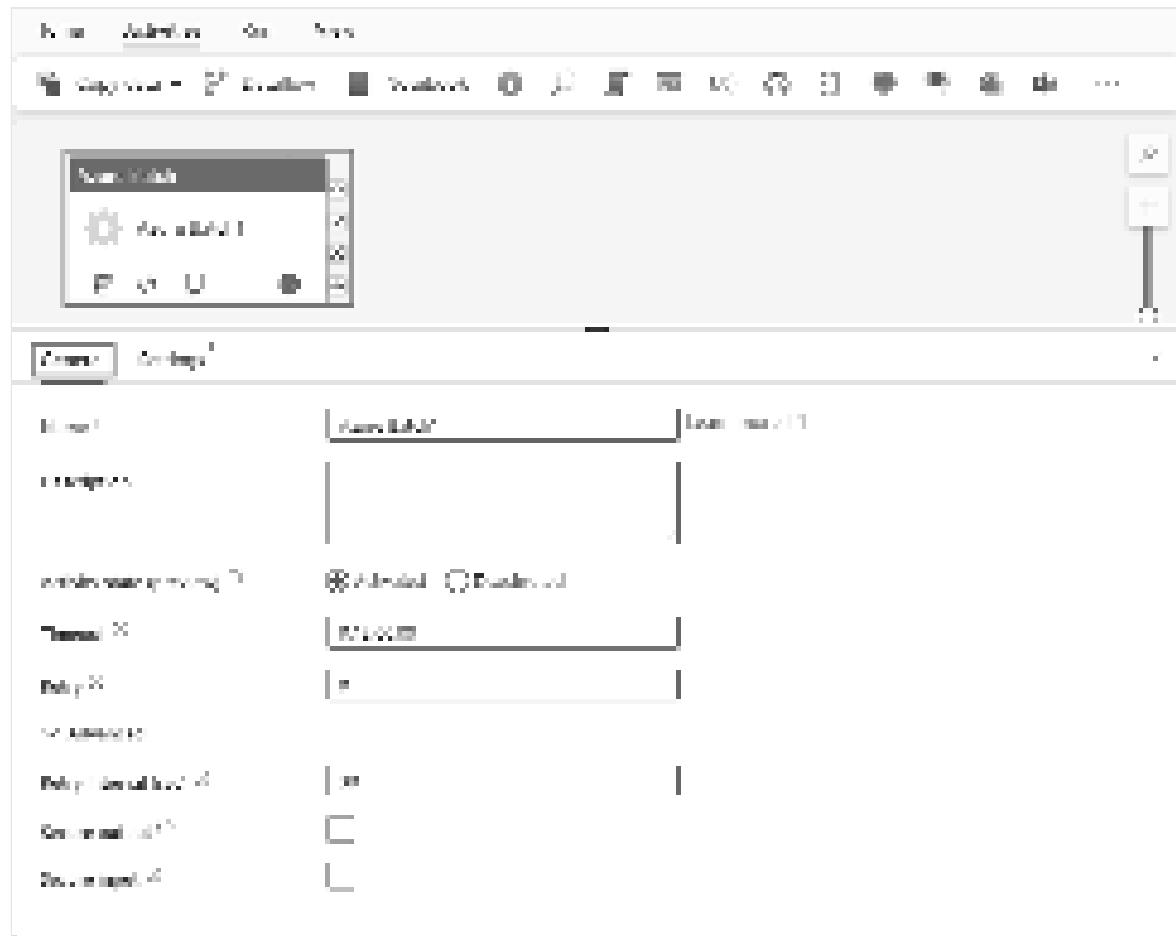
1. Create a new pipeline in your workspace.
2. Search for Azure Batch in the pipeline **Activities** pane, and select it to add it to the pipeline canvas.

ⓘ Note

You may need to expand the menu and scroll down to see the Azure Batch activity as highlighted in the following screenshot.



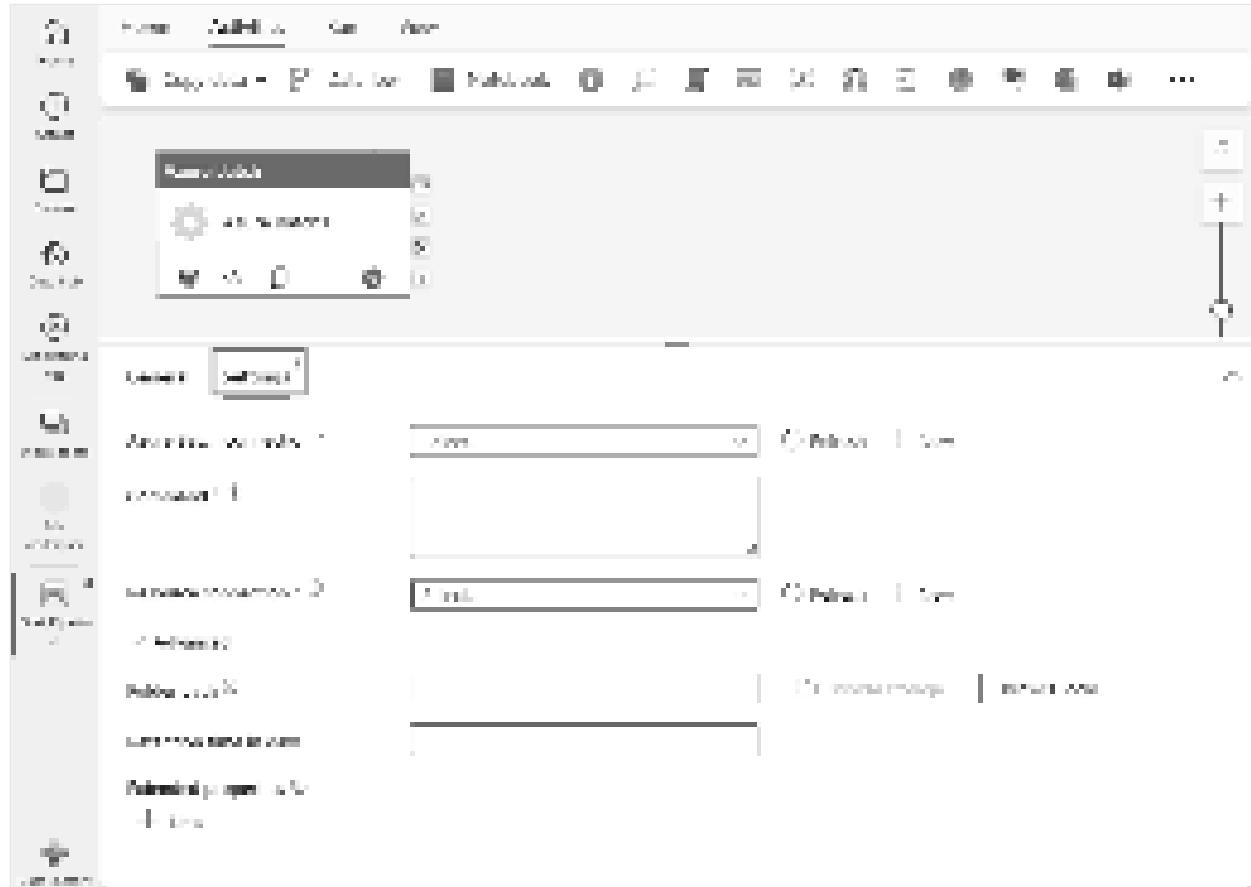
3. Select the new Azure Batch activity on the pipeline editor canvas if it isn't already selected.



Refer to the General settings guidance to configure the General settings tab.

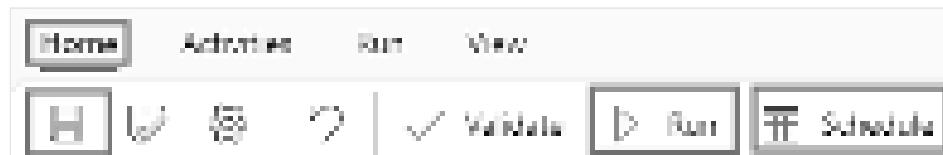
Azure Batch activity settings

Select the **Settings** tab, then you can choose an existing or create a new **Azure Batch connection**, provide a **Command** to be executed, and a **Resource connection** to a storage account. You can also specify a specific **Folder path** within the storage account and a **Retention time in days** for data to be retained there, as well as add extended properties of your own.



Save and run or schedule the pipeline

After you configure any other activities required for your pipeline, switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Transform data by running an Azure Databricks activity

Article • 12/05/2023

The Azure Databricks activity in Data Factory for Microsoft Fabric allows you to orchestrate the following Azure Databricks jobs:

- Notebook
- Jar
- Python

This article provides a step-by-step walkthrough that describes how to create an Azure Databricks activity using the Data Factory interface.

Prerequisites

To get started, you must complete the following prerequisites:

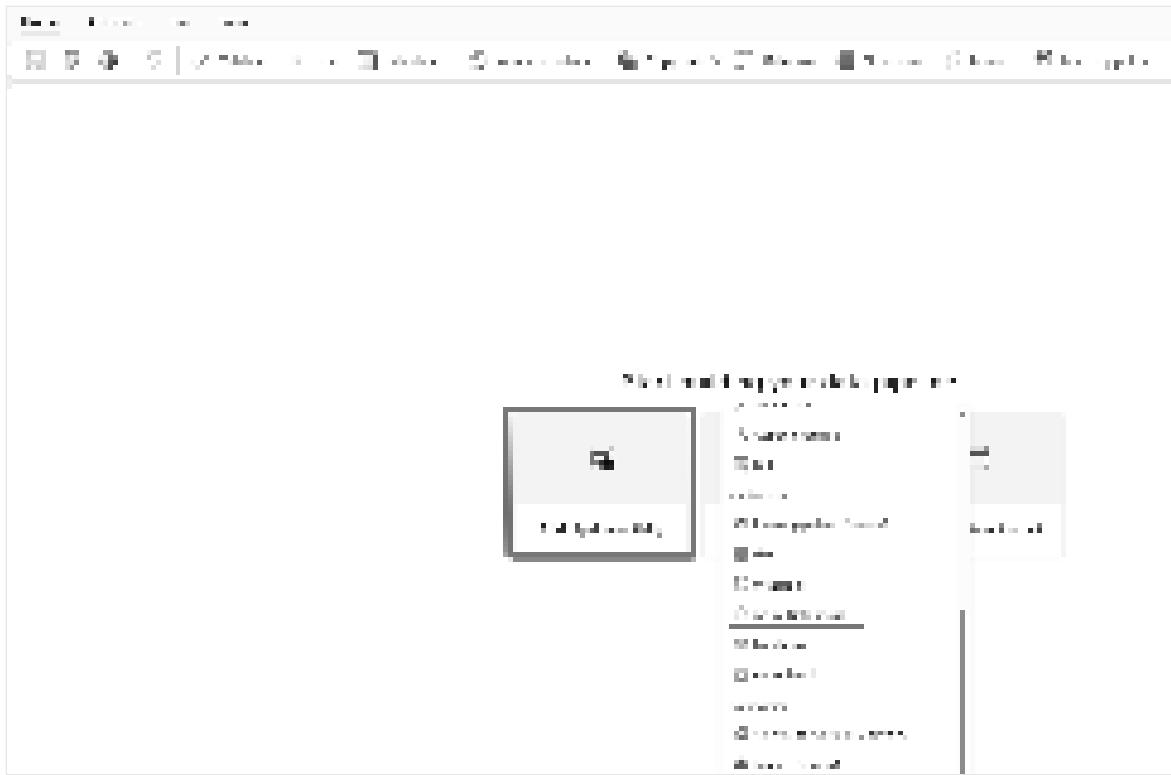
- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Configuring an Azure Databricks activity

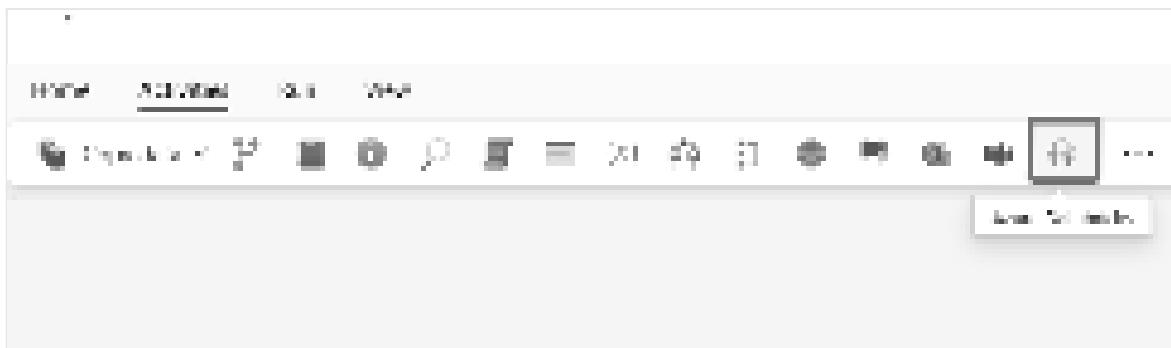
To use an Azure Databricks activity in a pipeline, complete the following steps:

Configuring connection

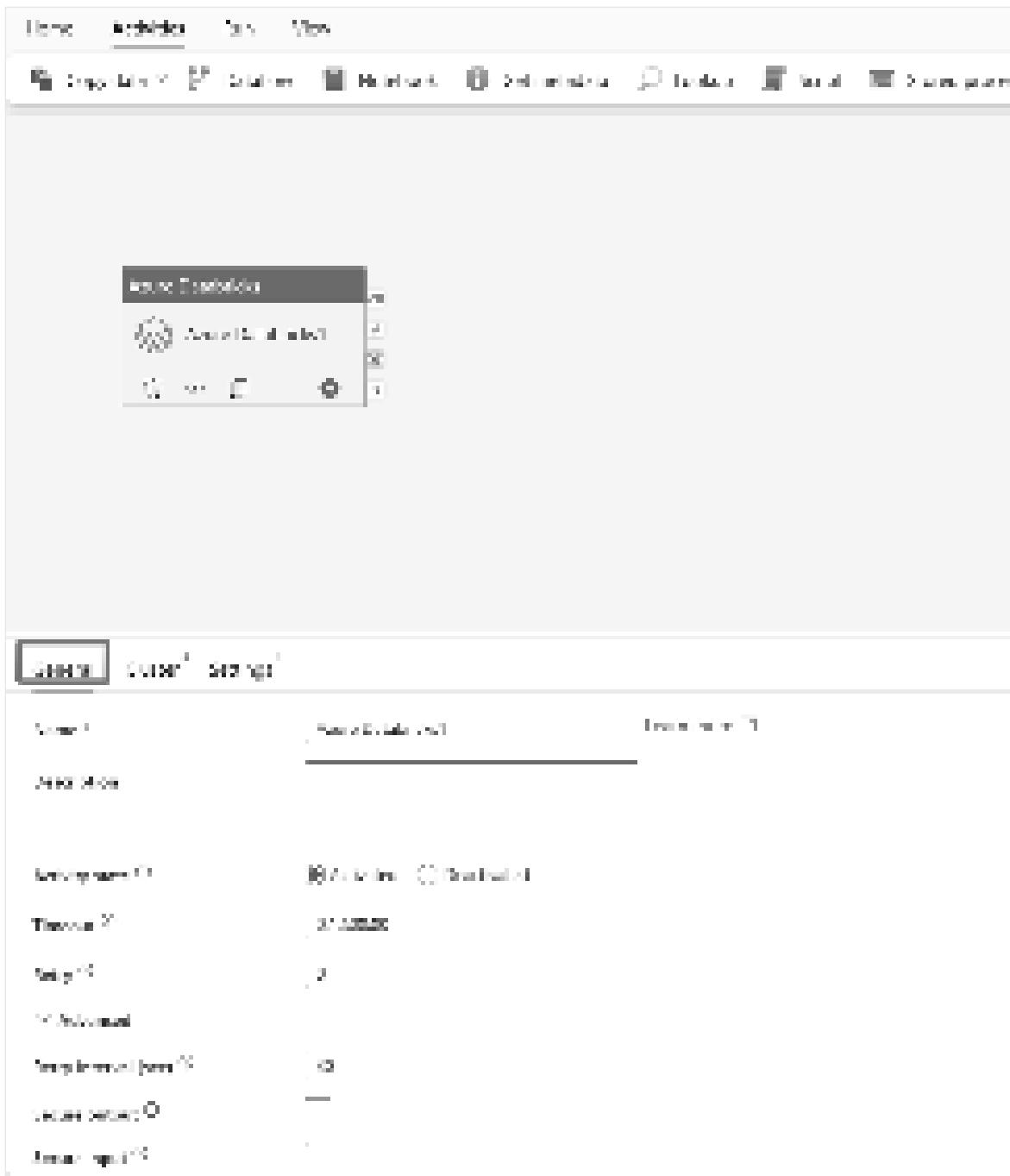
1. Create a new pipeline in your workspace.
2. Click on add a pipeline activity and search for Azure Databricks.



3. Alternately, you can search for Azure Databricks in the pipeline **Activities** pane, and select it to add it to the pipeline canvas.



4. Select the new Azure Databricks activity on the canvas if it isn't already selected.



Refer to the **General** settings guidance to configure the **General** settings tab.

Configuring clusters

1. Select the **Cluster** tab. Then you can choose an existing or create a new **Azure Databricks connection**, and then pick a **new job cluster**, an **existing interactive cluster**, or an **existing instance pool**.
2. Depending on what you pick for the cluster, fill out the corresponding fields as presented.
 - a. Under **new job cluster** and **existing instance pool**, you also have the ability to configure the number of **workers** and enable **spot instances**.

3. You can also specify additional cluster settings, such as **Cluster policy**, **Spark configuration**, **Spark environment variables**, and **custom tags**, as required for the cluster you are connecting to. **Databricks init scripts** and **Cluster Log destination path** can also be added under the additional cluster settings.

ⓘ Note

All advanced cluster properties and dynamic expressions supported in the Azure Data Factory Azure Databricks linked service are now also supported in the Azure Databricks activity in Microsoft Fabric under the 'Additional cluster configuration' section in the UI. As these properties are now included within the activity UI, they can be easily used with an expression (dynamic content) without the need for the Advanced JSON specification in the Azure Data Factory Azure Databricks linked service.

Azure Databricks

Validate Run Schedule View run history Copy

Azure Databricks1

General Cluster Settings

Connection: https://127.0.0.1:50070 + New

Cluster:

- New job cluster
- Existing interactive cluster
- Existing instance pool

Cluster service: 10.4 LTS (includes Apache Spark 3.2...)

Cluster node type: Standard_DS3_v2

Python Version: 3

Driver options: Killed

Driver memory: 1

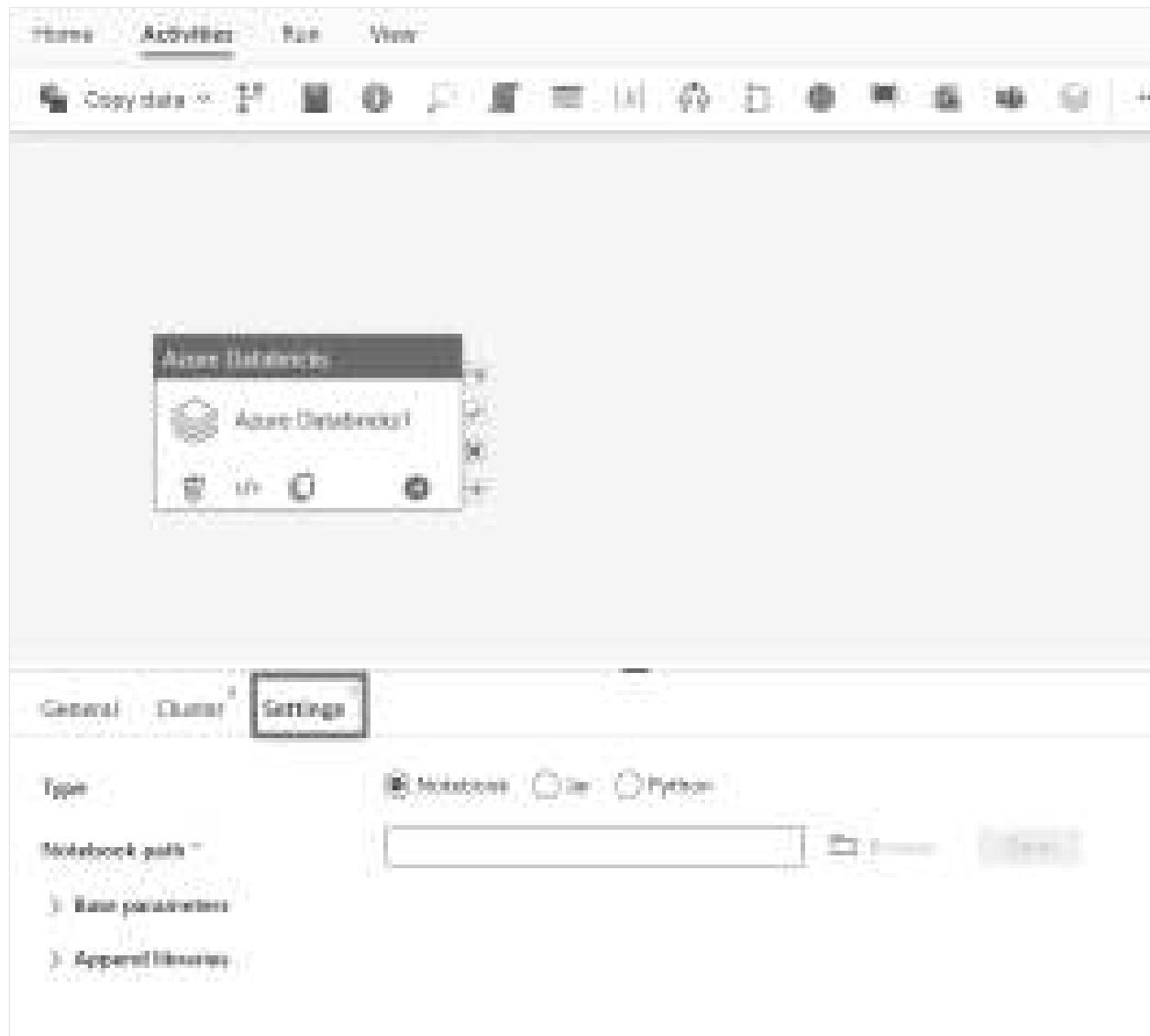
Spot instances:

Additional cluster settings:

- Cluster policy: Select...
- Cluster Spark conf: None
- Cluster spark environment variables: None
- Cluster custom tags: None

Configuring settings

Selecting the **Settings** tab, you can choose between 3 options which **Azure Databricks** type you would like to orchestrate.



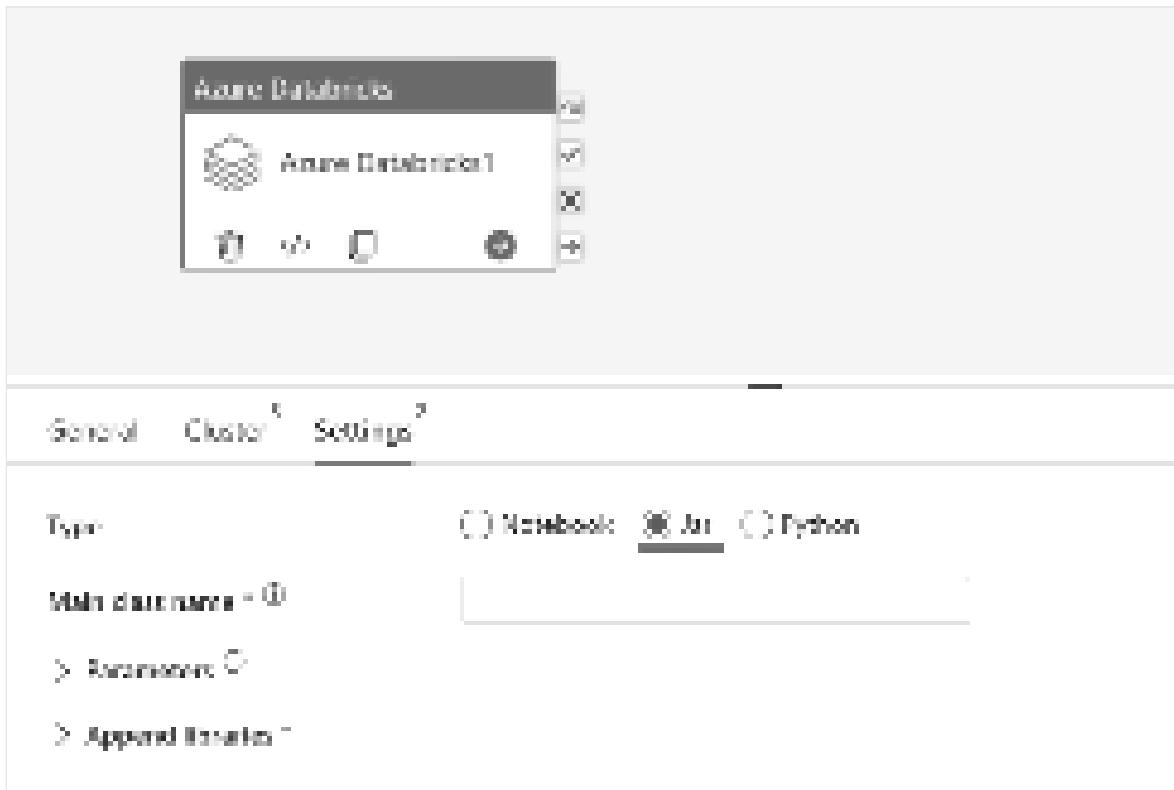
Orchestrating the Notebook type in Azure Databricks activity:

1. Under the **Settings** tab, you can choose the **Notebook** radio button to run a Notebook. You will need to specify the notebook path to be executed on Azure Databricks, optional base parameters to be passed to the notebook, and any additional libraries to be installed on the cluster to execute the job.



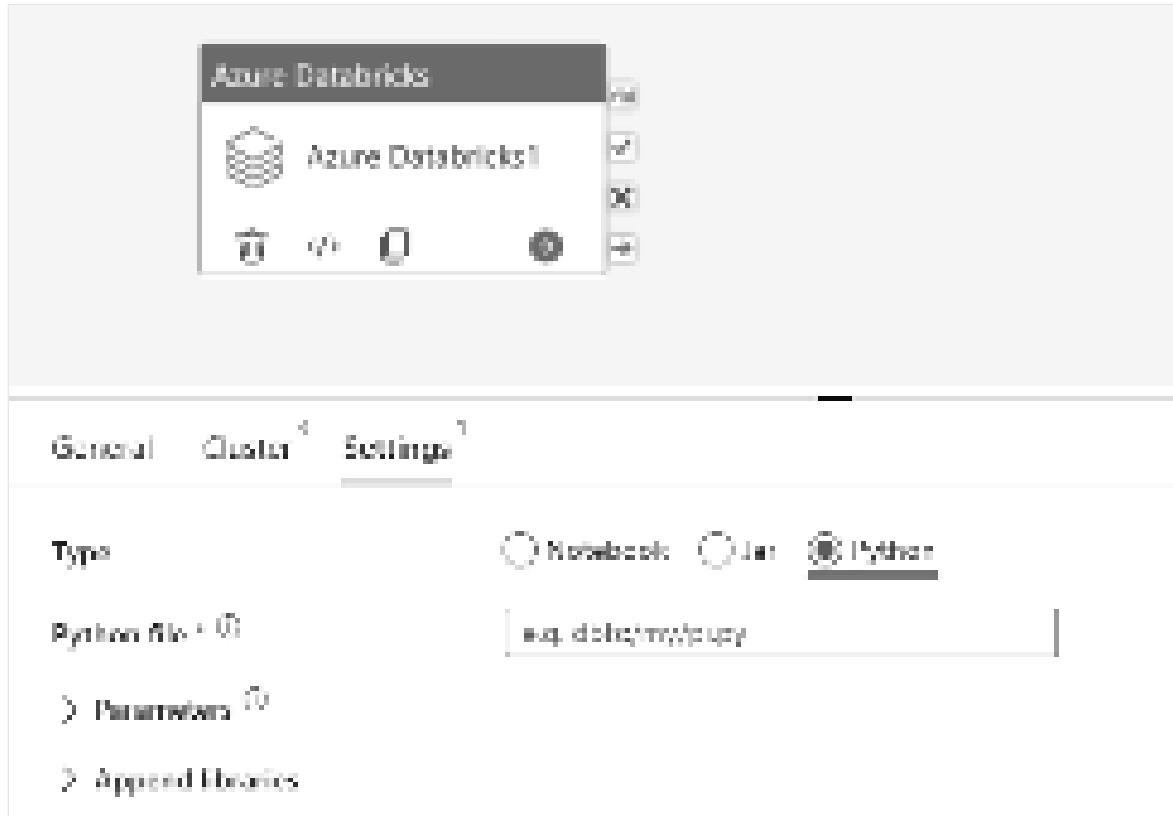
Orchestrating the Jar type in Azure Databricks activity:

1. Under the **Settings** tab, you can choose the **Jar** radio button to run a Jar. You will need to specify the class name to be executed on Azure Databricks, optional base parameters to be passed to the Jar, and any additional libraries to be installed on the cluster to execute the job.



Orchestrating the Python type in Azure Databricks activity:

1. Under the **Settings** tab, you can choose the **Python** radio button to run a Python file. You will need to specify the path within Azure Databricks to a Python file to be executed, optional base parameters to be passed, and any additional libraries to be installed on the cluster to execute the job.



Supported Libraries for the Azure Databricks activity

In the above Databricks activity definition, you can specify these library types: *jar, egg, whl, maven, pypi, cran*.

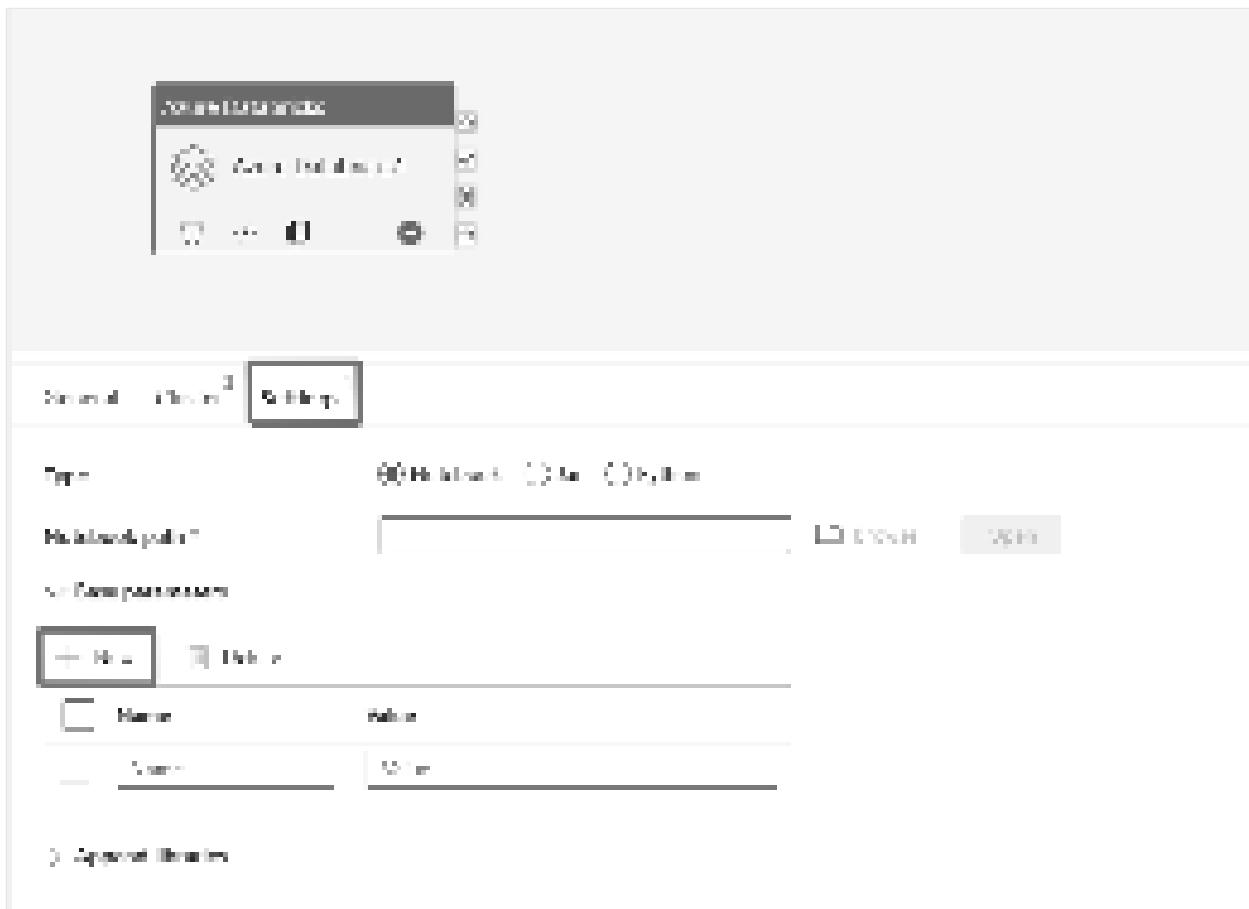
For more information, see the Databricks documentation for library types.

Passing parameters between Azure Databricks activity and pipelines

You can pass parameters to notebooks using *baseParameters* property in databricks activity.

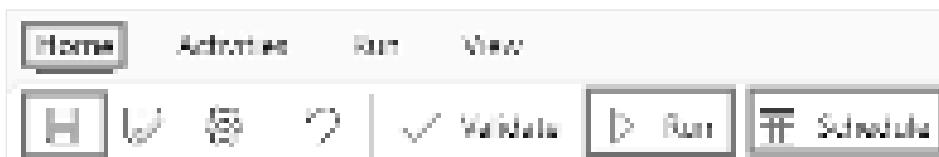
In certain cases, you might require to pass back certain values from notebook back to the service, which can be used for control flow (conditional checks) in the service or be consumed by downstream activities (size limit is 2 MB).

1. In your notebook, for example, you may call `dbutils.notebook.exit("returnValue")` and corresponding "returnValue" will be returned to the service.
2. You can consume the output in the service by using expression such as
`@{activity('databricks activity name').output.runOutput}`.



Save and run or schedule the pipeline

After you configure any other activities required for your pipeline, switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Use the Azure Machine Learning activity to run a job on an Azure Machine Learning instance

Article • 11/15/2023

The Azure Machine Learning activity in Data Factory for Microsoft Fabric allows you to run a job on an Azure Machine Learning instance.

Prerequisites

To get started, you must complete the following prerequisites:

- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add an Azure Machine Learning activity to a pipeline with UI

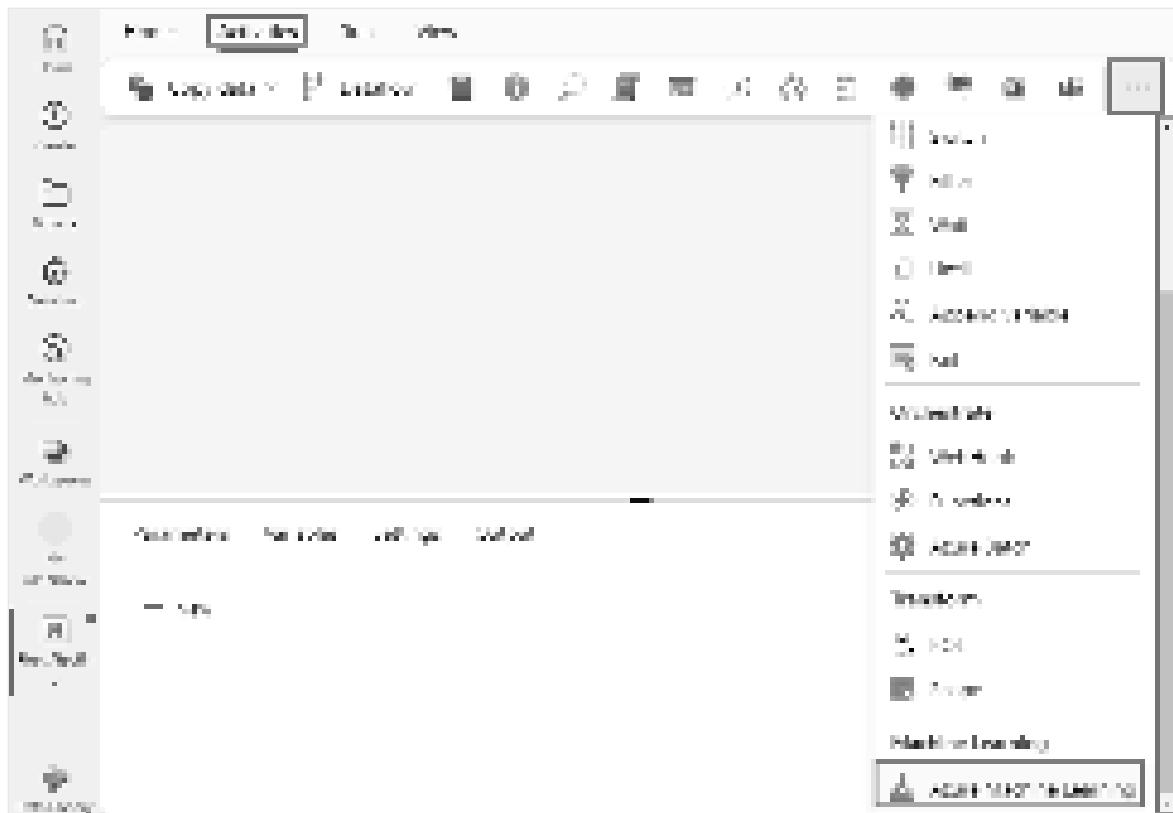
To use an Azure Machine Learning activity in a pipeline, complete the following steps:

Create the activity

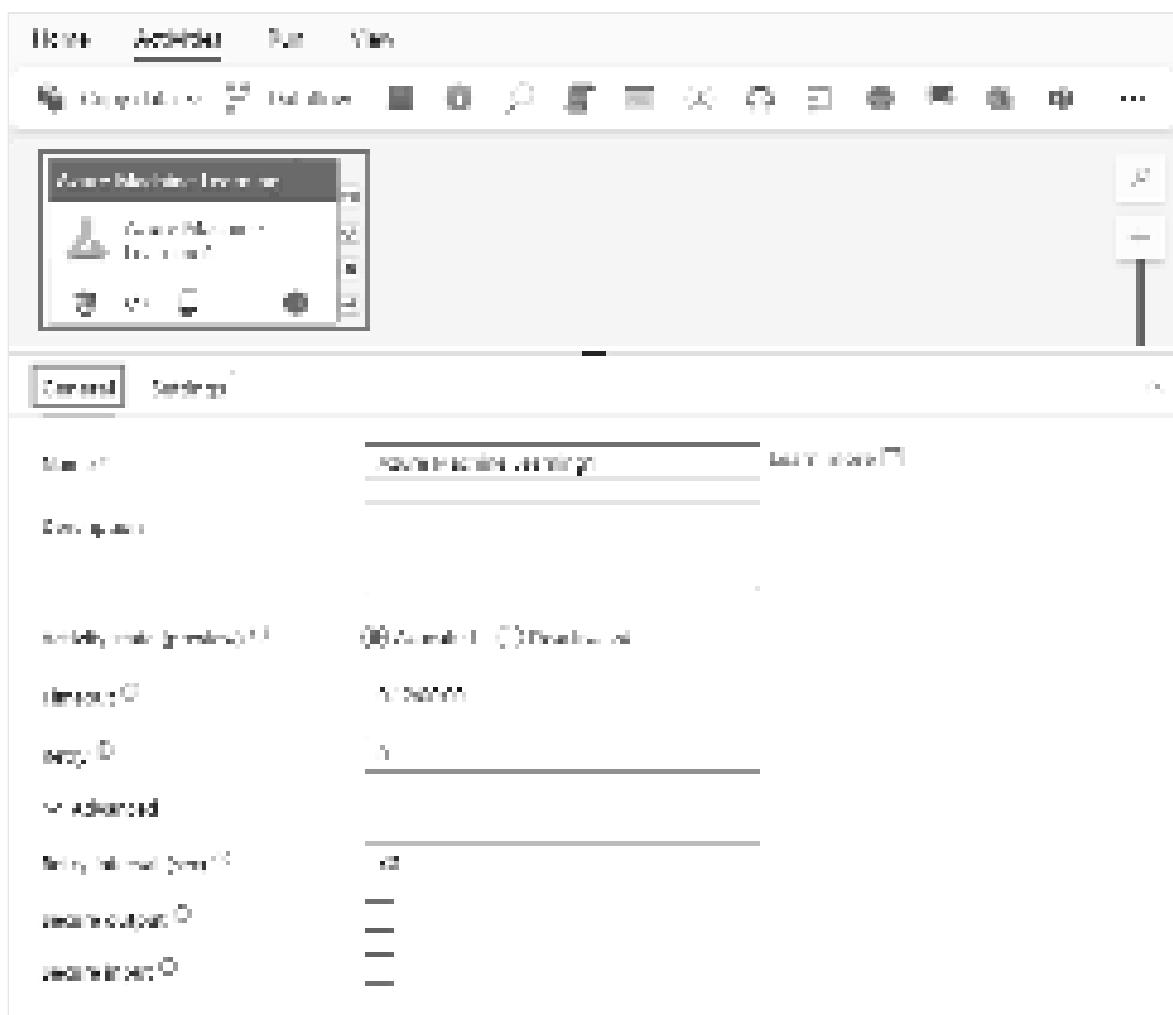
1. Create a new pipeline in your workspace.
2. Search for Azure Machine Learning in the pipeline **Activities** pane, and select it to add it to the pipeline canvas.

Note

You may need to expand the menu and scroll down to see the Azure Machine Learning activity as highlighted in following the screenshot.



3. Select the new Azure Batch activity on the pipeline editor canvas if it isn't already selected.



Refer to the General settings guidance to configure the General settings tab.

Azure Machine Learning activity settings

1. Select the **Settings** tab, then you can choose an existing or create a new **Azure Machine Learning connection**.
2. Choose and **Endpoint type**, either Batch Endpoint or Pipeline (v1).
3. Provide a **Batch endpoint** and **Batch deployment** and configure **Job settings for the Batch Endpoint type, or provide the pipeline details to run an Azure Machine Learning Pipeline (v1).

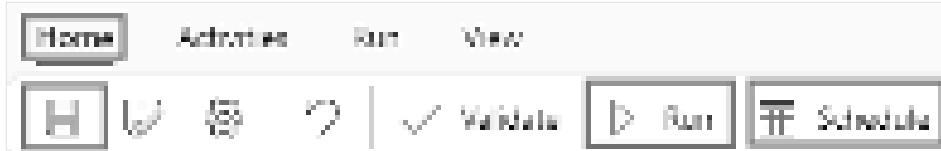
The screenshot shows the Azure Data Factory studio interface. At the top, there's a navigation bar with tabs: Home, Activities (which is selected), Run, and View. Below the navigation bar is a toolbar with various icons for operations like Copy data, Dataflow, and Transform. On the left, there's a sidebar titled "Azure Machine Learning" containing a list of activities: "Azure Machine Learning". The main workspace is titled "General" and has a sub-tab "Settings" which is currently selected. In the "Settings" tab, there are several configuration fields:

- "Azure Machine Learning connection": A dropdown menu labeled "Select..." with a refresh icon and a "New" button.
- "Endpoint Type": A radio button group where "Batch Endpoint" is selected, and "Pipeline (v1)" is an option.
- "Batch endpoint": A dropdown menu.
- "Batch deployment": A dropdown menu.
- "Job settings": A section with a "+ New" button.
- "Job inputs": A section with a "+ New" button, showing one entry: "name" and "value".
- "Job outputs": A section with a "+ New" button, showing one entry: "name" and "value".

On the right side of the workspace, there are two large buttons: a "P" button and a "+" button, and a vertical toolbar with various icons.

Save and run or schedule the pipeline

After you configure any other activities required for your pipeline, switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful? 👍 Yes 👎 No

Provide product feedback ↗ | Ask the community ↗

How to copy data using copy activity

Article • 11/15/2023

In Data Pipeline, you can use the Copy activity to copy data among data stores located in the cloud.

After you copy the data, you can use other activities to further transform and analyze it. You can also use the Copy activity to publish transformation and analysis results for business intelligence (BI) and application consumption.

To copy data from a source to a destination, the service that runs the Copy activity performs these steps:

1. Reads data from a source data store.
2. Performs serialization/deserialization, compression/decompression, column mapping, and so on. It performs these operations based on the configuration.
3. Writes data to the destination data store.

Prerequisites

To get started, you must complete the following prerequisites:

- A Microsoft Fabric tenant account with an active subscription. Create an account for free.
- Make sure you have a Microsoft Fabric enabled Workspace.

Add a copy activity using copy assistant

Follow these steps to set up your copy activity using copy assistant.

Start with copy assistant

1. Open an existing data pipeline or create a new data pipeline.
2. Select **Copy data** on the canvas to open the **Copy Assistant** tool to get started. Or select **Use copy assistant** from the **Copy data** drop down list under the **Activities** tab on the ribbon.



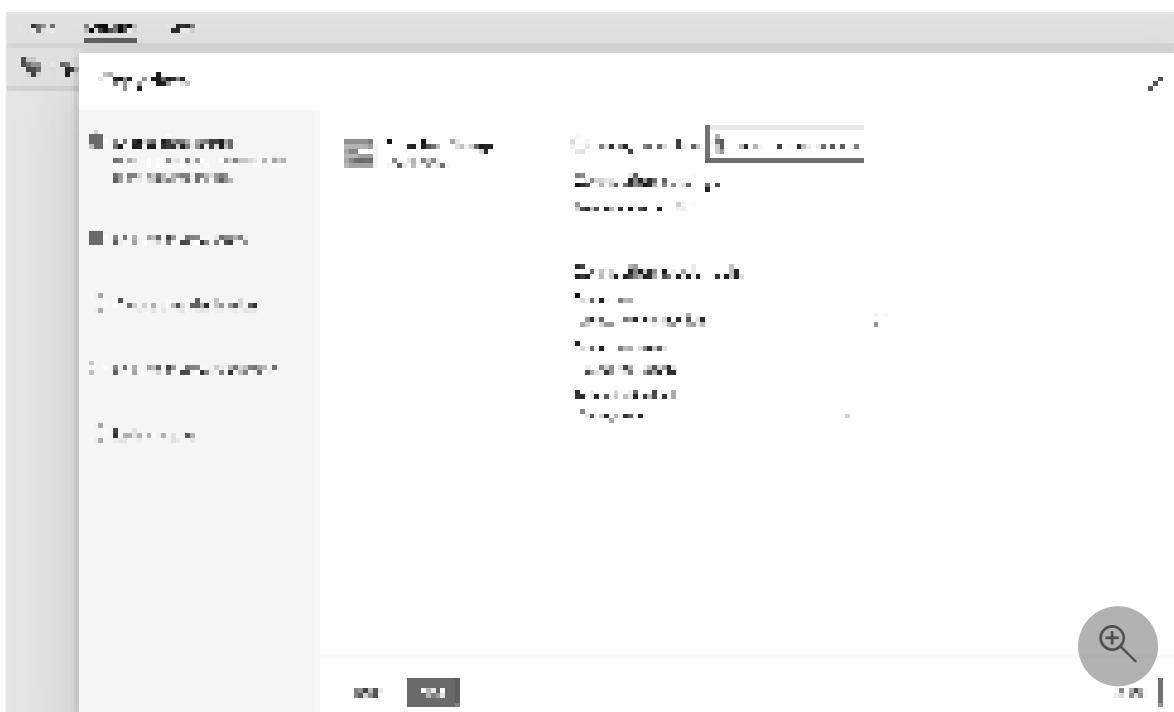
Configure your source

1. Select a data source type from the category. You'll use Azure Blob Storage as an example. Select **Azure Blob Storage** and then select **Next**.



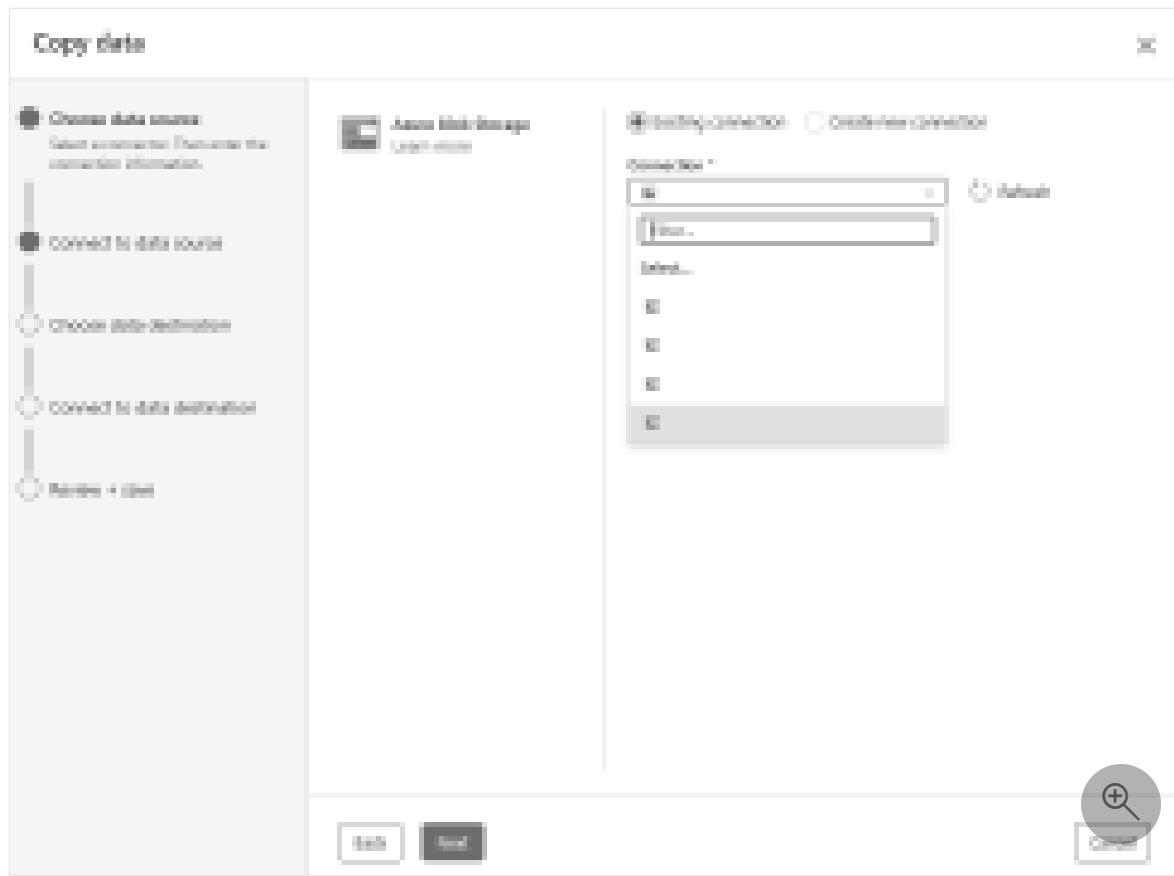


2. Create a connection to your data source by selecting **Create new connection**.

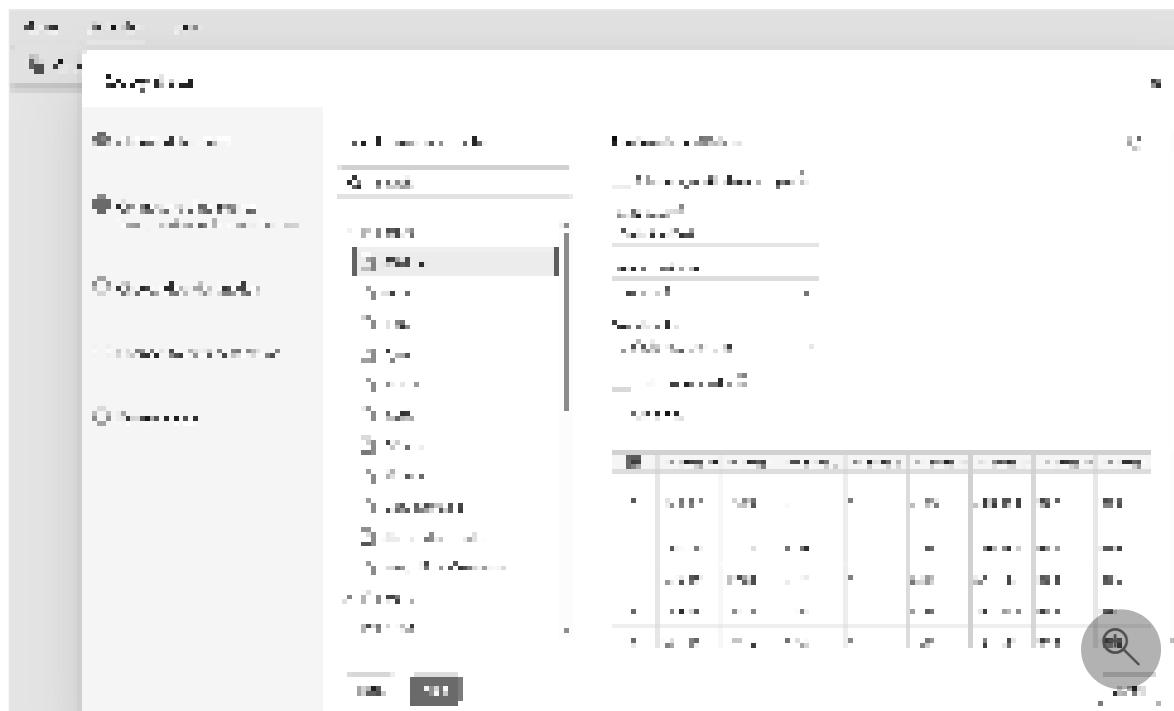


After you select **Create new connection**, fill in the required connection information and then select **Next**. For the details of connection creation for each type of data source, you can refer to each connector article.

If you have existing connections, you can select **Existing connection** and select your connection from the drop-down list.

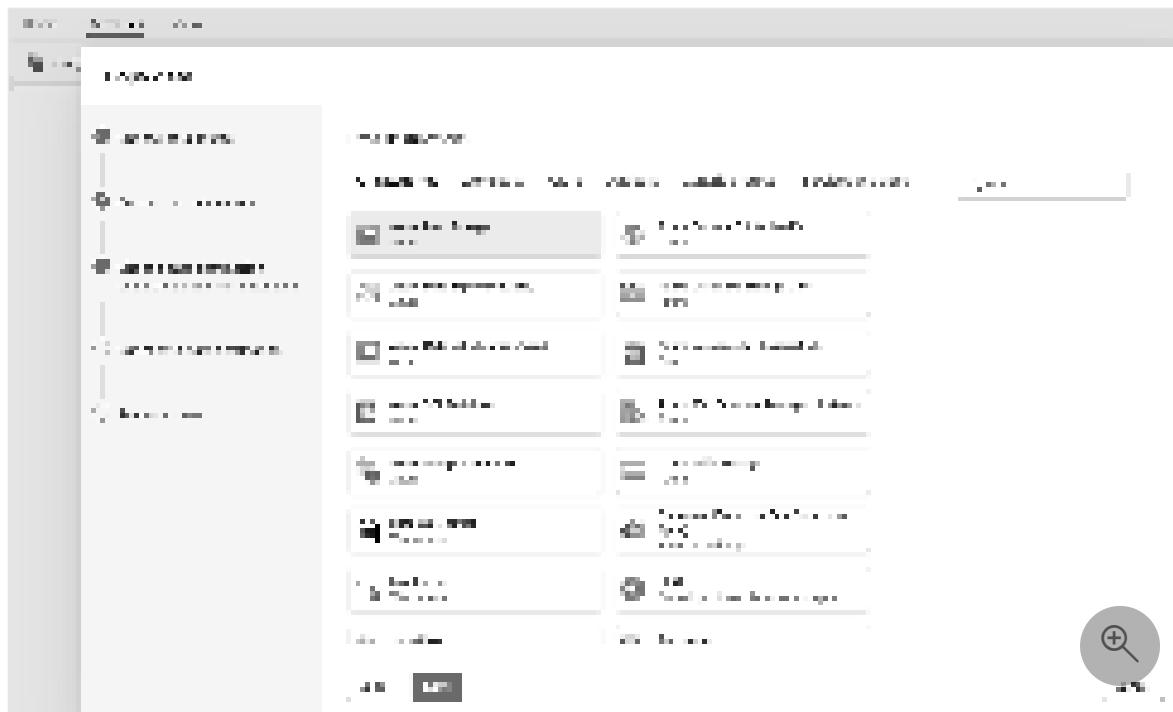


3. Choose the file or folder to be copied in this source configuration step, and then select **Next**.



Configure your destination

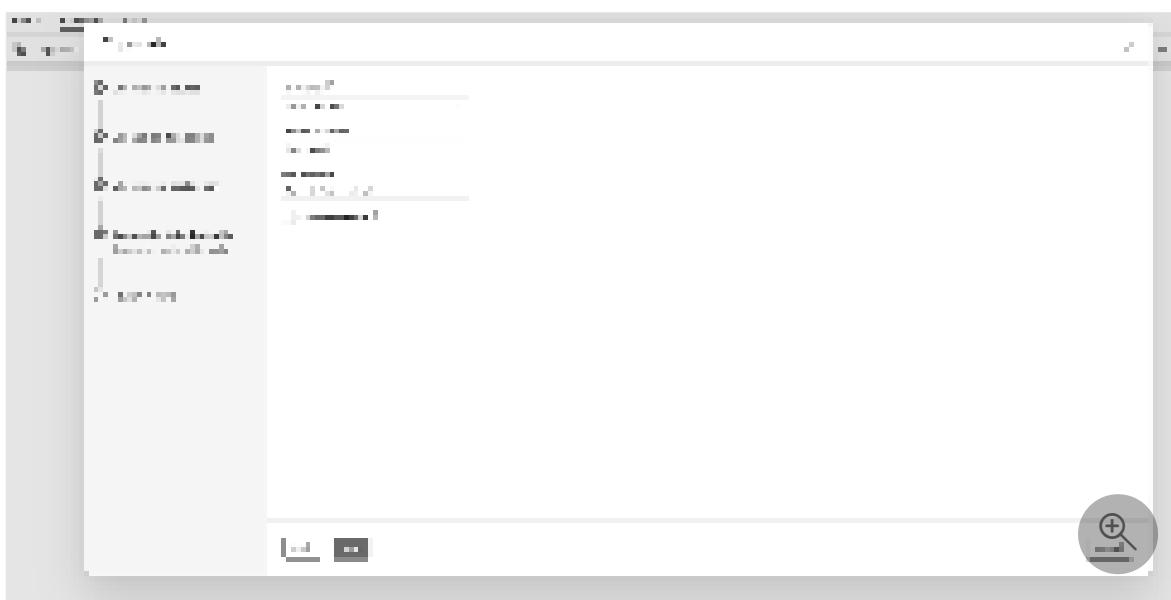
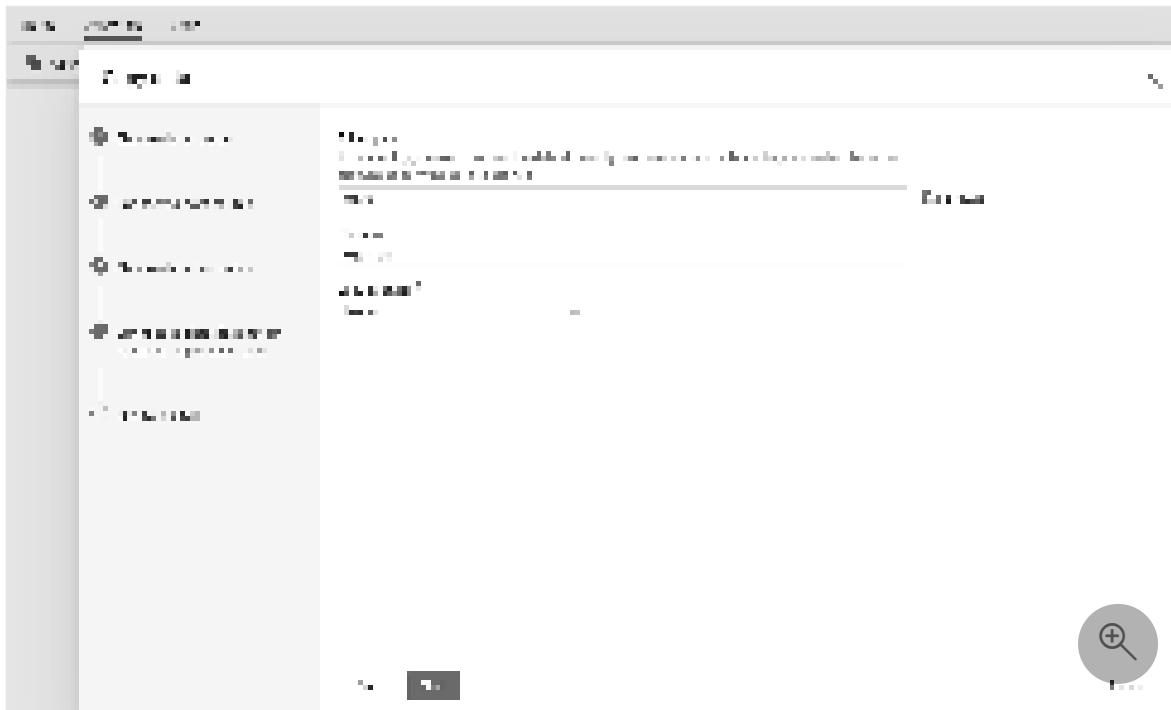
1. Select a data source type from the category. You'll use Azure Blob Storage as an example. Select **Azure Blob Storage**, and then select **Next**.



2. You can either create a new connection that links to a new Azure Blob Storage account by following the steps in the previous section or use an existing connection from the connection drop-down list. The capabilities of **Test connection** and **Edit** are available to each selected connection.

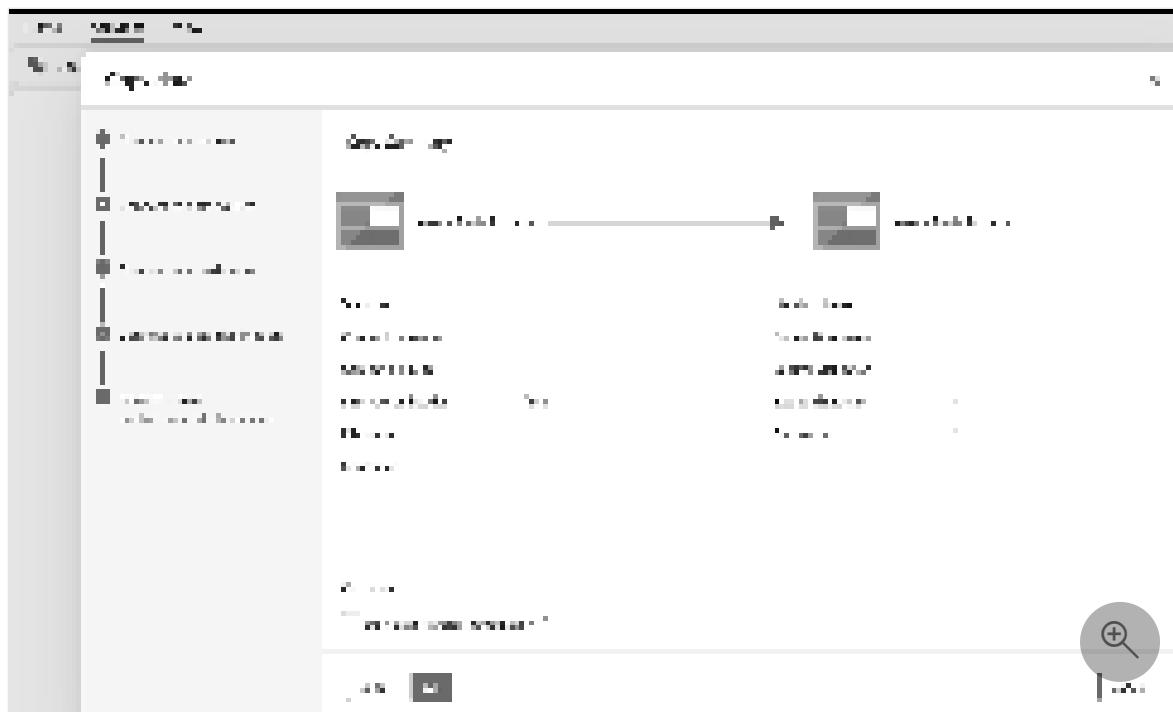


3. Configure and map your source data to your destination. Then select **Next** to finish your destination configurations.



Review and create your copy activity

1. Review your copy activity settings in the previous steps and select **OK** to finish. Or you can go back to the previous steps to edit your settings if needed in the tool.



Once finished, the copy activity will then be added to your data pipeline canvas. All settings, including advanced settings to this copy activity, are available under the tabs when it's selected.

The screenshot shows the 'Copy data' activity configuration screen. At the top, there's a navigation bar with 'Home', 'Activities' (which is underlined), and 'View'. Below the navigation bar are several icons: 'Copy data' (selected), 'Dataflow', 'Notebook', 'Get metadata', and 'Lookup'. The main area is titled 'Copy data' and contains a sub-section for 'Copy job'. There are tabs for 'General', 'Source', 'Destination', 'Mapping', and 'Settings'. The 'General' tab is selected. Under 'General', the 'Name' field is set to 'Copied' and has a 'Learn more' link. The 'Description' field is empty. The 'Timeout' field is set to '0. 20000'. The 'Retry' field is set to '0'. The 'Retries' dropdown is set to 'Unlimited'. The 'Retry interval (sec)' field is set to '20'. The 'Secure output' and 'Secure input' dropdowns are both set to 'None'. A search icon is located next to the 'Secure input' dropdown.

General Source Destination Mapping Settings

Name: Copied [Learn more](#)

Description:

Timeout: 0. 20000

Retry: 0

Retries: Unlimited

Retry interval (sec): 20

Secure output: None

Secure input: None

Now you can either save your data pipeline with this single copy activity or continue to design your data pipeline.

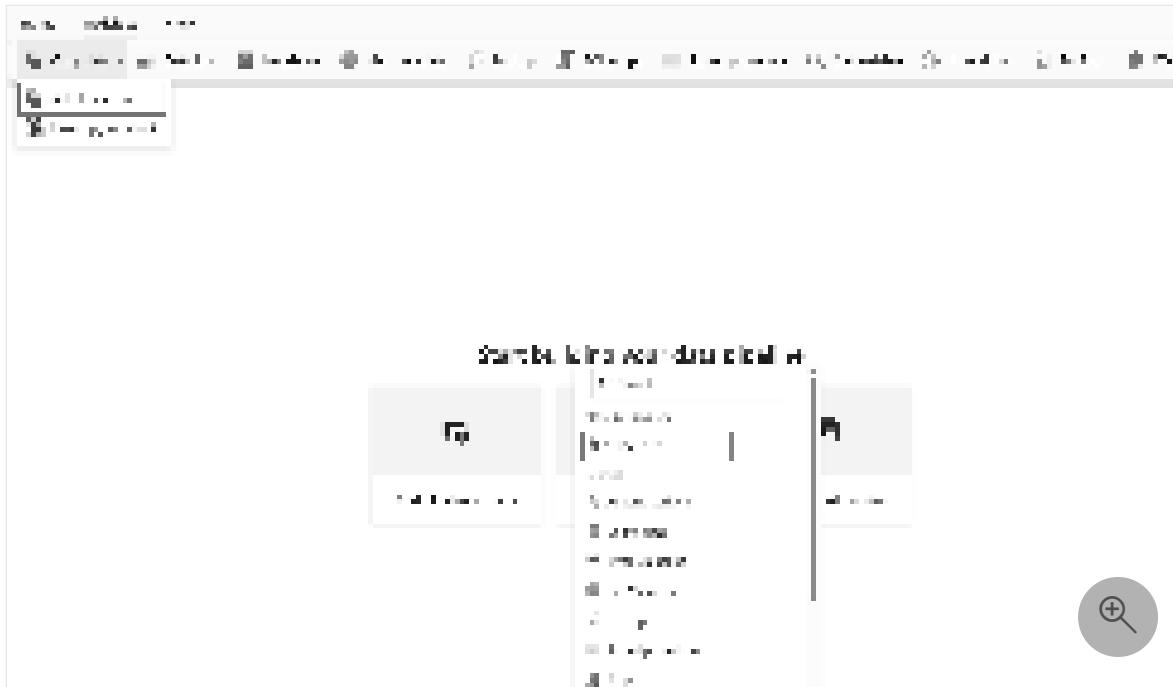
Add a copy activity directly

Follow these steps to add a copy activity directly.

Add a copy activity

1. Open an existing data pipeline or create a new data pipeline.

2. Add a copy activity either by selecting **Add pipeline activity > Copy activity** or by selecting **Copy data > Add to canvas** under the **Activities** tab.



Configure your general settings under general tab

To learn how to configure your general settings, see [General](#).

Configure your source under the source tab

1. Select **+ New** beside the **Connection** to create a connection to your data source.



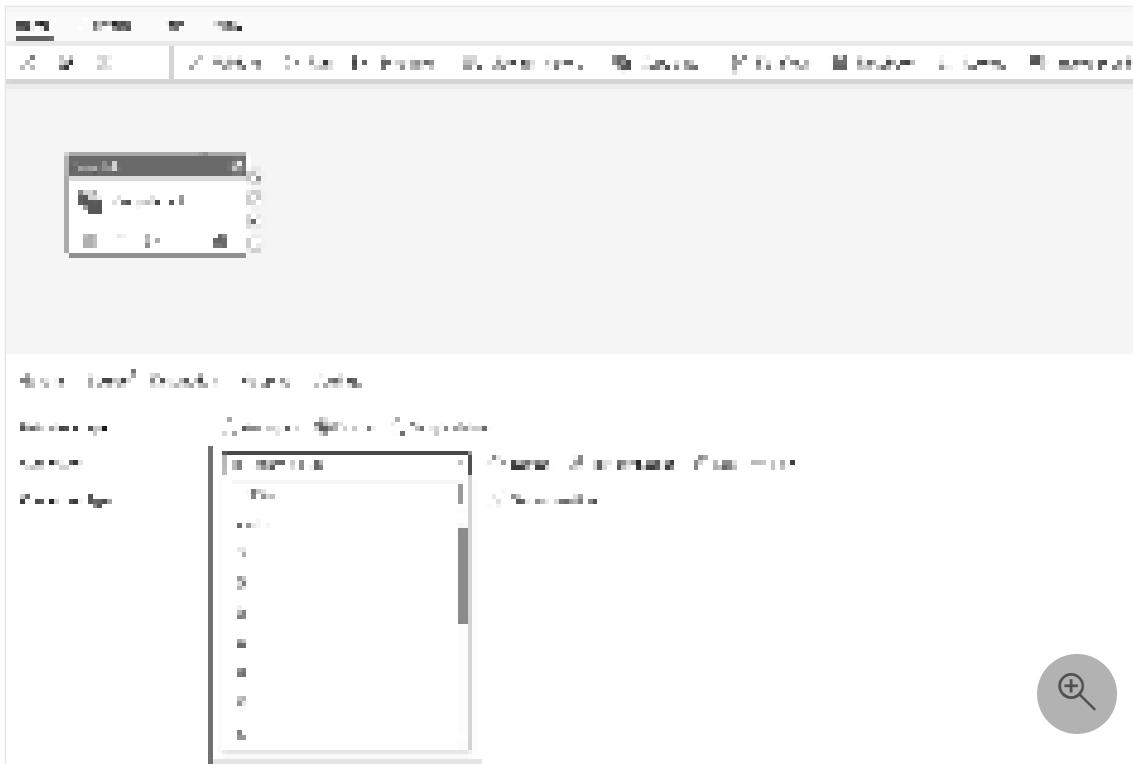
- a. Choose the data source type from the pop-up window. You'll use Azure SQL Database as an example. Select **Azure SQL Database**, and then select **Continue**.



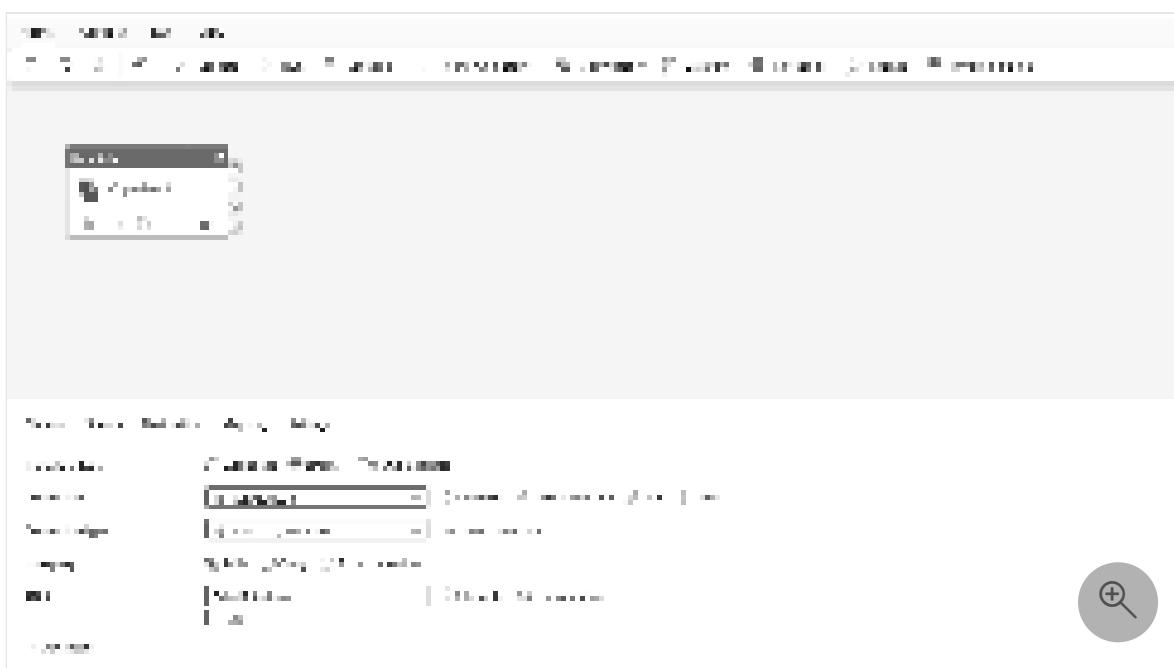
- b. It navigates to the connection creation page. Fill in the required connection information on the panel, and then select **Create**. For the details of connection creation for each type of data source, you can refer to each connector article.



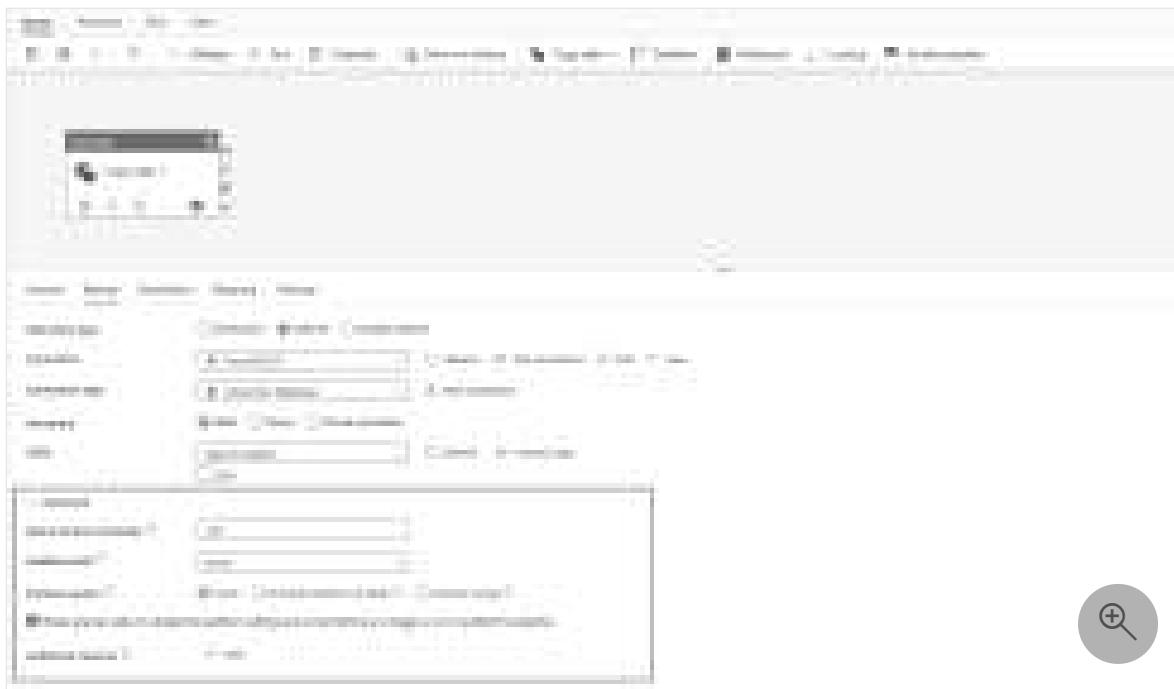
- c. Once your connection is created successfully, it takes you back to the data pipeline page. Then select **Refresh** to fetch the connection that you created from the drop-down list. You could also choose an existing Azure SQL Database connection from the drop-down directly if you already created it before. The capabilities of **Test connection** and **Edit** are available to each selected connection. Then select **Azure SQL Database** in **Connection type**.



2. Specify a table to be copied. Select **Preview data** to preview your source table. You can also use **Query** and **Stored procedure** to read data from your source.

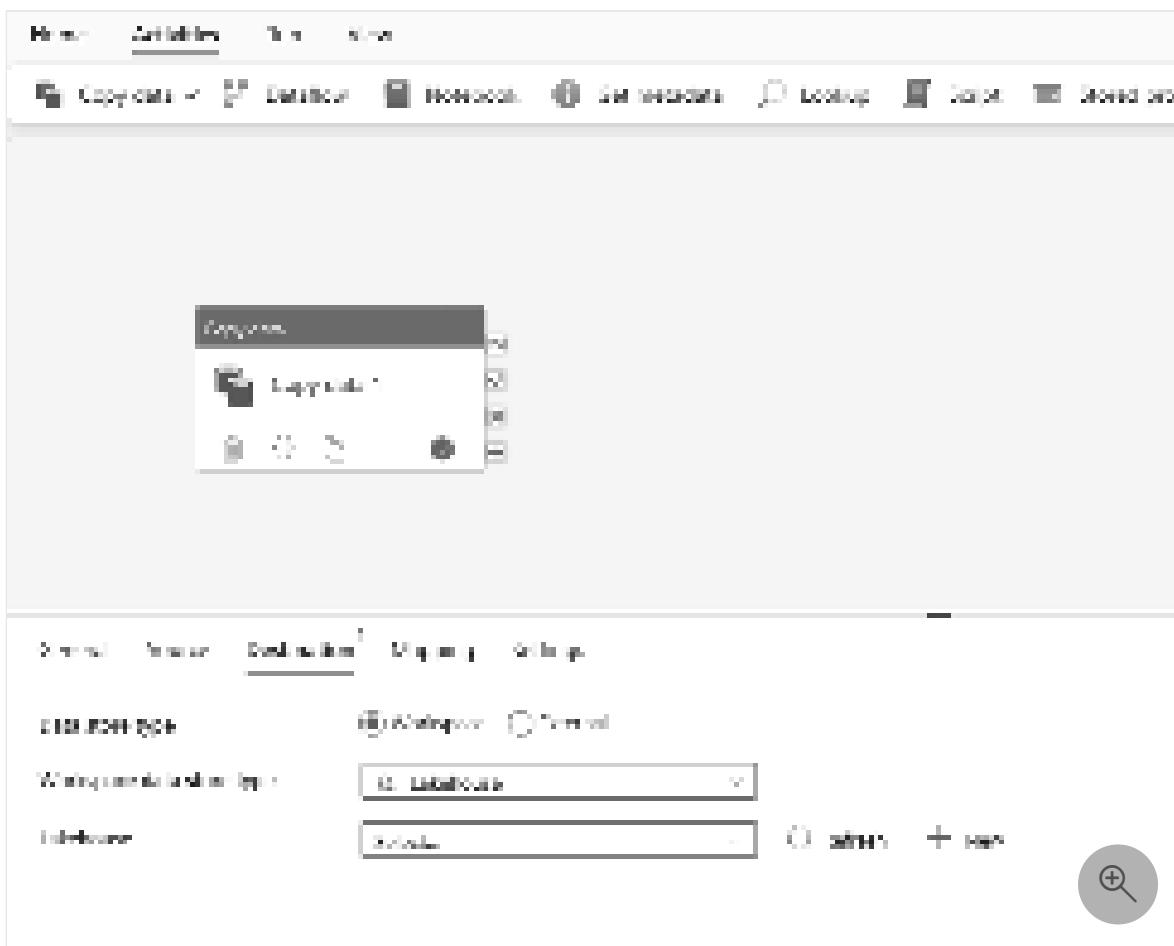


3. Expand **Advanced** for more advanced settings.



Configure your destination under the destination tab

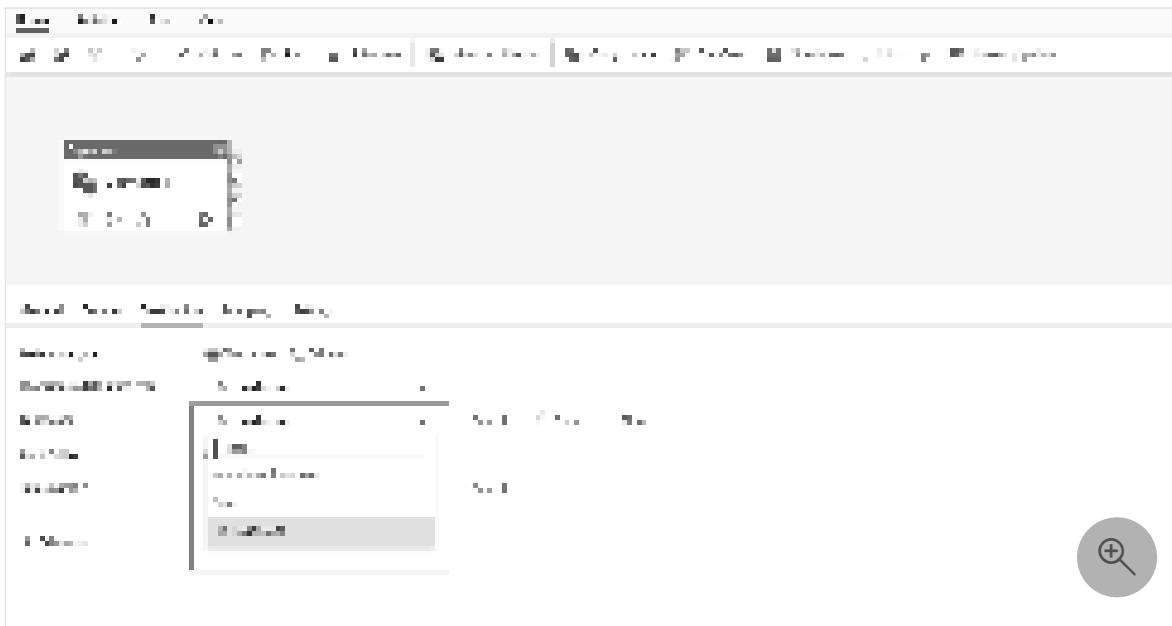
1. Choose your destination type. It could be either your internal first class data store from your workspace, such as Lakehouse, or your external data stores. You'll use Lakehouse as an example.



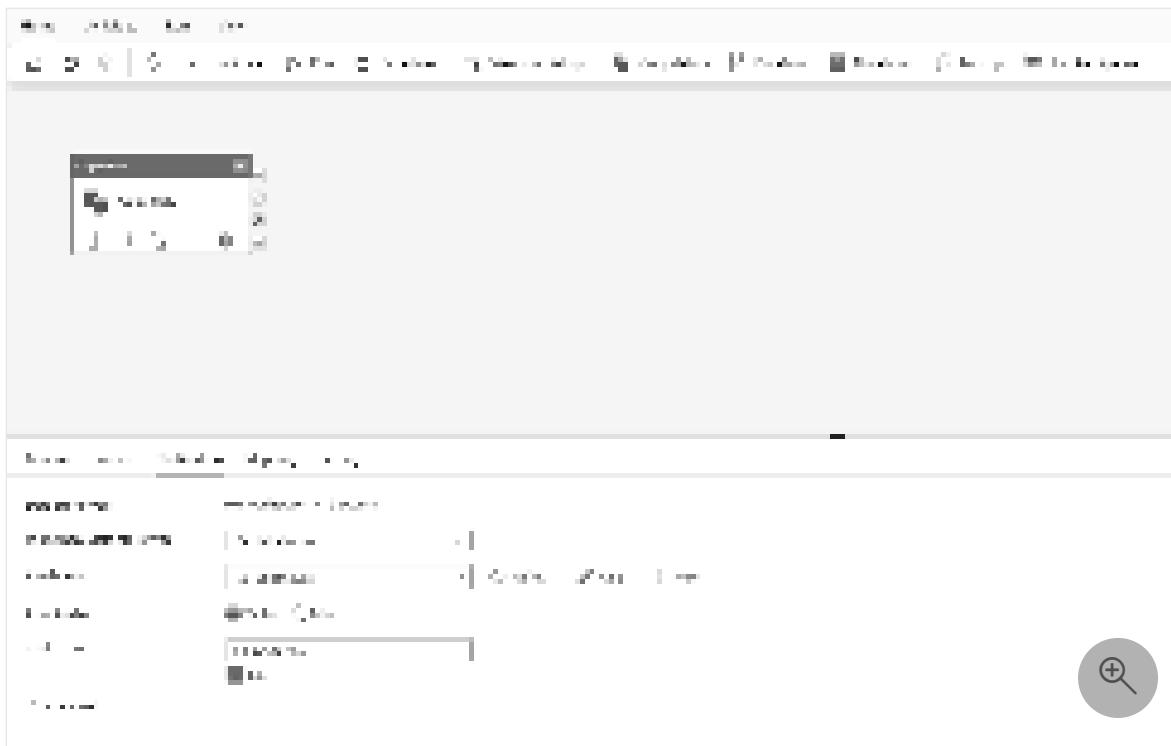
2. Choose to use **Lakehouse** in **Workspace data store type**. Select **+ New**, and it navigates you to the Lakehouse creation page. Specify your Lakehouse name and then select **Create**.



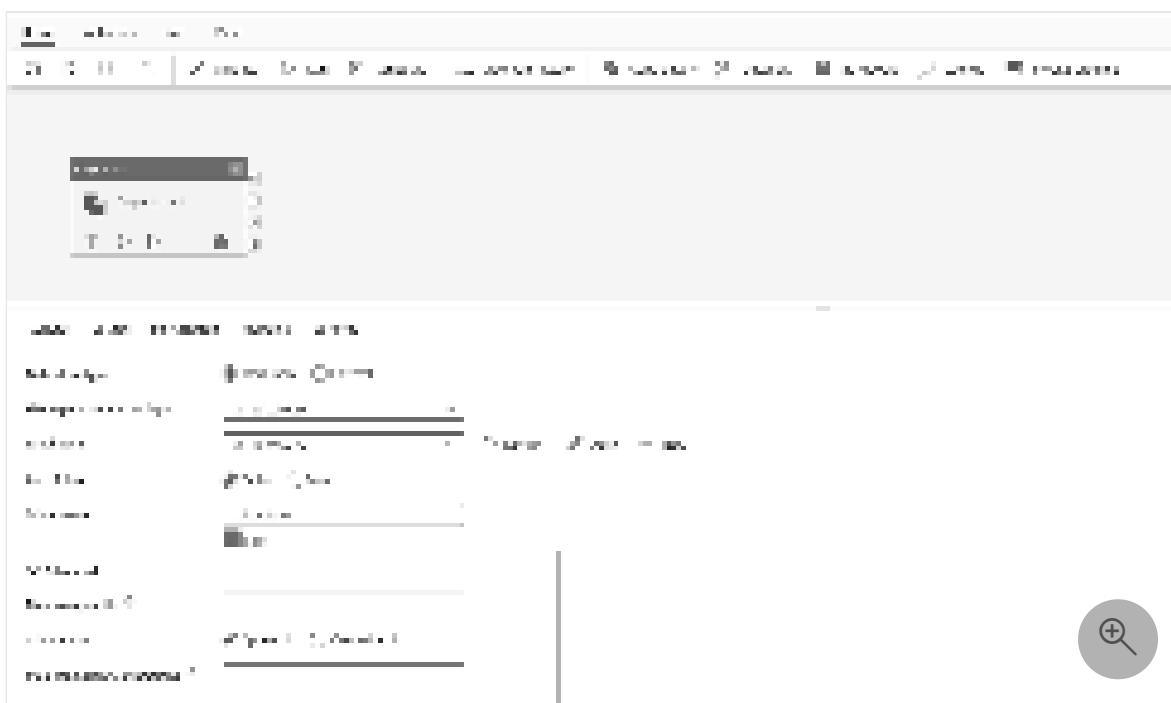
3. Once your connection is created successfully, it takes you back to the data pipeline page. Then select **Refresh** to fetch the connection that you created from the drop-down list. You could also choose an existing Lakehouse connection from the drop-down directly if you already created it before.



4. Specify a table or set up the file path to define the file or folder as the destination. Here select **Tables** and specify a table to write data.



5. Expand Advanced for more advanced settings.

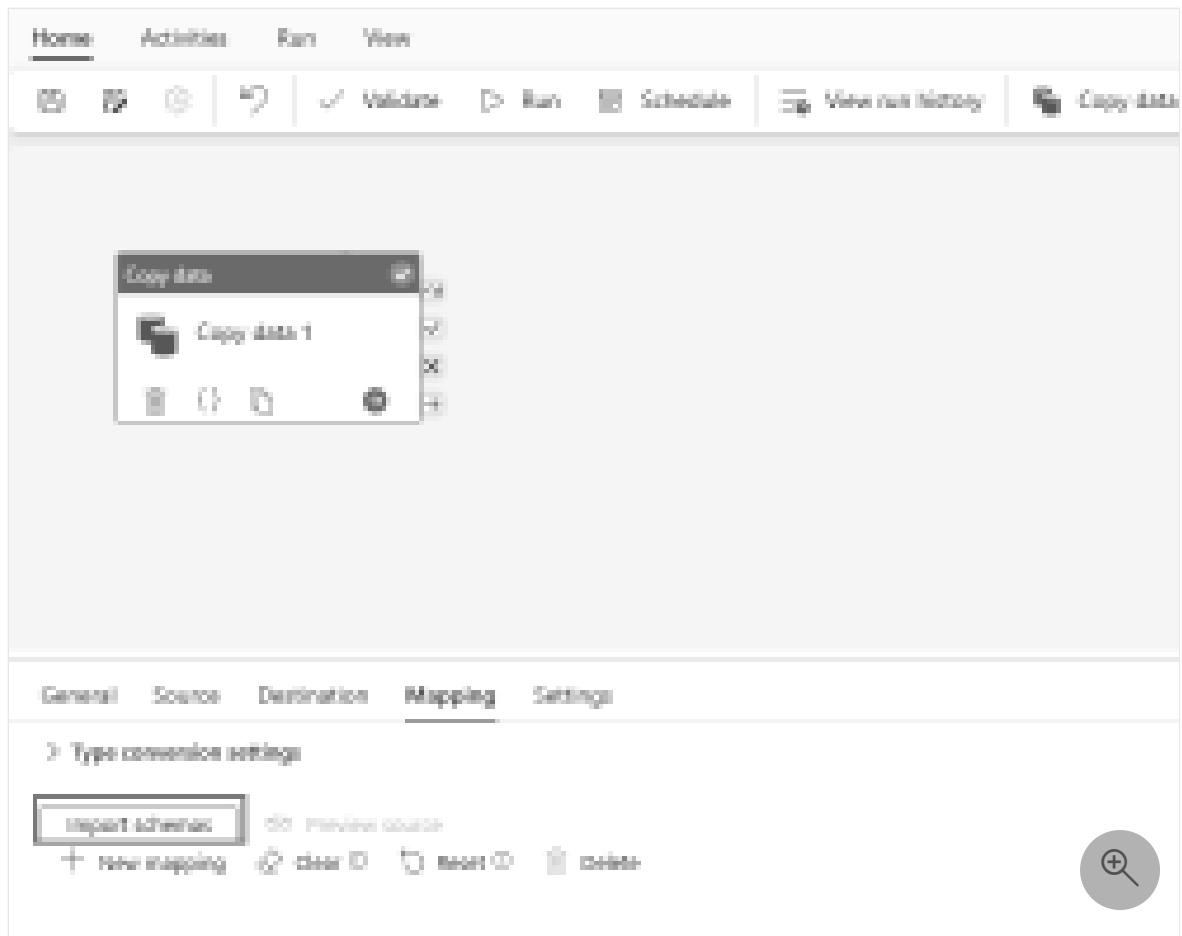


Now you can either save your data pipeline with this single copy activity or continue to design your data pipeline.

Configure your mappings under mapping tab

If the connector that you apply supports mapping, you can go to **Mapping** tab to configure your mapping.

1. Select **Import schemas** to import your data schema.



2. You can see the auto mapping is shown up. Specify your **Source** column and **Destination** column. If you create a new table in the destination, you can customize your **Destination** column name here. If you want to write data into the existing destination table, you can't modify the existing **Destination** column name. You can also view the **Type** of source and destination columns.

Mapping			
Type conversion settings			
<input type="button" value="Import schema"/> My_Warehouse			
<input type="button" value="+ New mapping"/> <input type="button" value="Clear"/> <input type="button" value="Reset"/> <input type="button" value="Delete"/>			
	Name	Type	Destination
<input type="checkbox"/>	Address	int	<input type="button" value="Address"/>
<input type="checkbox"/>	AddressLine1	int	<input type="button" value="AddressLine1"/>
<input type="checkbox"/>	AddressLine2	int	<input type="button" value="AddressLine2"/>
<input type="checkbox"/>	City	int	<input type="button" value="City"/>
<input type="checkbox"/>	StateProvince	int	<input type="button" value="StateProvince"/>
<input type="checkbox"/>	CountryRegion	int	<input type="button" value="CountryRegion"/>
<input type="checkbox"/>	PostalCode	int	<input type="button" value="PostalCode"/>
<input type="checkbox"/>	Region	int	<input type="button" value="Region"/>
<input type="checkbox"/>	RegionName	int	<input type="button" value="RegionName"/>

Besides, you can select **+ New mapping** to add new mapping, select **Clear** to clear all mapping settings, and select **Reset** to reset all mapping **Source** column.

Configure your type conversion

Expand Type conversion settings to configure your type conversion if needed.

The screenshot shows the 'Mapping' tab of the Data Flow Task Editor. At the top, there are tabs for General, Source, Destination, Mapping (which is selected), and Settings. Below the tabs, a section titled 'Type conversion settings' is expanded, indicated by a checkmark icon. This section contains five configuration items:

- Allow data truncation**: A checked checkbox.
- Treat boolean as number**: An unchecked checkbox.
- DateTime format**: An empty text input field.
- DateTimeOffset format**: An empty text input field.
- TimeSpan format**: An empty text input field.
- Culture**: An empty text input field.

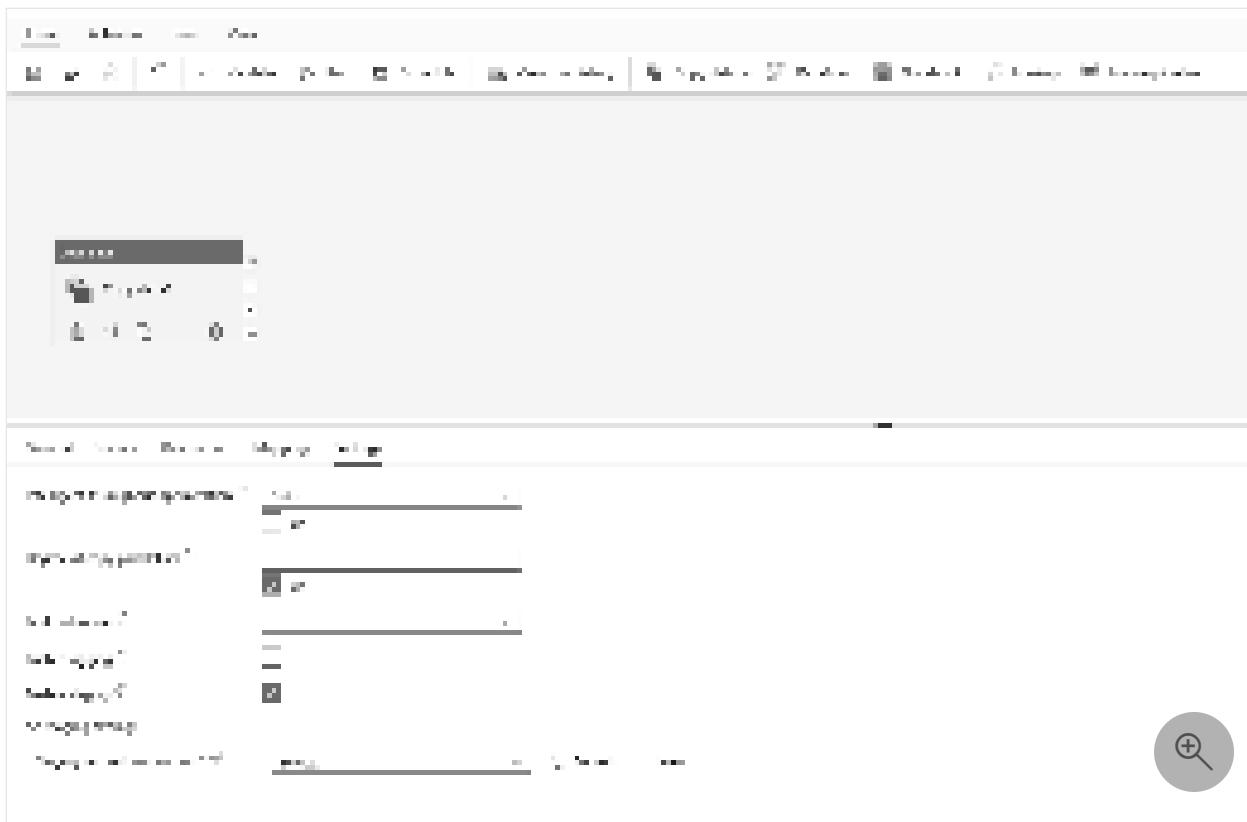
At the bottom of the 'Mapping' tab, there are several buttons: 'Import schema' (underlined), 'Preview source', '+ New mapping' (highlighted with a red box), 'Clear', 'Next', and 'Delete'. To the right of these buttons is a magnifying glass icon with a plus sign inside it.

See the following table for the setting details.

Setting	Description
Allow data truncation	Allow data truncation when converting source data to destination with different type during copy. For example, from decimal to integer, from DateTimeOffset to Datetime.
Treat boolean as number	Treat boolean as number. For example, treat true as 1.
DateTime format	Format string when converting between dates without time zone offset and strings. For example, "yyyy-MM-dd HH:mm:ss.fff".
DateTimeOffset format	Format string when converting between dates with time zone offset and strings. For example, "yyyy-MM-dd HH:mm:ss.fff zzz".
TimeSpan format	Format string when converting between time periods and strings. For example, "dd.hh:mm:ss".
Culture	Culture information to be used when convert types. For example, "en-us", "fr-fr".

Configure your other settings under settings tab

The **Settings** tab contains the settings of performance, staging, and so on.



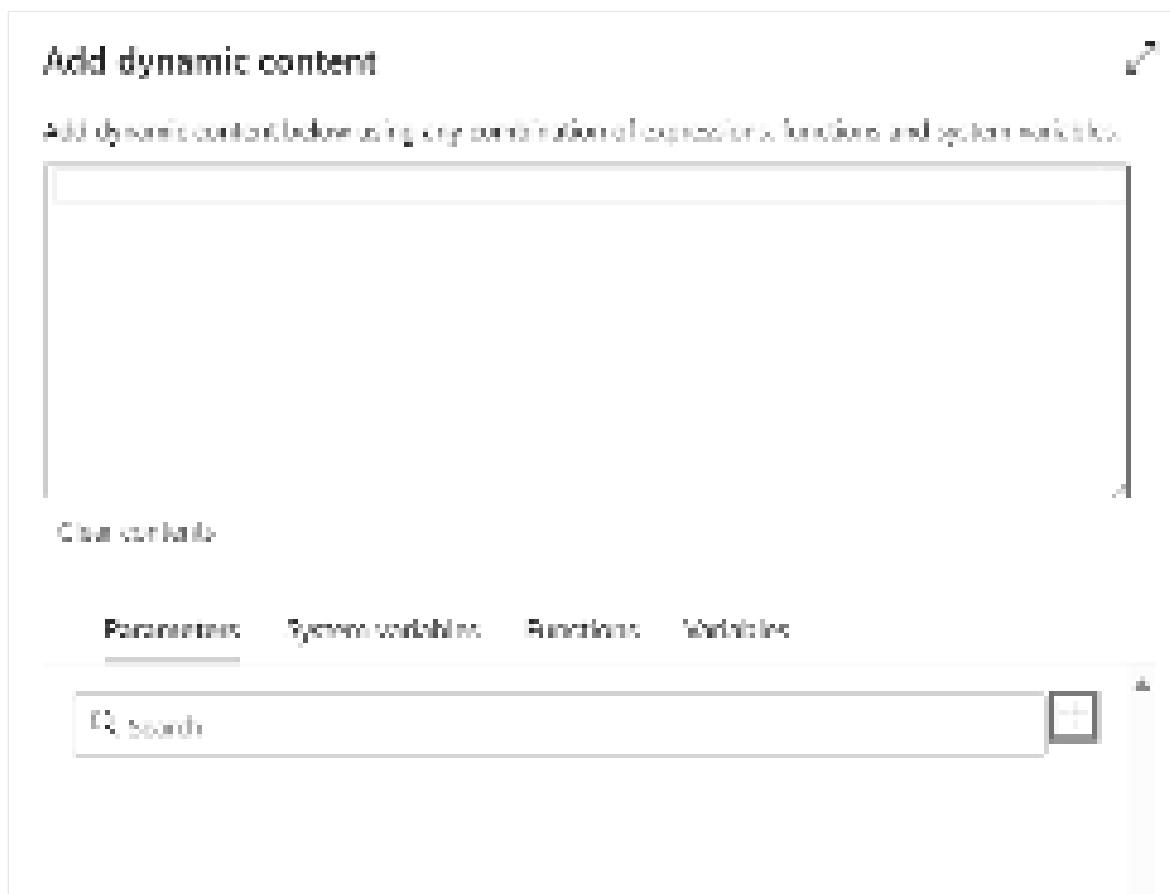
See the following table for the description of each setting.

Setting	Description
Intelligent throughput optimization	Specify to optimize the throughput. You can choose from: <ul style="list-style-type: none">• Auto• Standard• Balanced• Maximum When you choose Auto , the optimal setting is dynamically applied based on your source-destination pair and data pattern. You can also customize your throughput, and custom value can be 2-256 while higher value implies more gains.
Degree of copy parallelism	Specify the degree of parallelism that data loading would use.
Fault tolerance	When selecting this option, you can ignore some errors occurred in the middle of copy process. For example, incompatible rows between source and destination store, file being deleted during data movement, etc.
Enable logging	When selecting this option, you can log copied files, skipped files and rows
Enable staging	Specify whether to copy data via an interim staging store. Enable staging only for the beneficial scenarios.
Staging account connection	When selecting Enable staging , specify the connection of an Azure storage data source as an interim staging store. Select + New to create a staging connection if you don't have it.

Configure parameters in a copy activity

Parameters can be used to control the behavior of a pipeline and its activities. You can use **Add dynamic content** to specify parameters for your copy activity properties. Let's take specifying Lakehouse/Data Warehouse/KQL Database as an example to see how to use it.

1. In your source or destination, after selecting **Workspace** as data store type and specifying **Lakehouse/Data Warehouse/KQL Database** as workspace data store type, select **Add dynamic content** in the drop-down list of **Lakehouse or Data Warehouse or KQL Database**.
2. In the pop-up **Add dynamic content** pane, under **Parameters** tab, select **+**.



3. Specify the name for your parameter and give it a default value if you want, or you can specify the value for the parameter after selecting **Run** in the pipeline.

New parameter

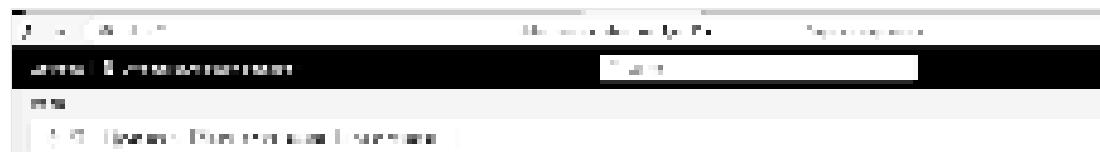
Name:

Type:

Default value:

Note that the parameter value should be Lakehouse/Data Warehouse/KQL Database object ID. To get your Lakehouse/Data Warehouse/KQL Database object ID, open your Lakehouse/Data Warehouse/KQL Database in your workspace, and the ID is after `/lakehouses/` or `/datawarehouses/` or `/databases/` in your URL.

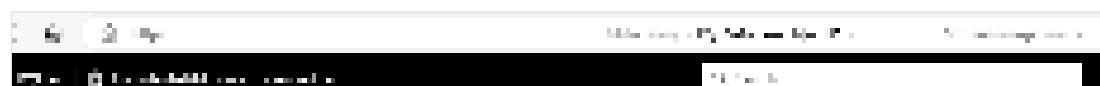
- **Lakehouse object ID:**



- **Data Warehouse object ID:**



- **KQL Database object ID:**



4. Select **Save** to go back to the **Add dynamic content** pane. Then select your parameter so it appears in the expression box. Then select **OK**. You'll go back to the pipeline page and can see the parameter expression is specified after **Lakehouse object ID/Data Warehouse object ID/KQL Database object ID**.

Add dynamic content

Add dynamic content below using any combination of extensions, functions and system variables.

```
[[pipeline().parameters.DatabaseName]]
```

Clear content

Parameters System variables Functions Variables

Q Search

Data Variables
Previous parameter

Next steps

- Connector overview
- How to monitor pipeline runs

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Use the Dataflow activity to run a Dataflow Gen2

Article • 12/11/2023

The Dataflow activity in Data Factory for Microsoft Fabric allows you to run a Dataflow Gen2.

Prerequisites

To get started, you must complete the following prerequisites:

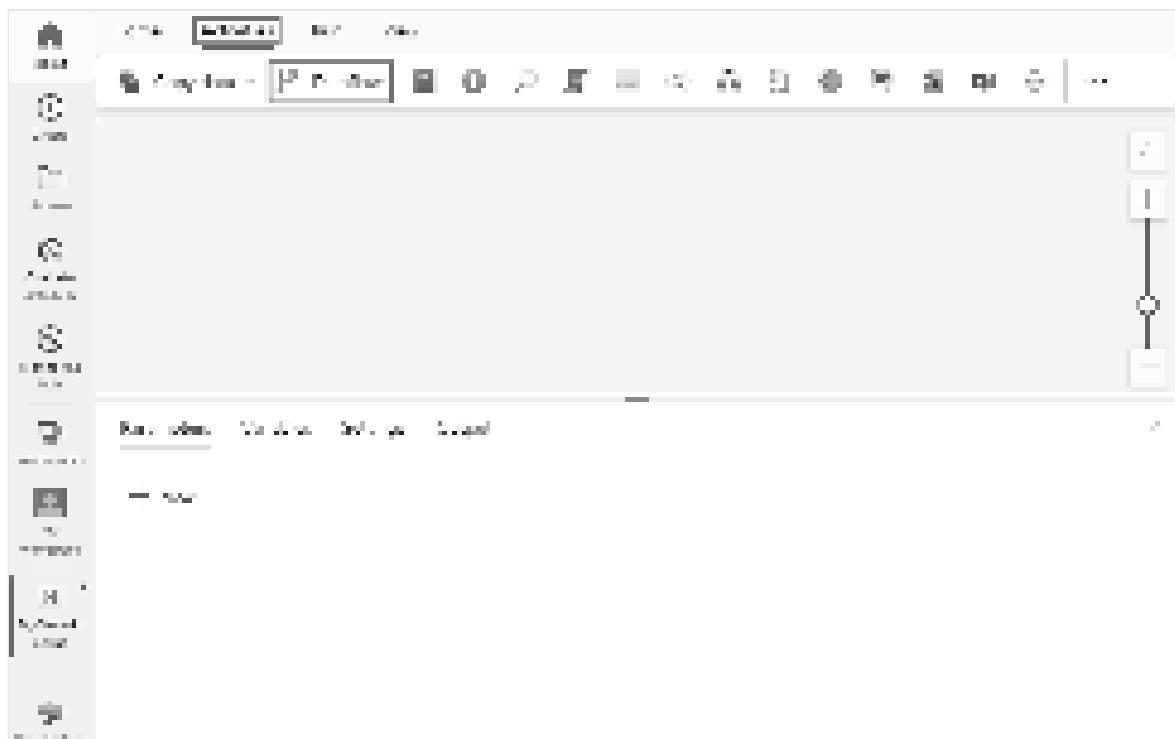
- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add a Dataflow activity to a pipeline with UI

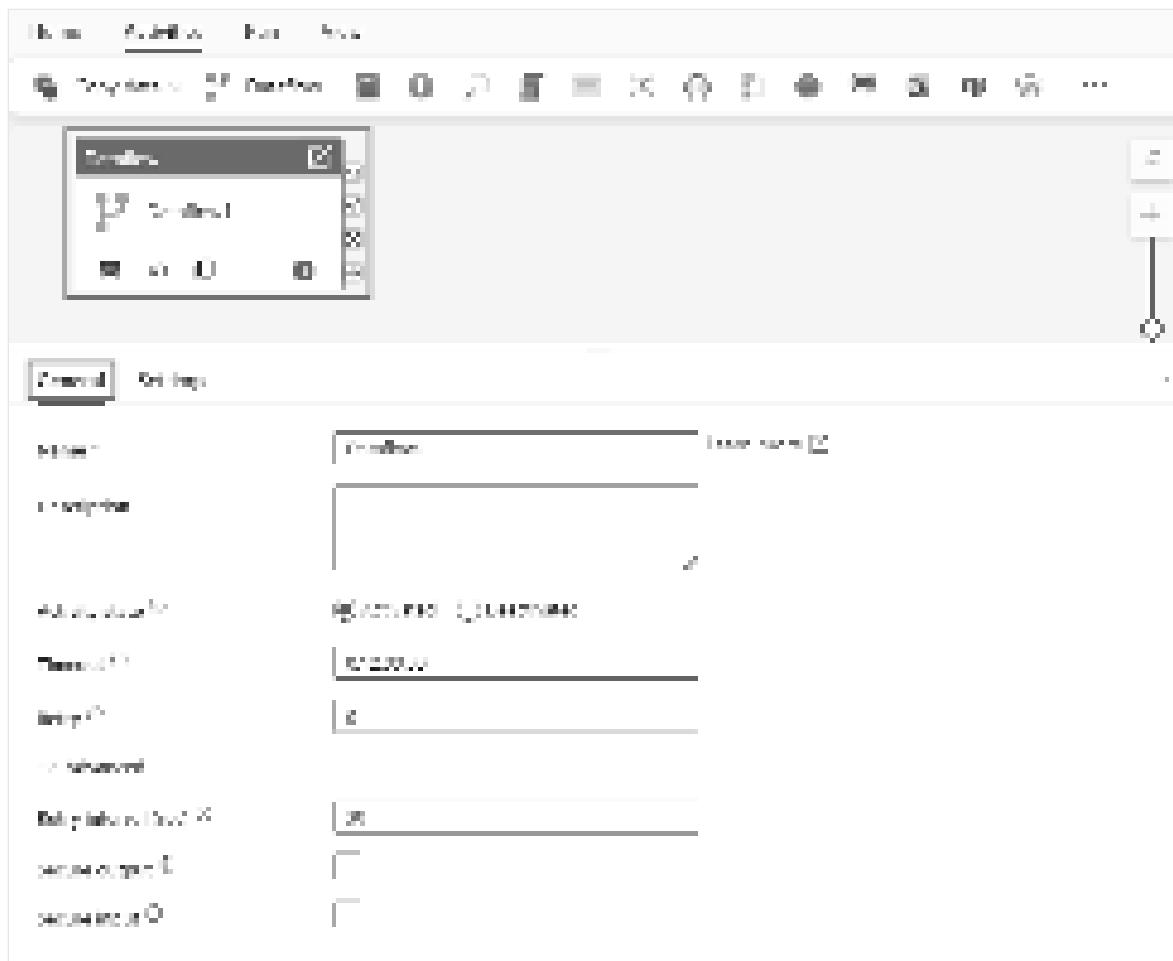
To use a Dataflow activity in a pipeline, complete the following steps:

Creating the activity

1. Create a new pipeline in your workspace.
2. Search for Dataflow in the pipeline **Activities** pane, and select it to add it to the pipeline canvas.



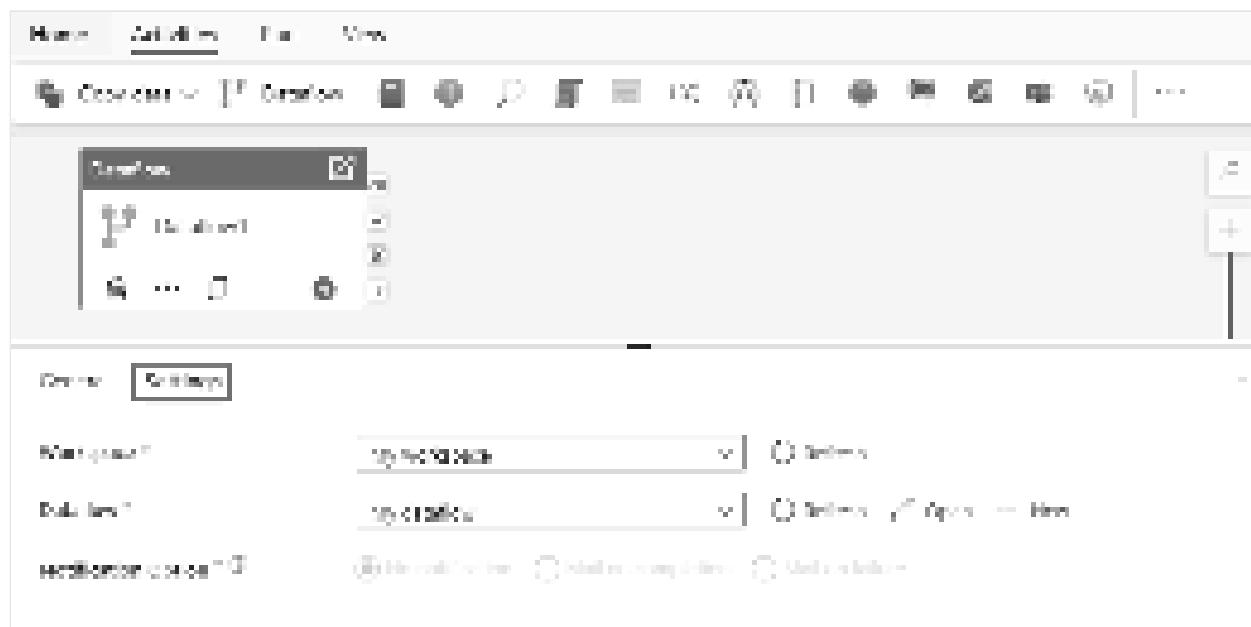
3. Select the new Dataflow activity on the canvas if it isn't already selected.



Refer to the General settings guidance to configure the **General** settings tab.

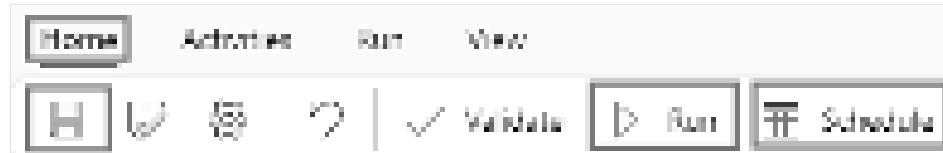
Dataflow activity settings

Select the **Settings** tab, then select an existing workspace and dataflow to run. The notification option is disabled for now, but the feature is coming soon.



Save and run or schedule the pipeline

After you configure any other activities required for your pipeline, switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Related content

How to monitor pipeline runs

Feedback

Was this page helpful?

Yes

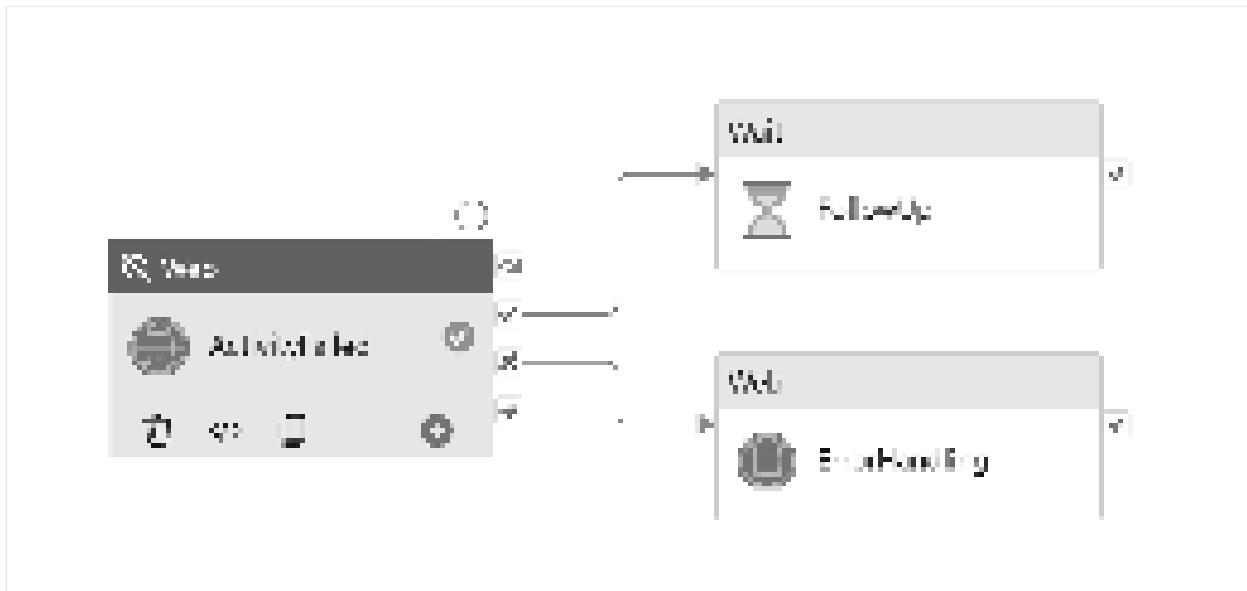
No

Provide product feedback | Ask the community

Deactivate an activity

Article • 11/15/2023

You can now deactivate one or more activities from a pipeline, and we skip them during validation and during pipeline run. This feature significantly improves pipeline developer efficiency, allowing customers to comment out part of the pipeline, without deleting it from the canvas. You may choose to reactivate them at a later time.

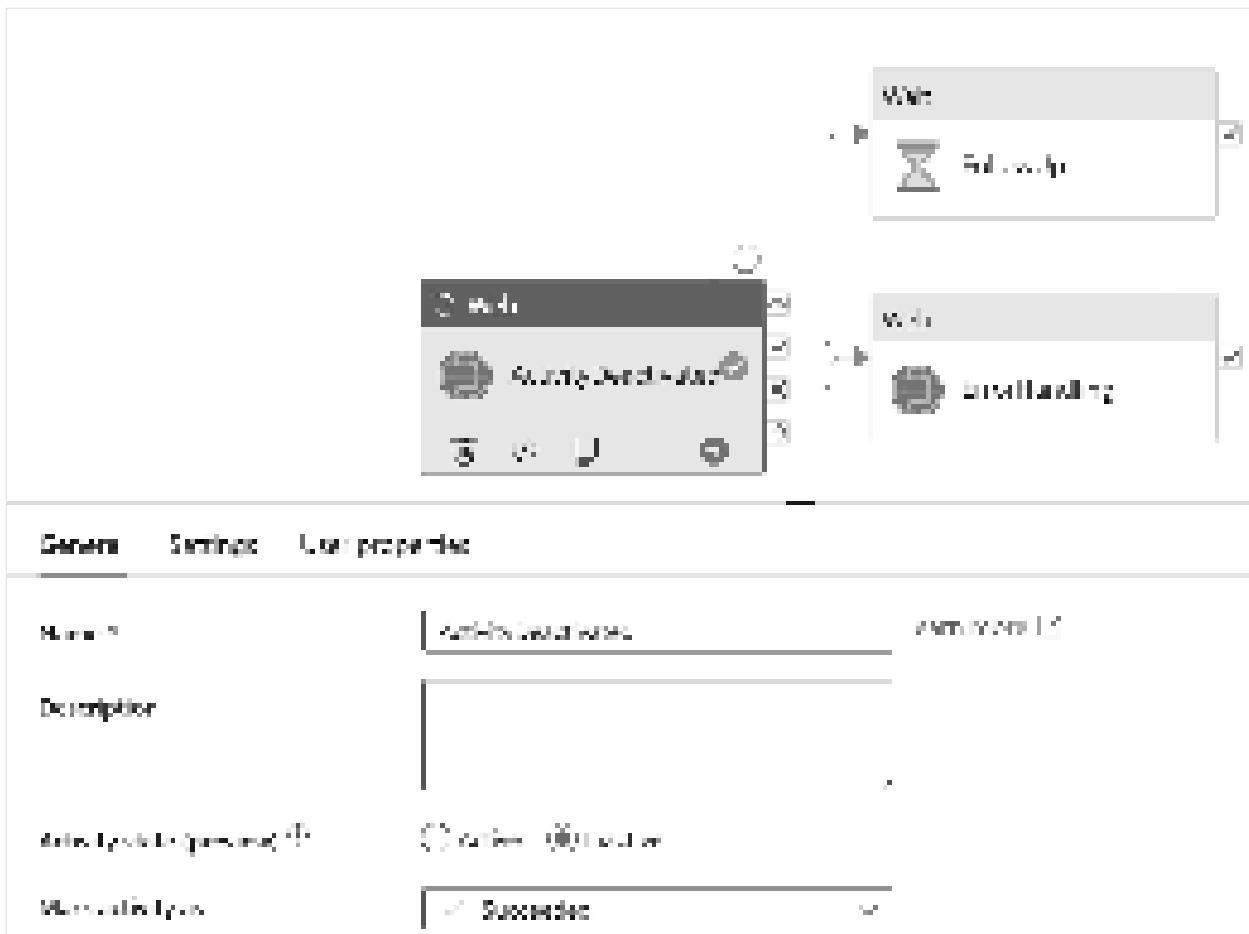


Deactivate and reactivate

There are two ways to deactivate an activity.

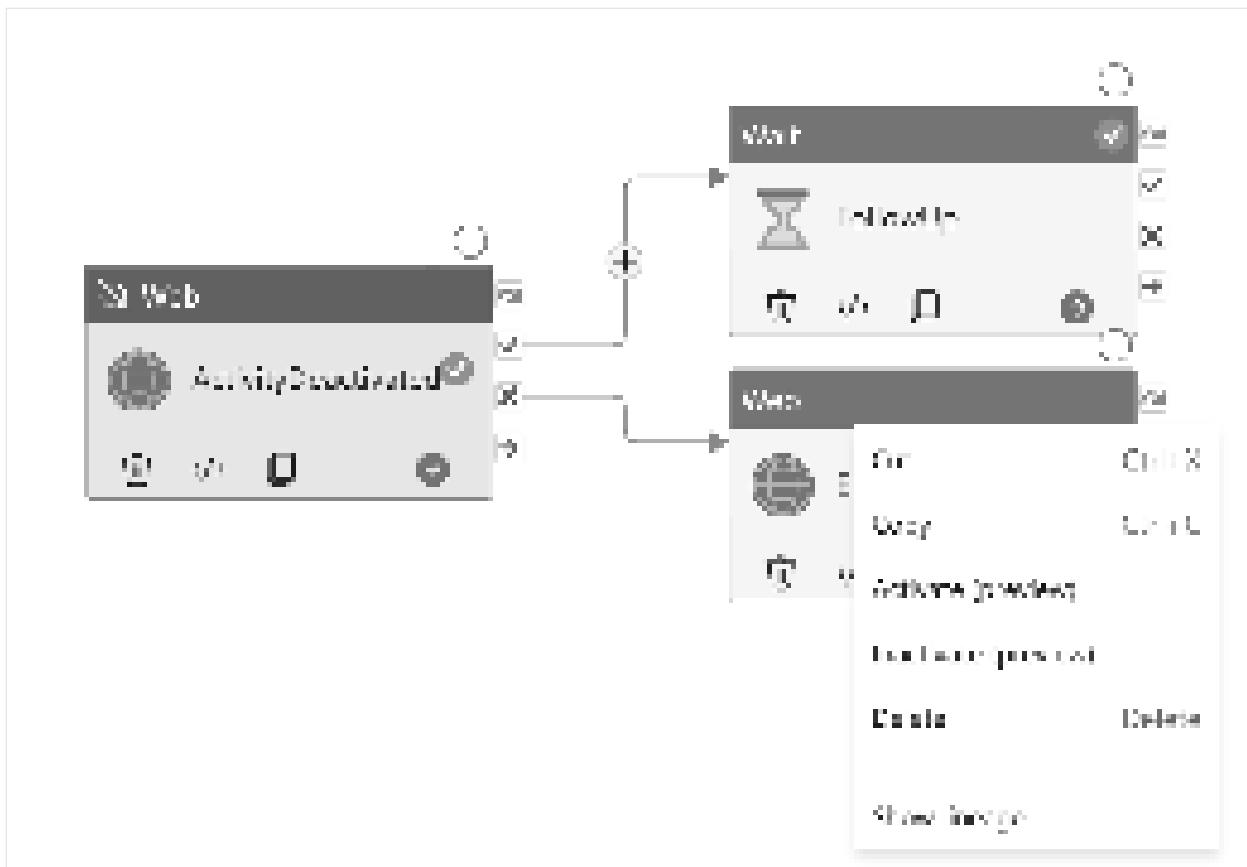
First, you may deactivate a single activity from its **General** tab.

1. Select the activity you want to deactivate
2. Under **General** tab, select *Inactive* for *Activity state*
3. Pick a state for *Mark activity as*. Choose from *Succeeded*, *Failed* or *Skipped*



Alternatively, you can deactivate multiple activities with right click.

1. Press down *Ctrl* key to multi-select. Using your mouse, left click on all activities you want to deactivate
2. Right click to bring up the drop down menu
3. Select *Deactivate* to deactivate them all
4. To fine tune the settings for *Mark activity as*, go to **General** tab of the activity, and make appropriate changes



In both cases, you do need to deploy the changes to deactivate the parts during pipeline run.

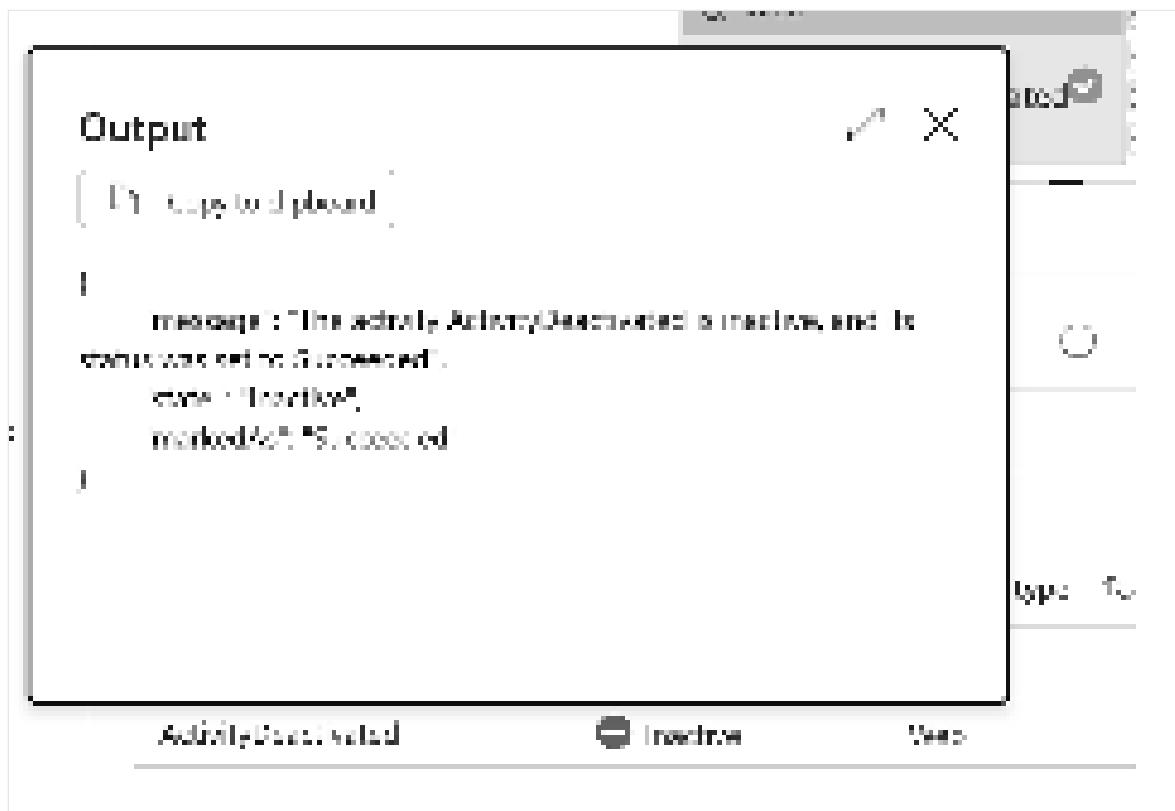
To reactivate the activities, choose *Active* for the *Activity State*, and they revert back to their previous behaviors, as expected.

Behaviors

An inactive activity behaves differently in a pipeline.

- On canvas, the inactive activity is grayed out, with *Inactive sign* placed next to the activity type
- On canvas, a status sign (Succeeded, Failed or Skipped) is placed on the box, to visualize the *Mark activity as* setting
- The activity is excluded from pipeline validation. Hence, you don't need to provide all required fields for an inactive activity.
- During debug run and pipeline run, the activity won't actually execute. Instead, it runs a place holder line item, with the reserved status **Inactive**
- The branching option is controlled by *Mark activity as* option. In other words:
 - If you mark the activity as *Succeeded*, the *UponSuccess* or *UponCompletion* branch runs

- If you mark the activity as *Failed*, the *UponFailure* or *UponCompletion* branch runs
- If you mark the activity as *Skipped*, the *UponSkip* branch runs



Best practices

Deactivation is a powerful tool for pipeline developer. It allows developers to "comment out" part of the code, without permanently deleting the activities. It shines in following scenarios:

- When developing a pipeline, developer can add place holder inactive activities before filling all the required fields. For instance, I need a Copy activity from SQL Server to Data warehouse, but I haven't set up all the connections yet. So I use an *inactive* copy activity as the place holder for iterative development process.
- After deployment, developer can comment out certain activities that are constantly causing troubles to avoid costly retries. For instance, my on-premises SQL server is having network connection issues, and I know my copy activities fail for certain. I may want to deactivate the copy activity, to avoid retry requests from flooding the brittle system.

Known limitations

An inactive activity never actually runs. This means the activity won't have an error field, or its typical output fields. Any references to missing fields may throw errors

downstream.

Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?



Yes



No

Provide product feedback | Ask the community

Use the Delete data activity to look up data from a data source

Article • 11/15/2023

The Fabric Delete data activity can delete data from any of the data sources supported by Microsoft Fabric.

You can use the Delete Activity in Data Factory to delete files or folders from any supported storage stores. Use this activity to clean up or archive files when they are no longer needed.

Prerequisites

To get started, you must complete the following prerequisites:

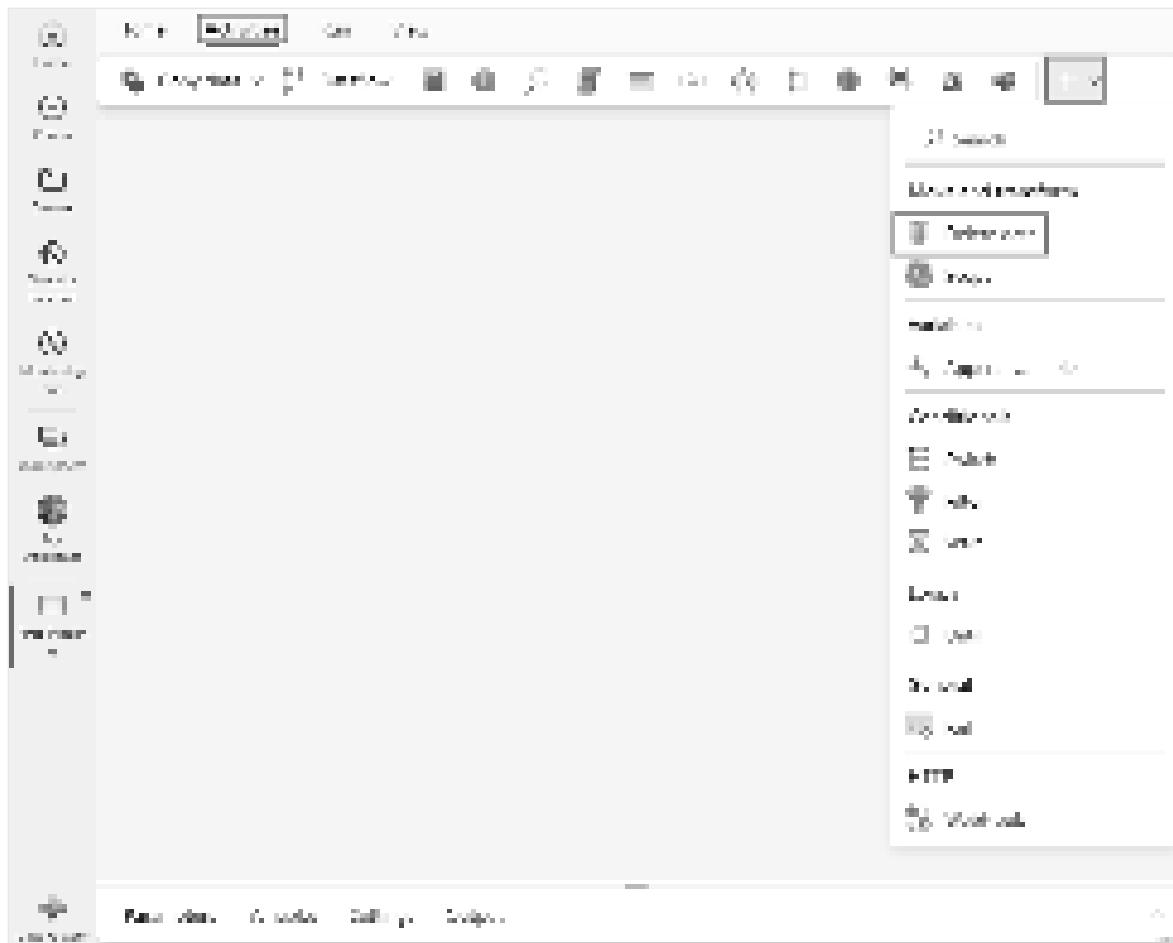
- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add a lookup activity to a pipeline with UI

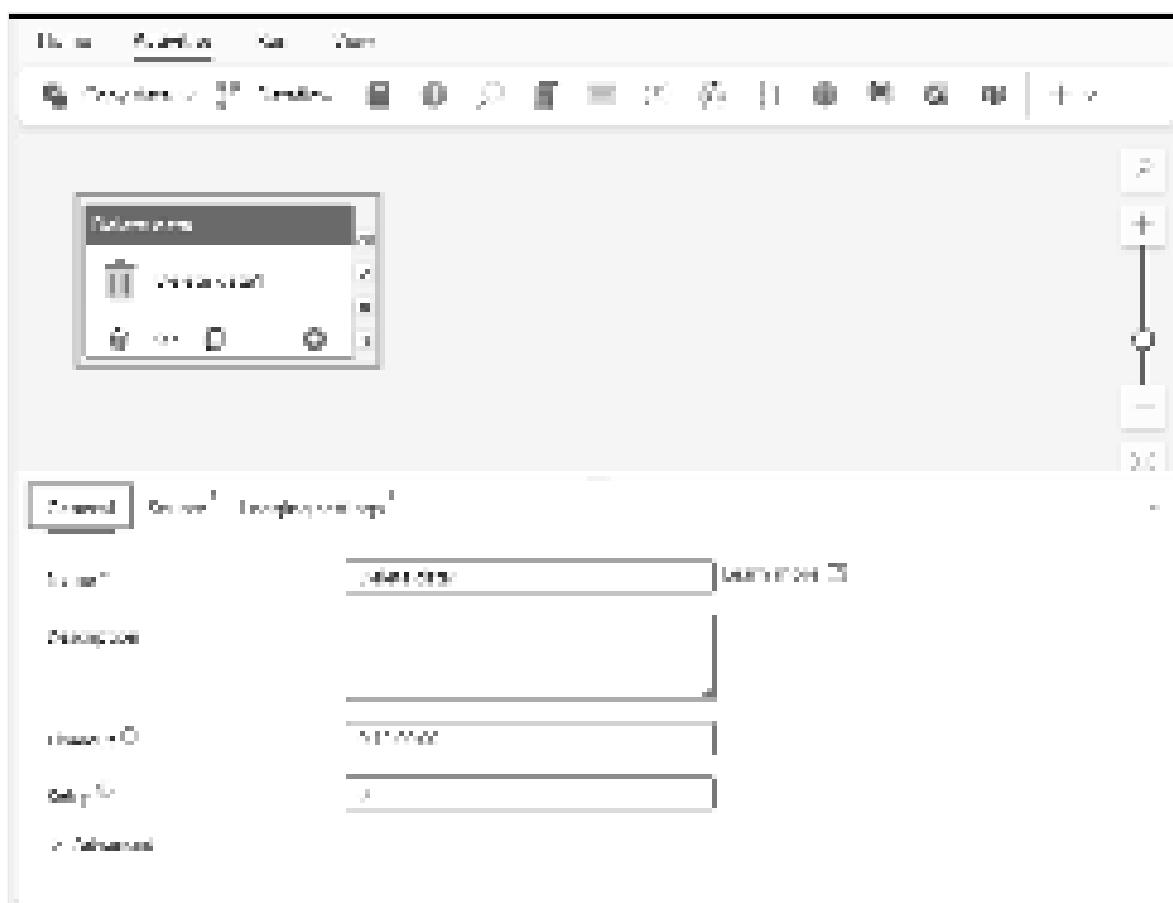
To use a Delete data activity in a pipeline, complete the following steps:

Creating the activity

1. Create a new pipeline in your workspace.
2. Search for Delete data in the pipeline **Activities** pane, and select it to add it to the pipeline canvas.



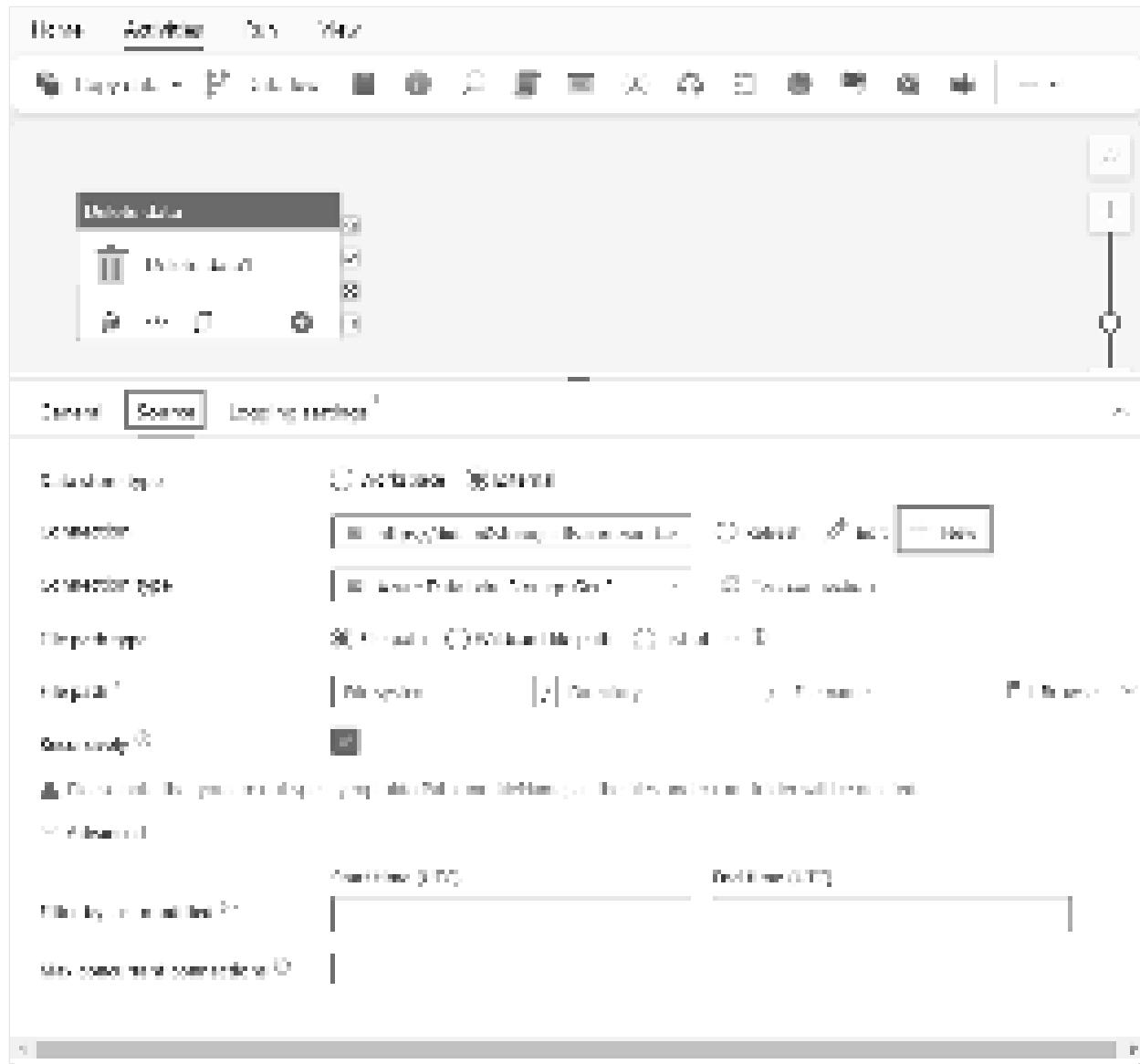
3. Select the new Delete data activity on the canvas if it isn't already selected.



Refer to the General settings guidance to configure the General settings tab.

Choose a data source

Select the **Source** tab, and select an existing connection from the **Connection** dropdown, or use the **+ New** button to create a new connection, and specify its configuration details.



The example in the previous image shows a blob storage connection, but each connection type has its own configuration details specific to the data source selected.

Supported data stores

Fabric supports the data stores listed in the Connector overview article. Any source that supports the Delete data activity can be used.

Examples of using the Delete activity

Delete specific folders or files

The store has the following folder structure:

```
Root/
  Folder_A_1/
    1.txt
    2.txt
    3.csv
  Folder_A_2/
    4.txt
    5.csv
  Folder_B_1/
    6.txt
    7.csv
  Folder_B_2/
    8.txt
```

Now you are using the Delete activity to delete folder or files by the combination of different property value from the dataset and the Delete activity:

FolderPath	fileName	recursive	Output
Root/ Folder_A_2	NULL	False	Root/ Folder_A_1/ 1.txt 2.txt 3.csv Folder_A_2/ 4.txt 5.csv Folder_B_1/ 6.txt 7.csv Folder_B_2/ 8.txt
Root/ Folder_A_2	NULL	True	Root/ Folder_A_1/ 1.txt 2.txt 3.csv Folder_A_2/ 4.txt 5.csv Folder_B_1/ 8.txt

FolderPath	fileName	recursive	Output
			6.txt 7.csv Folder_B_2/ 8.txt
Root/ Folder_A_2	*.txt	False	Root/ Folder_A_1/ 1.txt 2.txt 3.csv Folder_A_2/ 4.txt 5.csv Folder_B_1/ 6.txt 7.csv Folder_B_2/ 8.txt
Root/ Folder_A_2	*.txt	True	Root/ Folder_A_1/ 1.txt 2.txt 3.csv Folder_A_2/ 4.txt 5.csv Folder_B_1/ 6.txt 7.csv Folder_B_2/ 8.txt

Save and run or schedule the pipeline

1. Switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline.
2. Select **Run** to run it directly, or **Schedule** to schedule it.
3. You can also view the run history here or configure other settings.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Use the Fail activity to cause pipeline execution to fail with a customized error message and error code

Article • 12/11/2023

You might occasionally want to throw an error in a pipeline intentionally. A Lookup activity might return no matching data, or a Script activity might finish with an internal error. Whatever the reason might be, now you can use a Fail activity in a pipeline and customize both its error message and error code.

Prerequisites

To get started, you must complete the following prerequisites:

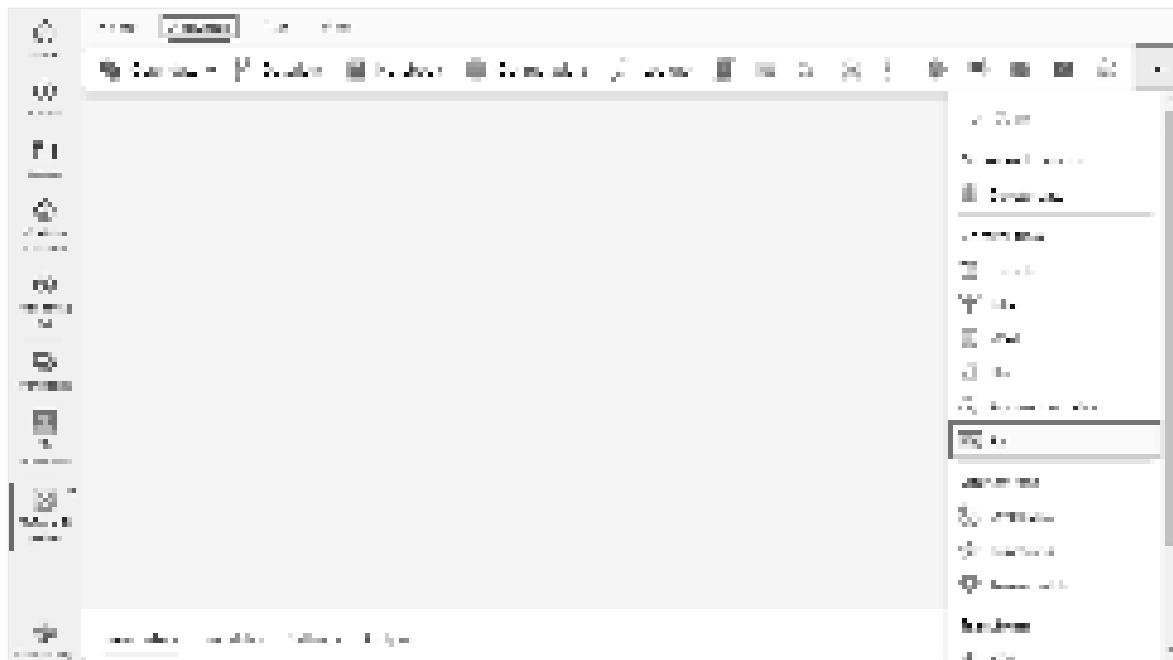
- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add a Fail activity to a pipeline with UI

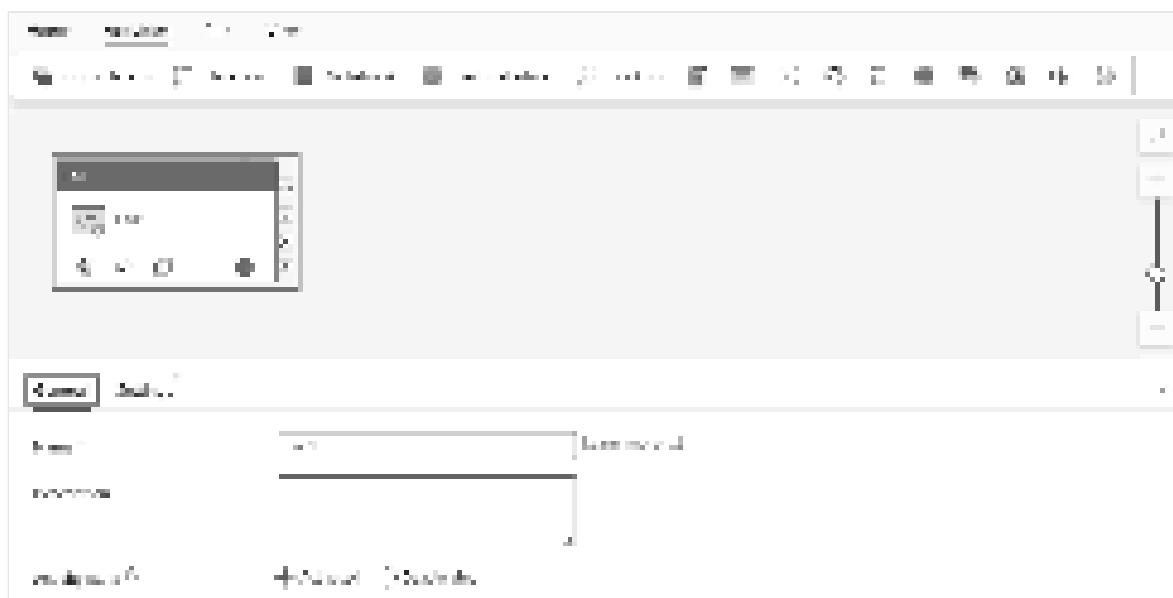
To use a Fail activity in a pipeline, complete the following steps:

Creating the activity

1. Create a new pipeline in your workspace.
2. Search for the Fail activity in the pipeline **Activities** pane, and select it to add it to the pipeline canvas. It might be necessary to expand the activities list on the far right side of the pane, or the Outlook icon can be compressed without labeling text beneath it, as shown in this image, depending on the window width of your browser.



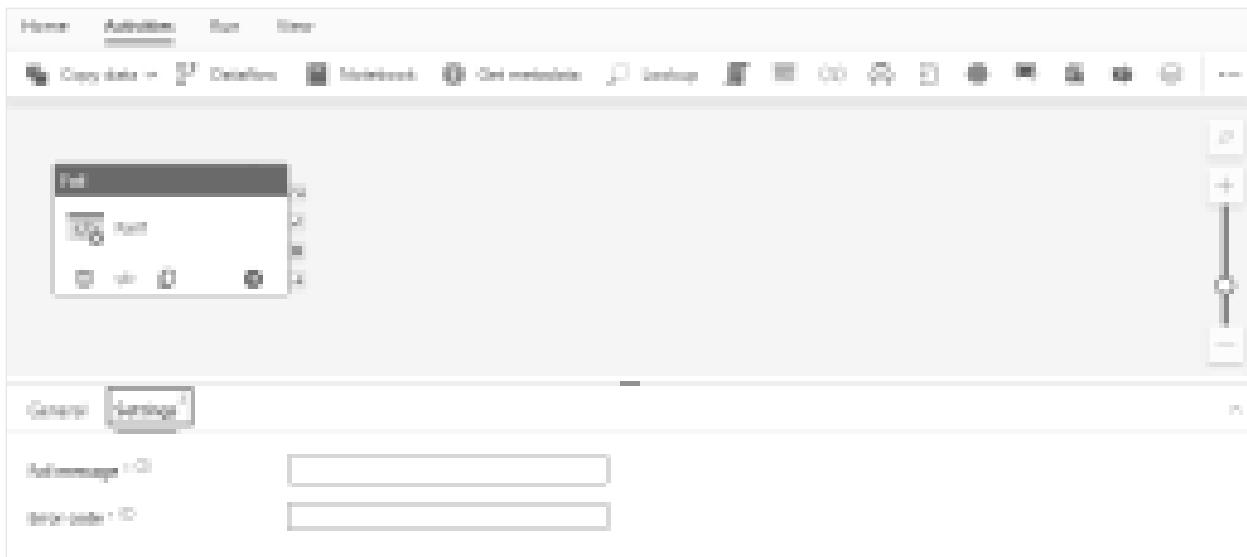
3. Select the new Fail activity on the canvas if it isn't already selected.



Refer to the **General** settings guidance to configure the **General** settings tab.

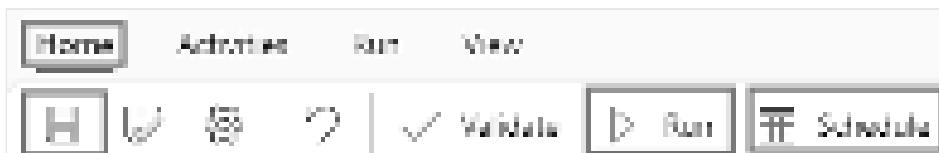
Fail activity settings

Select the **Settings** tab and provide a custom error message and error code you would like the pipeline to report when the activity is invoked.



Save and run or schedule the pipeline

The Fail activity is typically used with other activities. After you configure any other activities required for your pipeline, switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Related content

How to monitor pipeline runs

Feedback

Was this page helpful?

Yes

No

Provide product feedback | Ask the community

Transform data with a ForEach activity

Article • 11/15/2023

The ForEach Activity defines a repeating control flow in a Microsoft Fabric pipeline. This activity is used to iterate over a collection and executes specified activities in a loop. The loop implementation of this activity is similar to a ForEach looping structure in programming languages.

Add a ForEach activity to a pipeline

This section describes how to use a ForEach activity in a pipeline.

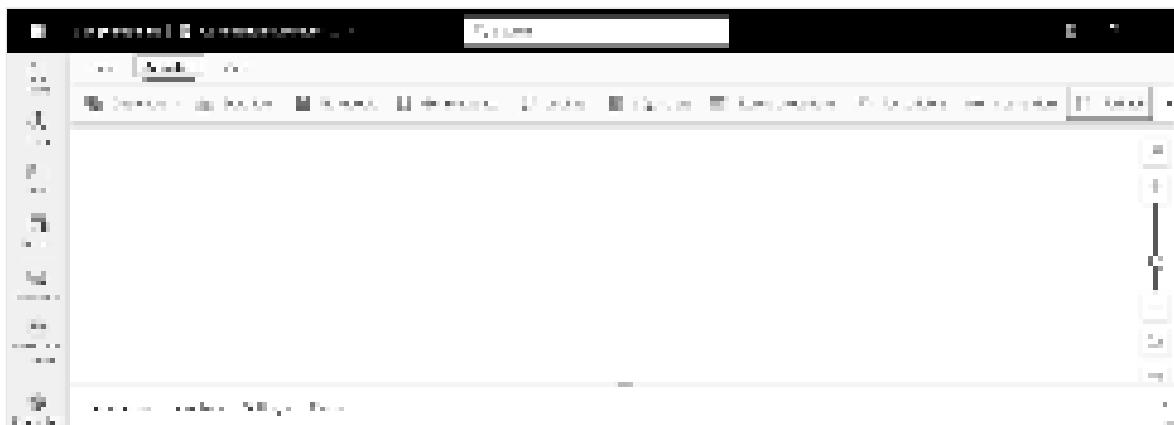
Prerequisites

To get started, you must complete the following prerequisites:

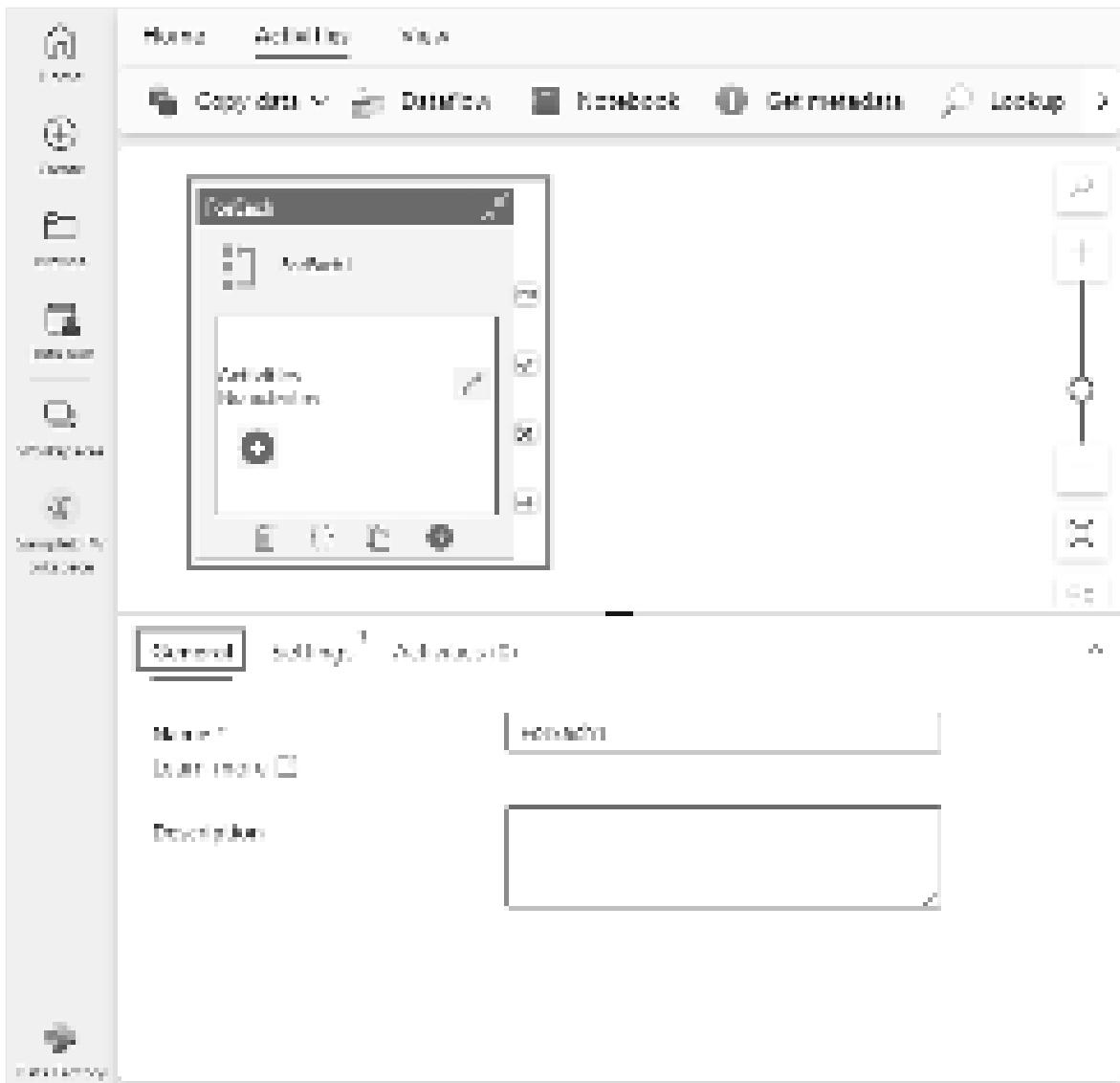
- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Creating the activity

1. Create a new pipeline in your workspace.
2. Search for **ForEach** in the pipeline **Activities** pane, and select it to add it to the pipeline canvas. If you can't see it initially, use the arrow on the right side of the activities toolbar to scroll to the right to find it.



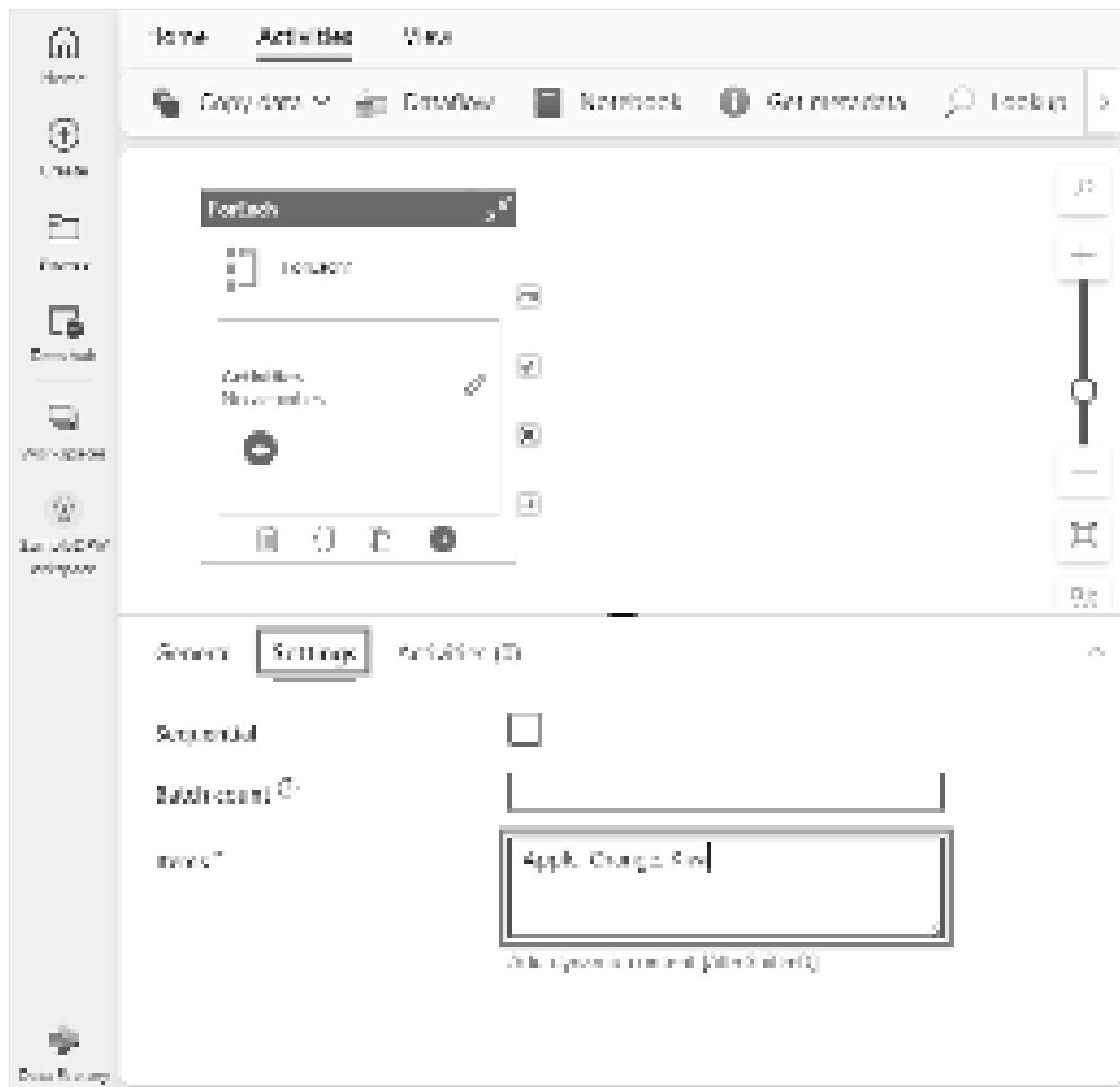
3. Select the new ForEach activity on the canvas if it isn't already selected.



4. In the **General** tab at the bottom of the canvas, enter a name for the activity in the Name property.
 5. (Optional) You can also enter a description.

ForEach settings

Select the **Settings** tab, where you can specify whether processing of the items in the batch should be **Sequential** (or otherwise in parallel). You can also specify a maximum number of items to process at the same time with **Batch count**. Finally, you must specify a list of comma delimited **Items**, which can be parameterized or include dynamic content. Add a few items to the **Items** list as shown in the example.

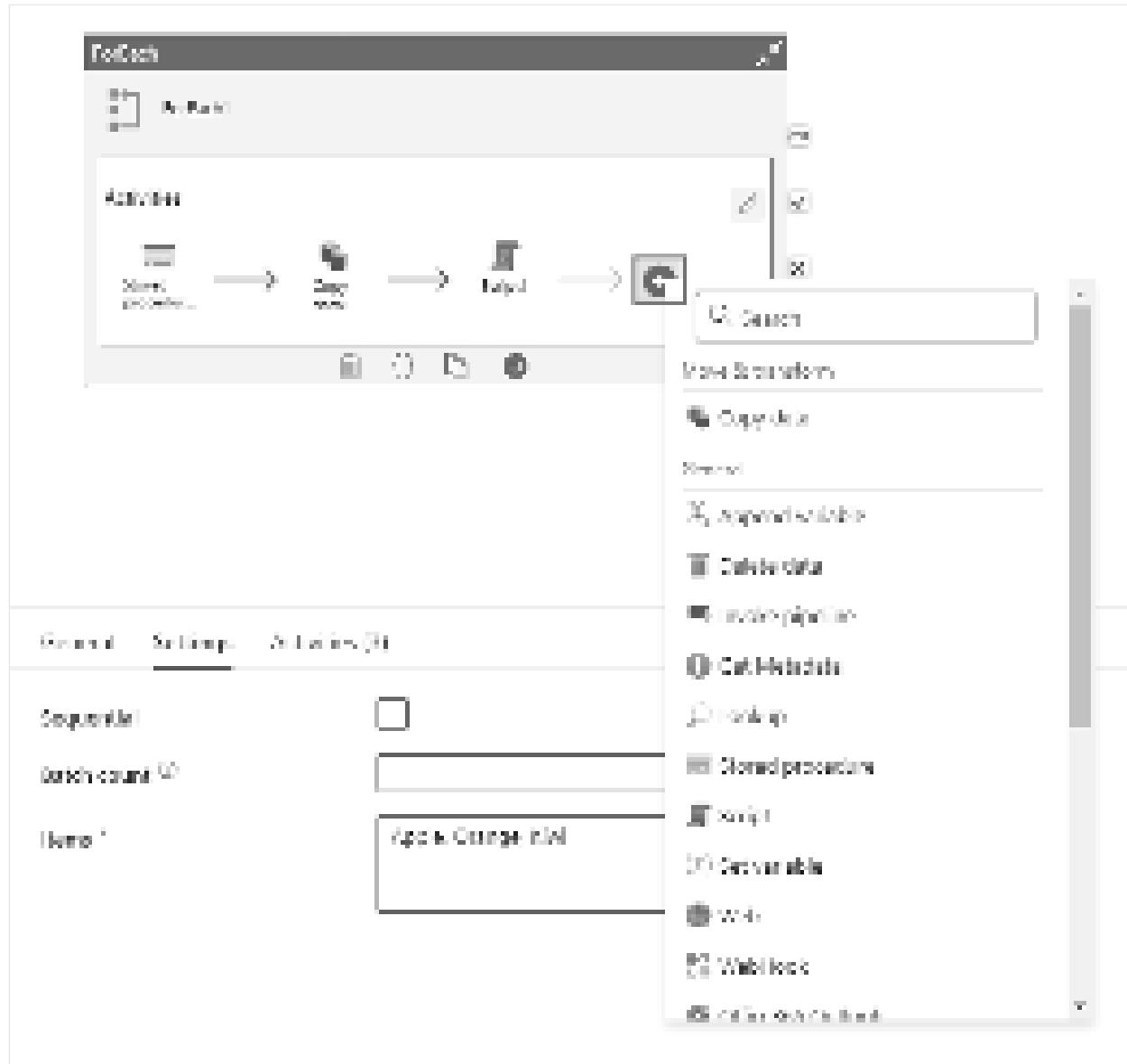


ForEach activities

You'll also define an activity or activities to be performed on each of the items in the list, in the **ForEach Activities** pane.



Select the + button to add a new activity to the pane. You'll see a list of activities to choose. You can add multiple activities to the **ForEach** activity, and each is run on each of the items in the **Items** list. Whether the **Sequential** option is selected in the **ForEach** settings or not, each of the child activities in the **ForEach** activities pane are processed sequentially to one another for each item. However, if **Sequential** isn't selected, multiple items are processed in parallel, each of them running sequentially through the list of child activities specified.



Referencing an item within an activity

Select one of the child activities in the **ForEach Activities** pane, and switch to its **Settings** tab. In this example, a **Stored Procedure** activity was selected. Populate the settings for the activity as you normally would select a connection and stored procedure. You can use the **@item()** iterator to refer to the current item being processed anywhere within an activity that supports dynamic content. Here the **@item()** was used as the value for the **FruitName** parameter that is passed to a stored procedure.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

Yes

No

Provide product feedback [↗](#) | Ask the community [↗](#)

Use the Functions activity to run Azure Functions

Article • 11/15/2023

The Functions activity in Data Factory for Microsoft Fabric allows you to run Azure Functions.

Prerequisites

To get started, you must complete the following prerequisites:

- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add a Functions activity to a pipeline with UI

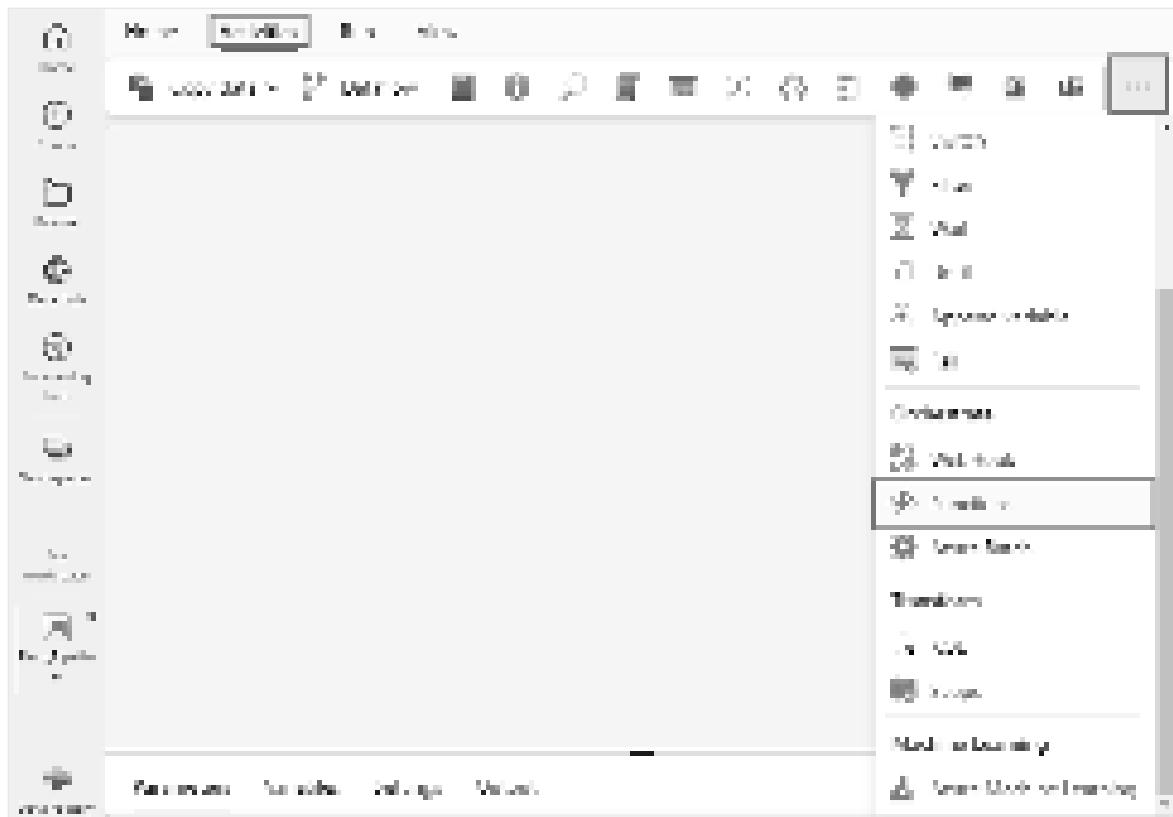
To use a Functions activity in a pipeline, complete the following steps:

Create the activity

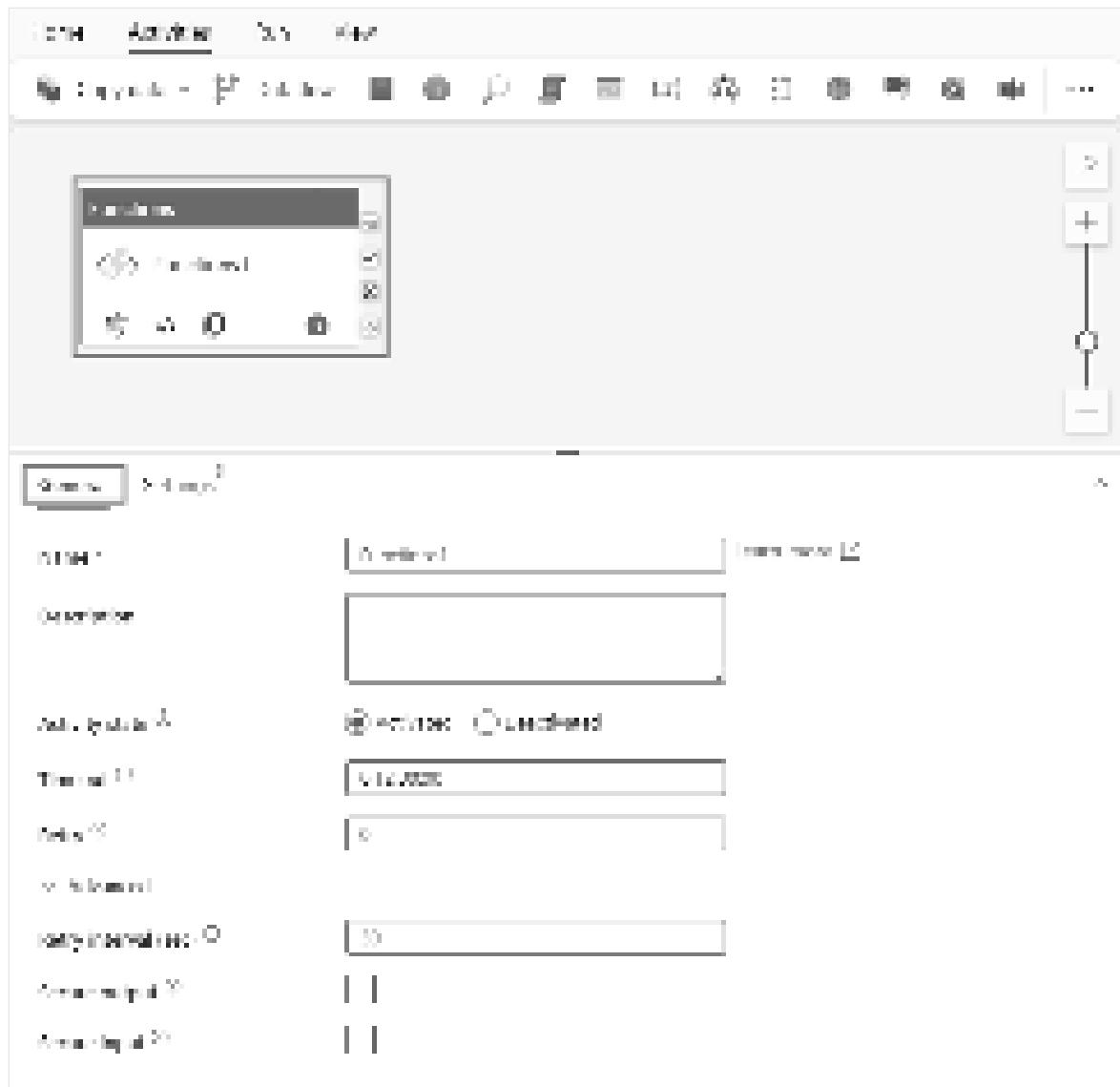
1. Create a new pipeline in your workspace.
2. Search for Functions in the pipeline **Activities** pane, and select it to add it to the pipeline canvas.

(!) Note

You may need to expand the menu and scroll down to see the Functions activity as highlighted in the following screenshot.



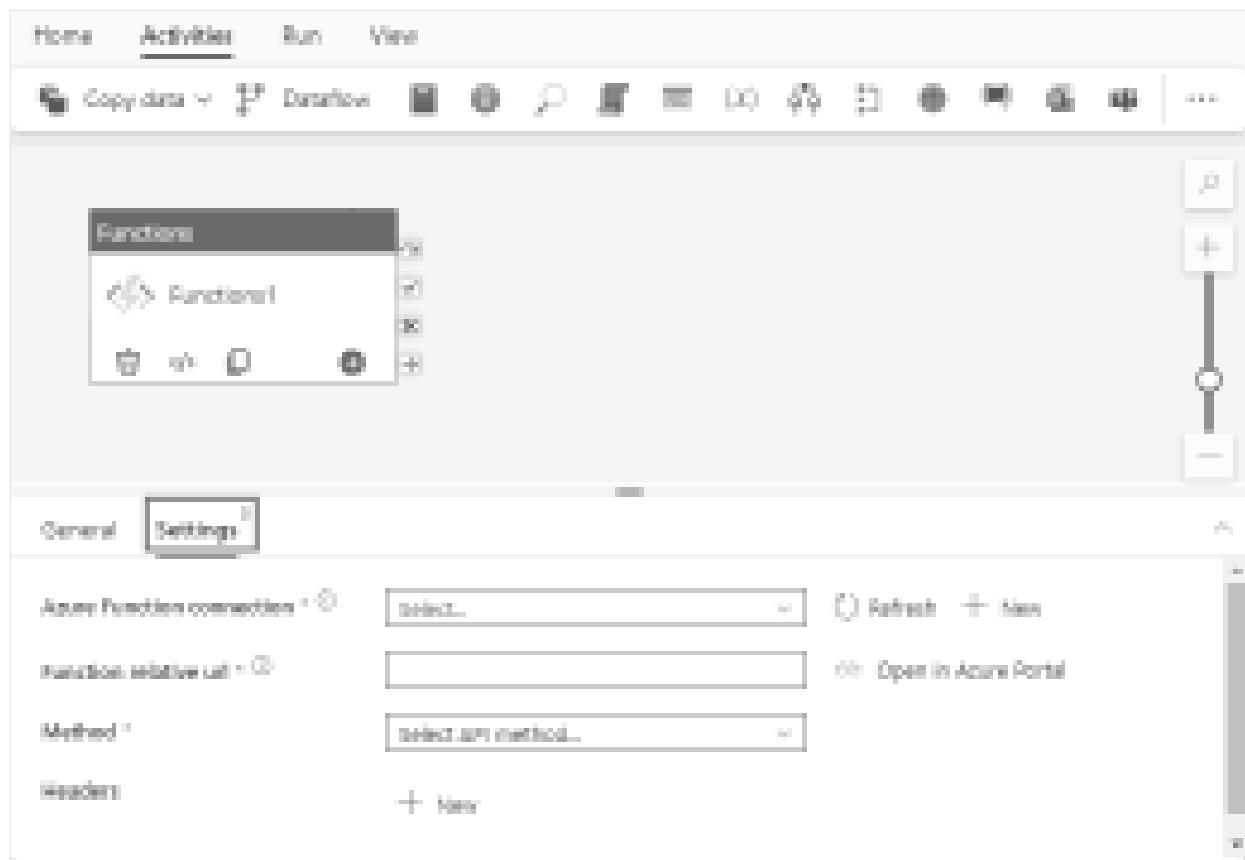
3. Select the new Functions activity on the pipeline editor canvas if it isn't already selected.



Refer to the **General** settings guidance to configure the **General** settings tab.

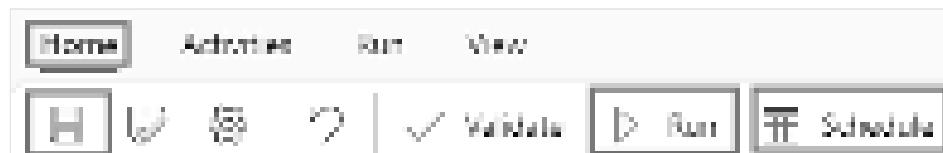
Functions activity settings

Select the **Settings** tab, then you can choose an existing or create a new **Azure Function connection**, provide a **Function relative URL** that points to the relative path to the Azure App function within the Azure Function connection, and an **HTTP Method** to be submitted to the URL. You can also specify as many additional **Headers** as required for the function you are executing.



Save and run or schedule the pipeline

After you configure any other activities required for your pipeline, switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

Yes

No

Provide product feedback ↗ | Ask the community ↗

Use the Get Metadata activity to look up data from a data source

Article • 11/15/2023

You can use the Get Metadata activity to retrieve the metadata of any data in a Fabric pipeline. You can use the output from the Get Metadata activity in conditional expressions to perform validation, or consume the metadata in subsequent activities.

Prerequisites

To get started, you must complete the following prerequisites:

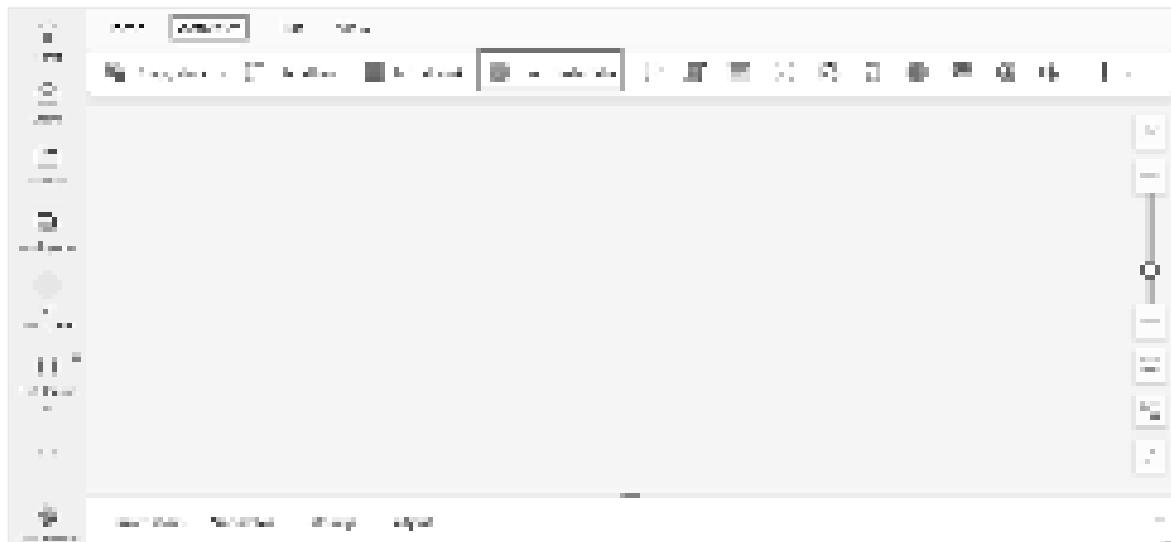
- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add a lookup activity to a pipeline with UI

To use a Lookup activity in a pipeline, complete the following steps:

Creating the activity

1. Create a new pipeline in your workspace.
2. Search for Get Metadata in the pipeline **Activities** pane, and select it to add it to the pipeline canvas.

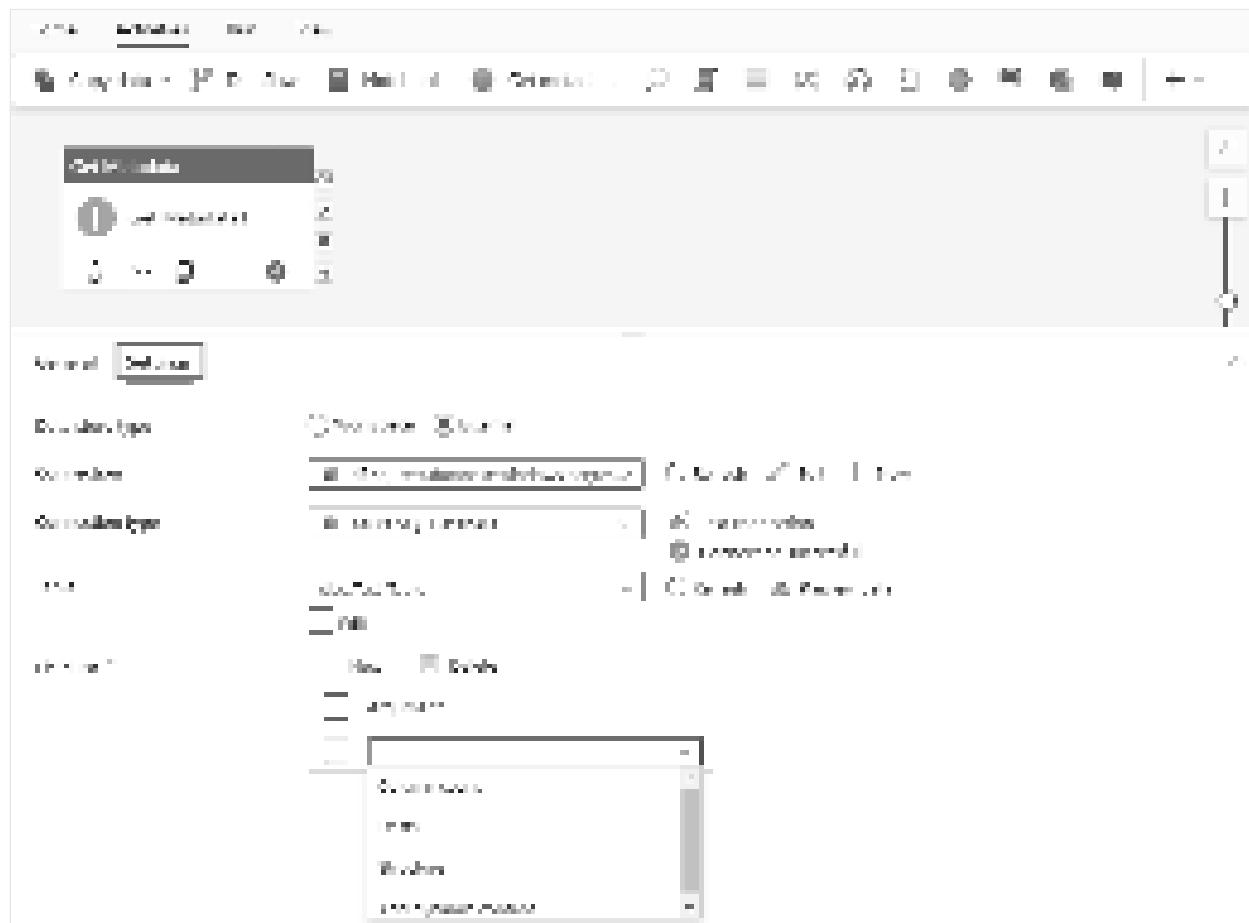


3. Select the new Get Metadata activity on the canvas if it isn't already selected.

Refer to the General settings guidance to configure the General settings tab.

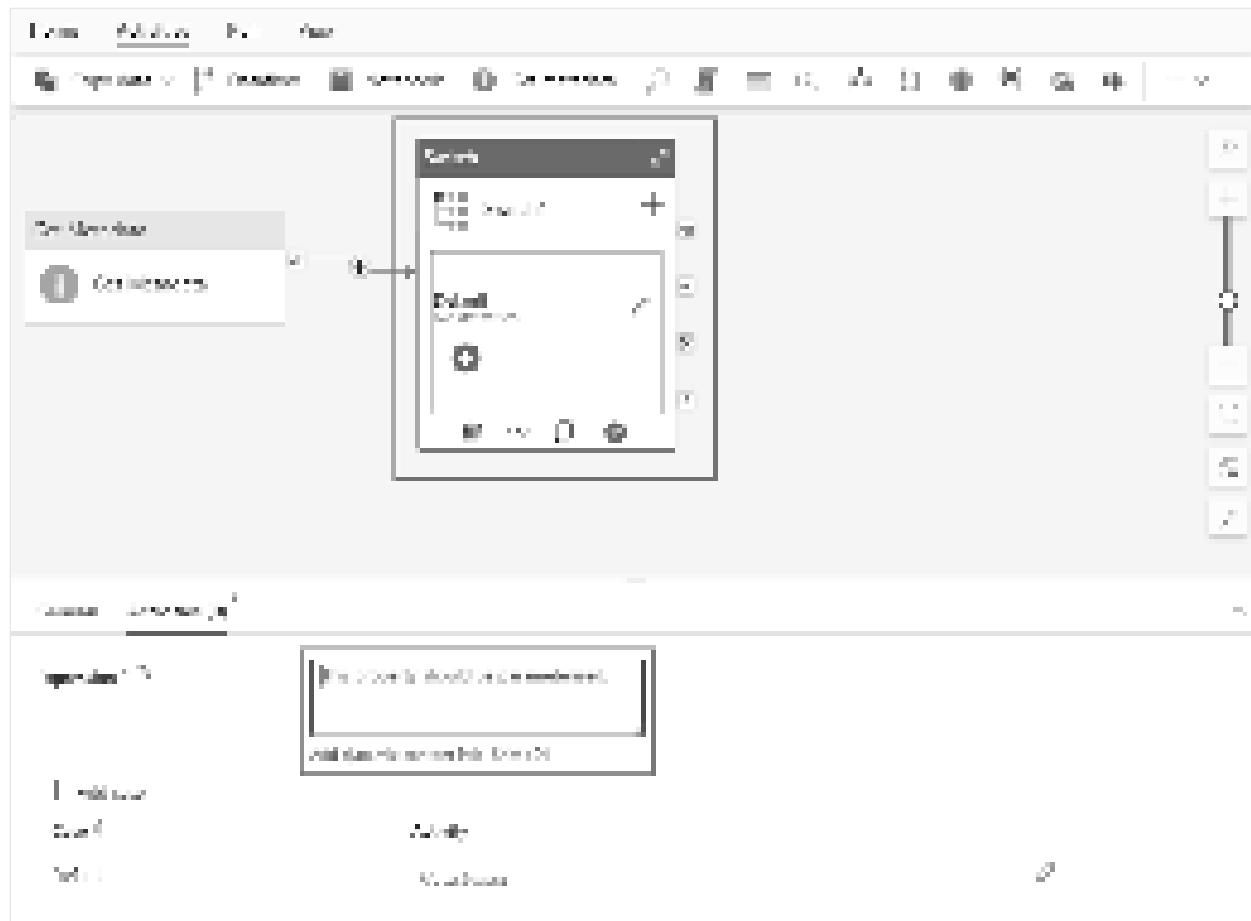
Get Metadata settings

Select the Settings tab, and then select an existing connection from the Connection dropdown, or use the + New button to create a new connection, and specify its configuration details. Then select a table, to choose from the various metadata fields available for the table, including column count, exists, structure, and dynamic content.

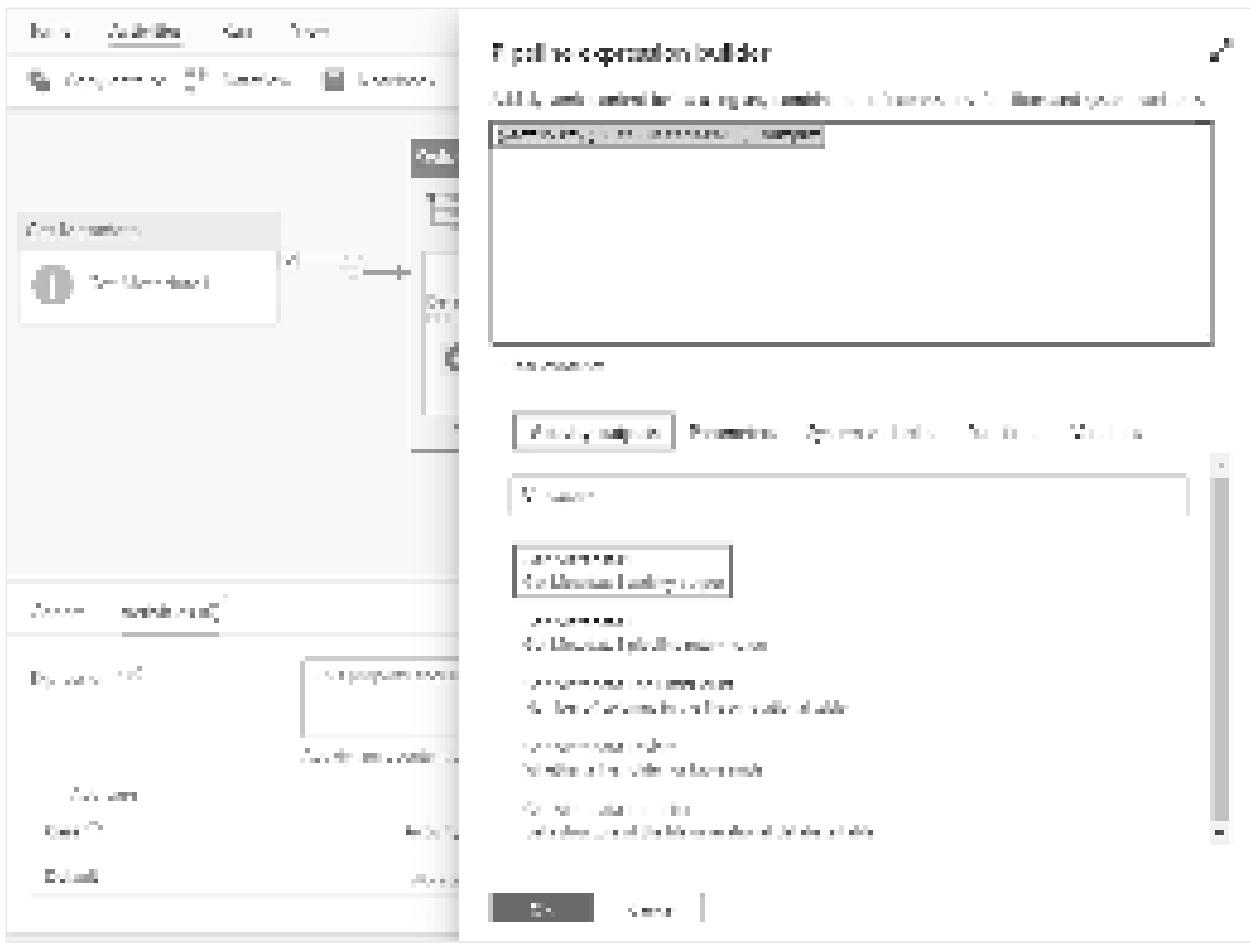


Using the Get Metadata activity

You can use the output of the Get Metadata activity in any other activities where dynamic content is supported. In this example, it's used as the **Expression** for a Switch activity.

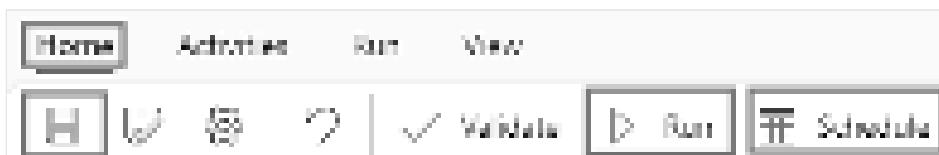


Select the **Add dynamic content** link that appears under the **Expression** text box for the Switch activity. Then you can browse the activity outputs in the expression builder and select them to add them to the expression.



Save and run or schedule the pipeline

After adding any other activities necessary to your pipeline, you can save and run it. Switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

Yes

1 No

[Provide product feedback](#) | [Ask the community](#)

Use the If Condition activity to execute activities based on an expression

Article • 11/15/2023

The If Condition activity in Data Factory for Microsoft Fabric provides the same functionality that an if statement provides in programming languages. It executes a set of activities when the condition evaluates to true and another set of activities when the condition evaluates to false.

Prerequisites

To get started, you must complete the following prerequisites:

- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add an If Condition activity to a pipeline with UI

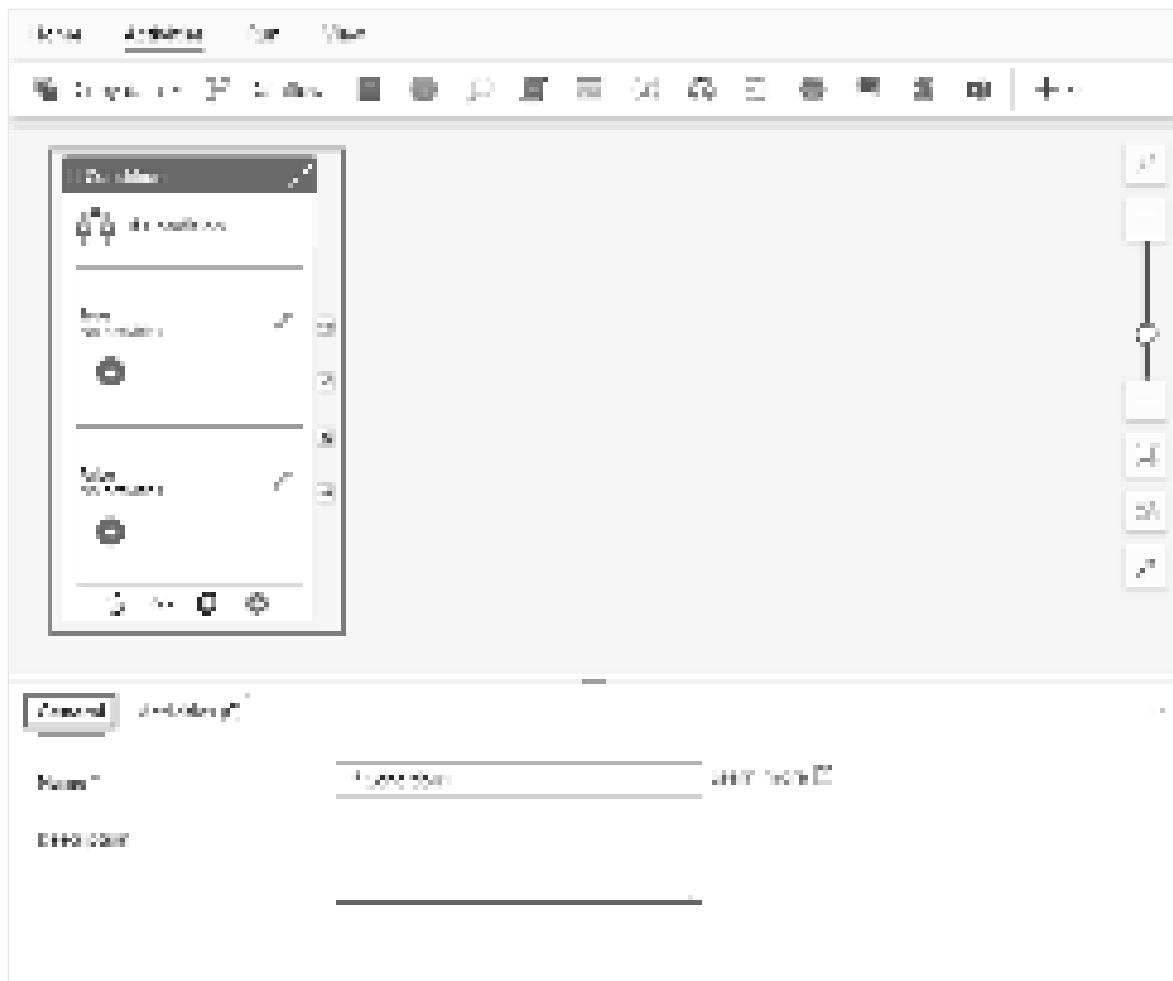
To use an If Condition activity in a pipeline, complete the following steps:

Creating the activity

1. Create a new pipeline in your workspace.
2. Search for If Condition in the pipeline **Activities** pane, and select it to add it to the pipeline canvas.



3. Select the new If Condition activity on the canvas if it isn't already selected.

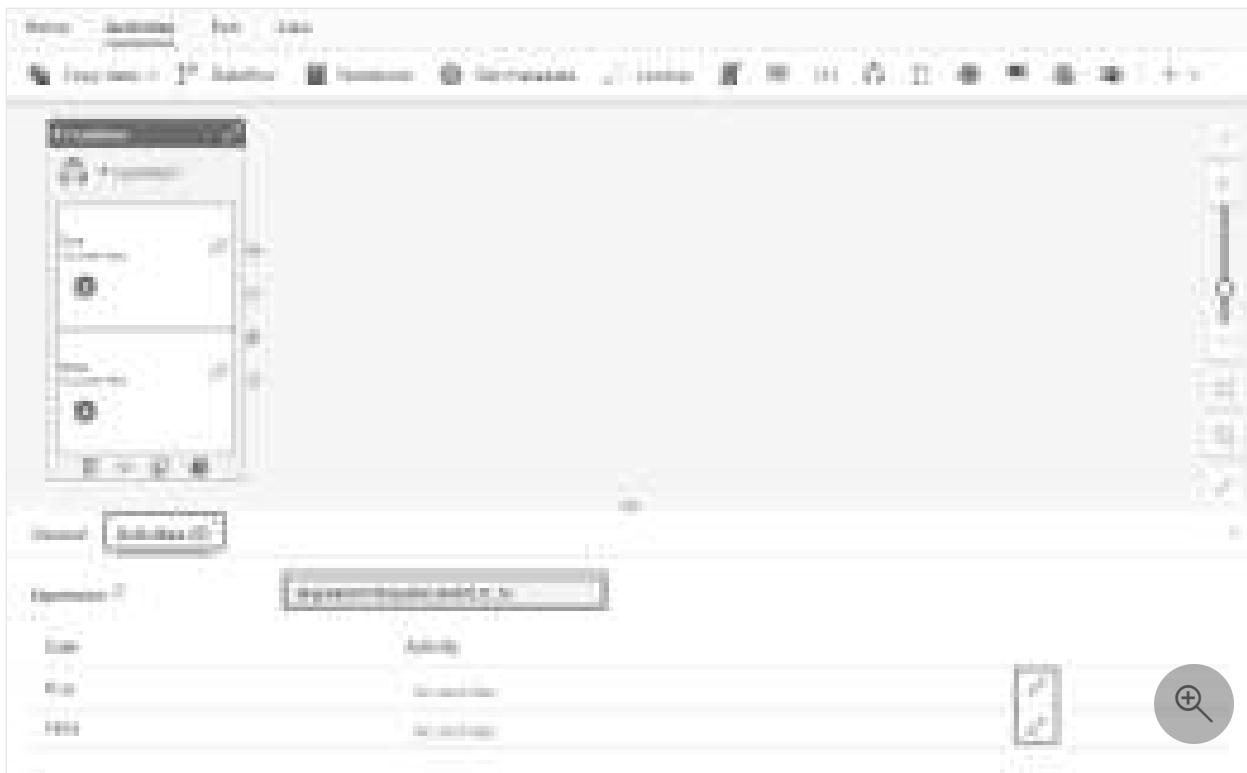


Refer to the General settings guidance to configure the **General** settings tab.

If Condition settings

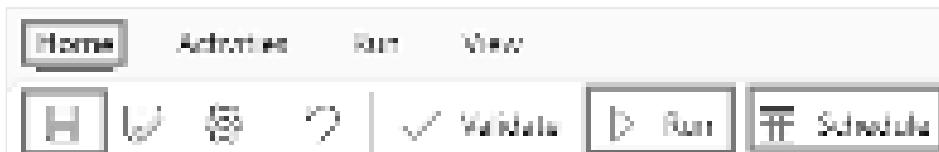
Select the **Activities** tab and provide a dynamic boolean **Expression** for the If activity. In this simple example, we randomly generate a number between 0 and 1, and return True if the number is greater than or equal to .5, or otherwise False. You can use any of the available functions in the Data Factory expression language or any parameters specified in the pipeline.

After providing the expression for the If Condition, selecting the pencil icon beside each case (True/False) allows you to add as many activities as necessary to be conditionally executed whenever the expression evaluates.



Save and run or schedule the pipeline

Switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

Yes

No

Provide product feedback | Ask the community

Use the Invoke pipeline activity to run another pipeline

Article • 11/15/2023

The Fabric Invoke pipeline activity can execute another Microsoft Fabric pipeline. You can use it to orchestrate the execution of one or multiple pipelines from within a single pipeline.

Prerequisites

To get started, you must complete the following prerequisites:

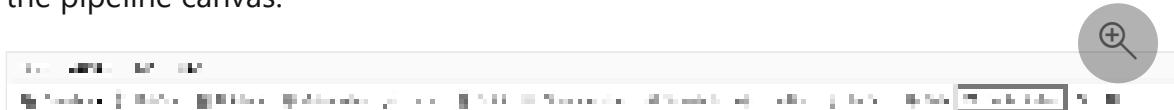
- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add an Invoke pipeline activity to a pipeline with UI

To use an Invoke pipeline activity in a pipeline, complete the following steps:

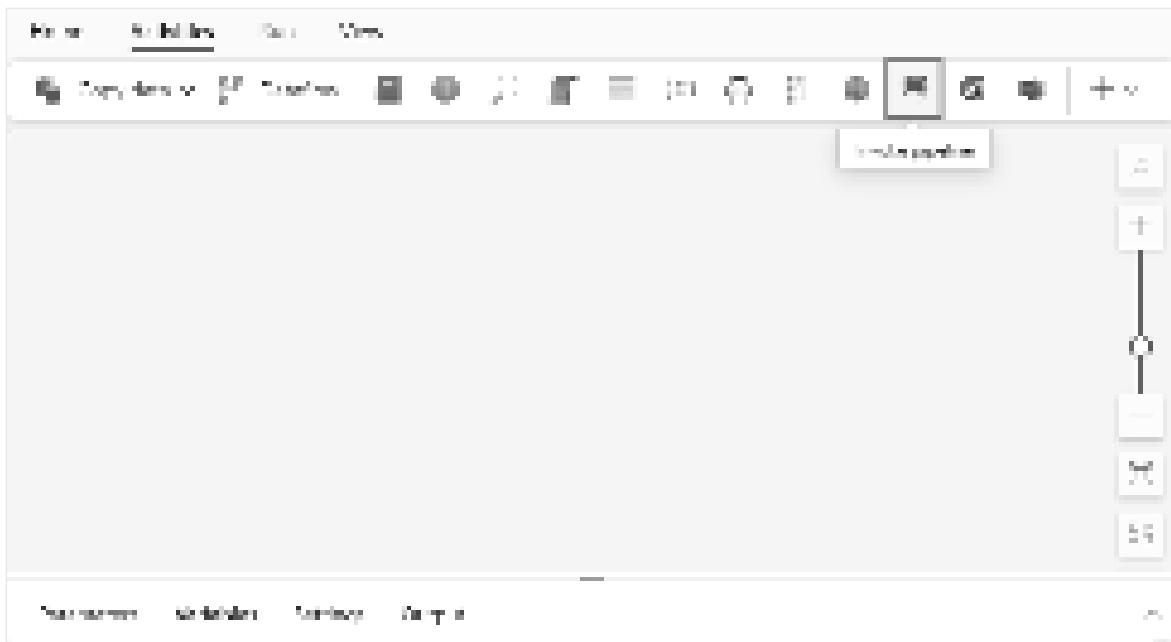
Create the activity

1. Create a new pipeline in your workspace.
2. Search for **Invoke pipeline** in the pipeline **Activities** pane, and select it to add it to the pipeline canvas.

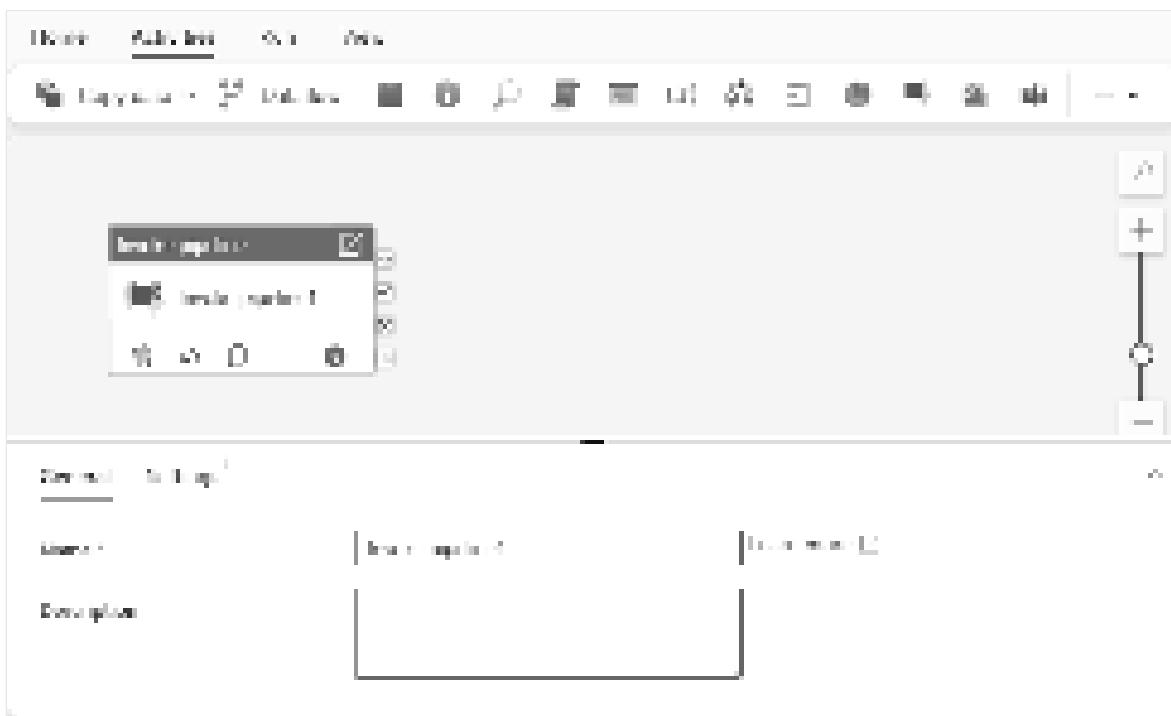


ⓘ Note

Information the user should notice even if skimmingIf your window size is not wide enough, the icon may appear without the text **Invoke pipeline** on the toolbar, although hovering over it will display its name.



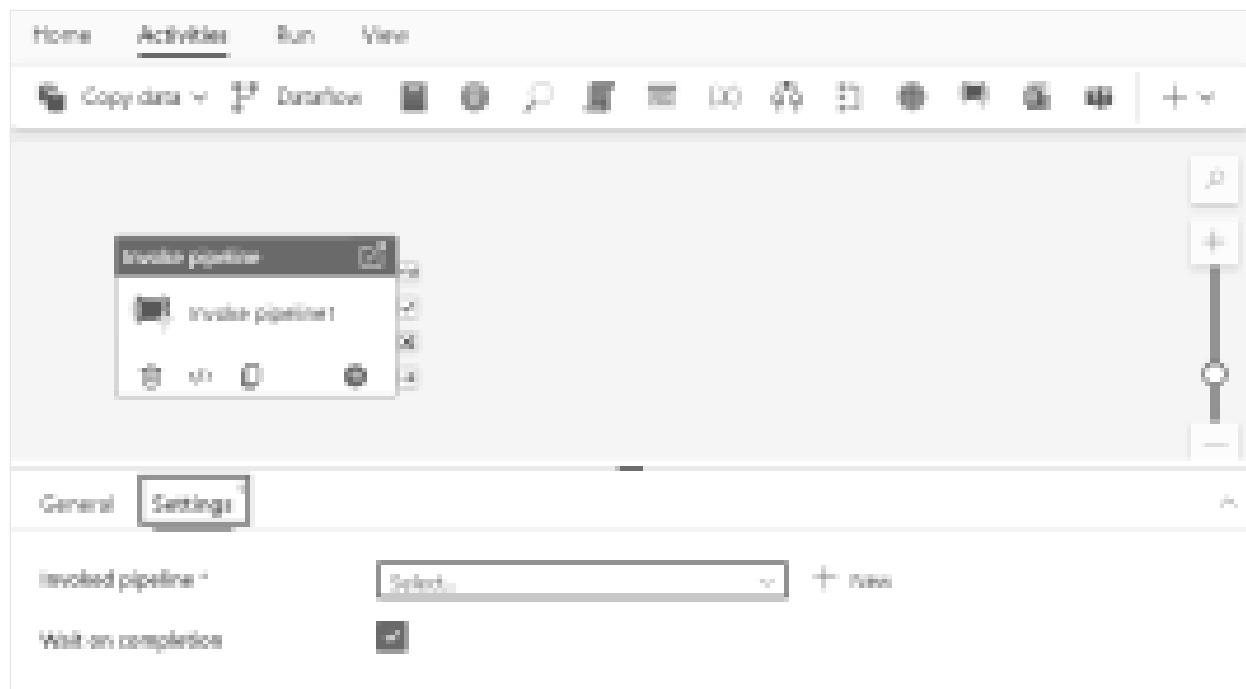
3. Select the new Invoke pipeline activity on the canvas if it isn't already selected.



Refer to the **General** settings guidance to configure the **General** settings tab.

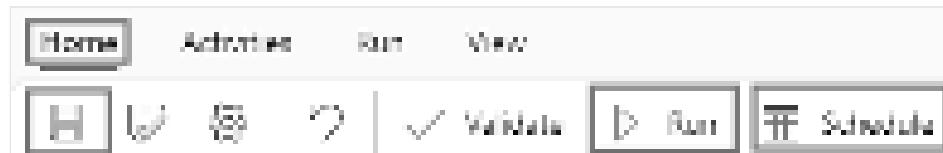
Invoke pipeline settings

Select the **Settings** tab, and choose an existing pipeline from the **Invoked pipeline** dropdown, or use the **+ New** button to create a new pipeline directly. You can choose to wait on completion, or continue directly, in which case the invoked pipeline executes in parallel with activities following it within the parent pipeline's execution flow.



Save and run or schedule the pipeline

Switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

Yes

No

Provide product feedback | Ask the community

Use the KQL activity to run a query

Article • 11/15/2023

The KQL activity in Data Factory for Microsoft Fabric allows you to run a query in Kusto Query Language (KQL) against an Azure Data Explorer instance.

Prerequisites

To get started, you must complete the following prerequisites:

- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add a KQL activity to a pipeline with UI

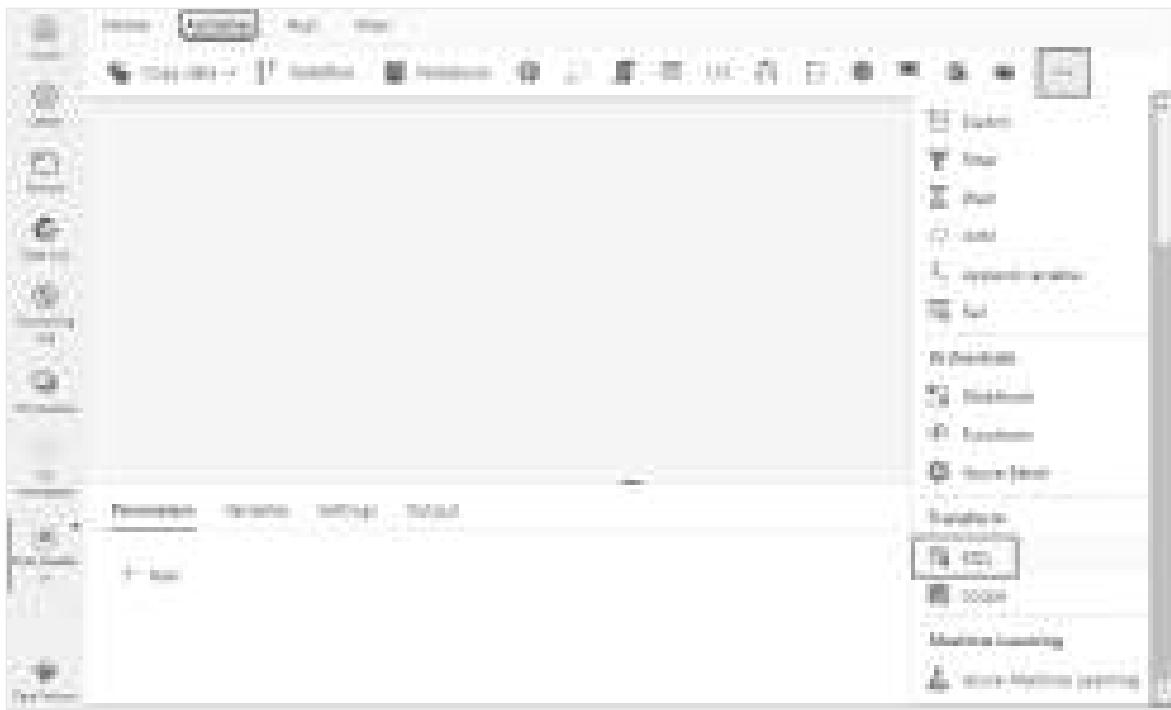
To use a KQL activity in a pipeline, complete the following steps:

Creating the activity

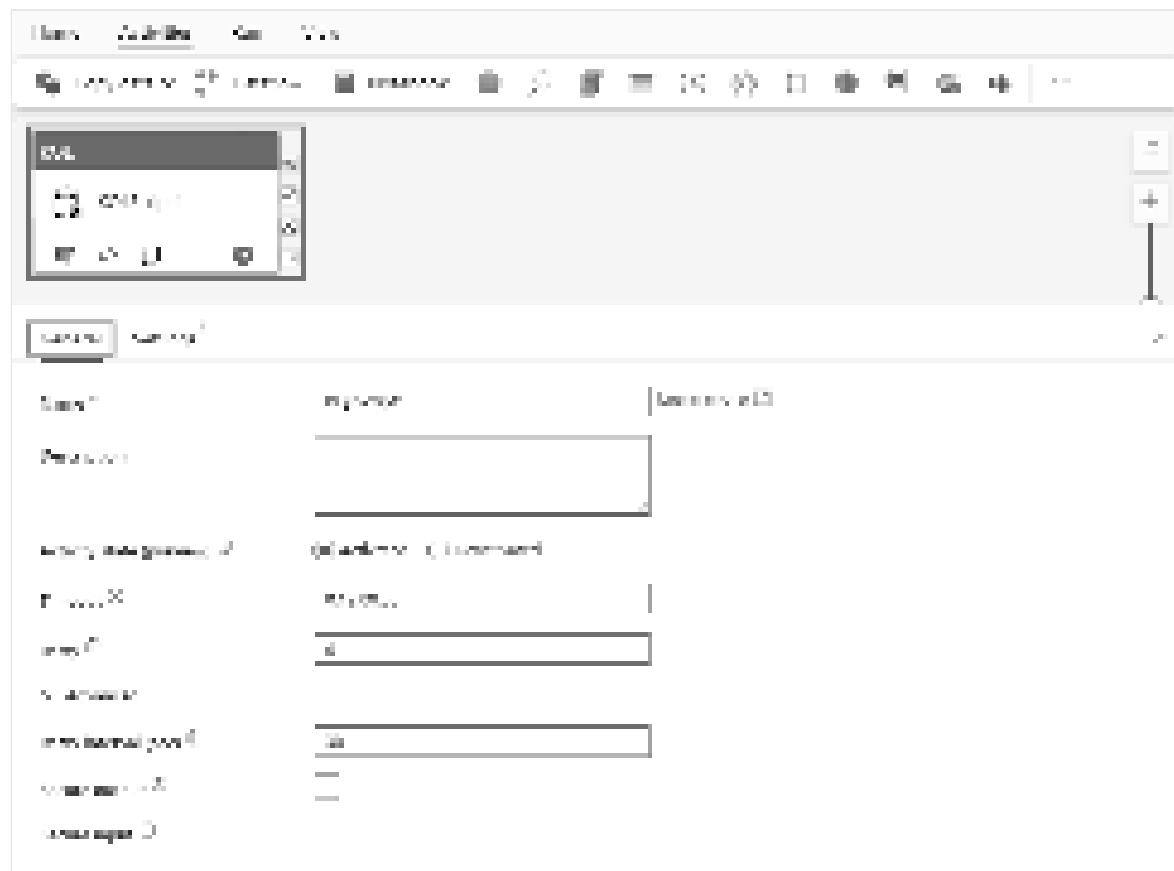
1. Create a new pipeline in your workspace.
2. Search for KQL in the pipeline **Activities** pane, and select it to add it to the pipeline canvas.

 **Note**

You may need to expand the menu and scroll down to see the KQL activity as highlighted in the screenshot below.



3. Select the new KQL activity on the pipeline editor canvas if it isn't already selected.



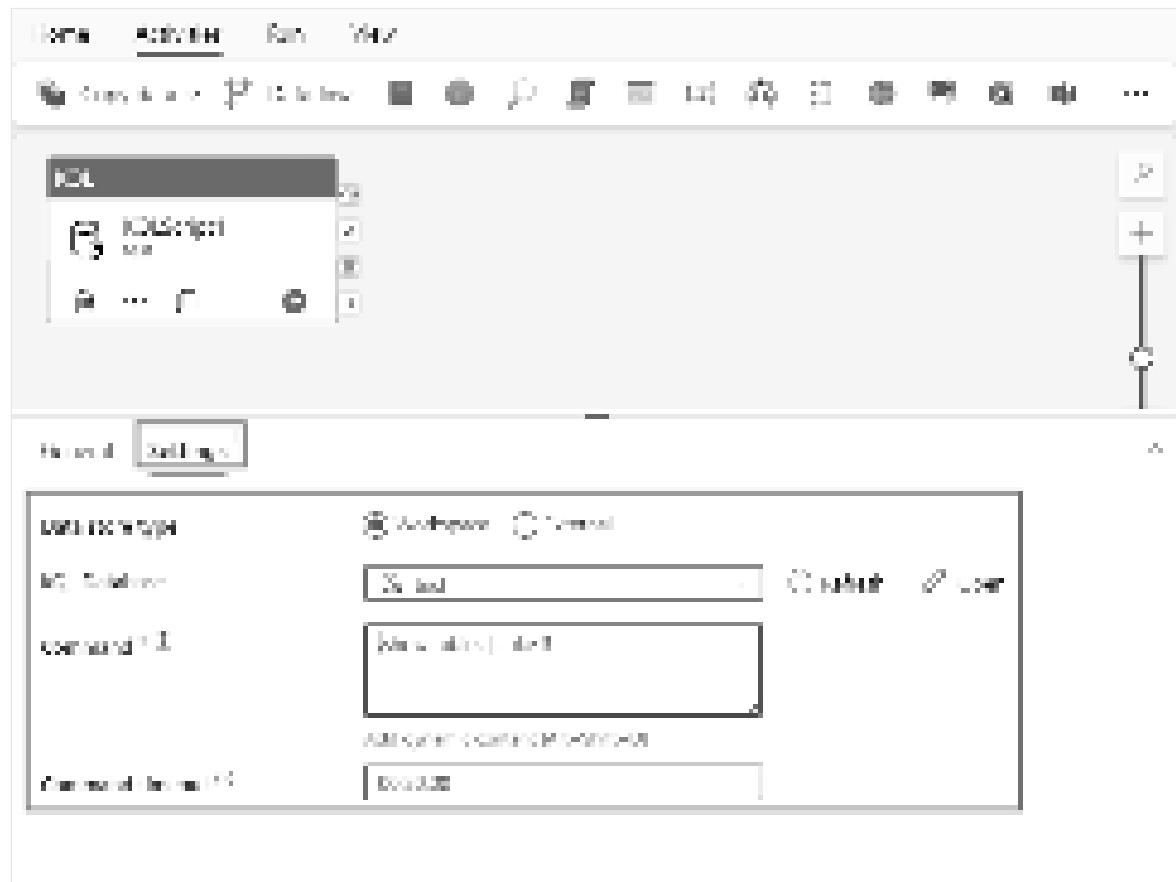
Refer to the **General** settings guidance to configure the **General** settings tab.

KQL activity settings

1. Select the **Settings** tab, then select a **Data store type**. You can use a KQL database created within your Fabric workspace, or connect to an external database with

Azure Data Explorer by providing a cluster URI, database, and credentials.

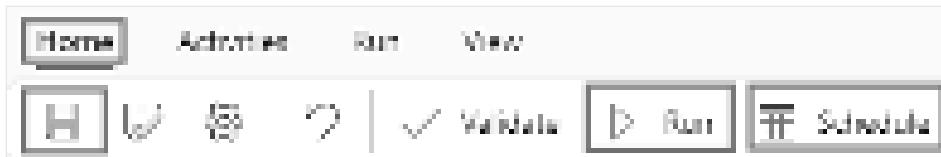
2. Next, select your **KQL Database** connection. If you select a workspace data store you can use dynamic content to parameterize the database selection by selecting the **Add dynamic content** option that appears in the dropdown.
3. Then provide a KQL query to execute against the selected database for the **Command** property. You can use dynamic content in the query by selecting the **Add dynamic content** link that appears when the text box is selected.



4. Finally, specify a command timeout or leave the default timeout of 20 minutes. You can use dynamic content for this property too.

Save and run or schedule the pipeline

The KQL activity might typically be used with other activities. After you configure any other activities required for your pipeline, switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

Yes

No

Provide product feedback | Ask the community

Use the Lookup activity to look up data from a data source

Article • 11/15/2023

The Fabric Lookup activity can retrieve a dataset from any of the data sources supported by Microsoft Fabric. You can use it to dynamically determine which objects to operate on in a subsequent activity, instead of hard coding the object name. Some object examples are files and tables.

Lookup activity reads and returns the content of a configuration file or table. It also returns the result of executing a query or stored procedure. The output can be a singleton value or an array of attributes, which can be consumed in a subsequent copy, transformation, or control flow activities like ForEach activity.

Prerequisites

To get started, you must complete the following prerequisites:

- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add a lookup activity to a pipeline with UI

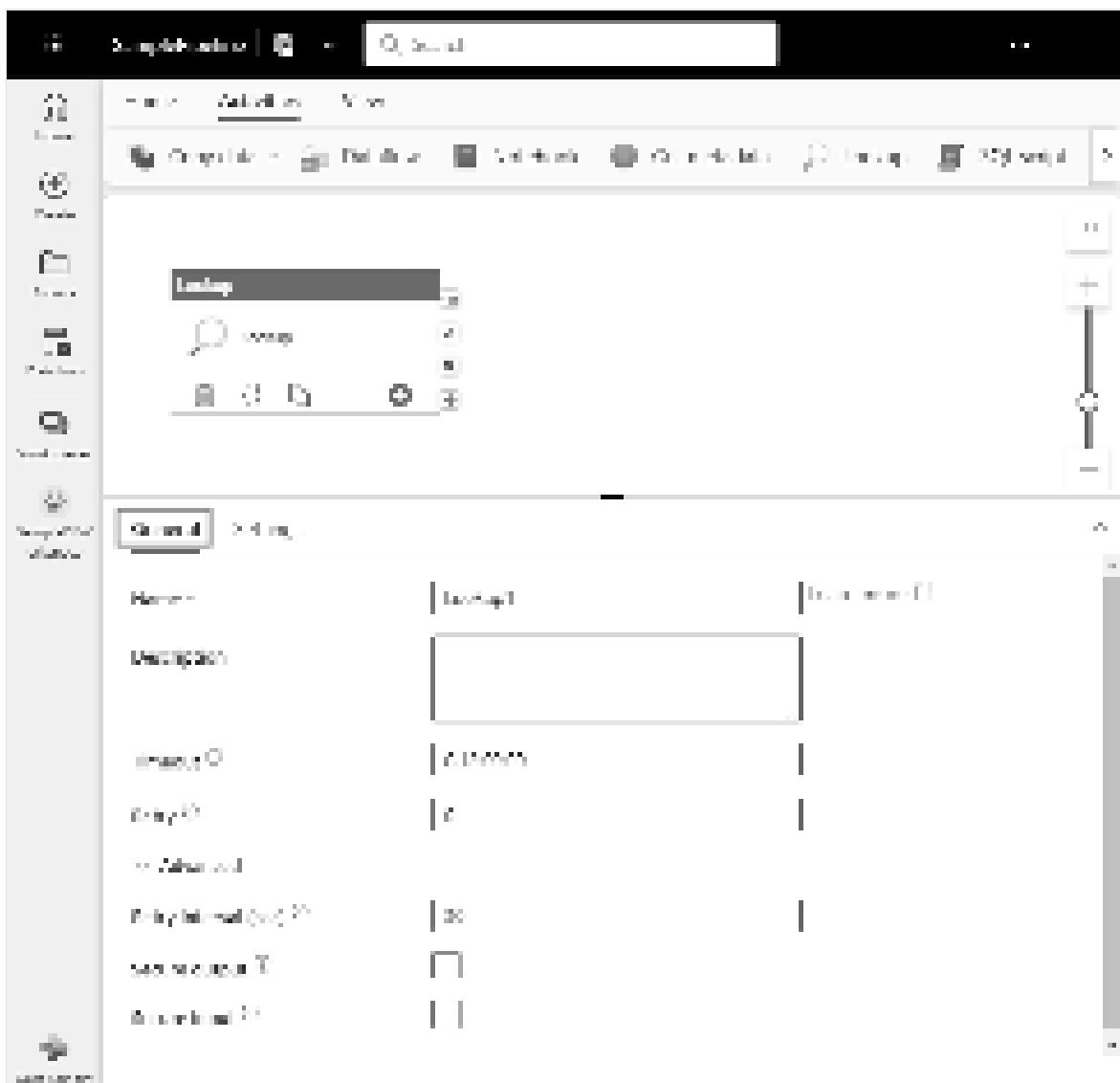
To use a Lookup activity in a pipeline, complete the following steps:

Creating the activity

1. Create a new pipeline in your workspace.
2. Search for **Lookup** in the pipeline **Activities** pane, and select it to add it to the pipeline canvas.



3. Select the new Lookup activity on the canvas if it isn't already selected.



Refer to the **General** settings guidance to configure the **General** settings tab.

Lookup settings

Select the **Settings** tab, select an existing connection from the **Connection** dropdown, or use the **+ New** button to create a new connection, and specify its configuration details.



The example in the previous image shows a blob storage connection, but each connection type has its own configuration details specific to the data source selected.

Supported capabilities

- The Lookup activity can return up to 5000 rows; if the result set contains more records, the first 5000 rows are returned.
- The Lookup activity output supports up to 4 MB in size; activity fails if the size exceeds the limit.
- The longest duration for Lookup activity before timeout is 24 hours.

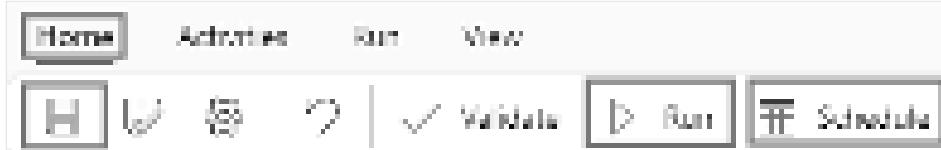
! Note

When you use query or stored procedure to look up data, make sure to return one and exact one result set. Otherwise, Lookup activity fails.

Fabric supports the data stores listed in the Connector overview article. Data from any source can be used.

Save and run or schedule the pipeline

Switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

Yes

No

Provide product feedback [↗](#) | Ask the community [↗](#)

Transform data by running a notebook

Article • 11/15/2023

The Notebook activity in pipeline allows you to run Notebook created in Microsoft Fabric. You can create a Notebook activity directly through the Fabric user interface. This article provides a step-by-step walkthrough that describes how to create a Notebook activity using the Data Factory user interface.

Add a Notebook activity to a pipeline

This section describes how to use a Notebook activity in a pipeline.

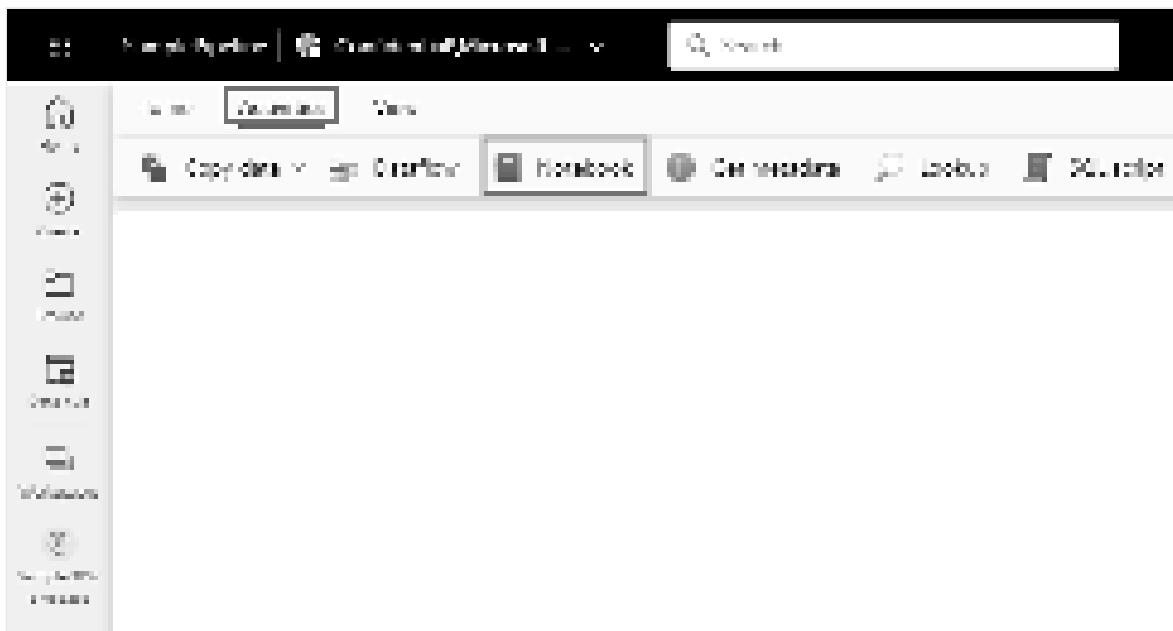
Prerequisites

To get started, you must complete the following prerequisites:

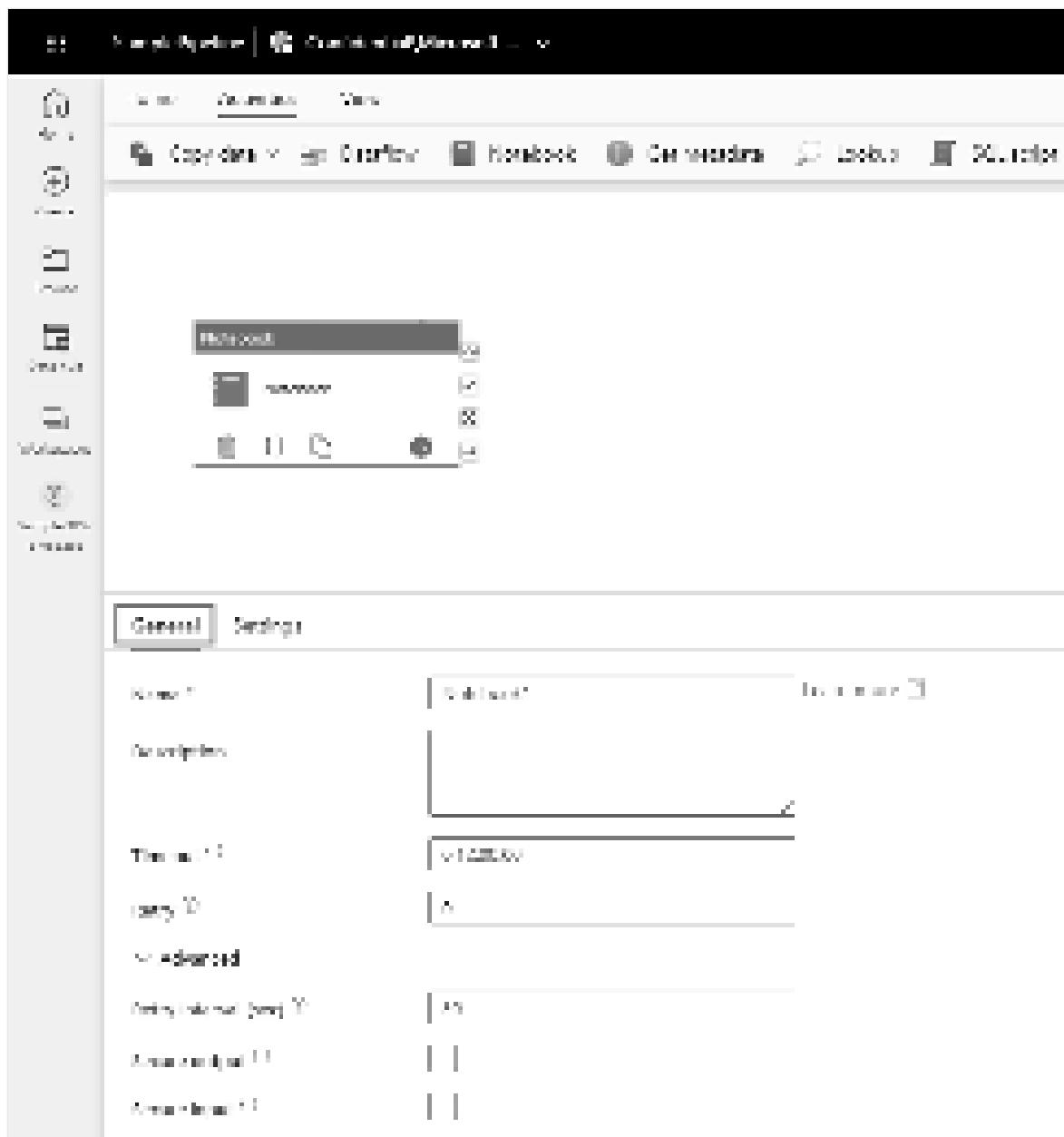
- A tenant account with an active subscription. Create an account for free.
- A workspace is created.
- A notebook is created in your workspace. To create a new notebook, refer to [How to create Microsoft Fabric notebooks](#).

Creating the activity

1. Create a new pipeline in your workspace.
2. Search for Notebook in the pipeline **Activities** pane, and select it to add it to the pipeline canvas.



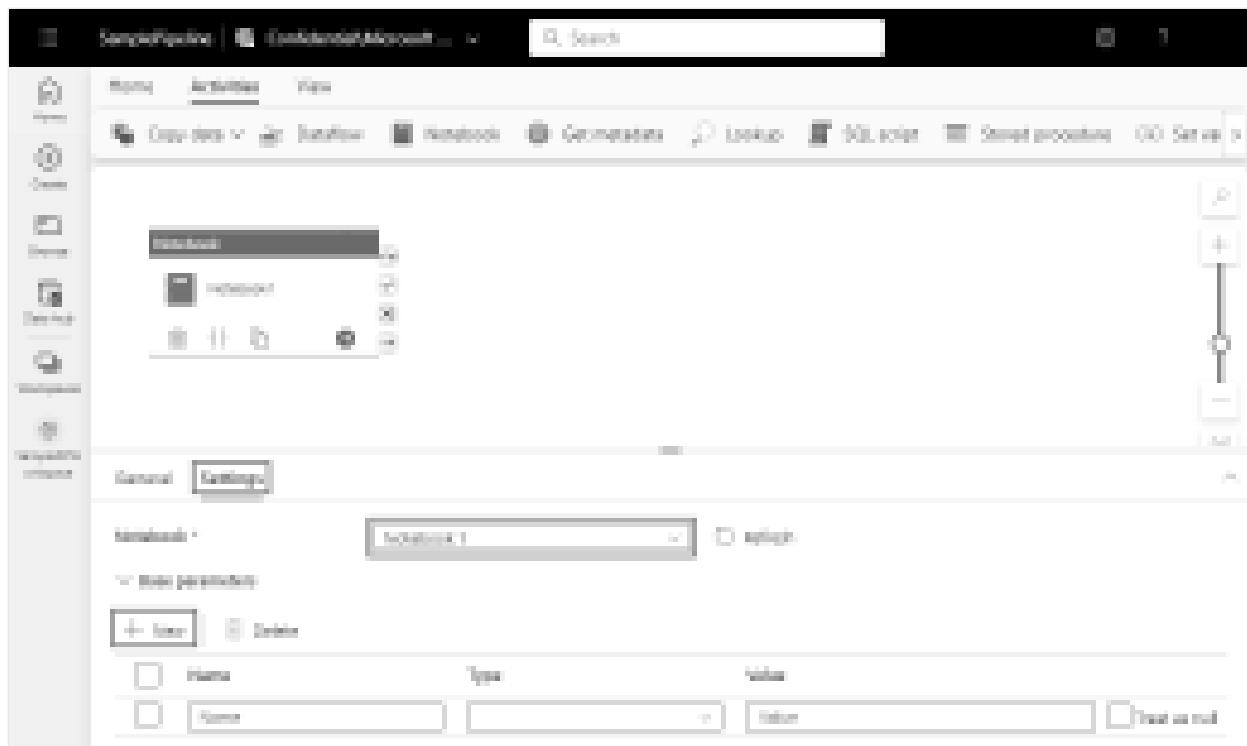
3. Select the new Notebook activity on the canvas if it isn't already selected.



Refer to the General settings guidance to configure the **General** settings tab.

Notebook settings

Select the **Settings** tab, select an existing notebook from the **Notebook** dropdown, and optionally specify any parameters to pass to the notebook.



Save and run or schedule the pipeline

Switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

Yes

No

Provide product feedback | Ask the community

Use the Office 365 Outlook activity to send an email with Outlook

Article • 12/07/2023

The Office 365 Outlook activity in Data Factory for Microsoft Fabric allows you to send an email with your Office 365 account. The message can include dynamic expressions to be customized as much as necessary.

Prerequisites

To get started, you must complete the following prerequisites:

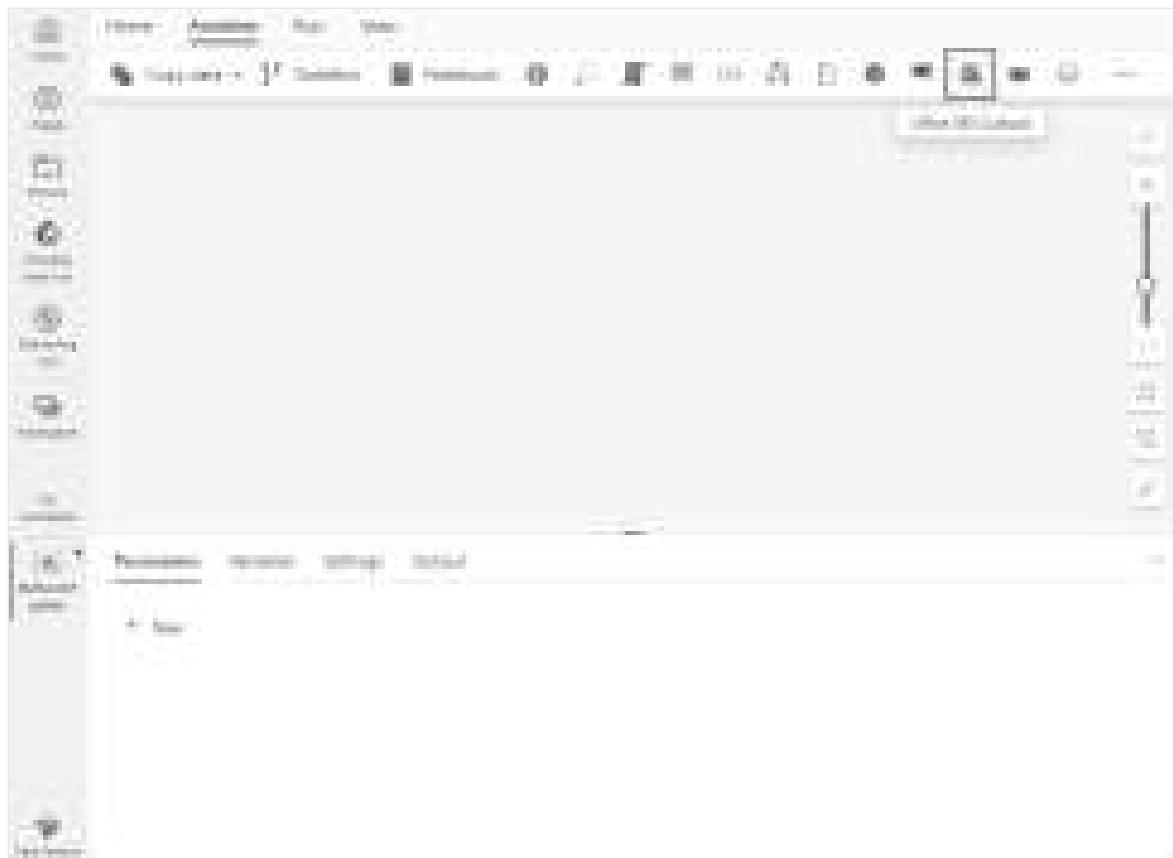
- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add an Office 365 Outlook activity to a pipeline with UI

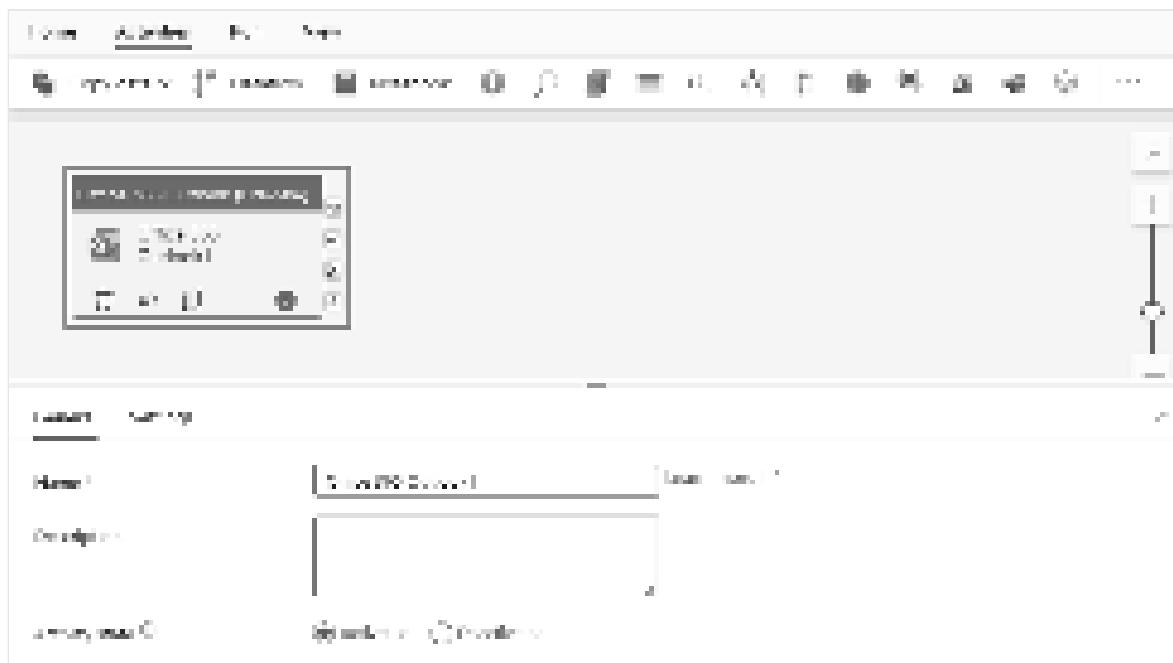
To use an Office 365 Outlook activity in a pipeline, complete the following steps:

Creating the activity

1. Create a new pipeline in your workspace.
2. Search for Office 365 Outlook in the pipeline **Activities** pane, and select it to add it to the pipeline canvas. It might be necessary to expand the activities list on the far right side of the pane, or the Outlook icon can be compressed without labeling text beneath it, as shown in this image, depending on the window width of your browser.



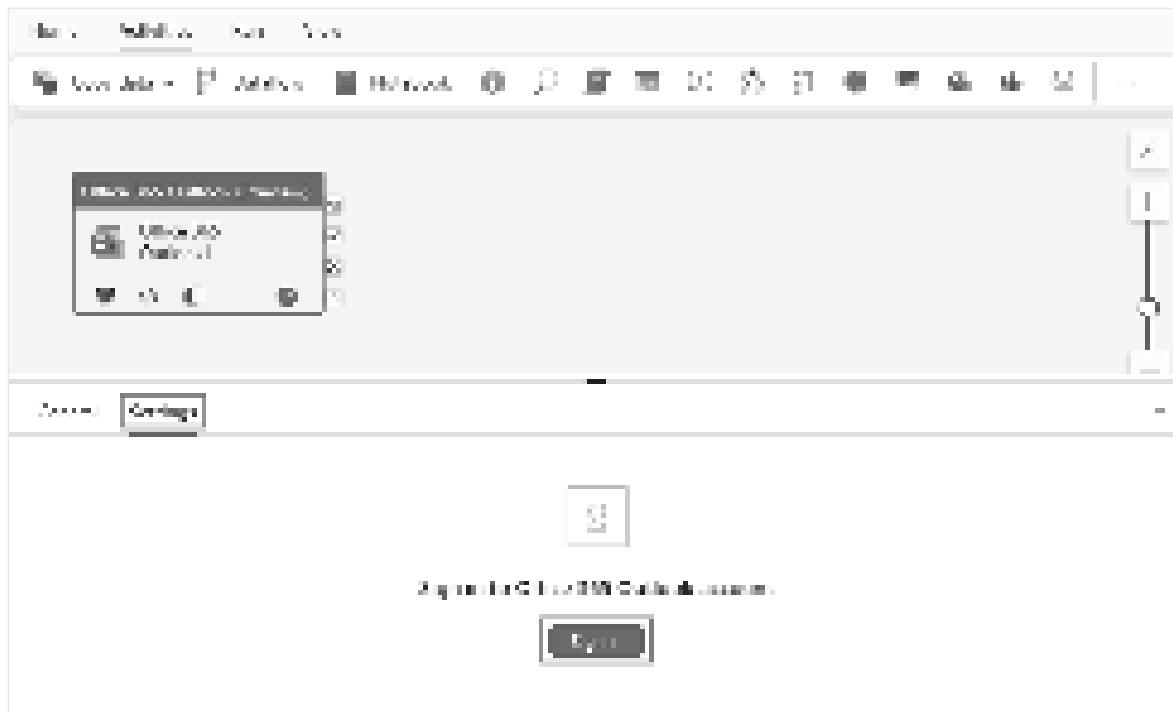
3. Select the new Outlook activity on the canvas if it isn't already selected.



Refer to the **General** settings guidance to configure the **General** settings tab.

Office 365 Outlook activity settings

1. Select the **Settings** tab, then select **Sign in** to sign in to your Office 365 account.



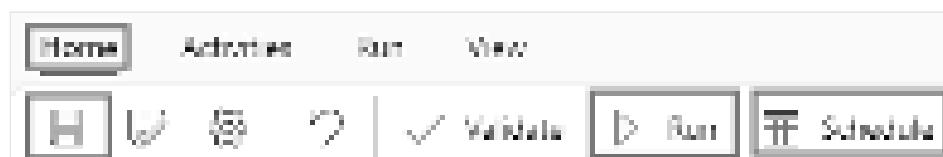
2. An authentication dialog appears for you to provide credentials for the account you want to use in Outlook. After that, a confirmation appears for you to allow access to Outlook from your pipeline. Select **Allow access** on the confirmation dialog to connect your Outlook activity to your account.
3. Once connected, you can choose to provide details for the email, including its recipients, subject, body. You can also include **Advanced** details such as a custom from address, CC and BCC recipients, sensitivity, and a custom reply-to address. All of the fields support dynamic expressions.

The screenshot shows the Microsoft Power Automate pipeline editor interface. At the top, there are tabs for 'General' and 'Settings'. The 'General' tab is selected. Below the tabs, there are several configuration sections:

- Subscription**: Set to 'John Doe (john.doe@contoso.com) - Personal account'.
- Title**: Set to 'Invoicing pipeline - john.doe@contoso.com'.
- Description**: Set to 'Payroll reporting'.
- Run ID**: Shows '1'.
- Last run**: Shows '2023-07-10 10:00:00'.
- Run history**: A button labeled 'View run history'.
- Run settings**: A section with dropdown menus for 'Run type', 'Run mode', 'Run schedule', and 'Run history'.

Save and run or schedule the pipeline

The Teams activity is typically used with other activities, often as a status notification for the outcome of prior steps in a pipeline. After you configure any other activities required for your pipeline, switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Related content

How to monitor pipeline runs

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

How to use Script activity

Article • 11/15/2023

In this article, you learn how to add a new Script activity, add a new connection, and configure script content.

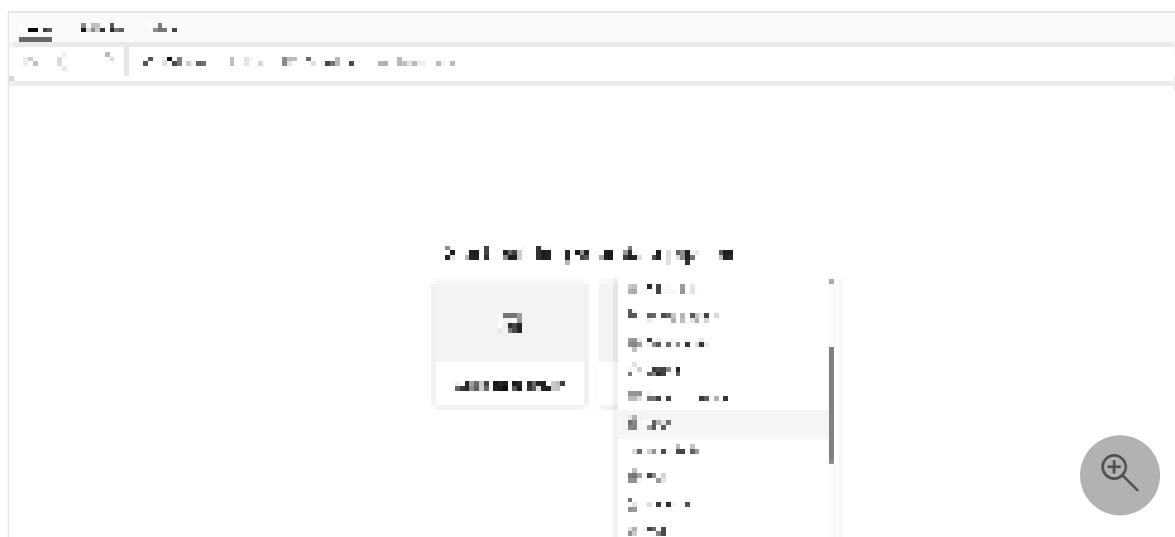
Prerequisites

To get started, you must complete the following prerequisites:

- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

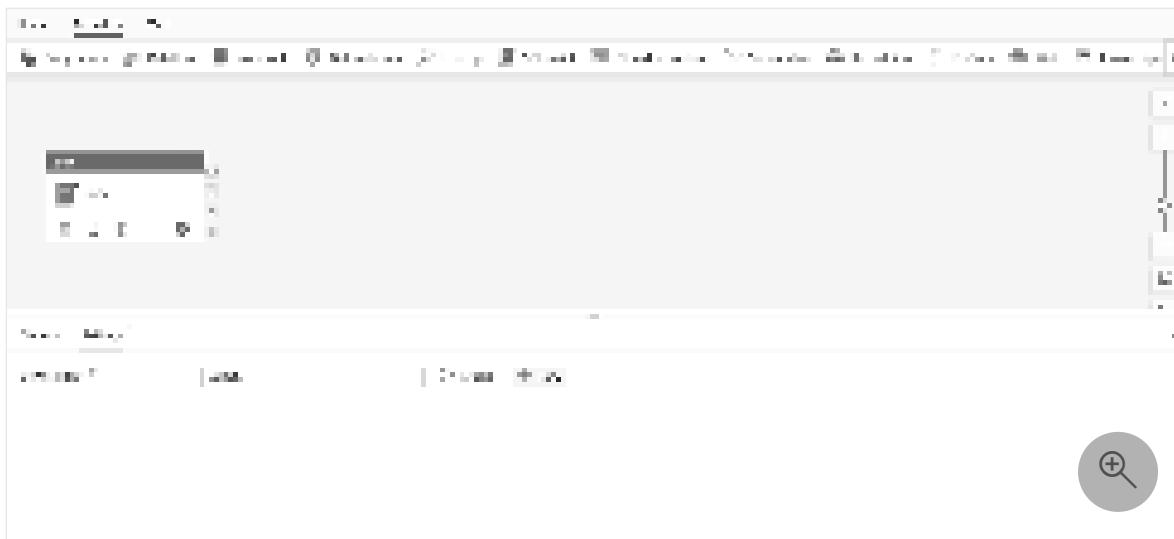
Step 1: Add a new Script activity in Pipeline canvas

1. Open an existing data pipeline or create a new data pipeline.
2. Select the **Script** activity.



Step 2: Add a new connection for SQL

1. Select the **Settings** tab. Select **New** to create a new connection.



2. In the new popup window, choose the target SQL source.



3. Create a new connection for the SQL source.

New connection



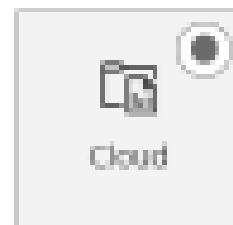
ⓘ Currently, these cloud connections are only supported for Data Pipelines and Kusto. In the future, other artifacts can also make use of the cloud connections. To create personal cloud connections in Datasets, Datamarts, and Dataflows, use the Power Query Online experience in "Get data".



On-premises



Virtual
network



Cloud

Connection name *

Connection type *

SQL Server

Server *

Ex: testazuresqlserver.database.windows.net

Database *

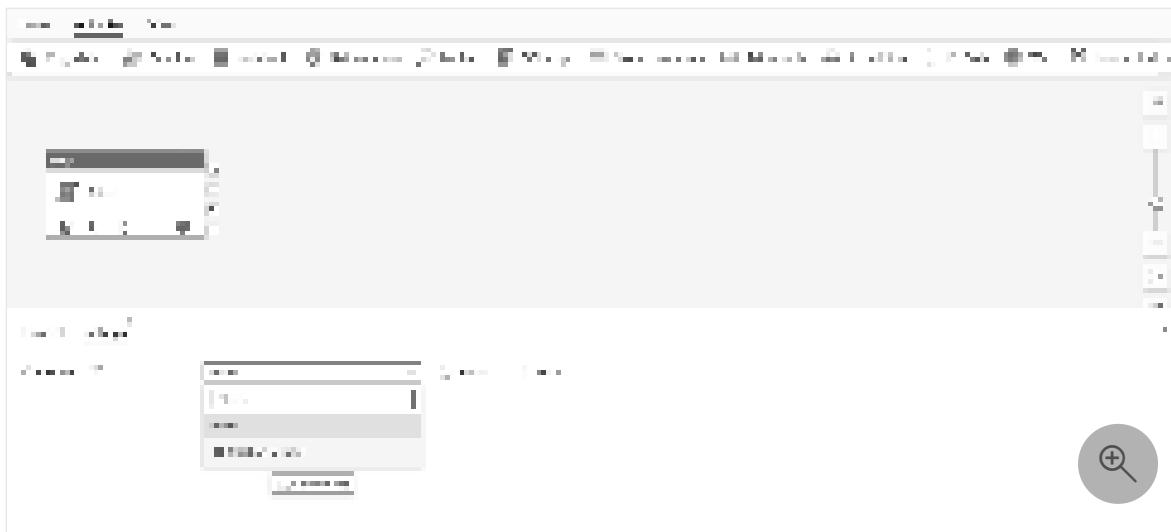
Ex: ContosoDB

Create

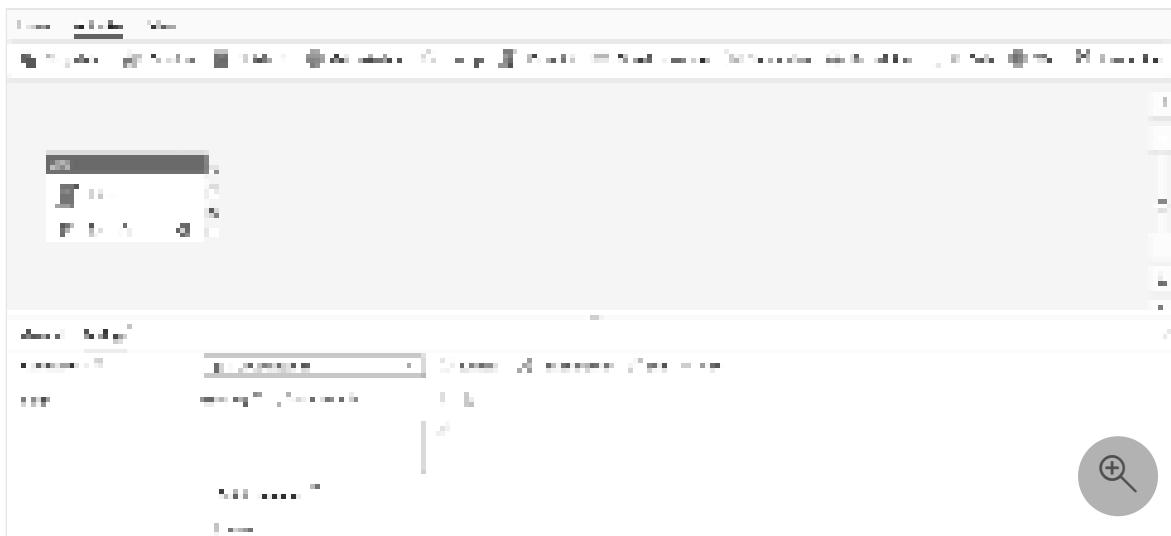
Close

Step 3: Configure script content

1. Select the connection you created in the previous step.



2. You can choose either **Query** to get a data result or **NonQuery** for any catalog operations.
3. Input the script content in the input box.
4. You can also define parameters for your script.



5. The Script activity is successfully created and you can run it directly.

Next steps

Monitor pipeline runs

Feedback

Was this page helpful?

Yes

No

[Provide product feedback](#) | [Ask the community](#)

Use the Set Variable activity in Fabric

Article • 11/15/2023

Use the Set Variable activity to set the value of an existing variable of type String, Bool, or Array defined in a pipeline in Data Factory for Microsoft Fabric or use the Set Variable activity to set a pipeline return value.

Prerequisites

To get started, you must complete the following prerequisites:

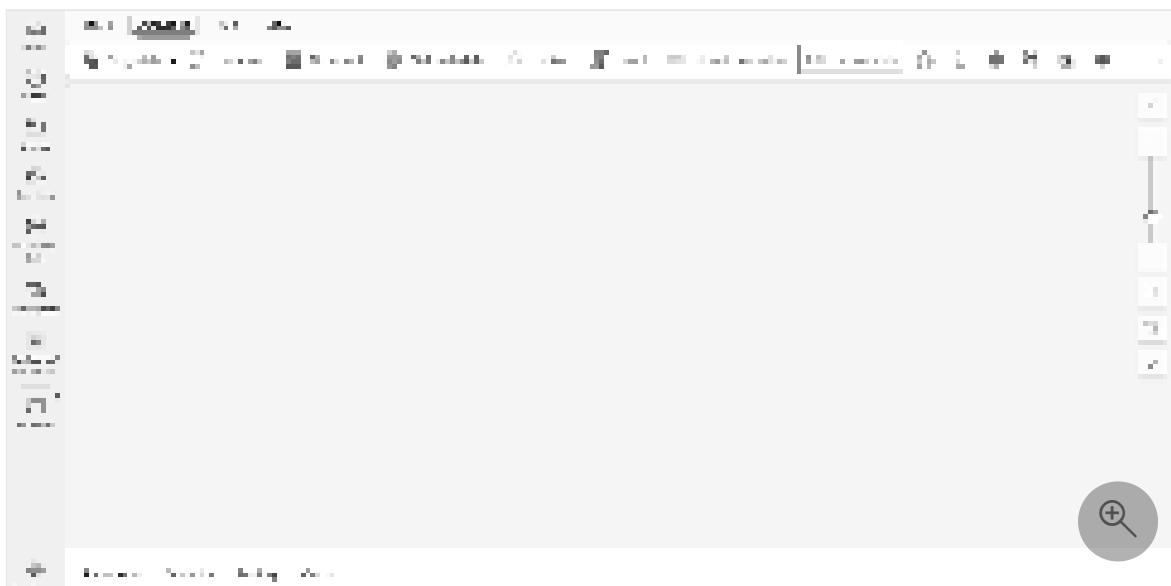
- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add a Set Variable activity to a pipeline with UI

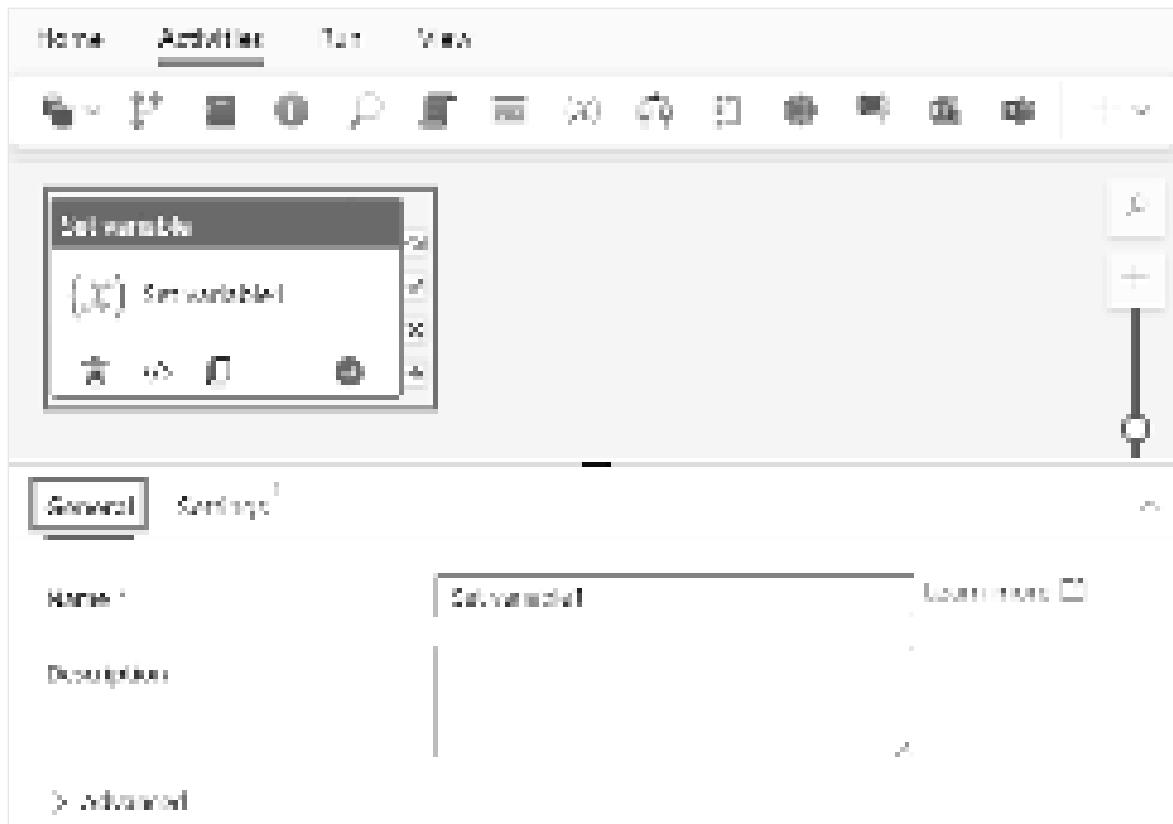
To use a Set Variable activity in a pipeline, complete the following steps:

Creating the activity

1. Create a new pipeline in your workspace.
2. Search for **Set Variable** in the pipeline **Activities** pane, and select it to add it to the pipeline canvas. You may need to expand the list of available activities using the dropdown + button at the far right of the toolbar if the window size isn't large enough to display it directly.



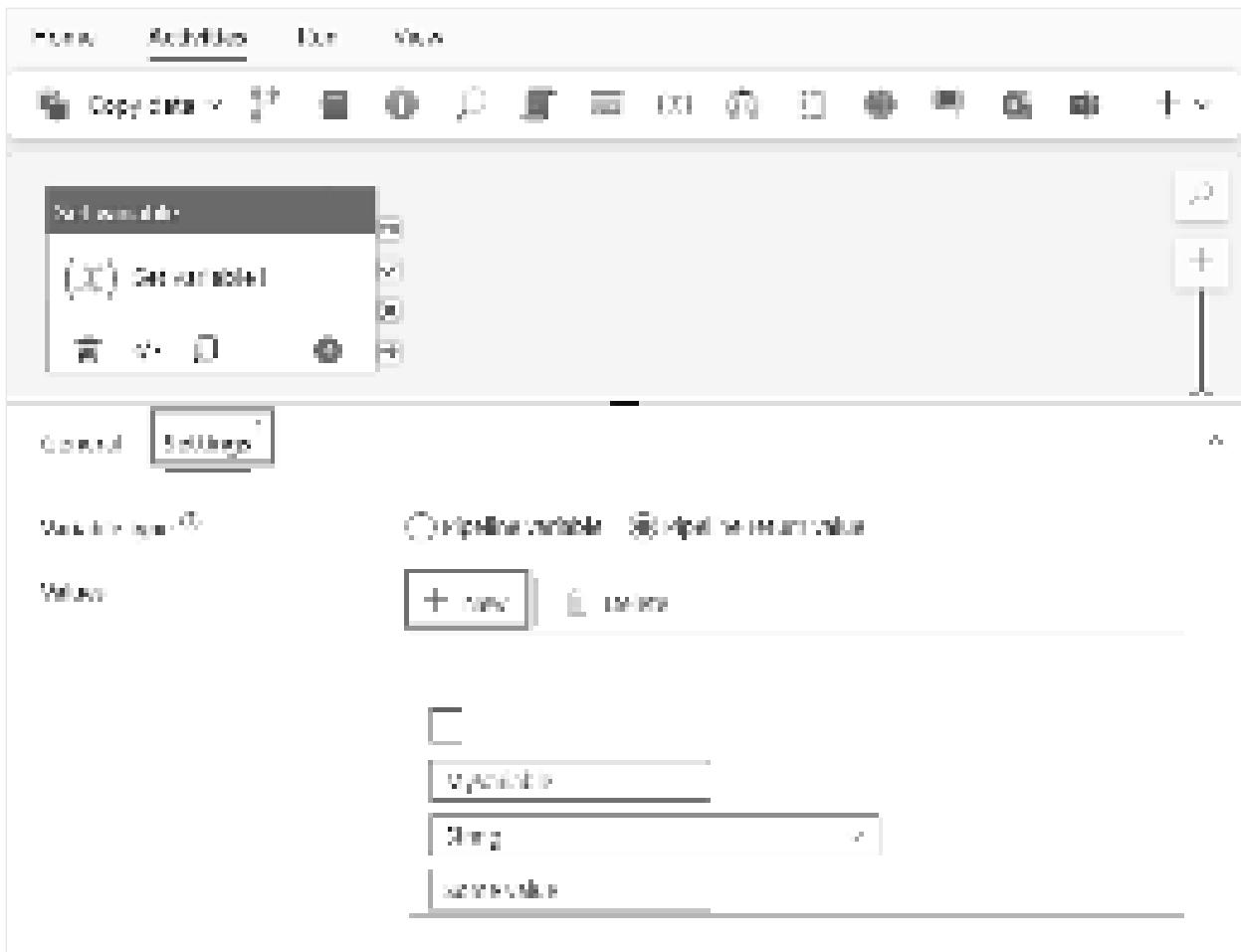
3. Select the new activity on the canvas if it isn't already selected.



Refer to the **General** settings guidance to configure the **General** settings tab.

Set Variable settings

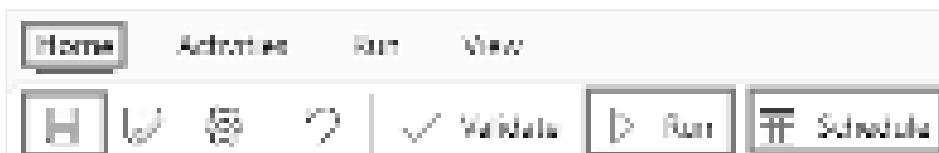
On the **Settings** tab, you can choose either a pipeline variable, or you can specify a return value for the pipeline directly. This example shows a single return value called MyVariable, with a string data type, and value "Some value".



You can add multiple values for the **Pipeline return value** using the **+ New** button (or the **Delete** button to remove them). Likewise, when setting a **Pipeline variable**, you can choose any previously defined variables from the **Name** dropdown box, or add new one(s) with the **+ New** button. If you want to create pipeline variables independently of the Set Variable activity, select the background in the pipeline canvas (to show the pipeline properties pages), and then use the **Variables** tab in the pipeline settings to create them. Then you can select them from the dropdown list in the Set Variable **Settings** when choosing **Pipeline variable**.

Save and run or schedule the pipeline

Although Set Variable is typically used with other activities, it can be run directly as is. To run the pipeline here, switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

How to use Stored procedure activity

Article • 11/15/2023

In this article, you learn how to add a new Stored procedure activity to a pipeline, add a new connection, and configure the activity to run.

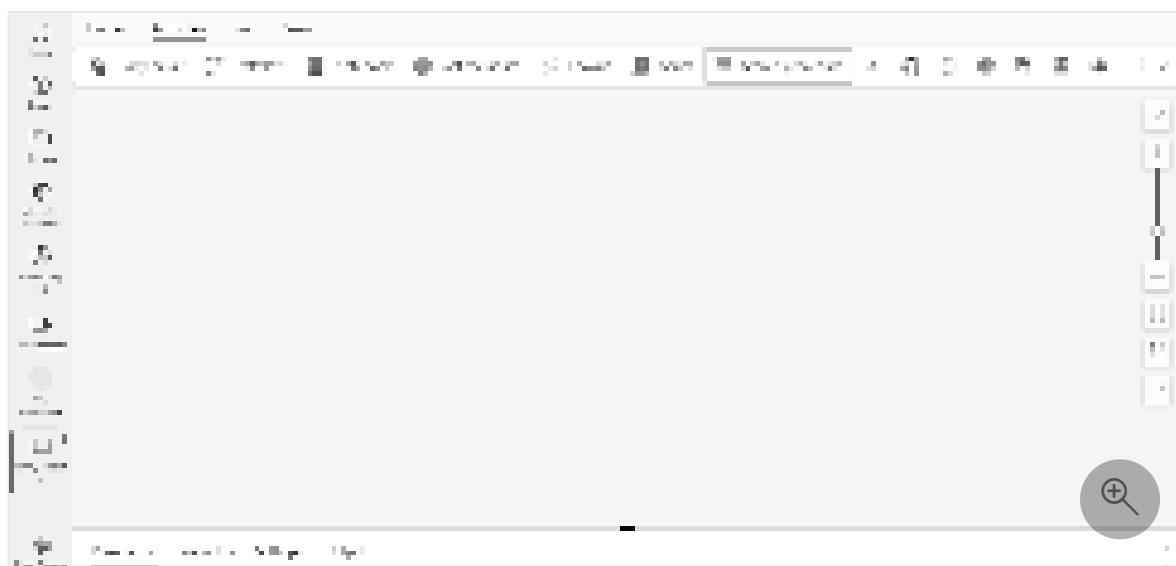
Prerequisites

To get started, you must complete the following prerequisites:

- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Step 1: Add a new Stored procedure activity in Pipeline canvas

1. Open an existing data pipeline or create a new data pipeline.
2. Select the **Stored procedure** activity.

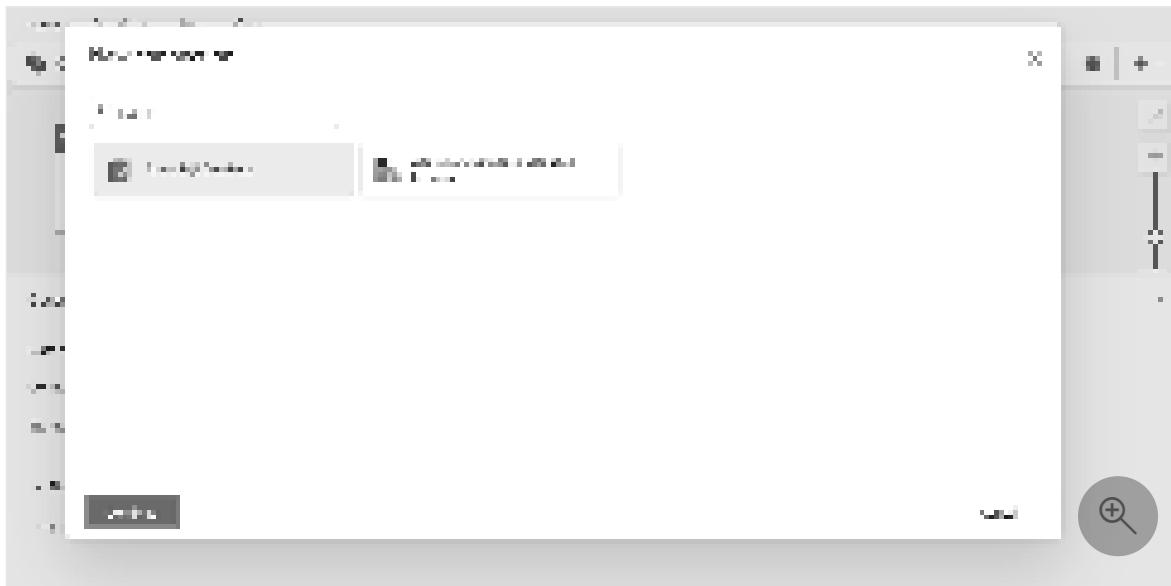


Step 2: Add a new connection for SQL

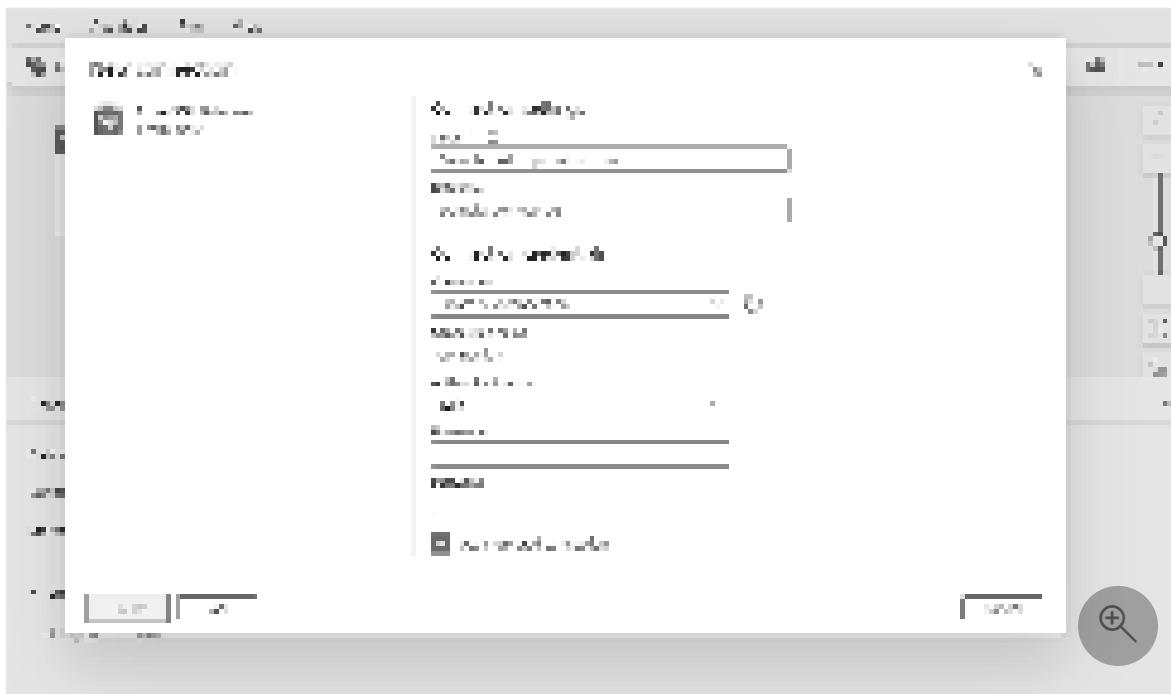
1. Select the **Settings** tab. Select **New** to create a new connection.



2. In the new popup window, choose the target SQL source type, then select **Continue**. The Stored procedure activity in Fabric currently supports Azure SQL and Azure SQL Managed instances.

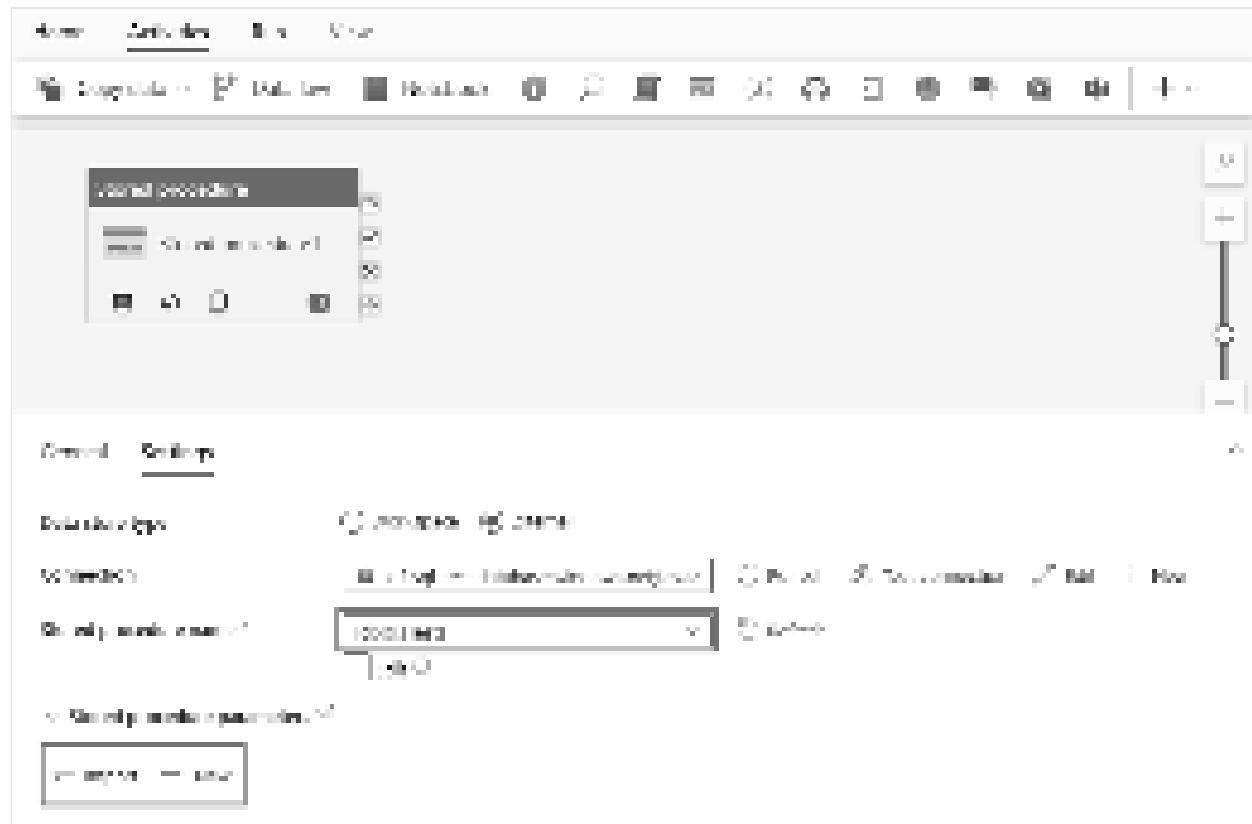


3. Provide the connection details for the new connection and select **Create**.



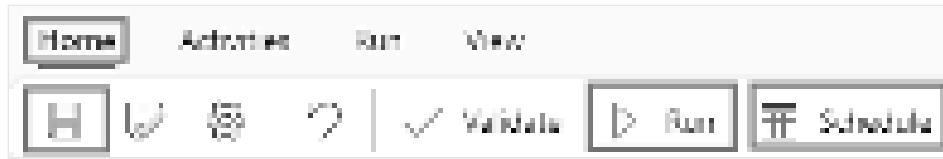
Step 3: Choose a stored procedure and configure parameters

Select a stored procedure, and optionally import its parameters or manually add parameters.



Step 4: Save and run or schedule the pipeline

Switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Next steps

Monitor pipeline runs

Feedback

Was this page helpful?

Yes No

Provide product feedback [↗](#) | Ask the community [↗](#)

Use the Switch activity to conditionally branch execution in a pipeline

Article • 11/15/2023

The Switch activity in Microsoft Fabric provides the same functionality that a switch statement provides in programming languages. It evaluates a set of activities corresponding to a case that matches the condition evaluation.

Prerequisites

To get started, you must complete the following prerequisites:

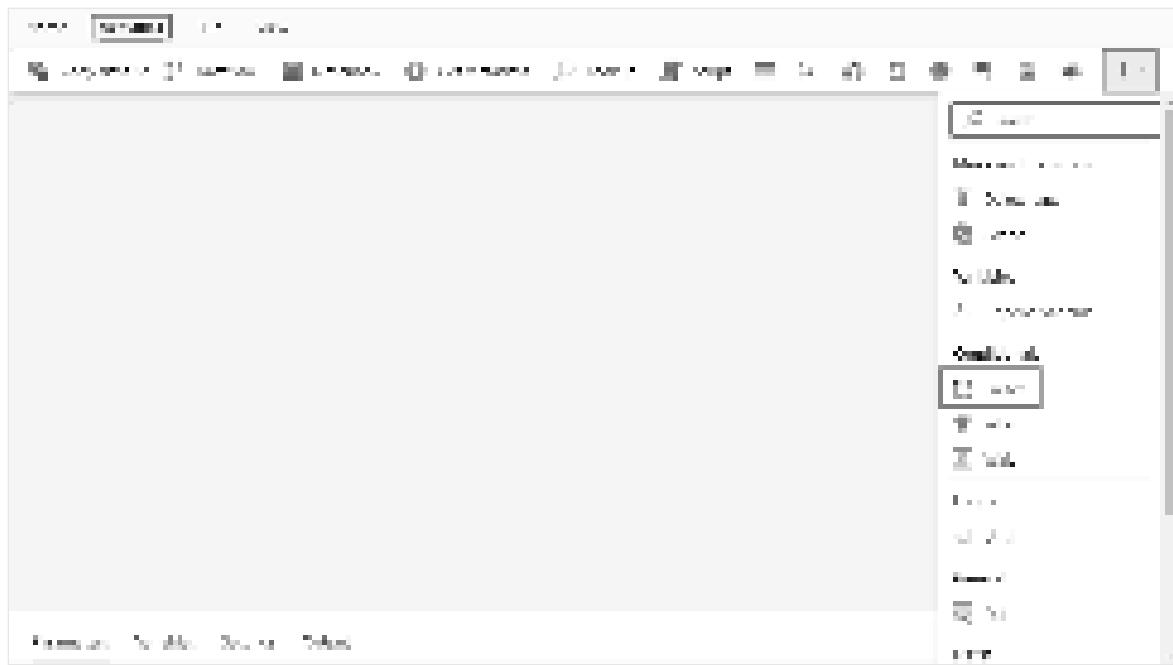
- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add a lookup activity to a pipeline with UI

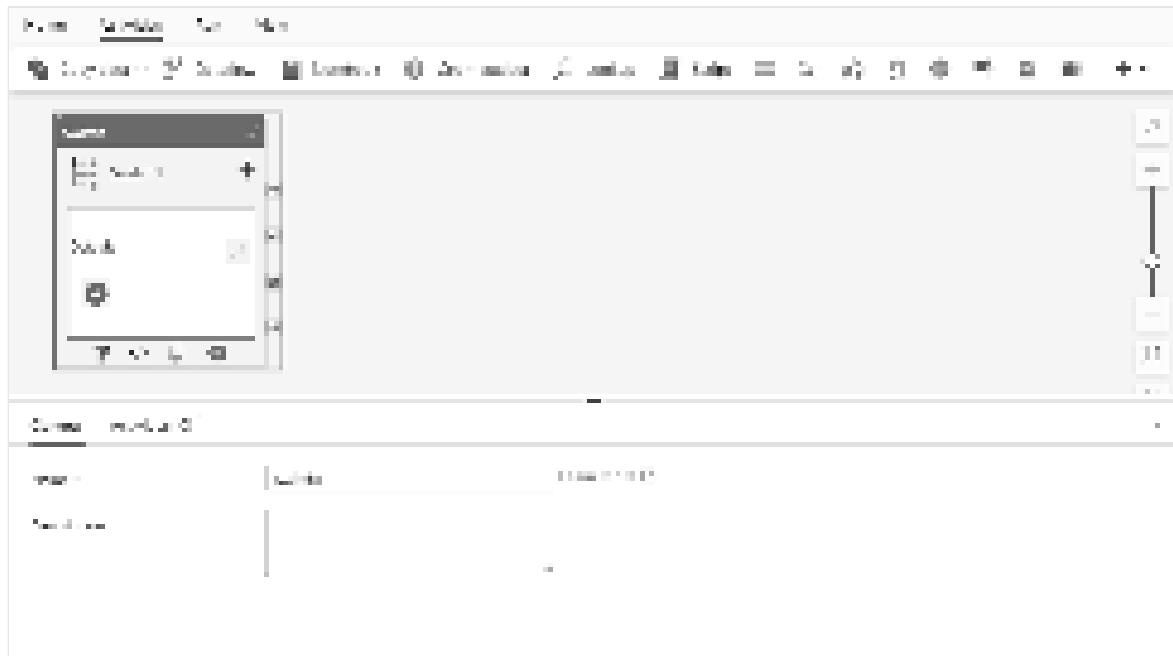
To use a Switch activity in a pipeline, complete the following steps:

Creating the activity

1. Create a new pipeline in your workspace.
2. Search for the Switch activity in the pipeline **Activities** toolbar, and select it to add it to the pipeline canvas. You may need to expand the list of activities using the + button since there are more activities available than will typically fit in the space for the toolbar.



3. Select the new Switch activity on the canvas if it isn't already selected.



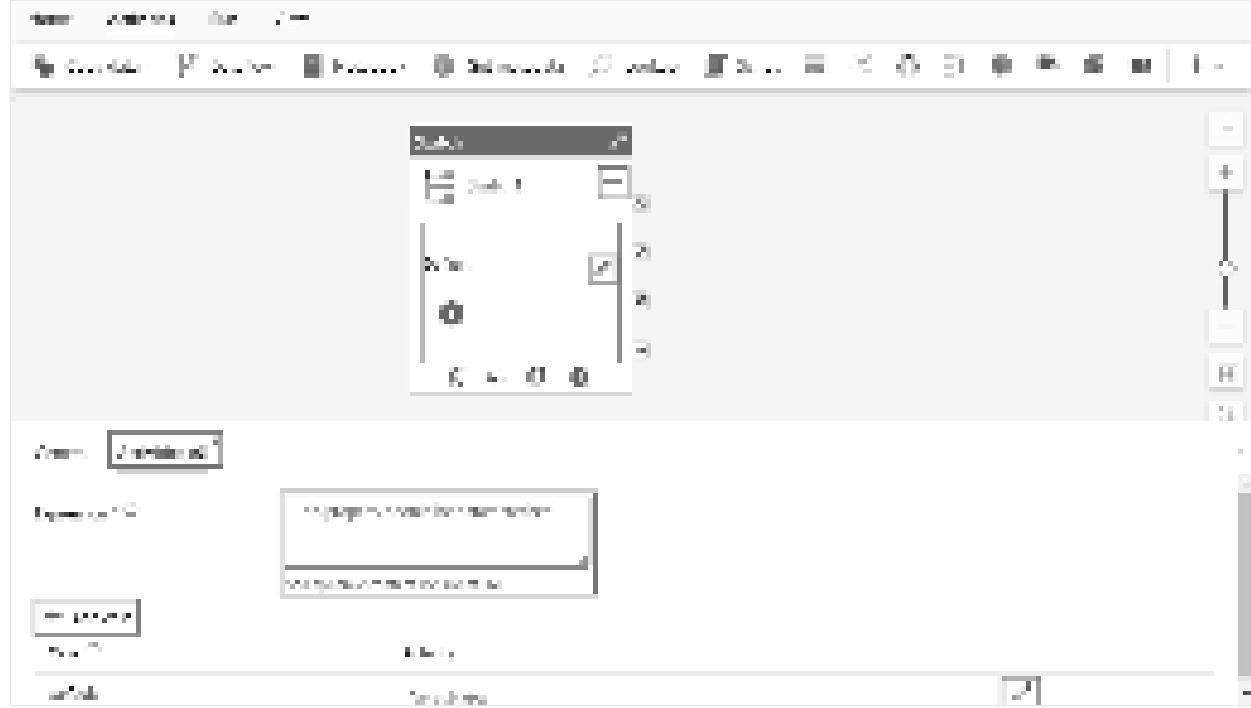
Refer to the General settings guidance to configure the General settings tab.

Activities settings

Select the Activities tab, and you will see the Default case already added to the list of cases.

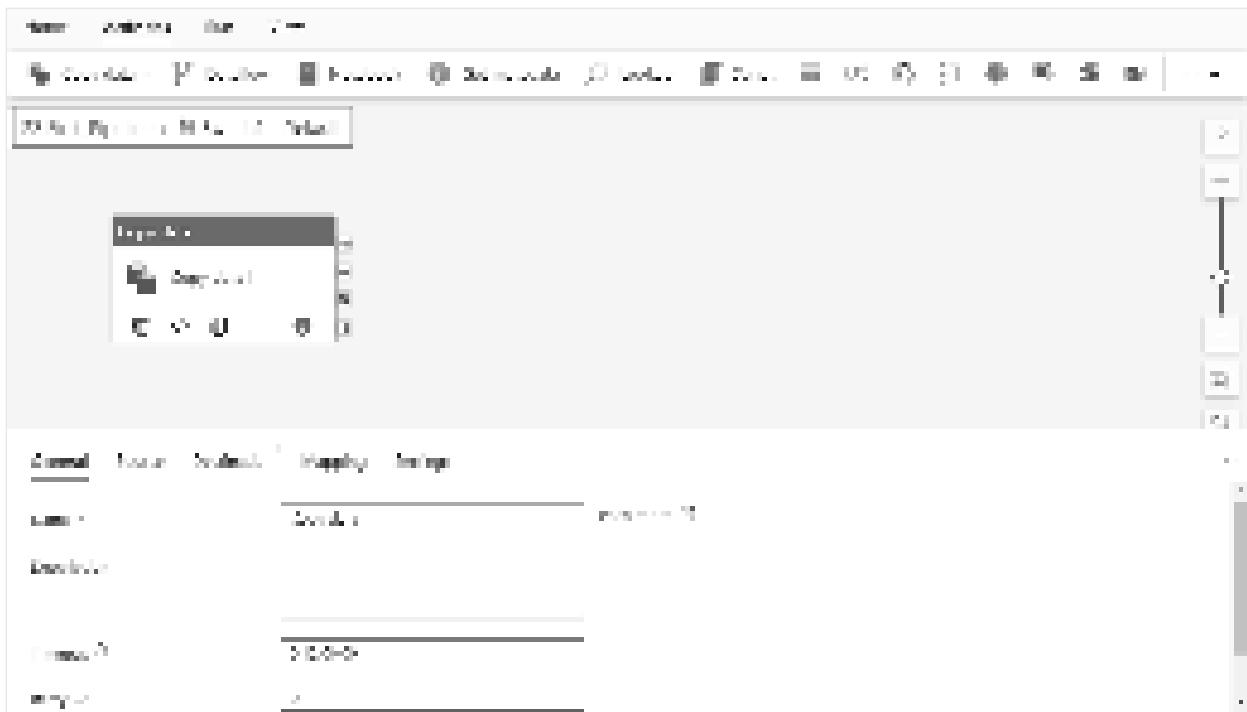
The **Expression** clause is where you provide an expression to be evaluated and compared to the **Case** options, and supports dynamic content that allows you to use parameters, system variables, functions, and local variables from your project to compare against the various cases.

You can use the pencil icon to the right of each case on the **Activities** tab to configure the activities that will execute when that case matches the **Expression**. You can use the **+ Add case** button to add new cases besides the default that executes if no other case matches the **Expression** result. You can also add cases and activities directly on the Switch activity interface on the pipeline canvas itself by selecting the **+** button to add a case, or the pencil icon beside the listed cases to configure activities for each case.



Configuring case activities

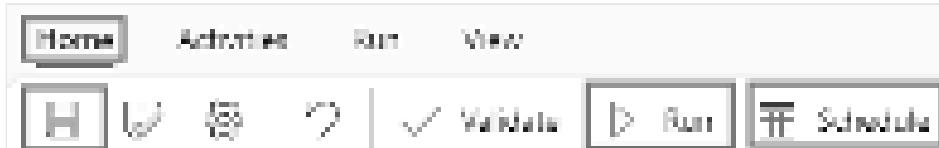
When you edit the activities for a case by selecting the pencil icon beside it, either in the **Activities** settings pane, or directly on the Switch activity interface on the pipeline canvas, you will see the case's activities editor. This is similar to the pipeline editor, but specific to the selected case. You can add any number of activities, just like with a pipeline, and these will be executed when the selected case matches the **Expression** for the Switch activity. In this example, a Copy activity was added to the default case, and will execute whenever none of the other cases match the **Expression** defined in the Switch activity.



Notice the pipeline and case in the top left corner of the activities editor for the case. When you finish configuring the case's activities, you can select the pipeline name link there to navigate back to the main pipeline editor again.

Save and run or schedule the pipeline

Switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

Yes

No

Provide product feedback ↗ | Ask the community ↗

Use the Teams activity to send a message in Teams

Article • 11/15/2023

The Teams activity in Data Factory for Microsoft Fabric allows you to send a message to a Teams channel or group chat. The message can include dynamic expressions to be customized as much as necessary.

Prerequisites

To get started, you must complete the following prerequisites:

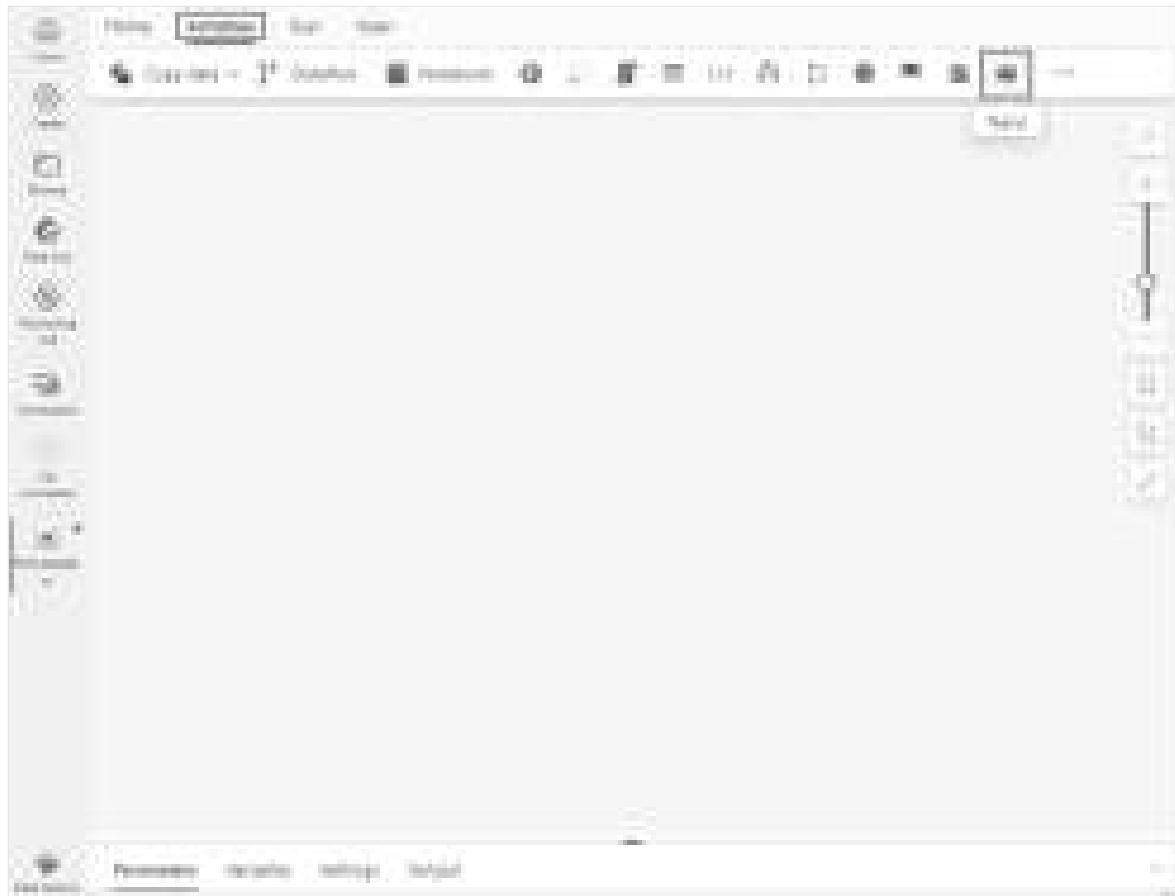
- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add a Teams activity to a pipeline with UI

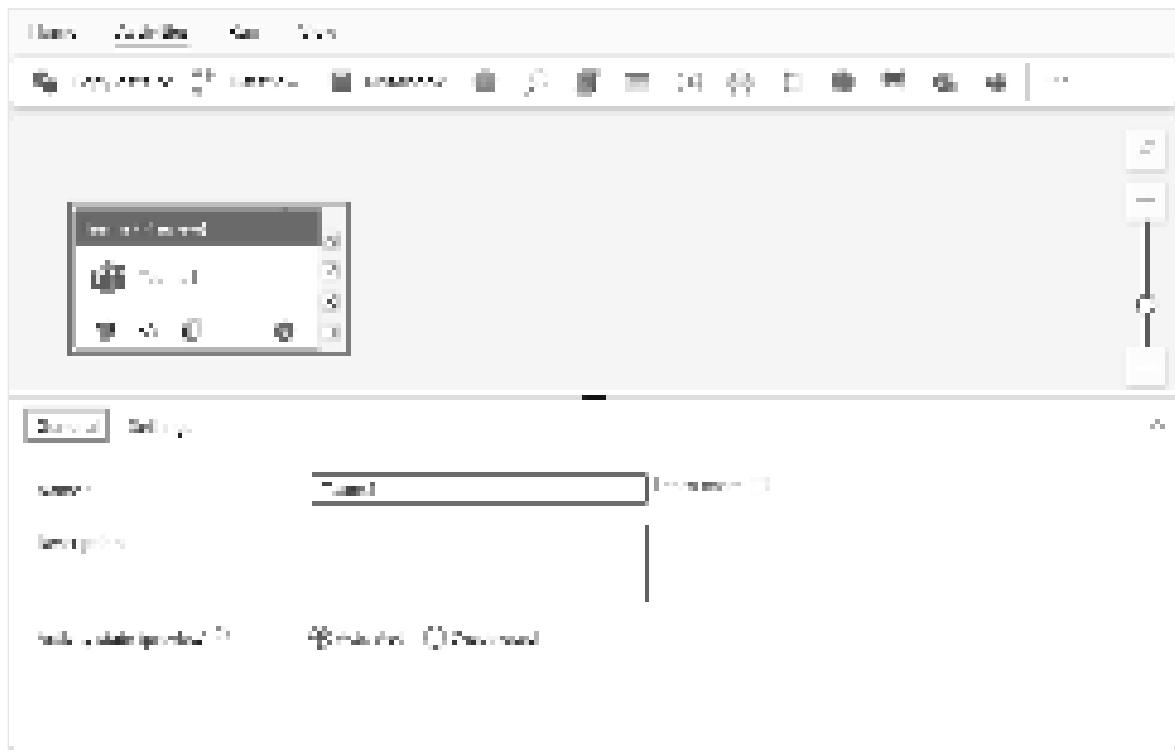
To use a Lookup activity in a pipeline, complete the following steps:

Creating the activity

1. Create a new pipeline in your workspace.
2. Search for Teams in the pipeline **Activities** pane, and select it to add it to the pipeline canvas.



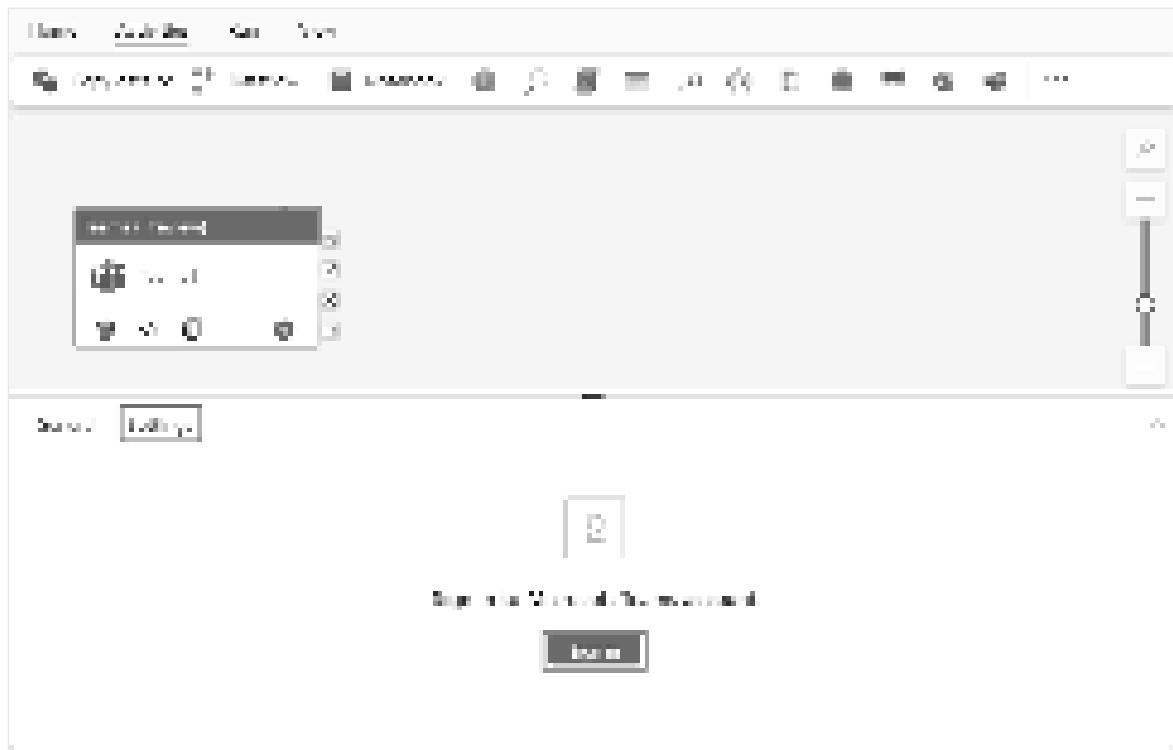
3. Select the new Teams activity on the canvas if it isn't already selected.



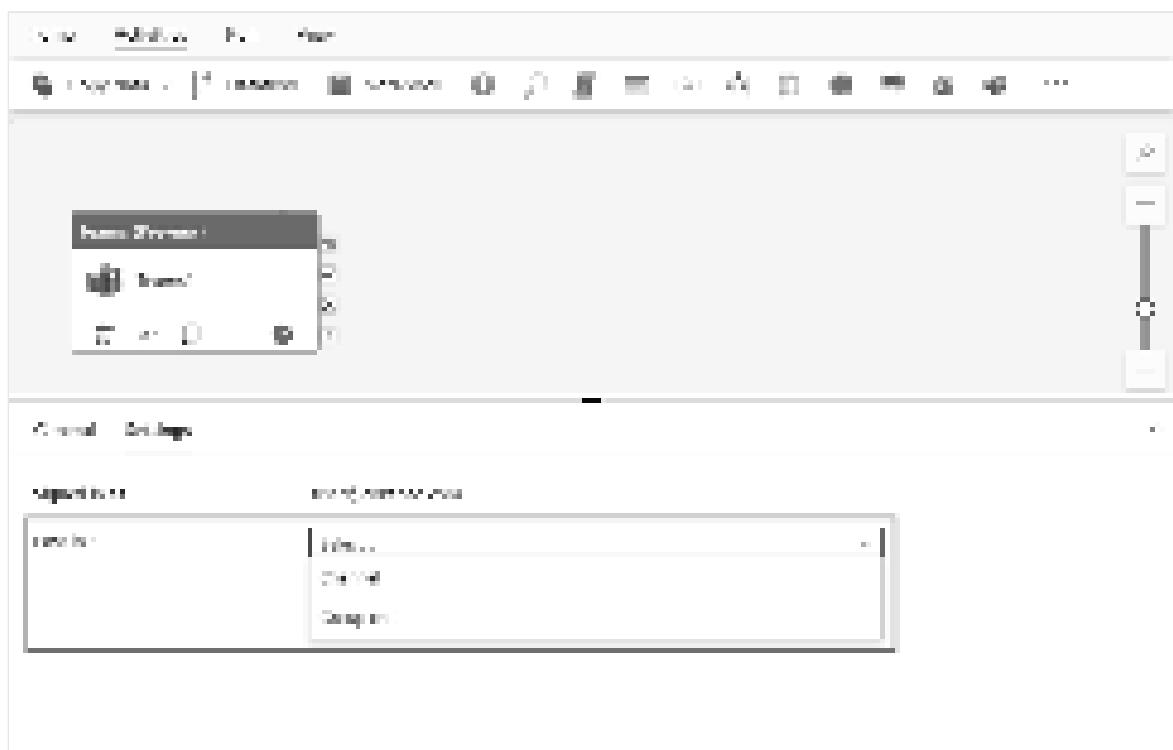
Refer to the General settings guidance to configure the **General** settings tab.

Teams activity settings

1. Select the **Settings** tab, then select **Sign in** to sign in to your Teams account.



2. An authentication dialog appears for you to provide credentials for the account you want to use in Teams. After that, a confirmation appears for you to allow access to Teams from your pipeline. Select **Allow access** on the confirmation dialog to connect your Teams activity to your account.
3. Once connected, you can choose where you want to post the message with the **Post in** setting. You can post to a channel or a group chat.



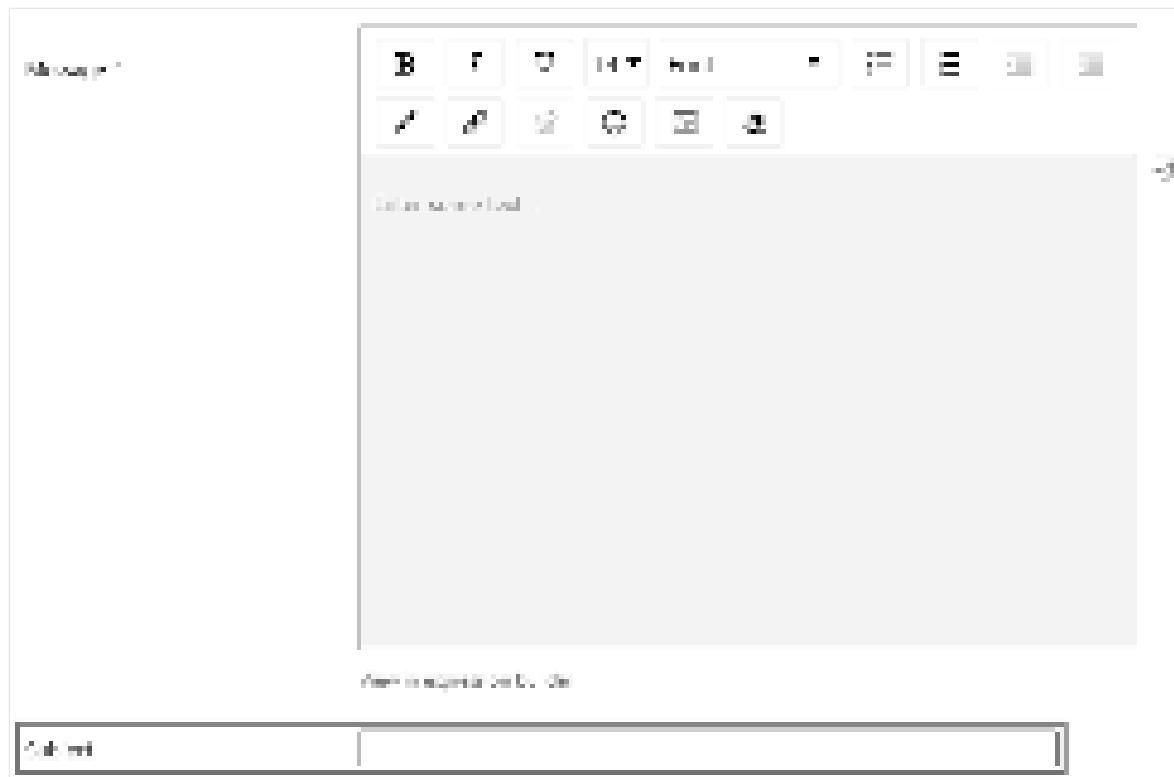
4. The **Group chat or Team** and **Channel** dropdowns appear after you select where to post. Use them to select a group chat or team and channel where you want to post

the message.

5. Use the **Message** area to create a message. Dynamic expressions are supported allowing you to incorporate any system or user variables, expressions, or functions to customize the message however necessary. To use dynamic expressions, select the **View in expression builder** link below the message area.

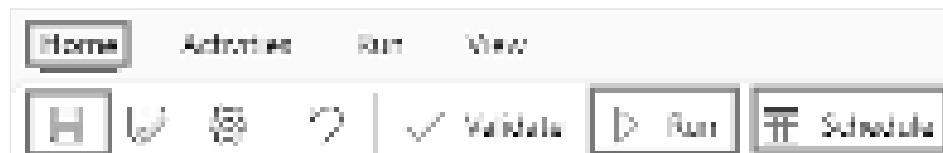


6. If you selected a channel for your post, you can also provide a subject for the message in the **Subject** text box that appears under the **Message** area. This setting is only available for messages to channels.



Save and run or schedule the pipeline

The Teams activity is typically used with other activities, often as a status notification for the outcome of prior steps in a pipeline. After you configure any other activities required for your pipeline, switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

Provide product feedback ↗ | Ask the community ↗

Use the Until activity to control execution flow

Article • 11/15/2023

The Until activity provides the same functionality that a do-until looping structure provides in programming languages. It executes a set of activities in a loop until the condition associated with the activity evaluates to true. If an inner activity fails, the Until activity doesn't stop. You can specify a timeout value for the until activity.

Prerequisites

To get started, you must complete the following prerequisites:

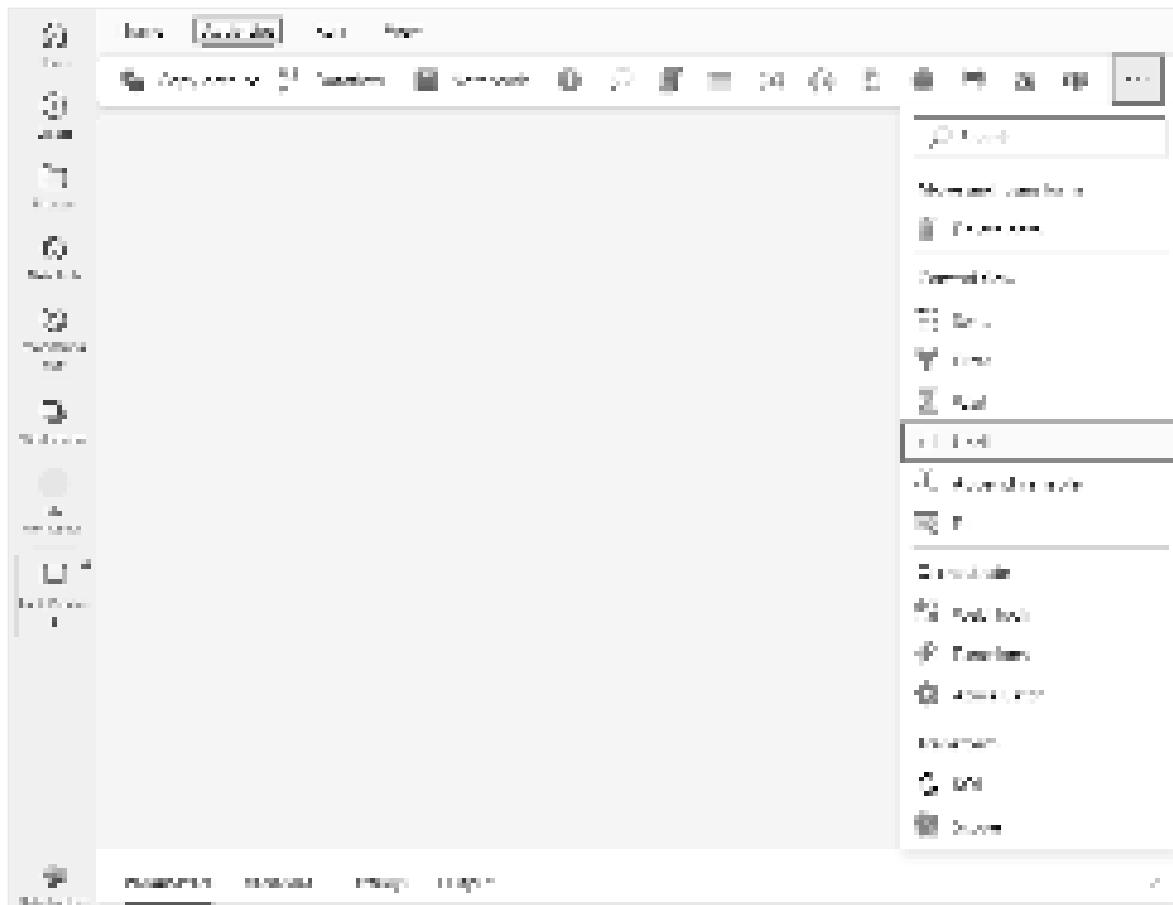
- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add an Until activity to a pipeline with UI

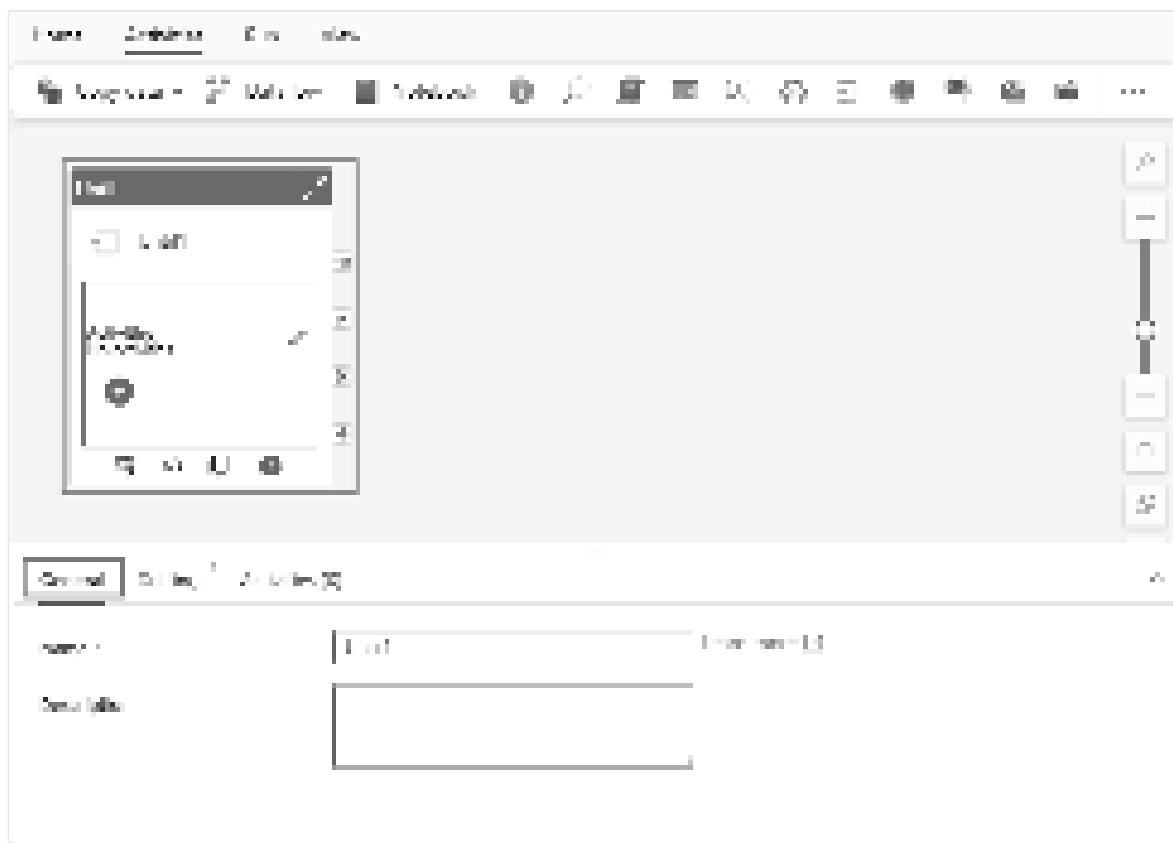
To use an Until activity in a pipeline, complete the following steps:

Create the activity

1. Create a new pipeline in your workspace.
2. Search for **Until** in the pipeline **Activities** pane, and select it to add it to the pipeline canvas.



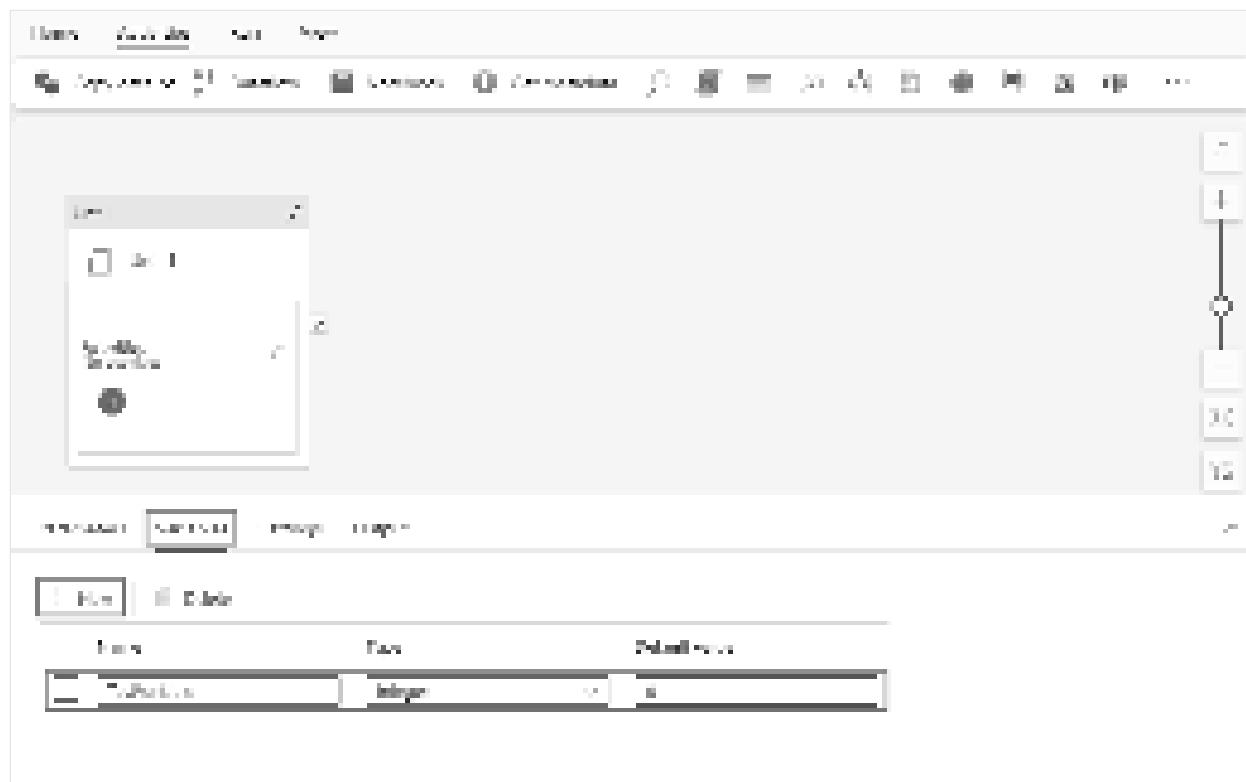
3. Select the new Until activity on the canvas if it isn't already selected.



Refer to the General settings guidance to configure the General settings tab.

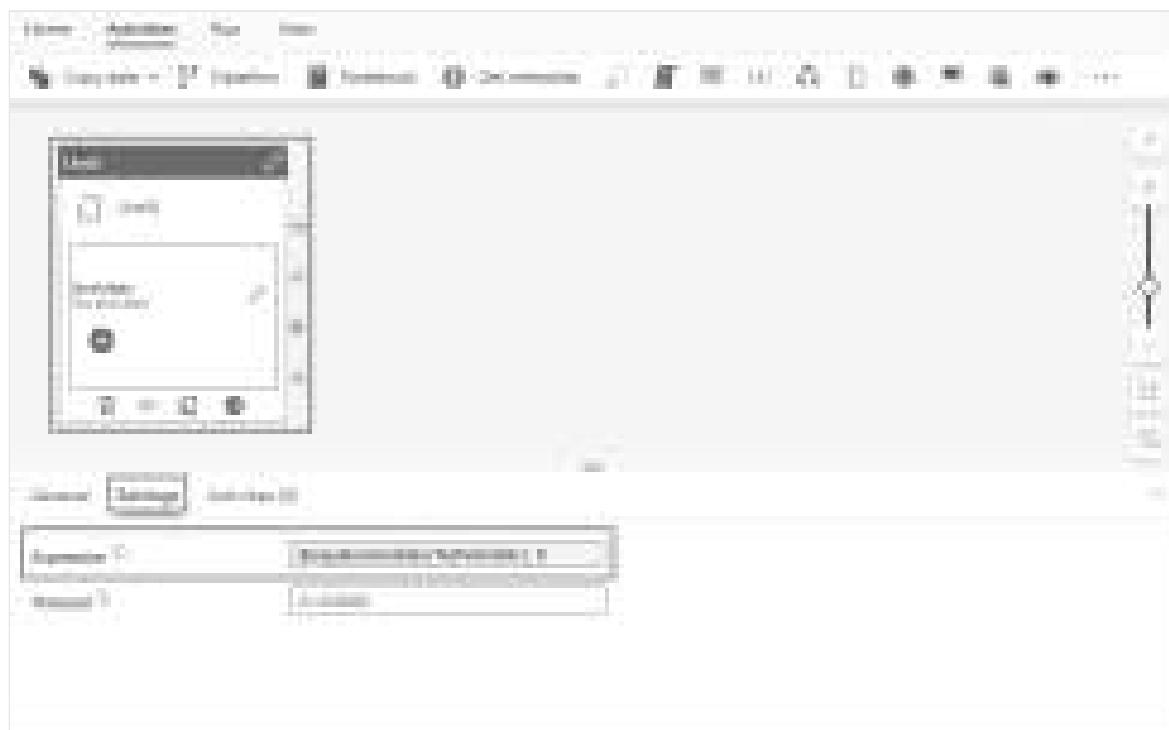
Add a pipeline variable

In this simple example, we test the value of a pipeline variable. Select the background of the pipeline editor canvas, then select the **Variables** tab, and add a new integer type variable called *TestVariable* with the value 0.

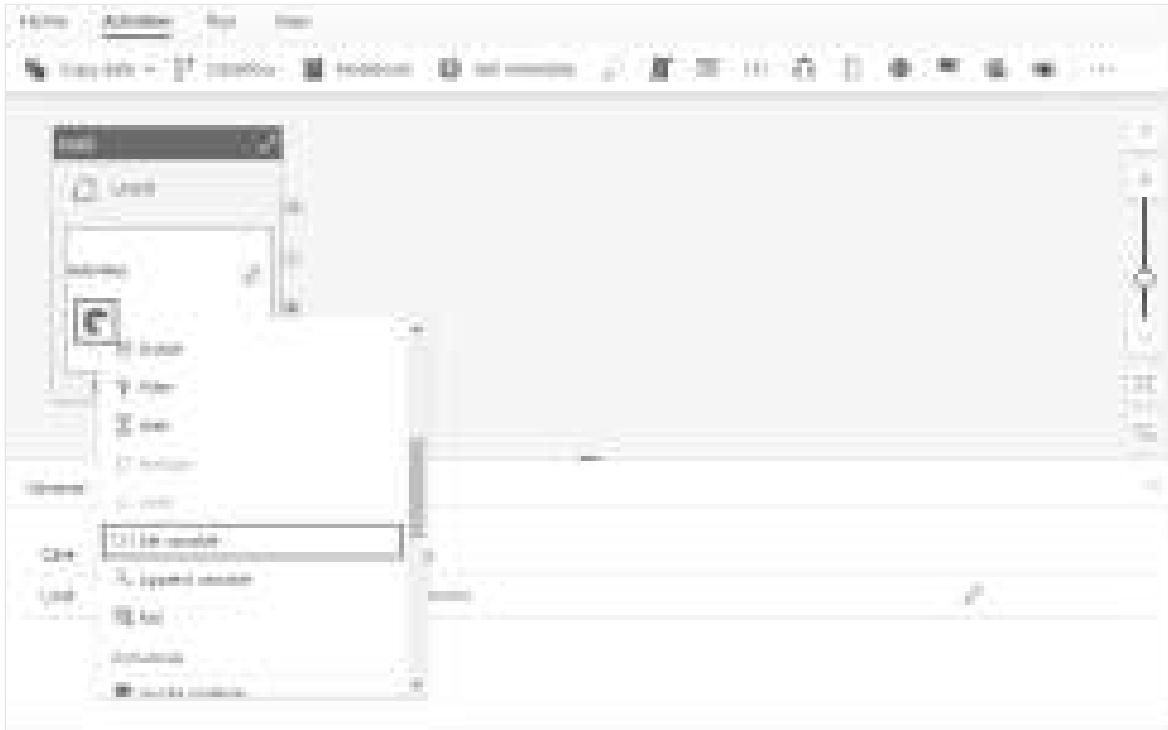


Until activity settings

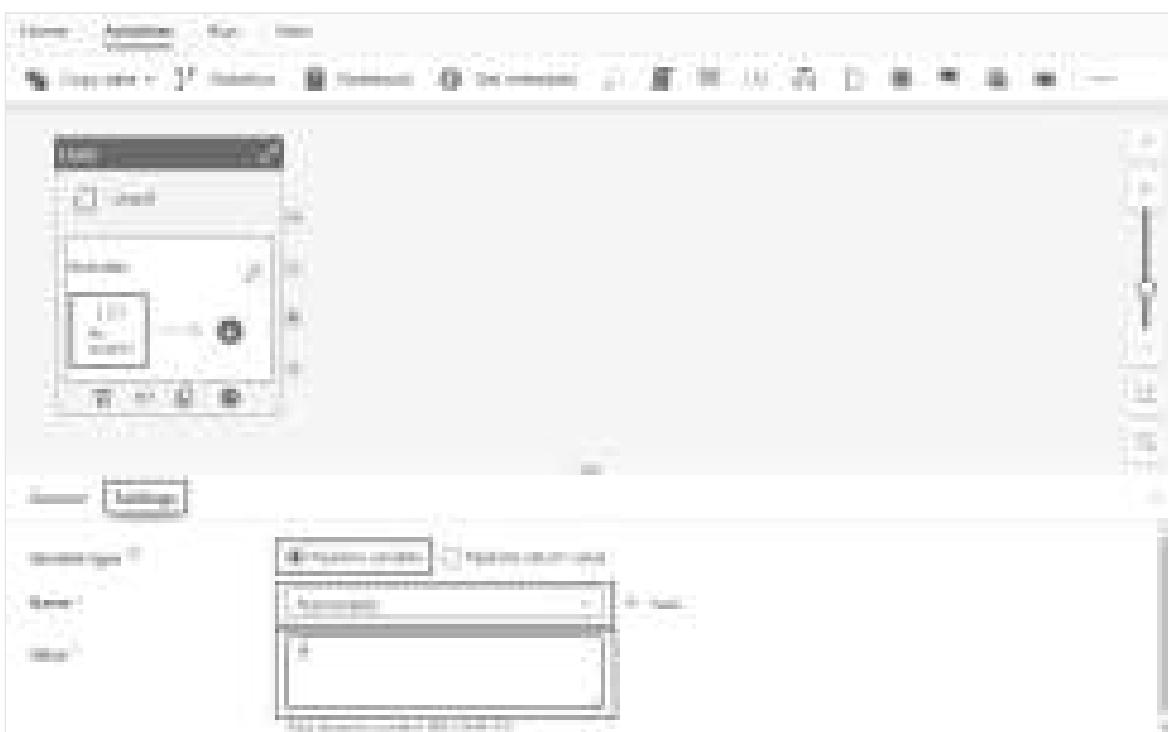
1. Select the **Until** activity again on the pipeline canvas, and then select the **Settings** tab. Select the **Expression** field, and then select **Add dynamic content**, and provide the following expression: `@equals(variables('TestVariable'), 1)`.



2. Select the **Activities** tab and then select the pencil icon to edit/add activities to the Until activity, or select the + icon on the Until activity on the pipeline editor canvas. Find the **Set Variable** activity and select it to add it to the list of child activities for the Until activity.

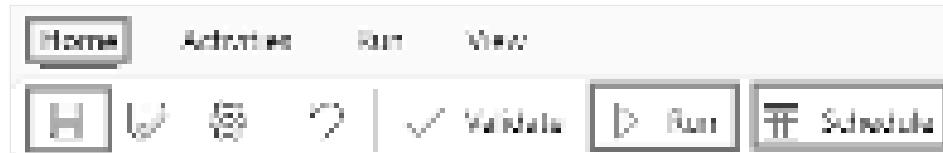


3. Select the newly added Set Variable activity from where it appears within the Until activity's **Activities** pane on the pipeline editor canvas, and then select its **Settings** tab from the activity properties pane. For **Variable type**, select **Pipeline variable**, and then choose your previously created *TestVariable* from the dropdown list. For this example, provide a **Value** of **1**.



Save and run or schedule the pipeline

Switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings. This simple pipeline will execute the child activity of the Until activity exactly 1 time, changing the pipeline variable value from 0 to 1, after which the Until expression evaluates to true and terminate.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

Yes

No

Provide product feedback [↗](#) | Ask the community [↗](#)

Use the Wait activity to control execution flow

Article • 11/15/2023

When you use a Wait activity in a pipeline, the pipeline waits for the specified period of time before continuing with execution of subsequent activities.

Prerequisites

To get started, you must complete the following prerequisites:

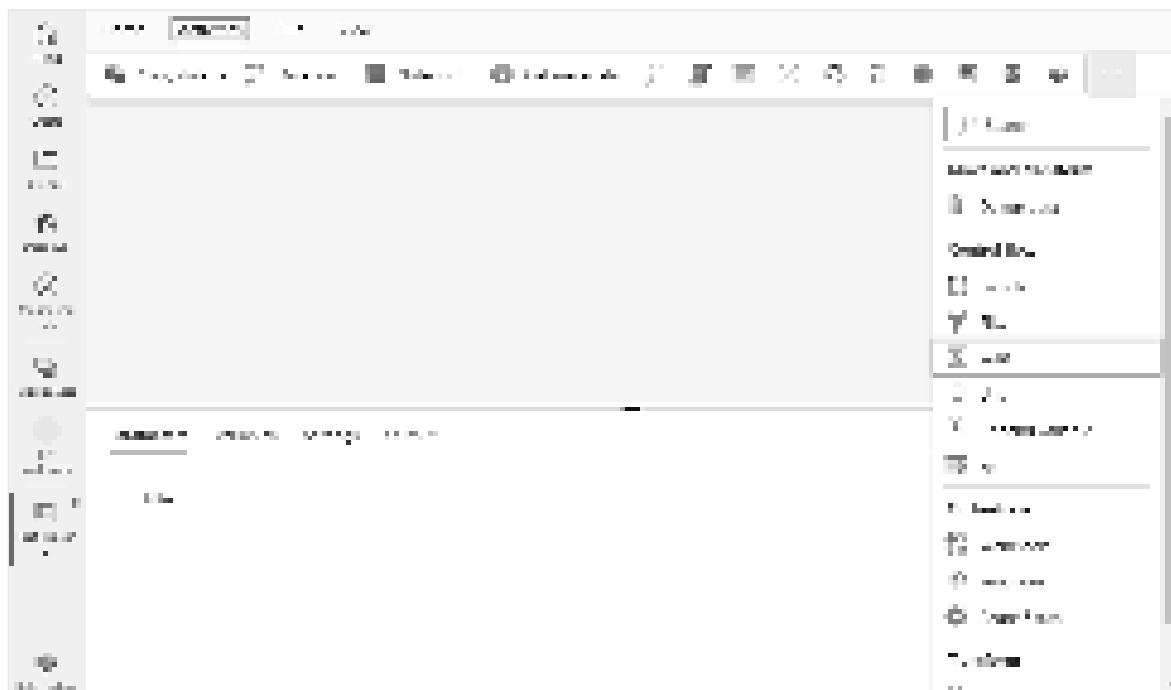
- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add a Wait activity to a pipeline with UI

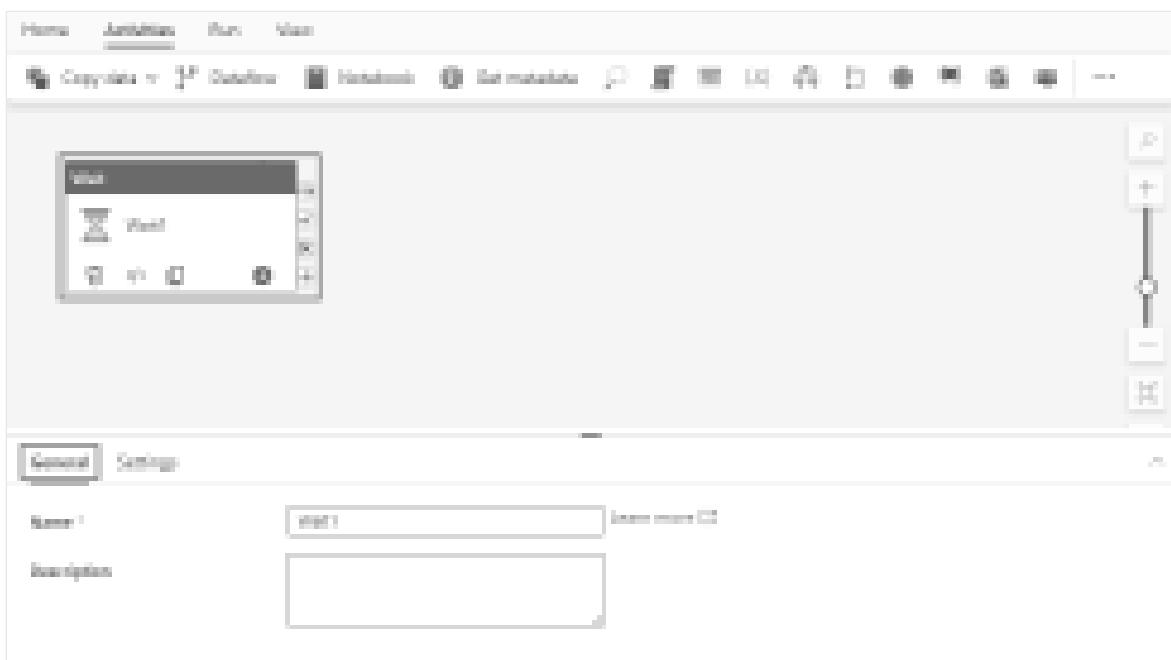
To use a Wait activity in a pipeline, complete the following steps:

Create the activity

1. Create a new pipeline in your workspace.
2. Search for **Wait** in the pipeline **Activities** pane, and select it to add it to the pipeline canvas.



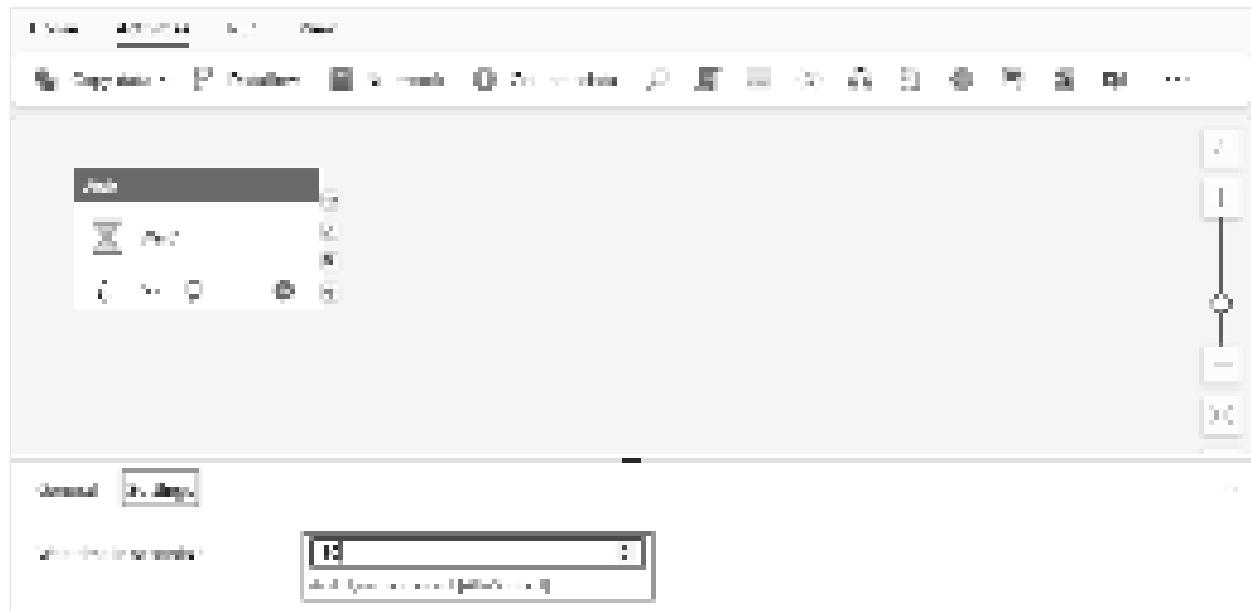
3. Select the new Wait activity on the canvas if it isn't already selected.



Refer to the **General** settings guidance to configure the **General** settings tab.

Wait activity settings

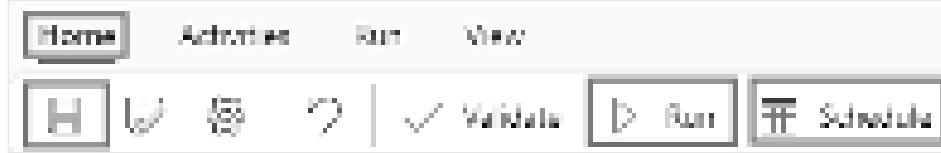
Select the **Settings** tab of the Wait activity. Specify a number of seconds for execution to wait before continuing. You can directly enter a number, or use a dynamic expression to derive a value from any of the available functions and variables for expressions.



Save and run or schedule the pipeline

This example will simply wait the specified period and terminate, although in a real-world setting, you would normally add other activities after or before the Wait activity to

achieve a more productive purpose. When your pipeline is finished, switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings. This simple pipeline will execute the child activity of the Until activity exactly 1 time, changing the pipeline variable value from 0 to 1, after which the Until expression evaluates to true and terminate.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful? 👍 Yes 👎 No

Provide product feedback ↗ | Ask the community ↗

Use the Web activity to call REST APIs in pipelines

Article • 11/15/2023

Web Activity can be used to call a custom REST endpoint from an Azure Data Factory or Synapse pipeline. You can pass datasets and linked services to be consumed and accessed by the activity.

ⓘ Note

The maximum supported output response payload size is 4 MB.

Prerequisites

To get started, you must complete the following prerequisites:

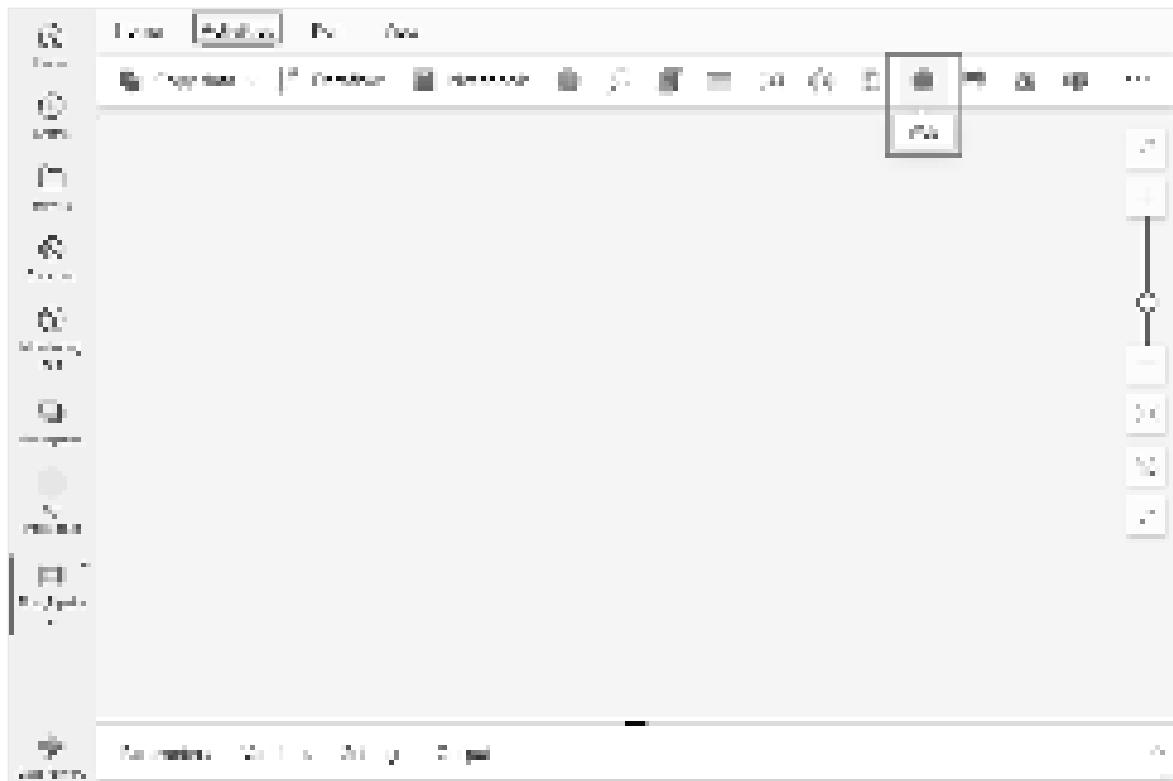
- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add a Web activity to a pipeline with UI

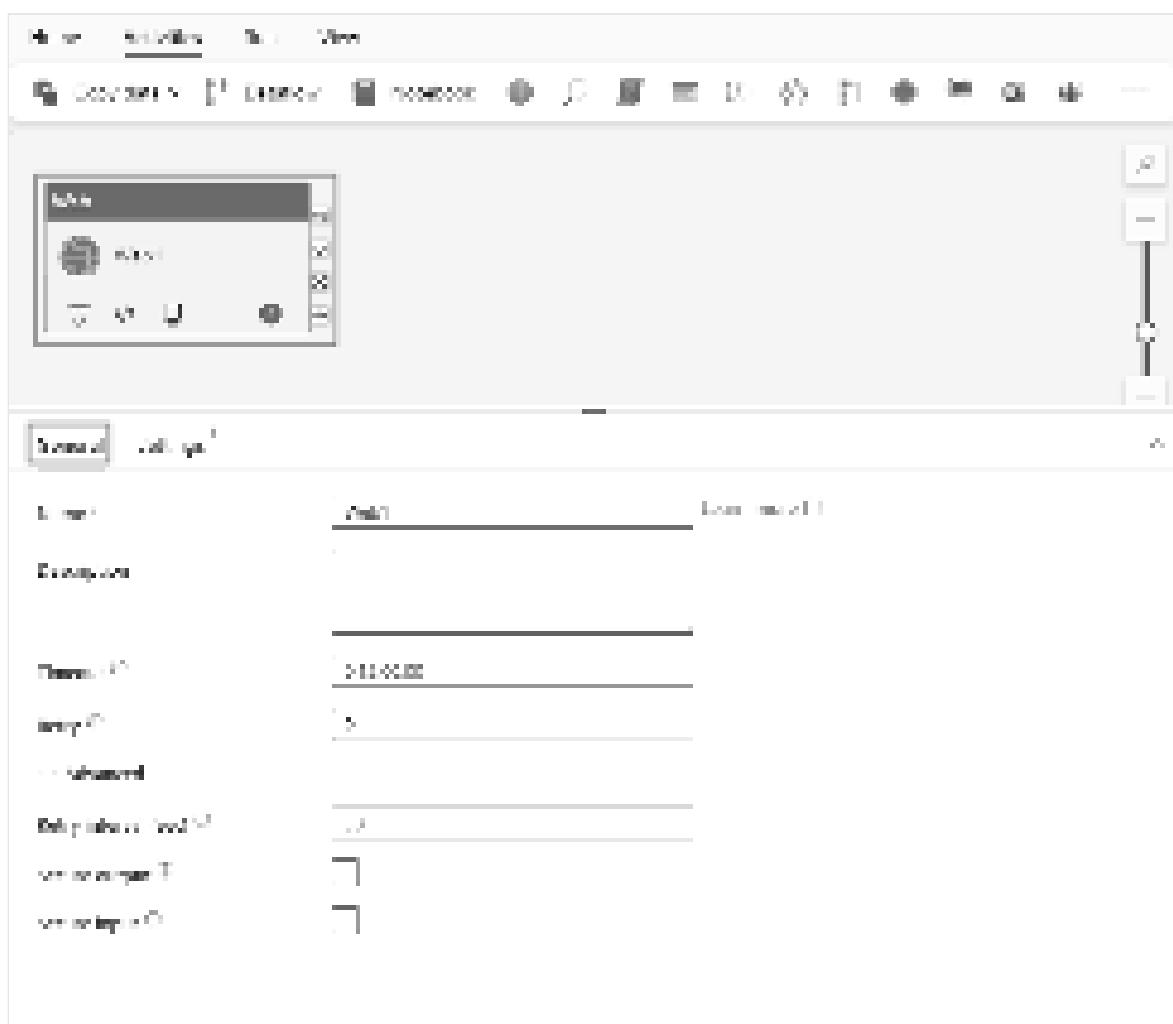
To use a Web activity in a pipeline, complete the following steps:

Create the activity

1. Create a new pipeline in your workspace.
2. Search for Web in the pipeline **Activities** pane, and select it to add it to the pipeline canvas.



3. Select the new Web activity on the canvas if it isn't already selected.



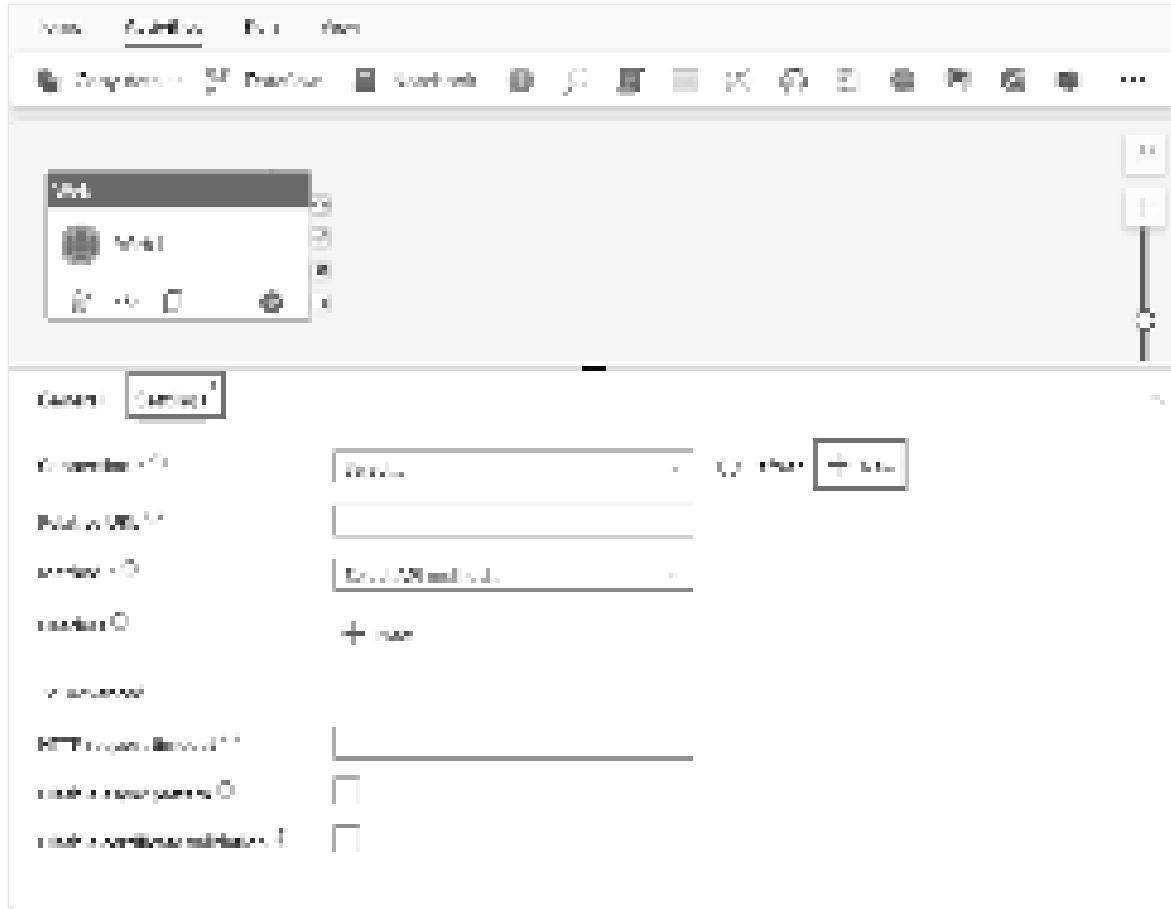
Note

The web activity may simply appear as an icon on the toolbar as shown in the previous image, if the screen resolution doesn't allow its name to be fully spelled out.

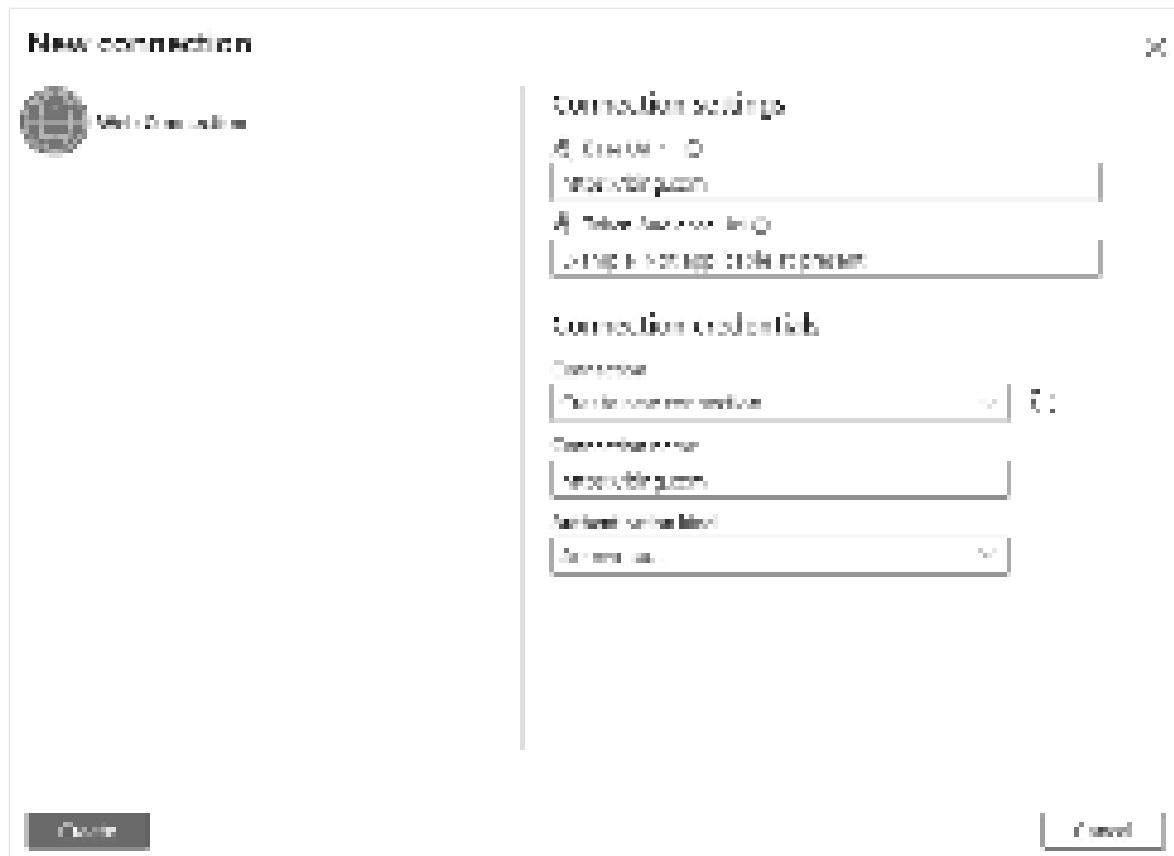
Refer to the General settings guidance to configure the **General** settings tab.

Web activity settings

1. Select the **Settings** tab, select an existing connection from the **Connection** dropdown, or use the **+ New** button to create a new connection, and specify its configuration details.



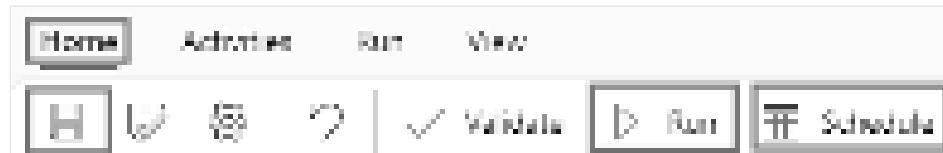
2. When you choose **+ New** to create a new connection, you see the connection creation dialog where you can provide the base URL and credentials to connect.



3. After choosing or creating your connection, complete the remaining required fields, add any required headers, or set any advanced settings. The Web activity supports GET, POST, PUT, DELETE, and PATCH methods.

Save and run or schedule the pipeline

Typically, you use the output of the Web activity with other activities, but once configured, it can be run directly without other activities, too. If you're running it to invoke a REST API that performs some action and don't require any output from the activity, your pipeline might contain only the Web activity, too. To run the activity, switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Use the WebHook activity to call an endpoint and wait for it to complete

Article • 11/15/2023

A webhook activity can control the execution of pipelines through custom code. With the webhook activity, code can call an endpoint and pass it a callback URL. The pipeline run waits for the callback invocation before it proceeds to the next activity.

Prerequisites

To get started, you must complete the following prerequisites:

- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Add a WebHook activity to a pipeline with UI

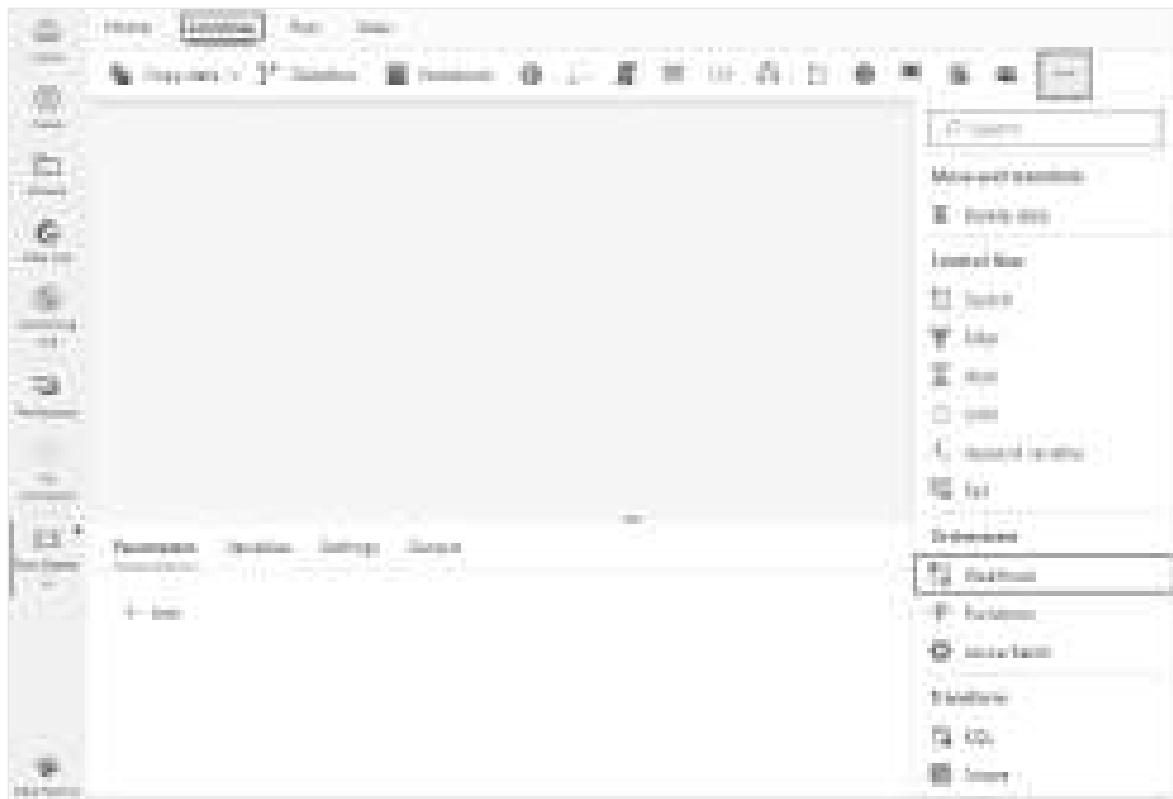
To use a WebHook activity in a pipeline, complete the following steps:

Creating the activity

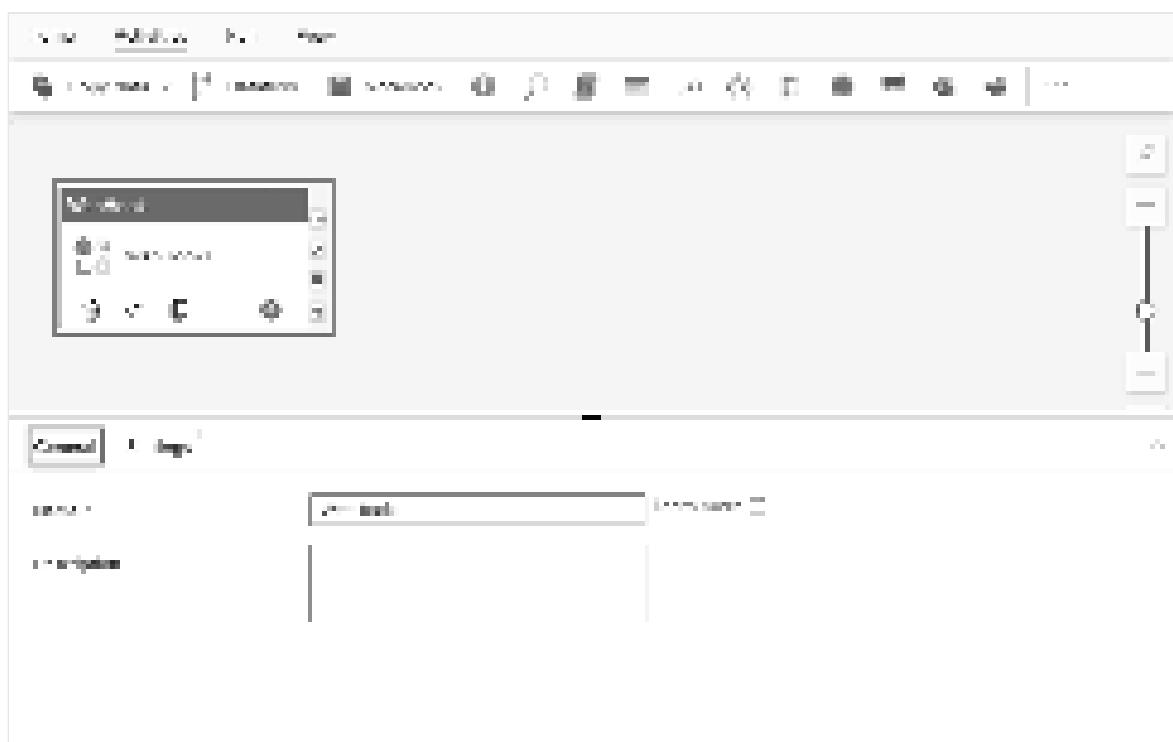
1. Create a new pipeline in your workspace.
2. Search for WebHook in the pipeline **Activities** pane, and select it to add it to the pipeline canvas.

ⓘ Note

Unless your screen has a very high resolution, you likely need to expand the list of activities from the toolbar using the ellipsis ... button to find the WebHook activity.



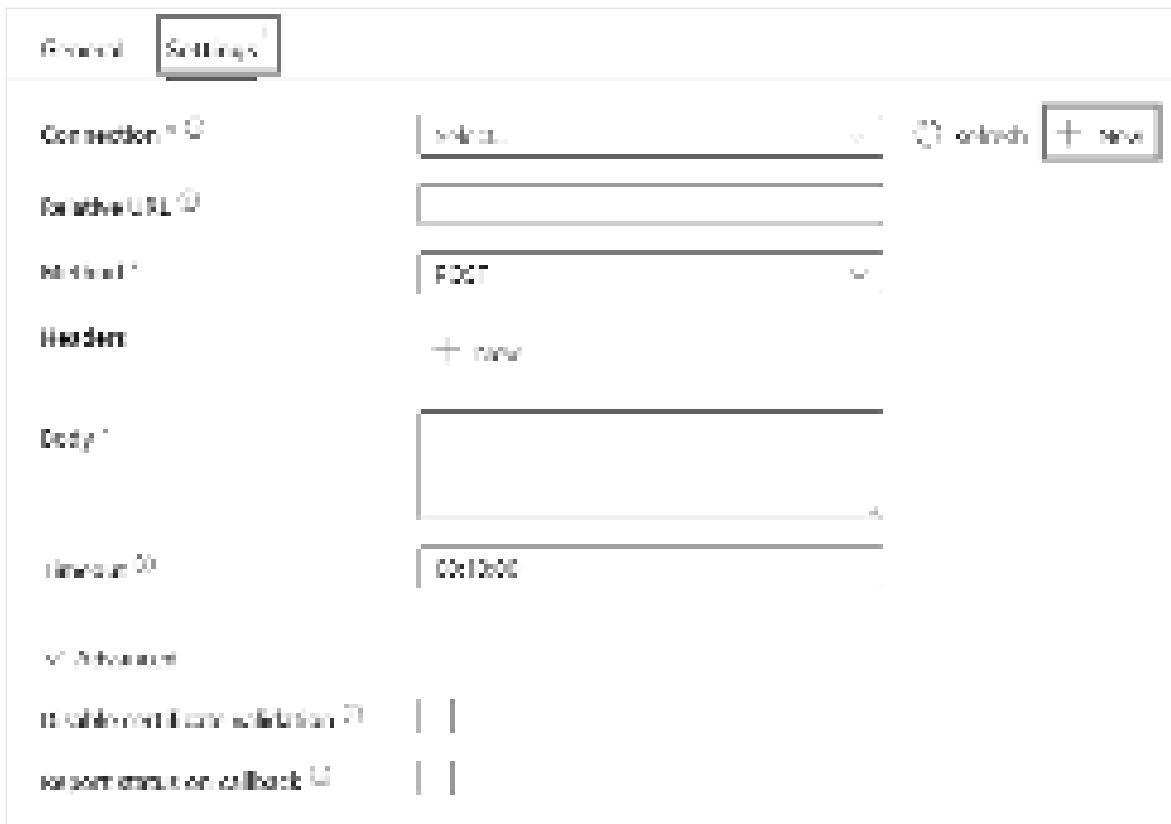
3. Select the new WebHook activity on the canvas if it isn't already selected.



Refer to the General settings guidance to configure the General settings tab.

Web activity settings

1. Select the **Settings** tab, select an existing connection from the **Connection** dropdown, or use the **+ New** button to create a new connection, and specify its configuration details.



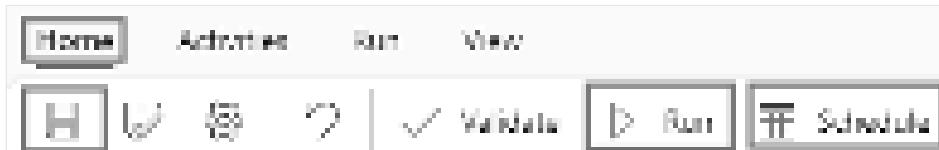
- When you choose **+ New** to create a new connection, you see the connection creation dialog where you can provide the base URL and credentials to connect.



- After choosing or creating your connection, complete the remaining required fields, add any required headers, or set any advanced settings. The WebHook activity only supports the POST method.
- Use the output from the activity as the input to any other activity, and reference the output anywhere dynamic content is supported in the destination activity.

Save and run or schedule the pipeline

Typically, you use the output of the WebHook activity with other activities, but once configured, it can be run directly without other activities, too. If you're running it to invoke a REST API that performs some action and you don't require any output from the activity, your pipeline might contain only the Web activity, too. To run the activity, switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

Yes

No

Provide product feedback [↗](#) | Ask the community [↗](#)

How to monitor data pipeline runs in Microsoft Fabric

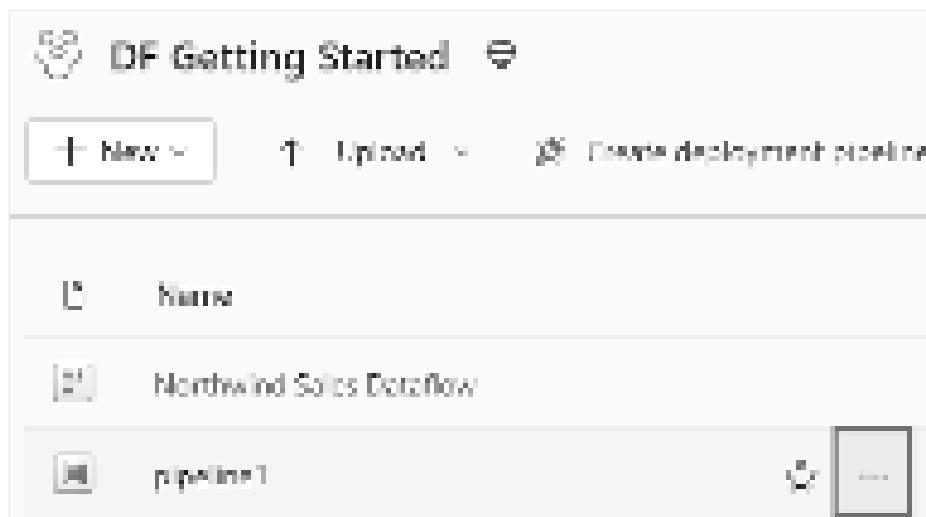
Article • 11/15/2023

In this how-to guide, you'll learn the different ways to review and monitor your pipeline runs.

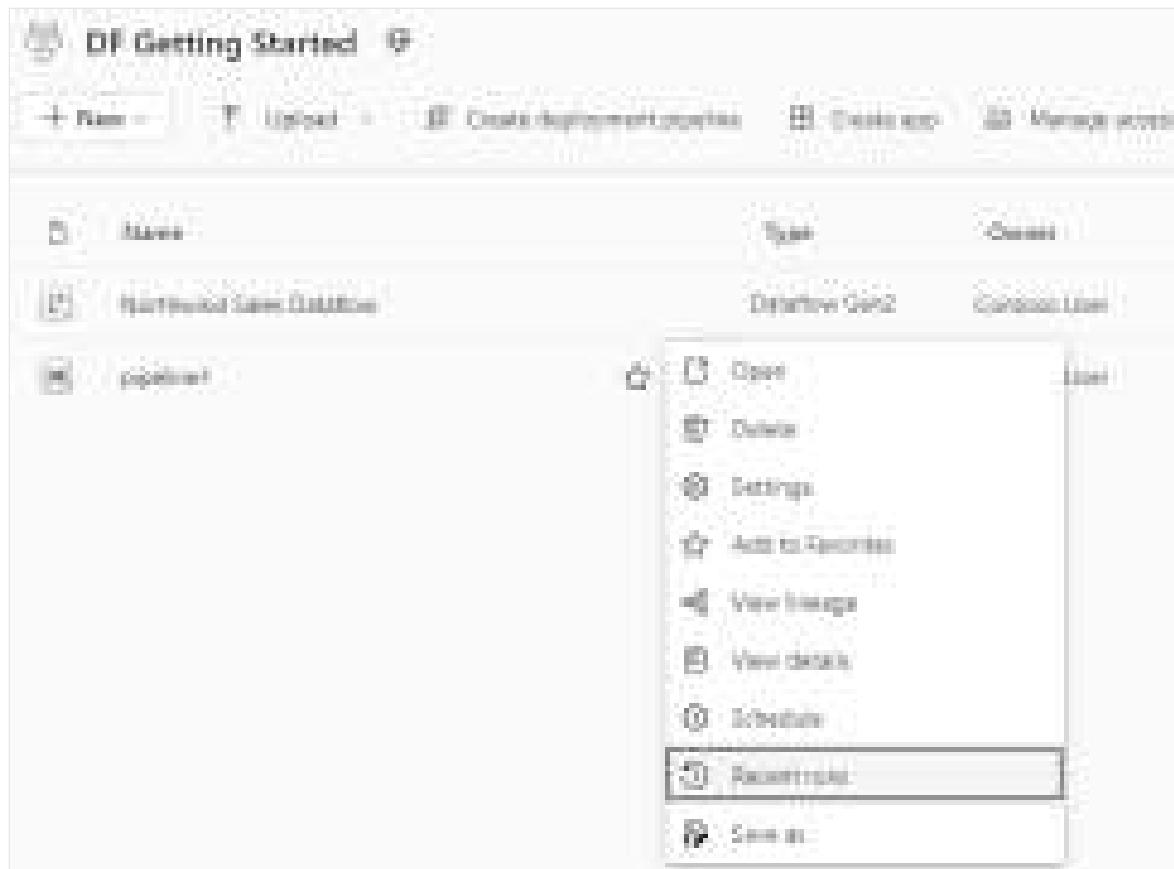
Monitor data pipeline runs

1. To monitor your data pipeline runs, hover over your pipeline in your workspace.

Doing so will bring up three dots to the right of your pipeline name.



2. Select the three dots to find a list of options. Then select **View run history**. This action opens a fly-out on the right side of your screen with all your recent runs and run statuses.



My Lambda Functions		Execution results for Pipeline			
	Name	Last Run	Duration	State	Logs
0	MyLambdaFunction	1 hour ago	00:00:00.000	Success	
1	MyLambdaFunction2	1 hour ago	00:00:00.000	Success	
2	MyLambdaFunction3	1 hour ago	00:00:00.000	Success	
3	MyLambdaFunction4	1 hour ago	00:00:00.000	Success	
4	MyLambdaFunction5	1 hour ago	00:00:00.000	Success	

3. Select **Go to monitoring hub** from the prior screenshot to view more details and filter results. Use the filter to find specific data pipeline runs based on several criteria.

Monitoring Hub					
Search for the name of the pipeline or run ID, or use the dropdown to select a specific run.					
Filter	Actions	Run ID	Run Type	Run Status	Run Details
Run ID	Run ID	11111111111111111111111111111111	Batch	Success	
Run ID	Run ID	22222222222222222222222222222222	Batch	Success	
Run ID	Run ID	33333333333333333333333333333333	Batch	Success	
Run ID	Run ID	44444444444444444444444444444444	Batch	Success	
Run ID	Run ID	55555555555555555555555555555555	Batch	Success	

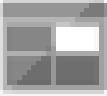
4. Select one of your pipeline runs to view detailed information. You'll be able to view what your pipeline looks like and view more properties like Run ID or errors if your pipeline run failed.



5. To find additional information on your pipeline runs **Input** and **Output**, select the input or output links to the right of the relevant row in the Activity Runs.
6. You can select **Update pipeline** to make changes to your pipeline from this screen. This selection will take you back in the pipeline canvas.
7. You can also **Rerun** your data pipeline. You can choose to rerun the entire pipeline or only rerun the pipeline from the failed activity.
8. To view performance details, select an activity from the list of **Activity Runs**. Performance details will pop up.

Copy data details

Copy detail

Source	Destination
 Azure Blob Storage	 Tabular
Data read: 11	Data written: 11
Files read: 10	File写入数: 1
Blocks read: 93,557	块写入数: 6,023

Status:  Succeeded

Start time: 10/27/2023, 3:21:34 PM

Close

More details can be found under **Duration breakdown** and **Advanced**.

Copy data details

 Azure Blob Storage →  Lakehouse

Data read:	504,515 KB	Data written:	2,053,011 KB
File read:	1	File written:	1
Rows read:	69,557	Rows written:	69,557

Status:	Success
Start time:	10/27/2023, 02:11:14 PM

Activity run ID:	09e794D3-e147-4f6c-95ca-041E0d924C09
Throughput:	65,765 KB/s
Total duration:	00:00:15

Duration breakdown

Start time:	10/27/2023, 02:11:14 PM
Optimized throughput:	Standard
Used parallel copies:	1

Copy: 

Transfer: 

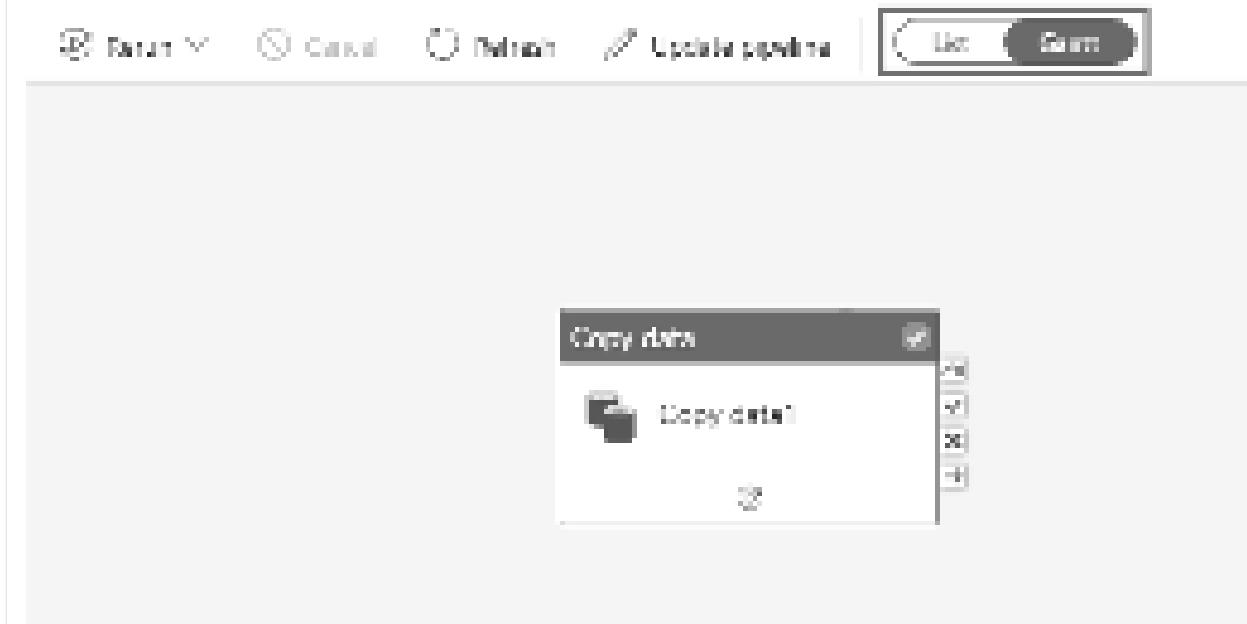
> Advanced

Close

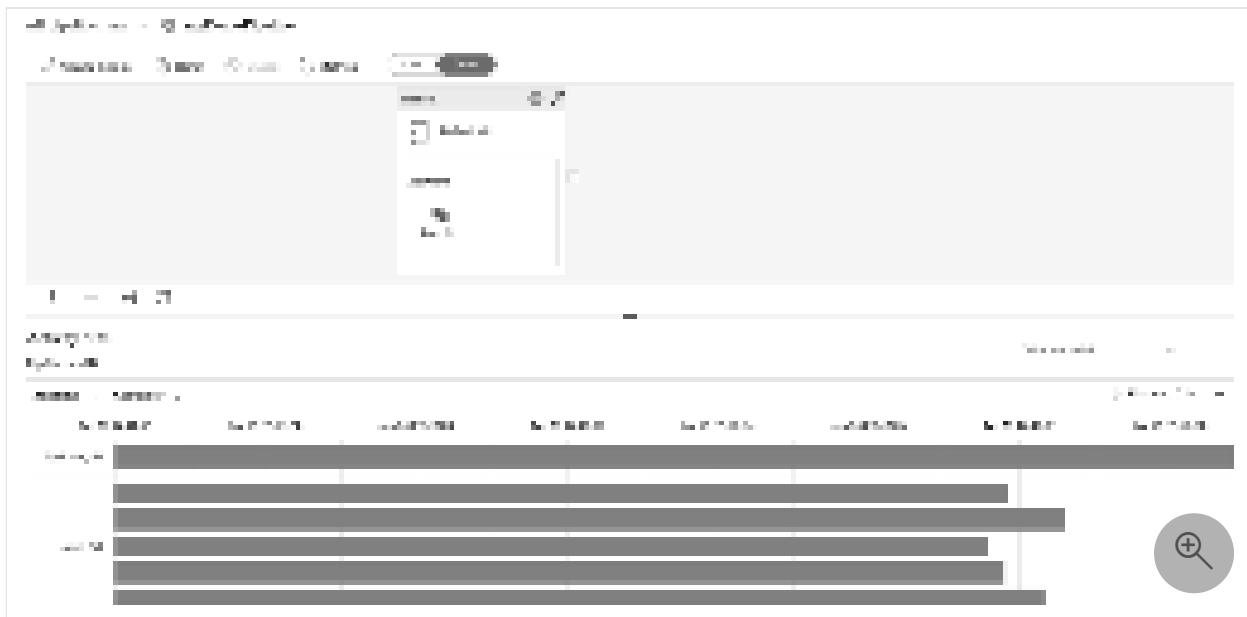
Gantt view

A Gantt chart is a view that lets you see the run history over a time range. If you switch to a Gantt view, all pipeline runs will be grouped by name, displayed as bars relative to how long the run took.

Monitoring hub > MyDemoPipeline



The length of the bar relates to the duration of the pipeline. You can select the bar to view more details.

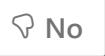


Next steps

- Quickstart: Create your first data pipeline to copy data
 - Quickstart: Create your first Dataflow Gen2 to get and transform data
-

Feedback

Was this page helpful?



Provide product feedback [↗](#) | Ask the community [↗](#)

Parameters for Data Factory in Microsoft Fabric

Article • 11/15/2023

This document describes how to use parameters in your pipelines for Data Factory in Fabric.

How to use parameters, expressions and functions in pipelines for Data Factory in Fabric

In this document, we focus on learning fundamental concepts with various examples to explore the ability to create parameterized data pipelines within Data Factory in Fabric. Parameterization and dynamic expressions can save a tremendous amount of time and allow for a much more flexible Extract, Transform, Load (ETL) or Extract, Load, Transform (ELT) solution, which will dramatically reduce the cost of solution maintenance and speed up the implementation of new features into existing pipelines. These gains are because parameterization minimizes the amount of hard coding and increases the number of reusable objects and processes in a solution.

Parameter and expression concepts

You can use parameters to pass external values into pipelines. Once the parameter has been passed into the resource, it cannot be changed. By parameterizing resources, you can reuse them with different values each time. Parameters can be used individually or as a part of expressions. Parameter values in the definition can be literal or expressions that are evaluated at runtime.

Expressions can appear anywhere in a string value and always generate another string value. Here, password is a pipeline parameter in the expression. If a parameter value is an expression, the body of the expression is extracted by removing the at-sign (@). If a literal string is needed that starts with @, it must be escaped by using @@. The following examples show how expressions are evaluated.

Parameter value	Result
"parameters"	The characters 'parameters' are returned.
"parameters[1]"	The characters 'parameters[1]' are returned.

Parameter value	Result
"@@"	A 1 character string that contains '@' is returned.
" @"	A 2 character string that contains ' @' is returned.

Expressions can also appear inside strings, using a feature called *string interpolation* where expressions are wrapped in `@{ ... }`. For example, the following string includes parameter values and literal string values:

```
"First Name: {@pipeline().parameters.firstName} Last Name:  
{@pipeline().parameters.lastName}"
```

Using string interpolation, the result is always a string. For example, if you defined `myNumber` as `42` and `myString` as `foo`:

Parameter value	Result
"@pipeline().parameters.myString"	Returns <code>foo</code> as a string.
"@{pipeline().parameters.myString}"	Returns <code>foo</code> as a string.
"@pipeline().parameters.myNumber"	Returns <code>42</code> as a <i>number</i> .
"@{pipeline().parameters.myNumber}"	Returns <code>42</code> as a <i>string</i> .
"Answer is: {@pipeline().parameters.myNumber}"	Returns the string <code>Answer is: 42</code> .
"@concat('Answer is:', string(pipeline().parameters.myNumber))"	Returns the string <code>Answer is: 42</code>
"Answer is: @@{@pipeline().parameters.myNumber}"	Returns the string <code>Answer is: {@pipeline().parameters.myNumber}</code> .

Examples of using parameters in expressions

Creating and using parameters

To create parameters, select the background of the pipeline editor canvas, and then the **Parameters** tab of the properties window at the bottom. Select the **+ New** button to add a new parameter to the pipeline, give it a name, a data type, and a default value:

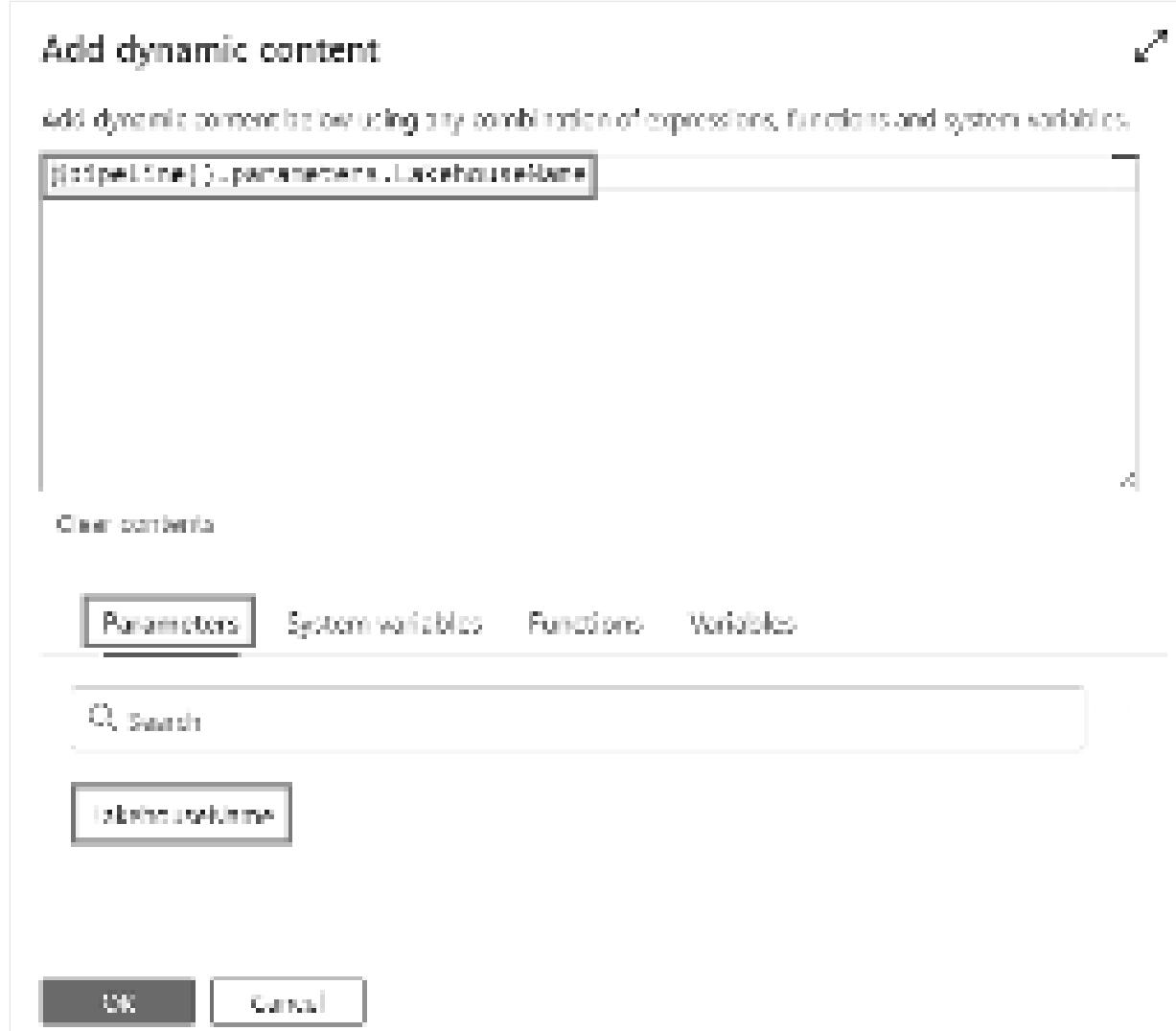
Parameters			
	Name	Type	Default Value
<input type="checkbox"/>	WarehouseName	String	WarehouseA

You can then use the parameter anywhere in your pipeline where dynamic content is supported. In this example, the parameter is used to dynamically provide the name of a Lakehouse data store on the **Source** tab of a copy activity's property pages.

The screenshot shows the Azure Data Factory studio interface. On the left is a navigation sidebar with icons for Home, Activities, Pipelines, Views, Datasets, Variables, Copy activities, and Run history. The main area has a toolbar with Home, Activities, Pipelines, Views, and a search bar. A central workspace contains a 'Copy' activity named 'My Copy Activity'. Below it, there are tabs for General, Source, Sink, Mapping, and Settings. The 'Source' tab is currently active. Under 'Source type', 'Lakehouse' is selected. In the 'Dataset' dropdown, 'WarehouseA' is highlighted. At the bottom of the dropdown menu, there is a button labeled 'Add dynamic content...'. The status bar at the bottom shows '1 item(s) found'.

The **Add dynamic content** window is displayed, allowing you to specify any kind of dynamic content, including parameters, system variables, functions, or pipeline variables. In this example, the previously defined parameter is selected, and the dynamic content

window is automatically populated with the correct expression to reference the parameter.



Complex expression example

The below example shows a complex example that references a deep sub-field of activity output. To reference a pipeline parameter that evaluates to a sub-field, use `[]` syntax instead of dot(.) operator (as in case of subfield1 and subfield2)

```
@activity('*activityName*').output.*subfield1*.*subfield2*  
[pipeline().parameters.*subfield3*].*subfield4*
```

Dynamic content editor

The dynamic content editor automatically escapes characters in your content when you finish editing. For example, the following content in the content editor is a string interpolation with an expression function:

```
@{toUpperCase('myData')}
```

The dynamic content editor converts the above content to the following expression:

MYDATA

Using functions and variables in expressions

You can call functions and use variables within expressions. The following sections provide information about the functions that can be used in an expression.

Pipeline scope variables

These system variables can be referenced anywhere in the pipeline JSON.

Variable Name	Description
@pipeline().DataFactory	Name of the data or Synapse workspace the pipeline run is running in
@pipeline().Pipeline	Name of the pipeline
@pipeline().RunId	ID of the specific pipeline run
@pipeline().TriggerId	ID of the trigger that invoked the pipeline
@pipeline().TriggerName	Name of the trigger that invoked the pipeline
@pipeline().TriggerTime	Time of the trigger run that invoked the pipeline. This is the time at which the trigger actually fired to invoke the pipeline run, and it may differ slightly from the trigger's scheduled time.
@pipeline().GroupId	ID of the group to which pipeline run belongs.
@pipeline()?.TriggeredByPipelineName	Name of the pipeline that triggers the pipeline run. Applicable when the pipeline run is triggered by an ExecutePipeline activity. Evaluate to <i>Null</i> when used in other circumstances. Note the question mark after @pipeline()
@pipeline()?.TriggeredByPipelineRunId	Run ID of the pipeline that triggers the pipeline run. Applicable when the pipeline run is triggered by an ExecutePipeline activity. Evaluate to <i>Null</i> when used in other circumstances. Note the question mark after @pipeline()

 **Note**

Trigger-related date/time system variables (in both pipeline and trigger scopes)
return UTC dates in ISO 8601 format, for example, 2017-06-01T22:20:00.4061448Z.

String functions

To work with strings, you can use these string functions and also some collection functions. String functions work only on strings.

String function	Task
concat	Combine two or more strings, and return the combined string.
endsWith	Check whether a string ends with the specified substring.
guid	Generate a globally unique identifier (GUID) as a string.
indexOf	Return the starting position for a substring.
lastIndexOf	Return the starting position for the last occurrence of a substring.
replace	Replace a substring with the specified string, and return the updated string.
split	Return an array that contains substrings, separated by commas, from a larger string based on a specified delimiter character in the original string.
startsWith	Check whether a string starts with a specific substring.
substring	Return characters from a string, starting from the specified position.
toLowerCase	Return a string in lowercase format.
toUpperCase	Return a string in uppercase format.
trim	Remove leading and trailing whitespace from a string, and return the updated string.

Collection functions

To work with collections, generally arrays, strings, and sometimes, dictionaries, you can use these collection functions.

Collection function	Task
contains	Check whether a collection has a specific item.

Collection function	Task
empty	Check whether a collection is empty.
first	Return the first item from a collection.
intersection	Return a collection that has <i>only</i> the common items across the specified collections.
join	Return a string that has <i>all</i> the items from an array, separated by the specified character.
last	Return the last item from a collection.
length	Return the number of items in a string or array.
skip	Remove items from the front of a collection, and return <i>all the other</i> items.
take	Return items from the front of a collection.
union	Return a collection that has <i>all</i> the items from the specified collections.

Logical functions

These functions are useful inside conditions, they can be used to evaluate any type of logic.

Logical comparison function	Task
and	Check whether all expressions are true.
equals	Check whether both values are equivalent.
greater	Check whether the first value is greater than the second value.
greaterOrEquals	Check whether the first value is greater than or equal to the second value.
if	Check whether an expression is true or false. Based on the result, return a specified value.
less	Check whether the first value is less than the second value.
lessOrEquals	Check whether the first value is less than or equal to the second value.
not	Check whether an expression is false.

Logical comparison function	Task
or	Check whether at least one expression is true.

Conversion functions

These functions are used to convert between each of the native types in the language:

- string
- integer
- float
- boolean
- arrays
- dictionaries

Conversion function	Task
array	Return an array from a single specified input. For multiple inputs, see <code>createArray</code> .
base64	Return the base64-encoded version for a string.
base64ToBinary	Return the binary version for a base64-encoded string.
base64ToString	Return the string version for a base64-encoded string.
binary	Return the binary version for an input value.
bool	Return the Boolean version for an input value.
coalesce	Return the first non-null value from one or more parameters.
createArray	Return an array from multiple inputs.
dataUri	Return the data URI for an input value.
dataUriToBinary	Return the binary version for a data URI.
dataUriToString	Return the string version for a data URI.
decodeBase64	Return the string version for a base64-encoded string.
decodeDataUri	Return the binary version for a data URI.
decodeUriComponent	Return a string that replaces escape characters with decoded versions.
encodeUriComponent	Return a string that replaces URL-unsafe characters with escape characters.

Conversion function	Task
float	Return a floating point number for an input value.
int	Return the integer version for a string.
json	Return the JavaScript Object Notation (JSON) type value or object for a string or XML.
string	Return the string version for an input value.
uriComponent	Return the URI-encoded version for an input value by replacing URL-unsafe characters with escape characters.
uriComponentToBinary	Return the binary version for a URI-encoded string.
uriComponentToString	Return the string version for a URI-encoded string.
xml	Return the XML version for a string.
xpath	Check XML for nodes or values that match an XPath (XML Path Language) expression, and return the matching nodes or values.

Math functions

These functions can be used for either types of numbers: **integers** and **floats**.

Math function	Task
add	Return the result from adding two numbers.
div	Return the result from dividing two numbers.
max	Return the highest value from a set of numbers or an array.
min	Return the lowest value from a set of numbers or an array.
mod	Return the remainder from dividing two numbers.
mul	Return the product from multiplying two numbers.
rand	Return a random integer from a specified range.
range	Return an integer array that starts from a specified integer.
sub	Return the result from subtracting the second number from the first number.

Date functions

Date or time function	Task
addDays	Add a number of days to a timestamp.
addHours	Add a number of hours to a timestamp.
addMinutes	Add a number of minutes to a timestamp.
addSeconds	Add a number of seconds to a timestamp.
addToTime	Add a number of time units to a timestamp. See also getFutureTime .
convertFromUtc	Convert a timestamp from Universal Time Coordinated (UTC) to the target time zone.
convertTimeZone	Convert a timestamp from the source time zone to the target time zone.
convertToUtc	Convert a timestamp from the source time zone to Universal Time Coordinated (UTC).
dayOfMonth	Return the day of the month component from a timestamp.
dayOfWeek	Return the day of the week component from a timestamp.
dayOfYear	Return the day of the year component from a timestamp.
formatDateTime	Return the timestamp as a string in optional format.
getFutureTime	Return the current timestamp plus the specified time units. See also addToTime .
getPastTime	Return the current timestamp minus the specified time units. See also subtractFromTime .
startOfDay	Return the start of the day for a timestamp.
startOfHour	Return the start of the hour for a timestamp.
startOfMonth	Return the start of the month for a timestamp.
subtractFromTime	Subtract a number of time units from a timestamp. See also getPastTime .
ticks	Return the <code>ticks</code> property value for a specified timestamp.
utcNow	Return the current timestamp as a string.

Next steps

[Expression language](#)

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Expressions and functions for Data Factory in Microsoft Fabric

Article • 11/15/2023

This article provides details about expressions and functions supported by Data Factory in Microsoft Fabric.

Expressions

Expression values in the definition can be literal or expressions that are evaluated at runtime. For example:

"value"

or

"@pipeline().parameters.password"

Expressions can appear anywhere in a string value and always result in another string value. If a value is an expression, the body of the expression is extracted by removing the at-sign (@). If a literal string is needed that starts with @, it must be escaped by using @@. The following examples show how expressions are evaluated.

Expression value	Result
"parameters"	The characters 'parameters' are returned.
"parameters[1]"	The characters 'parameters[1]' are returned.
"@@"	A 1 character string that contains '@' is returned.
" @"	A 2 character string that contains ' @' is returned.

Expressions can also appear inside strings, using a feature called *string interpolation* where expressions are wrapped in @{ ... }. For example: "First Name:

@{pipeline().parameters.firstName} Last Name: @{pipeline().parameters.lastName}"

Using string interpolation, the result is always a string. Say I have defined `myNumber` as 42 and `myString` as foo:

Expression value	Result
"@pipeline().parameters.myString"	Returns foo as a string.

Expression value	Result
<code>@{pipeline().parameters.myString}"</code>	Returns <code>foo</code> as a string.
<code>"@pipeline().parameters.myNumber"</code>	Returns <code>42</code> as a <i>number</i> .
<code>"@{pipeline().parameters.myNumber}"</code>	Returns <code>42</code> as a <i>string</i> .
<code>"Answer is: @{pipeline().parameters.myNumber}"</code>	Returns the string <code>Answer is: 42</code> .
<code>"@concat('Answer is: ', string(pipeline().parameters.myNumber))"</code>	Returns the string <code>Answer is: 42</code>
<code>"Answer is: @{pipeline().parameters.myNumber}"</code>	Returns the string <code>Answer is: @{pipeline().parameters.myNumber}</code> .

In the control flow activities like ForEach activity, you can provide an array to be iterated over for the property items and use `@item()` to iterate over a single enumeration in ForEach activity. For example, if items is an array: [1, 2, 3], `@item()` returns 1 in the first iteration, 2 in the second iteration, and 3 in the third iteration. You can also use `@range(0,10)` like expression to iterate 10 times starting with 0 ending with 9.

You can use `@activity('activity name')` to capture output of activity and make decisions. Consider a web activity called Web1. For placing the output of the first activity in the body of the second, the expression generally looks like: `@activity('Web1').output` or `@activity('Web1').output.data` or something similar depending upon what the output of the first activity looks like.

Examples

Complex expression example

The below example shows a complex example that references a deep sub-field of activity output. To reference a pipeline parameter that evaluates to a sub-field, use `[]` syntax instead of dot(.) operator (as in case of subfield1 and subfield2), as part of an activity output.

```
@activity('*activityName*').output.*subfield1*.*subfield2*
[pipeline().parameters.*subfield3*].*subfield4*
```

Creating files dynamically and naming them is common pattern. Let us explore few dynamic file naming examples.

- Append Date to a filename: `@concat('Test_', formatDateTime.UtcNow(), 'yyyy-dd-MM'))`

- Append DateTime in customer timezone: `@concat('Test_', convertFromUtc(utcnow(), 'Pacific Standard Time'))`
- Append Trigger Time: `@concat('Test_', pipeline().TriggerTime)`
- Output a custom filename in a Mapping Data Flow when outputting to a single file with date: `'Test_' + toString(currentDate()) + '.csv'`

In above cases, four dynamic filenames are created starting with Test_.

Dynamic content editor

The dynamic content editor automatically escapes characters in your content when you finish editing. For example, the following content in content editor is a string interpolation with an expression function.

```
"@{toUpperCase('myData')}"
```

The dynamic content editor converts above content to expression `"@{toUpperCase('myData')}"`. The result of this expression is the formatted string showed below.

```
"MYDATA"
```

Replacing special characters

The dynamic content editor automatically escapes characters like double quote, backslash in your content when you finish editing. This causes trouble if you want to replace line feed or tab by using \n, \t in replace() function. You can edit your dynamic content in code view to remove the extra \ in the expression, or you can follow below steps to replace special characters using expression language:

1. URL encoding against the original string value
2. Replace URL encoded string, for example, line feed (%0A), carriage return(%0D), horizontal tab(%09).
3. URL decoding

For example, variable `companyName` with a newline character in its value, expression `@uriComponentToString(replace(uriComponent(variables('companyName')), '%0A', ''))` can remove the newline character.

```
Contoso- Corporation
```

Escaping single quote character

Expression functions use single quote for string value parameters. Use two single quotes to escape a ' character in string functions. For example, expression `@concat('Baba', '''s ', 'book store')` will return below result.

```
Baba's book store
```

Pipeline scope variables

These system variables can be referenced anywhere in the pipeline.

Variable Name	Description
<code>@pipeline().DataFactory</code>	Name of the data or Synapse workspace the pipeline run is running in
<code>@pipeline().Pipeline</code>	Name of the pipeline
<code>@pipeline().RunId</code>	ID of the specific pipeline run
<code>@pipeline().TriggerId</code>	ID of the trigger that invoked the pipeline
<code>@pipeline().TriggerName</code>	Name of the trigger that invoked the pipeline
<code>@pipeline().TriggerTime</code>	Time of the trigger run that invoked the pipeline. This is the time at which the trigger actually fired to invoke the pipeline run, and it may differ slightly from the trigger's scheduled time.
<code>@pipeline().GroupId</code>	ID of the group to which pipeline run belongs.
<code>@pipeline()?.TriggeredByPipelineName</code>	Name of the pipeline that triggers the pipeline run. Applicable when the pipeline run is triggered by an ExecutePipeline activity. Evaluate to <i>Null</i> when used in other circumstances. Note the question mark after <code>@pipeline()</code>
<code>@pipeline()?.TriggeredByPipelineRunId</code>	Run ID of the pipeline that triggers the pipeline run. Applicable when the pipeline run is triggered by an ExecutePipeline activity. Evaluate to <i>Null</i> when used in other circumstances. Note the question mark after <code>@pipeline()</code>

Note

Trigger-related date/time system variables (in both pipeline and trigger scopes) return UTC dates in ISO 8601 format, for example, 2017-06-01T22:20:00.4061448Z.

Functions

You can call functions within expressions. The following sections provide information about the functions that can be used in an expression.

Date functions

Date or time function	Task
addDays	Add a number of days to a timestamp.
addHours	Add a number of hours to a timestamp.
addMinutes	Add a number of minutes to a timestamp.
addSeconds	Add a number of seconds to a timestamp.
addToTime	Add a number of time units to a timestamp. See also getFutureTime .
convertFromUtc	Convert a timestamp from Universal Time Coordinated (UTC) to the target time zone.
convertTimeZone	Convert a timestamp from the source time zone to the target time zone.
convertToUtc	Convert a timestamp from the source time zone to Universal Time Coordinated (UTC).
dayOfMonth	Return the day of the month component from a timestamp.
dayOfWeek	Return the day of the week component from a timestamp.
dayOfYear	Return the day of the year component from a timestamp.
formatDateTime	Return the timestamp as a string in optional format.
getFutureTime	Return the current timestamp plus the specified time units. See also addToTime .
getPastTime	Return the current timestamp minus the specified time units. See also subtractFromTime .
startOfDay	Return the start of the day for a timestamp.
startOfHour	Return the start of the hour for a timestamp.
startOfMonth	Return the start of the month for a timestamp.
subtractFromTime	Subtract a number of time units from a timestamp. See also getPastTime .
ticks	Return the <code>ticks</code> property value for a specified timestamp.

Date or time utcNow function	Task Return the current timestamp as a string.
--	---

String functions

To work with strings, you can use these string functions and also some collection functions. String functions work only on strings.

String function	Task
concat	Combine two or more strings, and return the combined string.
endsWith	Check whether a string ends with the specified substring.
guid	Generate a globally unique identifier (GUID) as a string.
indexOf	Return the starting position for a substring.
lastIndexOf	Return the starting position for the last occurrence of a substring.
replace	Replace a substring with the specified string, and return the updated string.
split	Return an array that contains substrings, separated by commas, from a larger string based on a specified delimiter character in the original string.
startsWith	Check whether a string starts with a specific substring.
substring	Return characters from a string, starting from the specified position.
toLowerCase	Return a string in lowercase format.
toUpperCase	Return a string in uppercase format.
trim	Remove leading and trailing whitespace from a string, and return the updated string.

Collection functions

To work with collections, generally arrays, strings, and sometimes, dictionaries, you can use these collection functions.

Collection function	Task
contains	Check whether a collection has a specific item.
empty	Check whether a collection is empty.

Collection function	Task
first	Return the first item from a collection.
intersection	Return a collection that has <i>only</i> the common items across the specified collections.
join	Return a string that has <i>all</i> the items from an array, separated by the specified character.
last	Return the last item from a collection.
length	Return the number of items in a string or array.
skip	Remove items from the front of a collection, and return <i>all the other</i> items.
take	Return items from the front of a collection.
union	Return a collection that has <i>all</i> the items from the specified collections.

Logical functions

These functions are useful inside conditions, they can be used to evaluate any type of logic.

Logical comparison function	Task
and	Check whether all expressions are true.
equals	Check whether both values are equivalent.
greater	Check whether the first value is greater than the second value.
greaterOrEquals	Check whether the first value is greater than or equal to the second value.
if	Check whether an expression is true or false. Based on the result, return a specified value.
less	Check whether the first value is less than the second value.
lessOrEquals	Check whether the first value is less than or equal to the second value.
not	Check whether an expression is false.
or	Check whether at least one expression is true.

Conversion functions

These functions are used to convert between each of the native types in the language:

- string
- integer
- float
- boolean
- arrays
- dictionaries

Conversion function	Task
array	Return an array from a single specified input. For multiple inputs, see createArray .
base64	Return the base64-encoded version for a string.
base64ToBinary	Return the binary version for a base64-encoded string.
base64ToString	Return the string version for a base64-encoded string.
binary	Return the binary version for an input value.
bool	Return the Boolean version for an input value.
coalesce	Return the first non-null value from one or more parameters.
createArray	Return an array from multiple inputs.
dataUri	Return the data URI for an input value.
dataUriToBinary	Return the binary version for a data URI.
dataUriToString	Return the string version for a data URI.
decodeBase64	Return the string version for a base64-encoded string.
decodeDataUri	Return the binary version for a data URI.
decodeUriComponent	Return a string that replaces escape characters with decoded versions.
encodeUriComponent	Return a string that replaces URL-unsafe characters with escape characters.
float	Return a floating point number for an input value.
int	Return the integer version for a string.
string	Return the string version for an input value.

Conversion function	Task
uriComponent	Return the URI-encoded version for an input value by replacing URL-unsafe characters with escape characters.
uriComponentToBinary	Return the binary version for a URI-encoded string.
uriComponentToString	Return the string version for a URI-encoded string.
xml	Return the XML version for a string.
xpath	Check XML for nodes or values that match an XPath (XML Path Language) expression, and return the matching nodes or values.

Math functions

These functions can be used for either types of numbers: **integers** and **floats**.

Math function	Task
add	Return the result from adding two numbers.
div	Return the result from dividing two numbers.
max	Return the highest value from a set of numbers or an array.
min	Return the lowest value from a set of numbers or an array.
mod	Return the remainder from dividing two numbers.
mul	Return the product from multiplying two numbers.
rand	Return a random integer from a specified range.
range	Return an integer array that starts from a specified integer.
sub	Return the result from subtracting the second number from the first number.

Function reference

This section lists all the available functions in alphabetical order.

add

Return the result from adding two numbers.

```
add(<summand_1>, <summand_2>)
```

Parameter	Required	Type	Description
<summand_1>, <summand_2>	Yes	Integer, Float, or mixed	The numbers to add

Return value	Type	Description
<result-sum>	Integer or Float	The result from adding the specified numbers

Example

This example adds the specified numbers:

```
add(1, 1.5)
```

And returns this result: 2.5

addDays

Add a number of days to a timestamp.

```
addDays('<timestamp>', <days>, '<format>'?)
```

Parameter	Required	Type	Description
<timestamp>	Yes	String	The string that contains the timestamp
<days>	Yes	Integer	The positive or negative number of days to add
<format>	No	String	Either a single format specifier or a custom format pattern. The default format for the timestamp is "o" (yyyy-MM-ddTHH:mm:ss.ffffffffK), which complies with ISO 8601 ↗ and preserves time zone information.

Return value	Type	Description
<updated-timestamp>	String	The timestamp plus the specified number of days

Example 1

This example adds 10 days to the specified timestamp:

```
addDays('2018-03-15T13:00:00Z', 10)
```

And returns this result: "2018-03-25T00:00:000000Z"

Example 2

This example subtracts five days from the specified timestamp:

```
addDays('2018-03-15T00:00:00Z', -5)
```

And returns this result: "2018-03-10T00:00:000000Z"

addHours

Add a number of hours to a timestamp.

```
addHours('<timestamp>', <hours>, '<format>'?)
```

Parameter	Required	Type	Description
<timestamp>	Yes	String	The string that contains the timestamp
<hours>	Yes	Integer	The positive or negative number of hours to add
<format>	No	String	Either a single format specifier or a custom format pattern. The default format for the timestamp is "o" (yyyy-MM-ddTHH:mm:ss.ffffffffK), which complies with ISO 8601 ↗ and preserves time zone information.

Return value	Type	Description
<updated-timestamp>	String	The timestamp plus the specified number of hours

Example 1

This example adds 10 hours to the specified timestamp:

```
addHours('2018-03-15T00:00:00Z', 10)
```

And returns this result: "2018-03-15T10:00:000000Z"

Example 2

This example subtracts five hours from the specified timestamp:

```
addHours('2018-03-15T15:00:00Z', -5)
```

And returns this result: "2018-03-15T10:00:000000Z"

addMinutes

Add a number of minutes to a timestamp.

```
addMinutes('<timestamp>', <minutes>, '<format>'?)
```

Parameter	Required	Type	Description
<timestamp>	Yes	String	The string that contains the timestamp
<minutes>	Yes	Integer	The positive or negative number of minutes to add
<format>	No	String	Either a single format specifier or a custom format pattern. The default format for the timestamp is "o" (yyyy-MM-ddTHH:mm:ss.ffffffffK), which complies with ISO 8601 ↗ and preserves time zone information.

Return value	Type	Description
<updated-timestamp>	String	The timestamp plus the specified number of minutes

Example 1

This example adds 10 minutes to the specified timestamp:

```
addMinutes('2018-03-15T00:10:00Z', 10)
```

And returns this result: "2018-03-15T00:20:00.000000Z"

Example 2

This example subtracts five minutes from the specified timestamp:

```
addMinutes('2018-03-15T00:20:00Z', -5)
```

And returns this result: "2018-03-15T00:15:00.000000Z"

addSeconds

Add a number of seconds to a timestamp.

```
addSeconds('<timestamp>', <seconds>, '<format>'?)
```

Parameter	Required	Type	Description
<timestamp>	Yes	String	The string that contains the timestamp
<seconds>	Yes	Integer	The positive or negative number of seconds to add
<format>	No	String	Either a single format specifier or a custom format pattern. The default format for the timestamp is "o" (yyyy-MM-ddTHH:mm:ss.ffffffffK), which complies with ISO 8601 and preserves time zone information.

Return value	Type	Description
<updated-timestamp>	String	The timestamp plus the specified number of seconds

Example 1

This example adds 10 seconds to the specified timestamp:

```
addSeconds('2018-03-15T00:00:00Z', 10)
```

And returns this result: "2018-03-15T00:00:10.000000Z"

Example 2

This example subtracts five seconds to the specified timestamp:

```
addSeconds('2018-03-15T00:00:30Z', -5)
```

And returns this result: "2018-03-15T00:00:25.000000Z"

addToTime

Add a number of time units to a timestamp. See also `getFutureTime()`.

```
addToTime('<timestamp>', <interval>, '<timeUnit>', '<format>'?)
```

Parameter	Required	Type	Description
<timestamp>	Yes	String	The string that contains the timestamp
<interval>	Yes	Integer	The number of specified time units to add
<timeUnit>	Yes	String	The unit of time to use with <i>interval</i> : "Second", "Minute", "Hour", "Day", "Week", "Month", "Year"
<format>	No	String	Either a single format specifier or a custom format pattern. The default format for the timestamp is "o" (yyyy-MM-ddTHH:mm:ss.ffffffffK), which complies with ISO 8601 ↗ and preserves time zone information.

Return value	Type	Description
<updated-timestamp>	String	The timestamp plus the specified number of time units

Example 1

This example adds one day to the specified timestamp:

```
addToTime('2018-01-01T00:00:00Z', 1, 'Day')
```

And returns this result: "2018-01-02T00:00:00.000000Z"

Example 2

This example adds one day to the specified timestamp:

```
addToTime('2018-01-01T00:00:00Z', 1, 'Day', 'D')
```

And returns the result using the optional "D" format: "Tuesday, January 2, 2018"

and

Check whether both expressions are true. Return true when both expressions are true, or return false when at least one expression is false.

```
and(<expression1>, <expression2>)
```

Parameter	Required	Type	Description
<expression1>, <expression2>	Yes	Boolean	The expressions to check

Return value	Type	Description
true or false	Boolean	Return true when both expressions are true. Return false when at least one expression is false.

Example 1

These examples check whether the specified Boolean values are both true:

```
and(true, true)
and(false, true)
and(false, false)
```

And returns these results:

- First example: Both expressions are true, so returns true.

- Second example: One expression is false, so returns `false`.
- Third example: Both expressions are false, so returns `false`.

Example 2

These examples check whether the specified expressions are both true:

```
and>equals(1, 1), equals(2, 2))  
and>equals(1, 1), equals(1, 2))  
and>equals(1, 2), equals(1, 3))
```

And returns these results:

- First example: Both expressions are true, so returns `true`.
- Second example: One expression is false, so returns `false`.
- Third example: Both expressions are false, so returns `false`.

array

Return an array from a single specified input. For multiple inputs, see `createArray()`.

```
array('<value>')
```

Parameter	Required	Type	Description
<code><value></code>	Yes	String	The string for creating an array

Return value	Type	Description
<code>[<value>]</code>	Array	An array that contains the single specified input

Example

This example creates an array from the "hello" string:

```
array('hello')
```

And returns this result: `["hello"]`

base64

Return the base64-encoded version for a string.

```
base64('<value>')
```

Parameter	Required	Type	Description
<value>	Yes	String	The input string

Return value	Type	Description
<base64-string>	String	The base64-encoded version for the input string

Example

This example converts the "hello" string to a base64-encoded string:

```
base64('hello')
```

And returns this result: "aGVsbG8="

base64ToBinary

Return the binary version for a base64-encoded string.

```
base64ToBinary('<value>')
```

Parameter	Required	Type	Description
<value>	Yes	String	The base64-encoded string to convert

Return value	Type	Description
<binary-for-base64-string>	String	The binary version for the base64-encoded string

Example

This example converts the "aGVsbG8=" base64-encoded string to a binary string:

```
base64ToBinary('aGVsbG8=')
```

And returns this result:

```
"0110000101000111010101100111001101100010010001110011100000111101"
```

base64ToString

Return the string version for a base64-encoded string, effectively decoding the base64 string. Use this function rather than decodeBase64(). Although both functions work the same way, `base64ToString()` is preferred.

```
base64ToString('<value>')
```

Parameter	Required	Type	Description
<code><value></code>	Yes	String	The base64-encoded string to decode

Return value	Type	Description
<code><decoded-base64-string></code>	String	The string version for a base64-encoded string

Example

This example converts the "aGVsbG8=" base64-encoded string to just a string:

```
base64ToString('aGVsbG8=')
```

And returns this result: "hello"

binary

Return the binary version for a string.

```
binary('<value>')
```

Parameter	Required	Type	Description
<value>	Yes	String	The string to convert

Return value	Type	Description
<binary-for-input-value>	String	The binary version for the specified string

Example

This example converts the "hello" string to a binary string:

```
binary('hello')
```

And returns this result:

```
"011010000110010101011011000110110001101111"
```

bool

Return the Boolean version for a value.

```
bool(<value>')
```

Parameter	Required	Type	Description
<value>	Yes	Any	The value to convert

Return value	Type	Description
true or false	Boolean	The Boolean version for the specified value

Example

These examples convert the specified values to Boolean values:

```
bool(1)  
bool(0)
```

And returns these results:

- First example: `true`
- Second example: `false`

coalesce

Return the first non-null value from one or more parameters. Empty strings, empty arrays, and empty objects aren't null.

```
coalesce(<object_1>, <object_2>, ...)
```

Parameter	Required	Type	Description
<code><object_1>, <object_2>, ...</code>	Yes	Any, can mix types	One or more items to check for null

Return value	Type	Description
<code><first-non-null-item></code>	Any	The first item or value that isn't null. If all parameters are null, this function returns null.

Example

These examples return the first non-null value from the specified values, or null when all the values are null:

```
coalesce(null, true, false)  
coalesce(null, 'hello', 'world')  
coalesce(null, null, null)
```

And returns these results:

- First example: `true`
- Second example: `"hello"`
- Third example: `null`

concat

Combine two or more strings, and return the combined string.

```
concat('<text1>', '<text2>', ...)
```

Parameter	Required	Type	Description
<text1>, <text2>, ...	Yes	String	At least two strings to combine

Return value	Type	Description
<text1text2...>	String	The string created from the combined input strings

Example

This example combines the strings "Hello" and "World":

```
concat('Hello', 'World')
```

And returns this result: "HelloWorld"

contains

Check whether a collection has a specific item. Return true when the item is found, or return false when not found. This function is case-sensitive.

```
contains('<collection>', '<value>')
contains([<collection>], '<value>')
```

Specifically, this function works on these collection types:

- A *string* to find a *substring*
- An *array* to find a *value*
- A *dictionary* to find a *key*

Parameter	Required	Type	Description
<collection>	Yes	String, Array, or Dictionary	The collection to check
<value>	Yes	String, Array, or Dictionary, respectively	The item to find

Return value	Type	Description
true or false	Boolean	Return true when the item is found. Return false when not found.

Example 1

This example checks the string "hello world" for the substring "world" and returns true:

```
contains('hello world', 'world')
```

Example 2

This example checks the string "hello world" for the substring "universe" and returns false:

```
contains('hello world', 'universe')
```

convertFromUtc

Convert a timestamp from Universal Time Coordinated (UTC) to the target time zone.

```
convertFromUtc('<timestamp>', '<destinationTimeZone>', '<format>'?)
```

Parameter	Required	Type	Description
<timestamp>	Yes	String	The string that contains the timestamp
<destinationTimeZone>	Yes	String	The name for the target time zone. For time zone names, see Microsoft Time Zone Values, but you might have to remove any punctuation from the time zone name.

Parameter	Required	Type	Description
<format>	No	String	Either a single format specifier or a custom format pattern. The default format for the timestamp is "o" (yyyy-MM-ddTHH:mm:ss.fffffffK), which complies with ISO 8601 and preserves time zone information.

Return value	Type	Description
<converted-timestamp>	String	The timestamp converted to the target time zone

Example 1

This example converts a timestamp to the specified time zone:

```
convertFromUtc('2018-01-01T08:00:00.000000Z', 'Pacific Standard Time')
```

And returns this result: "2018-01-01T00:00:00Z"

Example 2

This example converts a timestamp to the specified time zone and format:

```
convertFromUtc('2018-01-01T08:00:00.000000Z', 'Pacific Standard Time', 'D')
```

And returns this result: "Monday, January 1, 2018"

convertTimeZone

Convert a timestamp from the source time zone to the target time zone.

```
convertTimeZone('<timestamp>', '<sourceTimeZone>', '<destinationTimeZone>', '<format>'?)
```

Parameter	Required	Type	Description
<timestamp>	Yes	String	The string that contains the timestamp

Parameter	Required	Type	Description
<code><sourceTimeZone></code>	Yes	String	The name for the source time zone. For time zone names, see Microsoft Time Zone Values, but you might have to remove any punctuation from the time zone name.
<code><destinationTimeZone></code>	Yes	String	The name for the target time zone. For time zone names, see Microsoft Time Zone Values, but you might have to remove any punctuation from the time zone name.
<code><format></code>	No	String	Either a single format specifier or a custom format pattern. The default format for the timestamp is "o" (yyyy-MM-ddTHH:mm:ss.fffffffK), which complies with ISO 8601 ↗ and preserves time zone information.

Return value	Type	Description
<code><converted-timestamp></code>	String	The timestamp converted to the target time zone

Example 1

This example converts the source time zone to the target time zone:

```
convertTimeZone('2018-01-01T08:00:00.000000Z', 'UTC', 'Pacific Standard Time')
```

And returns this result: "2018-01-01T00:00:00.000000"

Example 2

This example converts a time zone to the specified time zone and format:

```
convertTimeZone('2018-01-01T08:00:00.000000Z', 'UTC', 'Pacific Standard Time', 'D')
```

And returns this result: "Monday, January 1, 2018"

convertToUtc

Convert a timestamp from the source time zone to Universal Time Coordinated (UTC).

```
convertToUtc('<timestamp>', '<sourceTimeZone>', '<format>'?)
```

Parameter	Required	Type	Description
<timestamp>	Yes	String	The string that contains the timestamp
<sourceTimeZone>	Yes	String	The name for the source time zone. For time zone names, see Microsoft Time Zone Values, but you might have to remove any punctuation from the time zone name.
<format>	No	String	Either a single format specifier or a custom format pattern. The default format for the timestamp is "o" (yyyy-MM-ddTHH:mm:ss.fffffffK), which complies with ISO 8601 ↗ and preserves time zone information.

Return value	Type	Description
<converted-timestamp>	String	The timestamp converted to UTC

Example 1

This example converts a timestamp to UTC:

```
convertToUtc('01/01/2018 00:00:00', 'Pacific Standard Time')
```

And returns this result: "2018-01-01T08:00:00.000000Z"

Example 2

This example converts a timestamp to UTC:

```
convertToUtc('01/01/2018 00:00:00', 'Pacific Standard Time', 'D')
```

And returns this result: "Monday, January 1, 2018"

createArray

Return an array from multiple inputs. For single input arrays, see `array()`.

```
createArray('<object1>', '<object2>', ...)
```

Parameter	Required	Type	Description
<code><object1>, <object2>, ...</code>	Yes	Any, but not mixed	At least two items to create the array

Return value	Type	Description
<code>[<object1>, <object2>, ...]</code>	Array	The array created from all the input items

Example

This example creates an array from these inputs:

```
createArray('h', 'e', 'l', 'l', 'o')
```

And returns this result: `["h", "e", "l", "l", "o"]`

dataUri

Return a data uniform resource identifier (URI) for a string.

```
dataUri('<value>')
```

Parameter	Required	Type	Description
<code><value></code>	Yes	String	The string to convert

Return value	Type	Description
<code><data-uri></code>	String	The data URI for the input string

Example

This example creates a data URI for the "hello" string:

```
dataUri('hello')
```

And returns this result: "data:text/plain;charset=utf-8;base64,aGVsbG8="

dataUriToBinary

Return the binary version for a data uniform resource identifier (URI). Use this function rather than decodeDataUri(). Although both functions work the same way, `dataUriBinary()` is preferred.

```
dataUriToBinary('<value>')
```

Parameter	Required	Type	Description
<code><value></code>	Yes	String	The data URI to convert

Return value	Type	Description
<code><binary-for-data-uri></code>	String	The binary version for the data URI

Example

This example creates a binary version for this data URI:

```
dataUriToBinary('data:text/plain;charset=utf-8;base64,aGVsbG8=')
```

And returns this result:

```
"0110010001100001011101000110000100111010011101000110010101111000011101000010111101  
1100000  
11011000110000101101001011011000111011010001101101000011000010111001001110011011010  
01010111  
0100001110101110101011101000110011000101101001110000011101101100010011000010111001  
10110010  
10011011000110100001011000110000101000111010101100111001101100010010001110011100000  
111101"
```

dataUriToString

Return the string version for a data uniform resource identifier (URI).

```
dataUriToString('<value>')
```

Parameter	Required	Type	Description
<value>	Yes	String	The data URI to convert

Return value	Type	Description
<string-for-data-uri>	String	The string version for the data URI

Example

This example creates a string for this data URI:

```
dataUriToString('data:text/plain;charset=utf-8;base64,aGVsbG8=')
```

And returns this result: "hello"

dayOfMonth

Return the day of the month from a timestamp.

```
dayOfMonth('<timestamp>')
```

Parameter	Required	Type	Description
<timestamp>	Yes	String	The string that contains the timestamp

Return value	Type	Description
<day-of-month>	Integer	The day of the month from the specified timestamp

Example

This example returns the number for the day of the month from this timestamp:

```
dayOfMonth('2018-03-15T13:27:36Z')
```

And returns this result: 15

dayOfWeek

Return the day of the week from a timestamp.

```
dayOfWeek('<timestamp>')
```

Parameter	Required	Type	Description
<timestamp>	Yes	String	The string that contains the timestamp

Return value	Type	Description
<day-of-week>	Integer	The day of the week from the specified timestamp where Sunday is 0, Monday is 1, and so on

Example

This example returns the number for the day of the week from this timestamp:

```
dayOfWeek('2018-03-15T13:27:36Z')
```

And returns this result: 3

dayOfYear

Return the day of the year from a timestamp.

```
dayOfYear('<timestamp>')
```

Parameter	Required	Type	Description
<timestamp>	Yes	String	The string that contains the timestamp
Return value	Type	Description	
<day-of-year>	Integer	The day of the year from the specified timestamp	

Example

This example returns the number of the day of the year from this timestamp:

```
dayOfYear('2018-03-15T13:27:36Z')
```

And returns this result: 74

decodeBase64

Return the string version for a base64-encoded string, effectively decoding the base64 string. Consider using `base64ToString()` rather than `decodeBase64()`. Although both functions work the same way, `base64ToString()` is preferred.

```
decodeBase64('<value>')
```

Parameter	Required	Type	Description
<value>	Yes	String	The base64-encoded string to decode
Return value	Type	Description	
<decoded-base64-string>	String	The string version for a base64-encoded string	

Example

This example creates a string for a base64-encoded string:

```
decodeBase64('aGVsbG8=')
```

And returns this result: "hello"

decodeDataUri

Return the binary version for a data uniform resource identifier (URI). Consider using `dataUriToBinary()`, rather than `decodeDataUri()`. Although both functions work the same way, `dataUriToBinary()` is preferred.

```
decodeDataUri('<value>')
```

Parameter	Required	Type	Description
<code><value></code>	Yes	String	The data URI string to decode

Return value	Type	Description
<code><binary-for-data-uri></code>	String	The binary version for a data URI string

Example

This example returns the binary version for this data URI:

```
decodeDataUri('data:text/plain;charset=utf-8;base64,aGVsbG8=')
```

And returns this result:

```
"011001000110000101110100011000010011101001110100011001010111000011101000010111101  
1100000  
11011000110000101101001011011000111011010001101101000010111001001110011011010  
01010111  
0100001110101110101011101000110011000101101001110000011101101100010011000010111001  
10110010  
100110110001101000010110001100001010001110101011001110001001000111001110000  
111101"
```

decodeUriComponent

Return a string that replaces escape characters with decoded versions.

```
decodeURIComponent('<value>')
```

Parameter	Required	Type	Description
<value>	Yes	String	The string with the escape characters to decode

Return value	Type	Description
<decoded-uri>	String	The updated string with the decoded escape characters

Example

This example replaces the escape characters in this string with decoded versions:

```
decodeURIComponent('http%3A%2F%2Fcontoso.com')
```

And returns this result: "https://contoso.com"

div

Return the integer result from dividing two numbers. To get the remainder result, see mod().

```
div(<dividend>, <divisor>)
```

Parameter	Required	Type	Description
<dividend>	Yes	Integer or Float	The number to divide by the <i>divisor</i>
<divisor>	Yes	Integer or Float	The number that divides the <i>dividend</i> , but can't be 0

Return value	Type	Description
<quotient-result>	Integer	The integer result from dividing the first number by the second number

Example

Both examples divide the first number by the second number:

```
div(10, 5)
div(11, 5)
```

And return this result: 2

encodeUriComponent

Return a uniform resource identifier (URI) encoded version for a string by replacing URL-unsafe characters with escape characters. Consider using `uriComponent()`, rather than `encodeUriComponent()`. Although both functions work the same way, `uriComponent()` is preferred.

```
encodeUriComponent('<value>')
```

Parameter	Required	Type	Description
<code><value></code>	Yes	String	The string to convert to URI-encoded format

Return value	Type	Description
<code><encoded-uri></code>	String	The URI-encoded string with escape characters

Example

This example creates a URI-encoded version for this string:

```
encodeUriComponent('https://contoso.com')
```

And returns this result: "http%3A%2F%2Fcontoso.com"

empty

Check whether a collection is empty. Return true when the collection is empty, or return false when not empty.

```
empty('<collection>')
empty([<collection>])
```

Parameter	Required	Type	Description
<collection>	Yes	String, Array, or Object	The collection to check

Return value	Type	Description
true or false	Boolean	Return true when the collection is empty. Return false when not empty.

Example

These examples check whether the specified collections are empty:

```
empty('')
empty('abc')
```

And returns these results:

- First example: Passes an empty string, so the function returns `true`.
- Second example: Passes the string "abc", so the function returns `false`.

endsWith

Check whether a string ends with a specific substring. Return true when the substring is found, or return false when not found. This function isn't case-sensitive.

```
endsWith('<text>', '<searchText>')
```

Parameter	Required	Type	Description
<text>	Yes	String	The string to check
<searchText>	Yes	String	The ending substring to find

Return value	Type	Description
true or false	Boolean	Return true when the ending substring is found. Return false when not found.

Example 1

This example checks whether the "hello world" string ends with the "world" string:

```
endsWith('hello world', 'world')
```

And returns this result: `true`

Example 2

This example checks whether the "hello world" string ends with the "universe" string:

```
endsWith('hello world', 'universe')
```

And returns this result: `false`

equals

Check whether both values, expressions, or objects are equivalent. Return true when both are equivalent, or return false when they're not equivalent.

```
equals('<object1>', '<object2>')
```

Parameter	Required	Type	Description
<code><object1>, <object2></code>	Yes	Various	The values, expressions, or objects to compare

Return value	Type	Description
true or false	Boolean	Return true when both are equivalent. Return false when not equivalent.

Example

These examples check whether the specified inputs are equivalent.

```
equals(true, 1)
equals('abc', 'abcd')
```

And returns these results:

- First example: Both values are equivalent, so the function returns `true`.
- Second example: Both values aren't equivalent, so the function returns `false`.

first

Return the first item from a string or array.

```
first('<collection>')
first([<collection>])
```

Parameter	Required	Type	Description
<code><collection></code>	Yes	String or Array	The collection where to find the first item

Return value	Type	Description
<code><first-collection-item></code>	Any	The first item in the collection

Example

These examples find the first item in these collections:

```
first('hello')
first(createArray(0, 1, 2))
```

And return these results:

- First example: `"h"`
- Second example: `0`

float

Convert a string version for a floating-point number to an actual floating point number.

```
float('<value>')
```

Parameter	Required	Type	Description
<value>	Yes	String	The string that has a valid floating-point number to convert

Return value	Type	Description
<float-value>	Float	The floating-point number for the specified string

Example

This example creates a string version for this floating-point number:

```
float('10.333')
```

And returns this result: 10.333

formatDateTime

Return a timestamp in the specified format.

```
formatDateTime('<timestamp>', '<format>'?)
```

Parameter	Required	Type	Description
<timestamp>	Yes	String	The string that contains the timestamp
<format>	No	String	Either a single format specifier or a custom format pattern. The default format for the timestamp is "o" (yyyy-MM-ddTHH:mm:ss.ffffffffK), which complies with ISO 8601 ↗ and preserves time zone information.

Return value	Type	Description
<reformatted-timestamp>	String	The updated timestamp in the specified format

Example

This example converts a timestamp to the specified format:

```
formatDateTime('03/15/2018 12:00:00', 'yyyy-MM-ddTHH:mm:ss')
```

And returns this result: "2018-03-15T12:00:00"

getFutureTime

Return the current timestamp plus the specified time units.

```
getFutureTime(<interval>, <timeUnit>, <format>?)
```

Parameter	Required	Type	Description
<interval>	Yes	Integer	The number of specified time units to add
<timeUnit>	Yes	String	The unit of time to use with <i>interval</i> : "Second", "Minute", "Hour", "Day", "Week", "Month", "Year"
<format>	No	String	Either a single format specifier or a custom format pattern. The default format for the timestamp is "o" (yyyy-MM-ddTHH:mm:ss.fffffffK), which complies with ISO 8601 and preserves time zone information.

Return value	Type	Description
<updated-timestamp>	String	The current timestamp plus the specified number of time units

Example 1

Suppose the current timestamp is "2018-03-01T00:00:00.0000000Z". This example adds five days to that timestamp:

```
getFutureTime(5, 'Day')
```

And returns this result: "2018-03-06T00:00:00.000000Z"

Example 2

Suppose the current timestamp is "2018-03-01T00:00:00.000000Z". This example adds five days and converts the result to "D" format:

```
getFutureTime(5, 'Day', 'D')
```

And returns this result: "Tuesday, March 6, 2018"

getPastTime

Return the current timestamp minus the specified time units.

```
getPastTime(<interval>, <timeUnit>, <format>?)
```

Parameter	Required	Type	Description
<interval>	Yes	Integer	The number of specified time units to subtract
<timeUnit>	Yes	String	The unit of time to use with <i>interval</i> : "Second", "Minute", "Hour", "Day", "Week", "Month", "Year"
<format>	No	String	Either a single format specifier or a custom format pattern. The default format for the timestamp is "o" (yyyy-MM-ddTHH:mm:ss.ffffffffK), which complies with ISO 8601 ↗ and preserves time zone information.

Return value	Type	Description
<updated-timestamp>	String	The current timestamp minus the specified number of time units

Example 1

Suppose the current timestamp is "2018-02-01T00:00:00.000000Z". This example subtracts five days from that timestamp:

```
getPastTime(5, 'Day')
```

And returns this result: "2018-01-27T00:00:00.000000Z"

Example 2

Suppose the current timestamp is "2018-02-01T00:00:00.000000Z". This example subtracts five days and converts the result to "D" format:

```
getPastTime(5, 'Day', 'D')
```

And returns this result: "Saturday, January 27, 2018"

greater

Check whether the first value is greater than the second value. Return true when the first value is more, or return false when less.

```
greater(<value>, <compareTo>)
greater('<value>', '<compareTo>')
```

Parameter	Required	Type	Description
<value>	Yes	Integer, Float, or String	The first value to check whether greater than the second value
<compareTo>	Yes	Integer, Float, or String, respectively	The comparison value

Return value	Type	Description
true or false	Boolean	Return true when the first value is greater than the second value. Return false when the first value is equal to or less than the second value.

Example

These examples check whether the first value is greater than the second value:

```
greater(10, 5)
greater('apple', 'banana')
```

And return these results:

- First example: `true`
- Second example: `false`

greaterOrEquals

Check whether the first value is greater than or equal to the second value. Return true when the first value is greater or equal, or return false when the first value is less.

```
greaterOrEquals(<value>, <compareTo>)
greaterOrEquals('<value>', '<compareTo>')
```

Parameter	Required	Type	Description
<code><value></code>	Yes	Integer, Float, or String	The first value to check whether greater than or equal to the second value
<code><compareTo></code>	Yes	Integer, Float, or String, respectively	The comparison value

Return value	Type	Description
<code>true</code> or <code>false</code>	Boolean	Return true when the first value is greater than or equal to the second value. Return false when the first value is less than the second value.

Example

These examples check whether the first value is greater or equal than the second value:

```
greaterOrEquals(5, 5)
greaterOrEquals('apple', 'banana')
```

And return these results:

- First example: `true`

- Second example: `false`

guid

Generate a globally unique identifier (GUID) as a string, for example, "c2ecc88d-88c8-4096-912c-d6f2e2b138ce":

```
guid()
```

Also, you can specify a different format for the GUID other than the default format, "D", which is 32 digits separated by hyphens.

```
guid('<format>')
```

Parameter	Required	Type	Description
<code><format></code>	No	String	A single format specifier for the returned GUID. By default, the format is "D", but you can use "N", "D", "B", "P", or "X".

Return value	Type	Description
<code><GUID-value></code>	String	A randomly generated GUID

Example

This example generates the same GUID, but as 32 digits, separated by hyphens, and enclosed in parentheses:

```
guid('P')
```

And returns this result: `"(c2ecc88d-88c8-4096-912c-d6f2e2b138ce)"`

if

Check whether an expression is true or false. Based on the result, return a specified value.

```
if(<expression>, <valueIfTrue>, <valueIfFalse>)
```

Parameter	Required	Type	Description
<expression>	Yes	Boolean	The expression to check
<valueIfTrue>	Yes	Any	The value to return when the expression is true
<valueIfFalse>	Yes	Any	The value to return when the expression is false

Return value	Type	Description
<specified-return-value>	Any	The specified value that returns based on whether the expression is true or false

Example

This example returns "yes" because the specified expression returns true. Otherwise, the example returns "no":

```
if>equals(1, 1), 'yes', 'no')
```

indexOf

Return the starting position or index value for a substring. This function isn't case-sensitive, and indexes start with the number 0.

```
indexOf('<text>', '<searchText>')
```

Parameter	Required	Type	Description
<text>	Yes	String	The string that has the substring to find
<searchText>	Yes	String	The substring to find

Return value	Type	Description
<index-value>	Integer	The starting position or index value for the specified substring. If the string isn't found, return the number -1.

Example

This example finds the starting index value for the "world" substring in the "hello world" string:

```
indexOf('hello world', 'world')
```

And returns this result: 6

int

Return the integer version for a string.

```
int('<value>')
```

Parameter	Required	Type	Description
<value>	Yes	String	The string to convert

Return value	Type	Description
<integer-result>	Integer	The integer version for the specified string

Example

This example creates an integer version for the string "10":

```
int('10')
```

And returns this result: 10

json

Return the JavaScript Object Notation (JSON) type value or object for a string or XML.

```
json('<value>')
```

Parameter	Required	Type	Description
<value>	Yes	String or XML	The string or XML to convert

Return value	Type	Description
<JSON-result>	JSON native type or object	The JSON native type value or object for the specified string or XML. If the string is null, the function returns an empty object.

Example 1

This example converts this string to the JSON value:

```
json('[1, 2, 3]')
```

And returns this result: [1, 2, 3]

Example 2

This example converts this string to JSON:

```
json('{"fullName": "Sophia Owen"}')
```

And returns this result:

```
{
  "fullName": "Sophia Owen"
}
```

Example 3

This example converts this XML to JSON:

```
json(xml('<?xml version="1.0"?> <root> <person id="1"> <name>Sophia Owen</name> <occupation>Engineer</occupation> </person> </root>'))
```

And returns this result:

JSON

```
{  
    "?xml": { "@version": "1.0" },  
    "root": {  
        "person": [ {  
            "@id": "1",  
            "name": "Sophia Owen",  
            "occupation": "Engineer"  
        } ]  
    }  
}
```

intersection

Return a collection that has *only* the common items across the specified collections. To appear in the result, an item must appear in all the collections passed to this function. If one or more items have the same name, the last item with that name appears in the result.

```
intersection([<collection1>], [<collection2>], ...)  
intersection('<collection1>', '<collection2>', ...)
```

Parameter	Required	Type	Description
<collection1>, <collection2>, ...	Yes	Array or Object, but not both	The collections from where you want <i>only</i> the common items

Return value	Type	Description
<common-items>	Array or Object, respectively	A collection that has only the common items across the specified collections

Example

This example finds the common items across these arrays:

```
intersection(createArray(1, 2, 3), createArray(101, 2, 1, 10),
createArray(6, 8, 1, 2))
```

And returns an array with *only* these items: [1, 2]

join

Return a string that has all the items from an array and has each character separated by a *delimiter*.

```
join([<collection>], '<delimiter>')
```

Parameter	Required	Type	Description
<collection>	Yes	Array	The array that has the items to join
<delimiter>	Yes	String	The separator that appears between each character in the resulting string

Return value	Type	Description
<char1><delimiter><char2> <delimiter>...	String	The resulting string created from all the items in the specified array

Example

This example creates a string from all the items in this array with the specified character as the delimiter:

```
join(createArray('a', 'b', 'c'), '.')
```

And returns this result: "a.b.c"

last

Return the last item from a collection.

```
last('<collection>')
last([<collection>])
```

Parameter	Required	Type	Description
<collection>	Yes	String or Array	The collection where to find the last item

Return value	Type	Description
<last-collection-item>	String or Array, respectively	The last item in the collection

Example

These examples find the last item in these collections:

```
last('abcd')
last(createArray(0, 1, 2, 3))
```

And returns these results:

- First example: "d"
- Second example: 3

lastIndexOf

Return the starting position or index value for the last occurrence of a substring. This function isn't case-sensitive, and indexes start with the number 0.

```
lastIndexOf('<text>', '<searchText>')
```

Parameter	Required	Type	Description
<text>	Yes	String	The string that has the substring to find
<searchText>	Yes	String	The substring to find

Return value	Type	Description
<ending-index-value>	Integer	The starting position or index value for the last occurrence of the specified substring.

Return value	Type	Description
		If the string isn't found, return the number -1.

Example

This example finds the starting index value for the last occurrence of the "world" substring in the "hello world" string:

```
lastIndexOf('hello world', 'world')
```

And returns this result: 6

length

Return the number of items in a collection.

```
length('<collection>')
length([<collection>])
```

Parameter	Required	Type	Description
<collection>	Yes	String or Array	The collection with the items to count

Return value	Type	Description
<length-or-count>	Integer	The number of items in the collection

Example

These examples count the number of items in these collections:

```
length('abcd')
length(createArray(0, 1, 2, 3))
```

And return this result: 4

less

Check whether the first value is less than the second value. Return true when the first value is less, or return false when the first value is more.

```
less(<value>, <compareTo>)
less('<value>', '<compareTo>')
```

Parameter	Required	Type	Description
<value>	Yes	Integer, Float, or String	The first value to check whether less than the second value
<compareTo>	Yes	Integer, Float, or String, respectively	The comparison item

Return value	Type	Description
true or false	Boolean	Return true when the first value is less than the second value. Return false when the first value is equal to or greater than the second value.

Example

These examples check whether the first value is less than the second value.

```
less(5, 10)
less('banana', 'apple')
```

And return these results:

- First example: `true`
- Second example: `false`

lessOrEquals

Check whether the first value is less than or equal to the second value. Return true when the first value is less than or equal, or return false when the first value is more.

```
lessOrEquals(<value>, <compareTo>)
lessOrEquals('<value>', '<compareTo>')
```

Parameter	Required	Type	Description
<value>	Yes	Integer, Float, or String	The first value to check whether less than or equal to the second value
<compareTo>	Yes	Integer, Float, or String, respectively	The comparison item

Return value	Type	Description
true or false	Boolean	Return true when the first value is less than or equal to the second value. Return false when the first value is greater than the second value.

Example

These examples check whether the first value is less or equal than the second value.

```
lessOrEquals(10, 10)
lessOrEquals('apply', 'apple')
```

And return these results:

- First example: `true`
- Second example: `false`

max

Return the highest value from a list or array with numbers that is inclusive at both ends.

```
max(<number1>, <number2>, ...)
max([<number1>, <number2>, ...])
```

Parameter	Required	Type	Description
<number1>, <number2>, ...	Yes	Integer, Float, or both	The set of numbers from which you want the highest value
[<number1>, <number2>, ...]	Yes	Array - Integer, Float, or both	The array of numbers from which you want the highest value

Return value	Type	Description
<max-value>	Integer or Float	The highest value in the specified array or set of numbers

Example

These examples get the highest value from the set of numbers and the array:

```
max(1, 2, 3)
max(createArray(1, 2, 3))
```

And return this result: 3

min

Return the lowest value from a set of numbers or an array.

```
min(<number1>, <number2>, ...)
min([<number1>, <number2>, ...])
```

Parameter	Required	Type	Description
<number1>, <number2>, ...	Yes	Integer, Float, or both	The set of numbers from which you want the lowest value
[<number1>, <number2>, ...]	Yes	Array - Integer, Float, or both	The array of numbers from which you want the lowest value

Return value	Type	Description
<min-value>	Integer or Float	The lowest value in the specified set of numbers or specified array

Example

These examples get the lowest value in the set of numbers and the array:

```
min(1, 2, 3)
```

```
min(createArray(1, 2, 3))
```

And return this result: 1

mod

Return the remainder from dividing two numbers. To get the integer result, see `div()`.

```
mod(<dividend>, <divisor>)
```

Parameter	Required	Type	Description
<code><dividend></code>	Yes	Integer or Float	The number to divide by the <i>divisor</i>
<code><divisor></code>	Yes	Integer or Float	The number that divides the <i>dividend</i> , but can't be 0.

Return value	Type	Description
<code><modulo-result></code>	Integer or Float	The remainder from dividing the first number by the second number

Example

This example divides the first number by the second number:

```
mod(3, 2)
```

And return this result: 1

mul

Return the product from multiplying two numbers.

```
mul(<multiplicand1>, <multiplicand2>)
```

Parameter	Required	Type	Description
<multiplicand1>	Yes	Integer or Float	The number to multiply by <i>multiplicand2</i>
<multiplicand2>	Yes	Integer or Float	The number that multiples <i>multiplicand1</i>

Return value	Type	Description
<product-result>	Integer or Float	The product from multiplying the first number by the second number

Example

These examples multiple the first number by the second number:

```
mul(1, 2)
mul(1.5, 2)
```

And return these results:

- First example: 2
- Second example 3

not

Check whether an expression is false. Return true when the expression is false, or return false when true.

JSON

```
not(<expression>)
```

Parameter	Required	Type	Description
<expression>	Yes	Boolean	The expression to check

Return value	Type	Description
true or false	Boolean	Return true when the expression is false. Return false when the expression is true.

Example 1

These examples check whether the specified expressions are false:

JSON

```
not(false)  
not(true)
```

And return these results:

- First example: The expression is false, so the function returns `true`.
- Second example: The expression is true, so the function returns `false`.

Example 2

These examples check whether the specified expressions are false:

JSON

```
not(equals(1, 2))  
not(equals(1, 1))
```

And return these results:

- First example: The expression is false, so the function returns `true`.
- Second example: The expression is true, so the function returns `false`.

Or

Check whether at least one expression is true. Return true when at least one expression is true, or return false when both are false.

```
or(<expression1>, <expression2>)
```

Parameter	Required	Type	Description
<code><expression1>, <expression2></code>	Yes	Boolean	The expressions to check

Return value	Type	Description
<code>true</code> or <code>false</code>	Boolean	Return true when at least one expression is true. Return false when both expressions are false.

Example 1

These examples check whether at least one expression is true:

JSON

```
or(true, false)
or(false, false)
```

And return these results:

- First example: At least one expression is true, so the function returns `true`.
- Second example: Both expressions are false, so the function returns `false`.

Example 2

These examples check whether at least one expression is true:

JSON

```
or>equals(1, 1), equals(1, 2))
or>equals(1, 2), equals(1, 3))
```

And return these results:

- First example: At least one expression is true, so the function returns `true`.
- Second example: Both expressions are false, so the function returns `false`.

rand

Return a random integer from a specified range, which is inclusive only at the starting end.

```
rand(<minValue>, <maxValue>)
```

Parameter	Required	Type	Description
<code><minValue></code>	Yes	Integer	The lowest integer in the range
<code><maxValue></code>	Yes	Integer	The integer that follows the highest integer in the range that the function can return

Return value	Type	Description
<random-result>	Integer	The random integer returned from the specified range

Example

This example gets a random integer from the specified range, excluding the maximum value:

```
rand(1, 5)
```

And returns one of these numbers as the result: 1, 2, 3, or 4

range

Return an integer array that starts from a specified integer.

```
range(<startIndex>, <count>)
```

Parameter	Required	Type	Description
<startIndex>	Yes	Integer	An integer value that starts the array as the first item
<count>	Yes	Integer	The number of integers in the array

Return value	Type	Description
[<range-result>]	Array	The array with integers starting from the specified index

Example

This example creates an integer array that starts from the specified index and has the specified number of integers:

```
range(1, 4)
```

And returns this result: [1, 2, 3, 4]

replace

Replace a substring with the specified string, and return the result string. This function is case-sensitive.

```
replace('<text>', '<oldText>', '<newText>')
```

Parameter	Required	Type	Description
<text>	Yes	String	The string that has the substring to replace
<oldText>	Yes	String	The substring to replace
<newText>	Yes	String	The replacement string

Return value	Type	Description
<updated-text>	String	The updated string after replacing the substring If the substring isn't found, return the original string.

Example

This example finds the "old" substring in "the old string" and replaces "old" with "new":

```
replace('the old string', 'old', 'new')
```

And returns this result: "the new string"

skip

Remove items from the front of a collection, and return *all the other* items.

```
skip([<collection>], <count>)
```

Parameter	Required	Type	Description
<collection>	Yes	Array	The collection whose items you want to remove

Parameter	Required	Type	Description
<count>	Yes	Integer	A positive integer for the number of items to remove at the front

Return value	Type	Description
[<updated-collection>]	Array	The updated collection after removing the specified items

Example

This example removes one item, the number 0, from the front of the specified array:

```
skip(createArray(0, 1, 2, 3), 1)
```

And returns this array with the remaining items: [1,2,3]

split

Return an array that contains substrings, separated by commas, based on the specified delimiter character in the original string.

```
split('<text>', '<delimiter>')
```

Parameter	Required	Type	Description
<text>	Yes	String	The string to separate into substrings based on the specified delimiter in the original string
<delimiter>	Yes	String	The character in the original string to use as the delimiter

Return value	Type	Description
[<substring1>, <substring2>, ...]	Array	An array that contains substrings from the original string, separated by commas

Example

This example creates an array with substrings from the specified string based on the specified character as the delimiter:

```
split('a_b_c', '_')
```

And returns this array as the result: ["a", "b", "c"]

startOfDay

Return the start of the day for a timestamp.

```
startOfDay('<timestamp>', '<format>'?)
```

Parameter	Required	Type	Description
<timestamp>	Yes	String	The string that contains the timestamp
<format>	No	String	Either a single format specifier or a custom format pattern. The default format for the timestamp is "o" (yyyy-MM-ddTHH:mm:ss.fffffffK), which complies with ISO 8601 ↗ and preserves time zone information.

Return value	Type	Description
<updated-timestamp>	String	The specified timestamp but starting at the zero-hour mark for the day

Example

This example finds the start of the day for this timestamp:

```
startOfDay('2018-03-15T13:30:30Z')
```

And returns this result: "2018-03-15T00:00:00.0000000Z"

startOfHour

Return the start of the hour for a timestamp.

```
startOfHour('<timestamp>', '<format>'?)
```

Parameter	Required	Type	Description
<timestamp>	Yes	String	The string that contains the timestamp
<format>	No	String	Either a single format specifier or a custom format pattern. The default format for the timestamp is "o" (yyyy-MM-ddTHH:mm:ss.fffffffK), which complies with ISO 8601 ↗ and preserves time zone information.

Return value	Type	Description
<updated-timestamp>	String	The specified timestamp but starting at the zero-minute mark for the hour

Example

This example finds the start of the hour for this timestamp:

```
startOfHour('2018-03-15T13:30:30Z')
```

And returns this result: `"2018-03-15T13:00:00.0000000Z"`

startOfMonth

Return the start of the month for a timestamp.

```
startOfMonth('<timestamp>', '<format>'?)
```

Parameter	Required	Type	Description
<timestamp>	Yes	String	The string that contains the timestamp
<format>	No	String	Either a single format specifier or a custom format pattern. The default format for the timestamp is "o" (yyyy-MM-ddTHH:mm:ss.fffffffK), which complies with ISO 8601 ↗ and preserves time zone information.

Return value	Type	Description
<code><updated-timestamp></code>	String	The specified timestamp but starting on the first day of the month at the zero-hour mark

Example

This example returns the start of the month for this timestamp:

```
startOfMonth('2018-03-15T13:30:30Z')
```

And returns this result: `"2018-03-01T00:00:00.000000Z"`

startsWith

Check whether a string starts with a specific substring. Return true when the substring is found, or return false when not found. This function isn't case-sensitive.

```
startsWith('<text>', '<searchText>')
```

Parameter	Required	Type	Description
<code><text></code>	Yes	String	The string to check
<code><searchText></code>	Yes	String	The starting string to find

Return value	Type	Description
true or false	Boolean	Return true when the starting substring is found. Return false when not found.

Example 1

This example checks whether the "hello world" string starts with the "hello" substring:

```
startsWith('hello world', 'hello')
```

And returns this result: `true`

Example 2

This example checks whether the "hello world" string starts with the "greetings" substring:

```
startsWith('hello world', 'greetings')
```

And returns this result: `false`

string

Return the string version for a value.

```
string(<value>)
```

Parameter	Required	Type	Description
<code><value></code>	Yes	Any	The value to convert

Return value	Type	Description
<code><string-value></code>	String	The string version for the specified value

Example 1

This example creates the string version for this number:

```
string(10)
```

And returns this result: `"10"`

Example 2

This example creates a string for the specified JSON object and uses the backslash character (\) as an escape character for the double-quotation mark (").

```
string( { "name": "Sophie Owen" } )
```

And returns this result: `"{ \\\"name\\\": \\"Sophie Owen\\" }"`

sub

Return the result from subtracting the second number from the first number.

```
sub(<minuend>, <subtrahend>)
```

Parameter	Required	Type	Description
<code><minuend></code>	Yes	Integer or Float	The number from which to subtract the <i>subtrahend</i>
<code><subtrahend></code>	Yes	Integer or Float	The number to subtract from the <i>minuend</i>

Return value	Type	Description
<code><result></code>	Integer or Float	The result from subtracting the second number from the first number

Example

This example subtracts the second number from the first number:

```
sub(10.3, .3)
```

And returns this result: `10`

substring

Return characters from a string, starting from the specified position, or index. Index values start with the number 0.

```
substring('<text>', <startIndex>, <length>)
```

Parameter	Required	Type	Description
<text>	Yes	String	The string whose characters you want
<startIndex>	Yes	Integer	A positive number equal to or greater than 0 that you want to use as the starting position or index value
<length>	Yes	Integer	A positive number of characters that you want in the substring

Return value	Type	Description
<substring-result>	String	A substring with the specified number of characters, starting at the specified index position in the source string

Example

This example creates a five-character substring from the specified string, starting from the index value 6:

```
substring('hello world', 6, 5)
```

And returns this result: "world"

subtractFromTime

Subtract a number of time units from a timestamp. See also [getPastTime](#).

```
subtractFromTime('<timestamp>', <interval>, '<timeUnit>', '<format>'?)
```

Parameter	Required	Type	Description
<timestamp>	Yes	String	The string that contains the timestamp
<interval>	Yes	Integer	The number of specified time units to subtract
<timeUnit>	Yes	String	The unit of time to use with <i>interval</i> : "Second", "Minute", "Hour", "Day", "Week", "Month", "Year"
<format>	No	String	Either a single format specifier or a custom format pattern. The default format for the timestamp is "o" (yyyy-MM-

Parameter	Required	Type	Description
			ddTHH:mm:ss.ffffffffK), which complies with ISO 8601 ↗ and preserves time zone information.

Return value	Type	Description
<updated-timestamp>	String	The timestamp minus the specified number of time units

Example 1

This example subtracts one day from this timestamp:

```
subtractFromTime('2018-01-02T00:00:00Z', 1, 'Day')
```

And returns this result: "2018-01-01T00:00:00:000000Z"

Example 2

This example subtracts one day from this timestamp:

```
subtractFromTime('2018-01-02T00:00:00Z', 1, 'Day', 'D')
```

And returns this result using the optional "D" format: "Monday, January, 1, 2018"

take

Return items from the front of a collection.

```
take('<collection>', <count>)
take([<collection>], <count>)
```

Parameter	Required	Type	Description
<collection>	Yes	String or Array	The collection whose items you want
<count>	Yes	Integer	A positive integer for the number of items that you want from the front

Return value	Type	Description
<subset> or [<subset>]	String or Array, respectively	A string or array that has the specified number of items taken from the front of the original collection

Example

These examples get the specified number of items from the front of these collections:

```
take('abcde', 3)
take(createArray(0, 1, 2, 3, 4), 3)
```

And return these results:

- First example: "abc"
- Second example: [0, 1, 2]

ticks

Return the `ticks` property value for a specified timestamp. A *tick* is a 100-nanosecond interval.

```
ticks('<timestamp>')
```

Parameter	Required	Type	Description
<timestamp>	Yes	String	The string for a timestamp

Return value	Type	Description
<ticks-number>	Integer	The number of ticks that have elapsed since 12:00:00 midnight, January 1, 0001 in the Gregorian calendar since the input timestamp

toLower

Return a string in lowercase format. If a character in the string doesn't have a lowercase version, that character stays unchanged in the returned string.

```
toLowerCase('<text>')
```

Parameter	Required	Type	Description
<text>	Yes	String	The string to return in lowercase format

Return value	Type	Description
<lowercase-text>	String	The original string in lowercase format

Example

This example converts this string to lowercase:

```
toLowerCase('Hello World')
```

And returns this result: "hello world"

toUpperCase

Return a string in uppercase format. If a character in the string doesn't have an uppercase version, that character stays unchanged in the returned string.

```
toUpperCase('<text>')
```

Parameter	Required	Type	Description
<text>	Yes	String	The string to return in uppercase format

Return value	Type	Description
<uppercase-text>	String	The original string in uppercase format

Example

This example converts this string to uppercase:

```
toUpper('Hello World')
```

And returns this result: "HELLO WORLD"

trim

Remove leading and trailing whitespace from a string, and return the updated string.

```
trim('<text>')
```

Parameter	Required	Type	Description
<text>	Yes	String	The string that has the leading and trailing whitespace to remove

Return value	Type	Description
<updatedText>	String	An updated version for the original string without leading or trailing whitespace

Example

This example removes the leading and trailing whitespace from the string " Hello World ":

```
trim(' Hello World  ')
```

And returns this result: "Hello World"

union

Return a collection that has *all* the items from the specified collections. To appear in the result, an item can appear in any collection passed to this function. If one or more items have the same name, the last item with that name appears in the result.

```
union('<collection1>', '<collection2>', ...)
```

```
union([<collection1>], [<collection2>], ...)
```

Parameter	Required	Type	Description
<collection1>, <collection2>, ...	Yes	Array or Object, but not both	The collections from where you want <i>all</i> the items

Return value	Type	Description
<updatedCollection>	Array or Object, respectively	A collection with all the items from the specified collections - no duplicates

Example

This example gets *all* the items from these collections:

```
union(createArray(1, 2, 3), createArray(1, 2, 10, 101))
```

And returns this result: [1, 2, 3, 10, 101]

uriComponent

Return a uniform resource identifier (URI) encoded version for a string by replacing URL-unsafe characters with escape characters. Use this function rather than `encodeUriComponent()`. Although both functions work the same way, `uriComponent()` is preferred.

```
uriComponent('<value>')
```

Parameter	Required	Type	Description
<value>	Yes	String	The string to convert to URI-encoded format

Return value	Type	Description
<encoded-uri>	String	The URI-encoded string with escape characters

Example

This example creates a URI-encoded version for this string:

```
uriComponent('https://contoso.com')
```

And returns this result: "http%3A%2F%2Fcontoso.com"

uriComponentToBinary

Return the binary version for a uniform resource identifier (URI) component.

```
uriComponentToBinary('<value>')
```

Parameter	Required	Type	Description
<value>	Yes	String	The URI-encoded string to convert

Return value	Type	Description
<binary-for-encoded-uri>	String	The binary version for the URI-encoded string. The binary content is base64-encoded and represented by \$content.

Example

This example creates the binary version for this URI-encoded string:

```
uriComponentToBinary('http%3A%2F%2Fcontoso.com')
```

And returns this result:

```
"001000100110100001110100011101000111000000100101001100  
11010000010010010100110010010001100010010010100110010010001  
10011000110110111101101110011101000110111101110011011011  
110010111001100011011011110110110100100010"
```

uriComponentToString

Return the string version for a uniform resource identifier (URI) encoded string, effectively decoding the URI-encoded string.

```
uriComponentToString('<value>')
```

Parameter	Required	Type	Description
<value>	Yes	String	The URI-encoded string to decode

Return value	Type	Description
<decoded-uri>	String	The decoded version for the URI-encoded string

Example

This example creates the decoded string version for this URL-encoded string:

```
uriComponentToString('http%3A%2F%2Fcontoso.com')
```

And returns this result: "https://contoso.com"

utcNow

Return the current timestamp.

```
utcNow('<format>')
```

Optionally, you can specify a different format with the <format> parameter.

Parameter	Required	Type	Description
<format>	No	String	Either a single format specifier or a custom format pattern. The default format for the timestamp is "o" (yyyy-MM-ddTHH:mm:ss.fffffffK), which complies with ISO 8601 and preserves time zone information.

Return value	Type	Description
<current-timestamp>	String	The current date and time

Example 1

Suppose today is April 15, 2018 at 1:00:00 PM. This example gets the current timestamp:

```
utcNow()
```

And returns this result: "2018-04-15T13:00:00.000000Z"

Example 2

Suppose today is April 15, 2018 at 1:00:00 PM. This example gets the current timestamp using the optional "D" format:

```
utcNow('D')
```

And returns this result: "Sunday, April 15, 2018"

xml

Return the XML version for a string that contains a JSON object.

```
xml('<value>')
```

Parameter	Required	Type	Description
<value>	Yes	String	The string with the JSON object to convert The JSON object must have only one root property, which can't be an array. Use the backslash character (\) as an escape character for the double quotation mark (").

Return value	Type	Description
<xml-version>	Object	The encoded XML for the specified string or JSON object

Example 1

This example creates the XML version for this string, which contains a JSON object:

```
xml(json('{ \"name\": \"Sophia Owen\" }'))
```

And returns this result XML:

```
XML
```

```
<name>Sophia Owen</name>
```

Example 2

Suppose you have this JSON object:

```
JSON
```

```
{
  "person": {
    "name": "Sophia Owen",
    "city": "Seattle"
  }
}
```

This example creates XML for a string that contains this JSON object:

```
xml(json('{"person": {"name": "Sophia Owen", "city": "Seattle"}'))
```

And returns this result XML:

```
XML
```

```
<person>
  <name>Sophia Owen</name>
  <city>Seattle</city>
<person>
```

xpath

Check XML for nodes or values that match an XPath (XML Path Language) expression, and return the matching nodes or values. An XPath expression, or just "XPath", helps you navigate an XML document structure so that you can select nodes or compute values in the XML content.

```
xpath('<xml>', '<xpath>')
```

Parameter	Required	Type	Description
<xml>	Yes	Any	The XML string to search for nodes or values that match an XPath expression value
<xpath>	Yes	Any	The XPath expression used to find matching XML nodes or values

Return value	Type	Description
<xml-node>	XML	An XML node when only a single node matches the specified XPath expression
<value>	Any	The value from an XML node when only a single value matches the specified XPath expression
[<xml-node1>, <xml-node2>, ...] -or- [<value1>, <value2>, ...]	Array	An array with XML nodes or values that match the specified XPath expression

Example 1

Following on Example 1, this example finds nodes that match the <count></count> node and adds those node values with the `sum()` function:

```
xpath(xml(parameters('items')), 'sum(/produce/item/count)')
```

And returns this result: 30

Example 2

For this example, both expressions find nodes that match the <location></location> node, in the specified arguments, which include XML with a namespace. The expressions use the backslash character (\) as an escape character for the double quotation mark (").

- *Expression 1*

```
xpath(xml(body('Http')), '/*[name()=\"file\"]/*[name()=\"location\"]')
```

- *Expression 2*

```
xpath(xml(body('Http')), '/*[local-name()=\"file\" and namespace-uri()=\"http://contoso.com\"]/*[local-name()=\"location\"]')
```

Here are the arguments:

- This XML, which includes the XML document namespace,

```
xmlns="http://contoso.com":
```

XML

```
<?xml version="1.0"?> <file xmlns="http://contoso.com">  
<location>Paris</location> </file>
```

- Either XPath expression here:

- `/*[name()=\"file\"]/*[name()=\"location\"]`
- `/*[local-name()=\"file\" and namespace-uri()=\"http://contoso.com\"]/*[local-name()=\"location\"]`

Here's the result node that matches the `<location></location>` node:

XML

```
<location xmlns="https://contoso.com">Paris</location>
```

Example 3

Following on Example 3, this example finds the value in the `<location></location>` node:

```
xpath(xml(body('Http')), 'string(/*[name()=\"file\"]/*[name()=\"location\"]))')
```

And returns this result: "Paris"

① Note

One can add comments to data flow expressions, but not in pipeline expressions.

Next steps

For instructions on general parameter usage, refer to Parameters for Data Factory in Fabric

Feedback

Was this page helpful?

 Yes

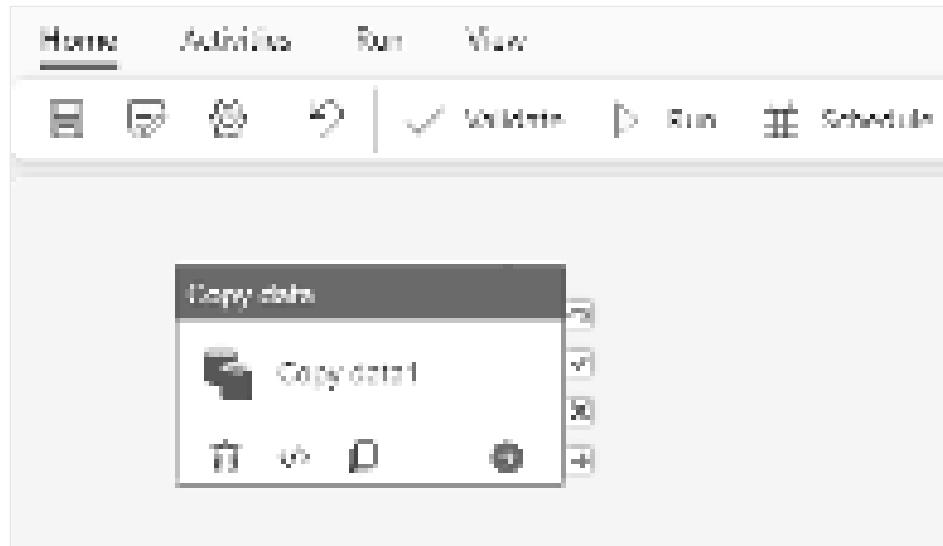
 No

Provide product feedback  | Ask the community 

Concept: Data pipeline Runs

Article • 11/15/2023

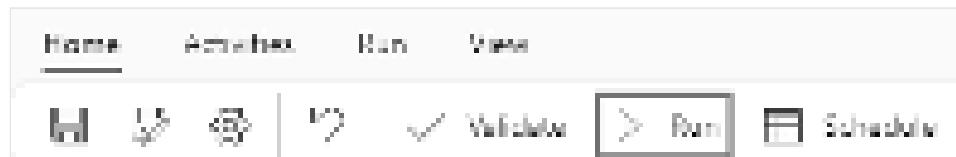
A data pipeline run occurs when a data pipeline is executed. This means that the activities in your data pipeline will run and be executed to completion. For example, running a data pipeline with a **Copy data** activity will perform that action and copy your data. Each data pipeline run will have its own unique pipeline run ID.



A data pipeline run can be triggered one of two ways, either on-demand or by setting up a schedule. A scheduled pipeline will be able to run based on the time and frequency that you set.

On-demand data pipeline run

To manually trigger a data pipeline run, select **Run** found in the top banner of the **Home** tab.



You'll be prompted to save your changes before triggering the pipeline run. Select **Save and run** to continue.



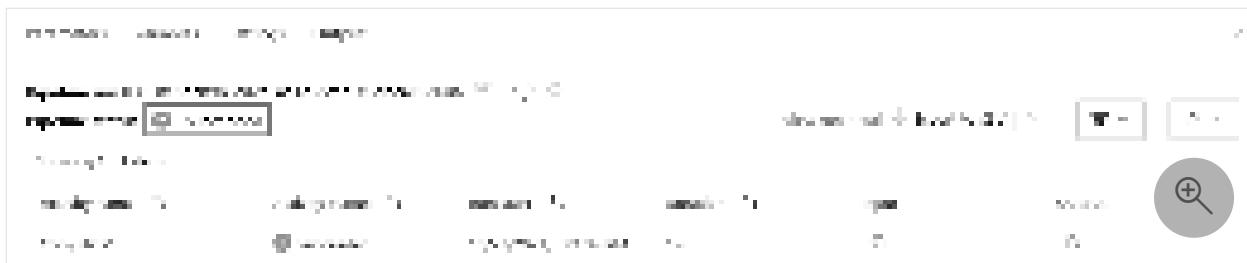
After your changes are saved, your pipeline will run. You can view the progress of the run in the **Output** tab found at the bottom of the canvas.



Once an activity has completed in a run, a green check mark appears in the corner of the activity.

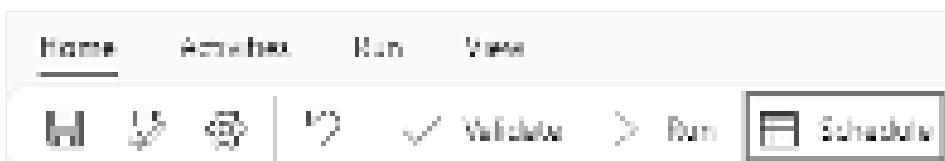


Once the entire pipeline has been executed and the output status updates to **Succeeded**, you've had a successful pipeline run!

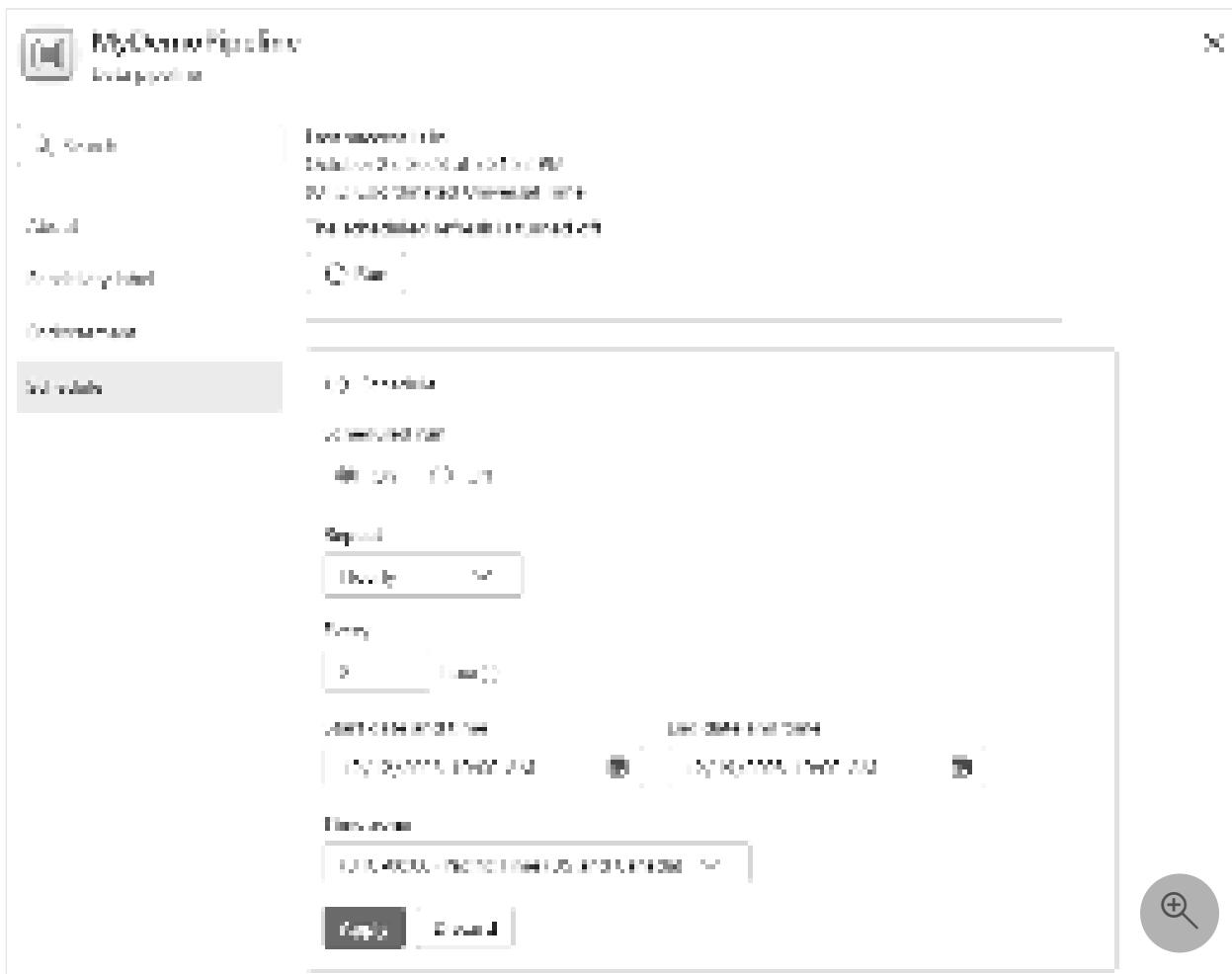


Scheduled data pipeline runs

When you schedule a data pipeline run, you can choose the frequency that your pipeline runs. Select **Schedule**, found in the top banner of the **Home** tab, to view your options. By default, your data pipeline won't be set on a schedule.



On the Schedule configuration page, you can specify a schedule frequency, start and end dates and times, and time zone.



Once configured, select **Apply** to set your schedule. You can view or edit the schedule again anytime by selecting the **Schedule** button again.

Next steps

- How to monitor data pipeline runs in Microsoft Fabric
 - Quickstart: Create your first data pipeline to copy data
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Templates for Data Factory in Microsoft Fabric

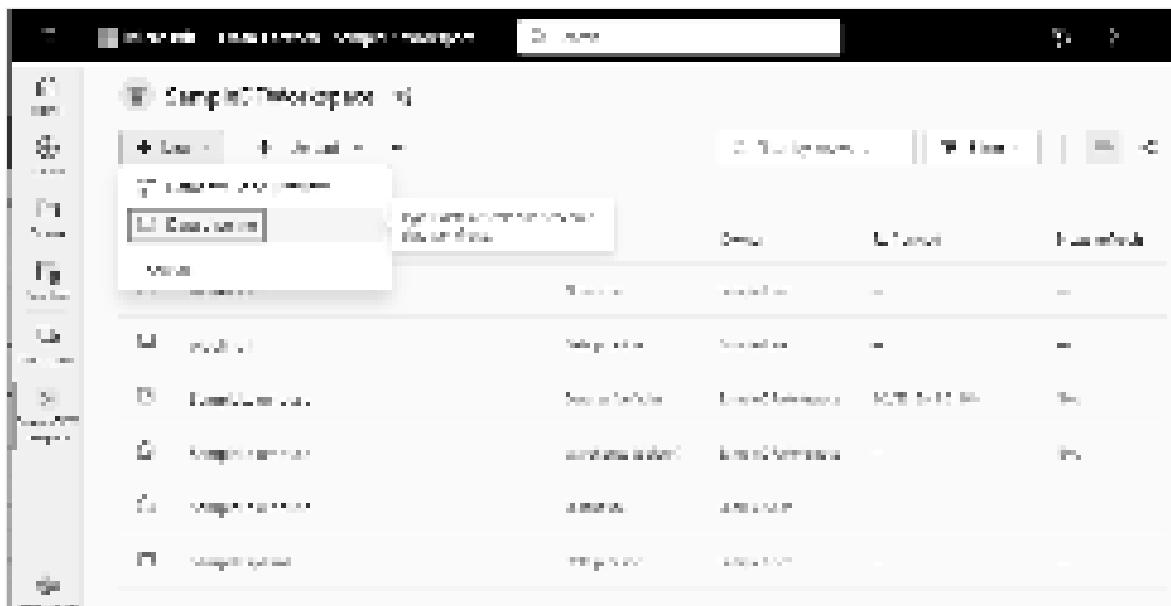
Article • 11/15/2023

Templates are pre-defined pipelines that allow you to get started quickly with Data Factory. These templates help to reduce development time by providing an easy way to create pipelines for common data integration scenarios.



How to build a pipeline from a template

1. To get started with a template, start by selecting New and then Data pipeline.

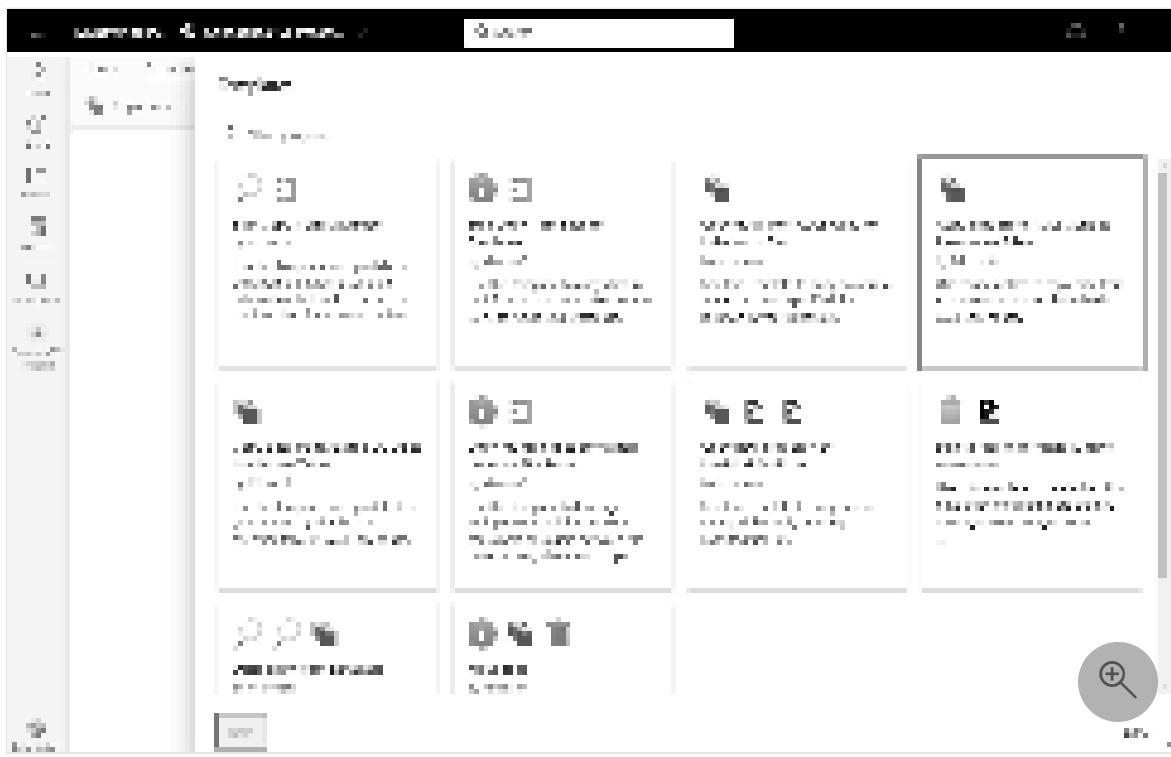


2. Set a name for your new pipeline and select **Create**. The pipeline editor window opens.

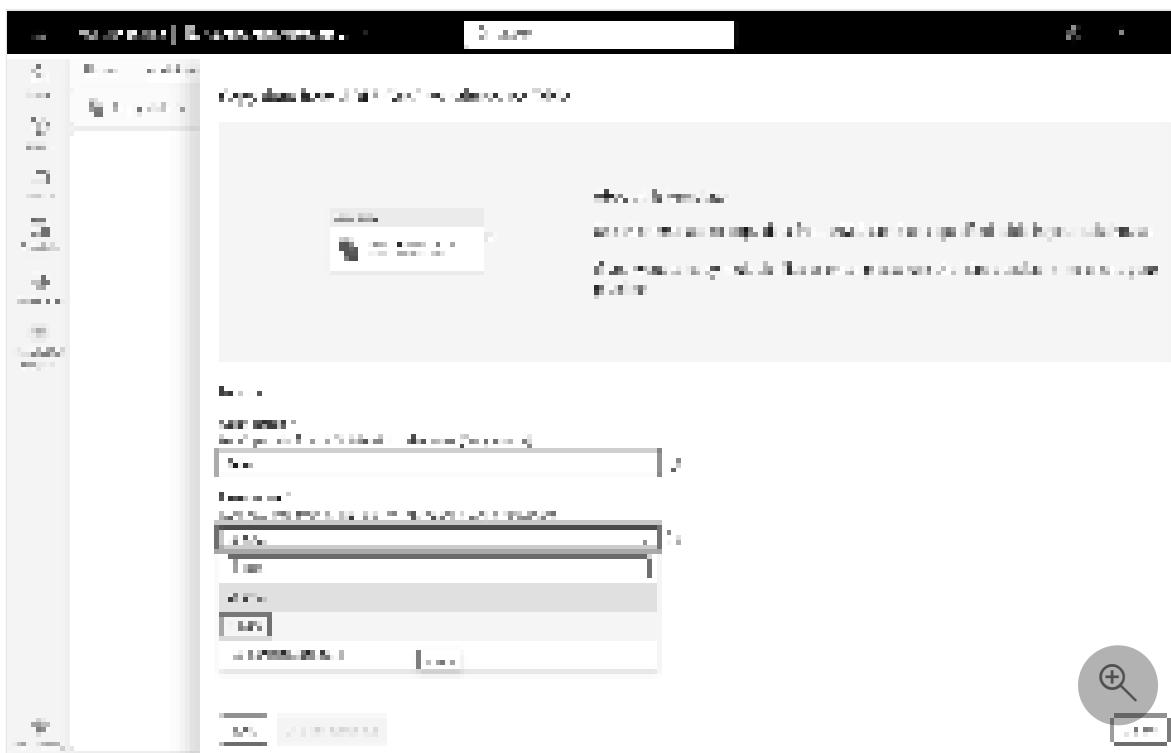
3. Select **Choose a task to start**.



4. The template browser appears. Select the **Copy data from ADLS Gen2 to Lakehouse Table** template, and then select **Next**.



5. Select from the drop-down list your source and destination connections or use + New to create a new connection.



6. After making your selections or creating new connections, select the **Use this template** button to generate a new pipeline directly.
7. The pipeline is created using the connections you set. You can view the newly created pipeline in the pipeline editor, where you can use the **Run** and **Schedule** buttons to control its execution.



8. Initially the new Copy activity is highlighted on the editor canvas, and its properties shown in the properties pane at the bottom of the editor.
9. When you select the background of the pipeline canvas, you can see the general pipeline properties in the properties pane, where you can add or edit existing parameters.

A screenshot of the Azure Data Factory pipeline editor focusing on the 'Parameters' pane. It shows a table with four rows of parameters. Each row has three columns: 'Name' (checkbox), 'Type' (dropdown set to 'String'), and 'Default value' (text input field). The parameters listed are FilesystemName, DirectoryName, FileName, and UriTableName.

10. When you're done, save your edits by selecting the save button on the toolbar of the **Home** tab.



11. Now you can run your pipeline, providing values for any parameters as required.



Next steps

How to monitor pipeline runs in Microsoft Fabric

Feedback

Was this page helpful?

Yes

No

Provide product feedback [↗](#) | Ask the community [↗](#)

How to monitor data pipeline runs in Microsoft Fabric

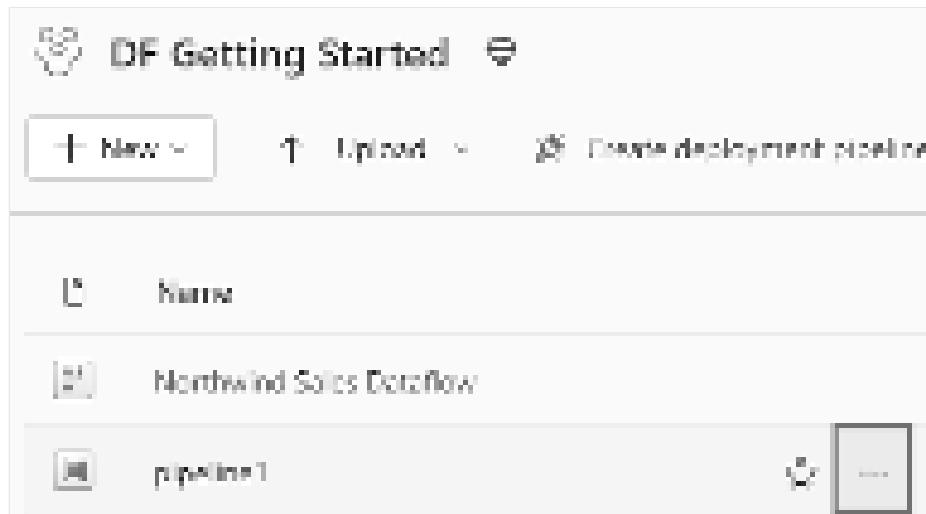
Article • 11/15/2023

In this how-to guide, you'll learn the different ways to review and monitor your pipeline runs.

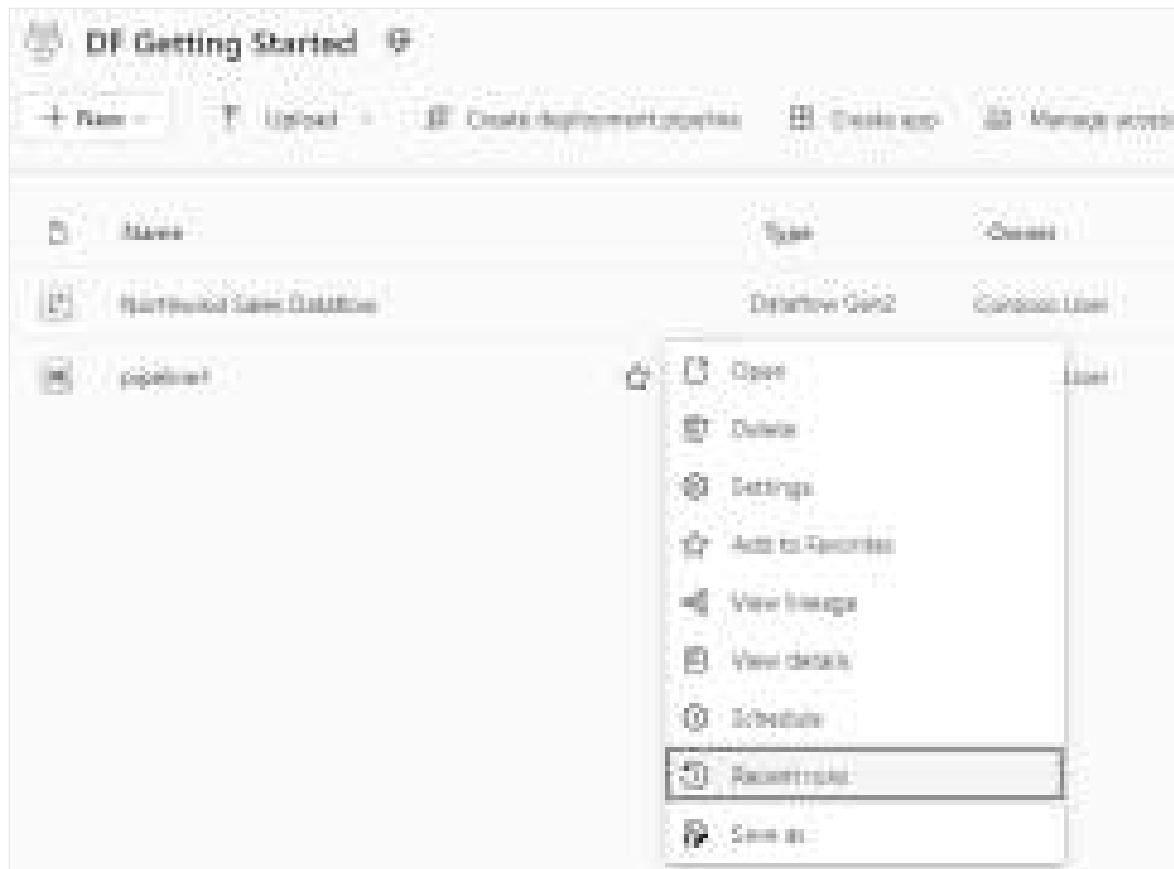
Monitor data pipeline runs

1. To monitor your data pipeline runs, hover over your pipeline in your workspace.

Doing so will bring up three dots to the right of your pipeline name.



2. Select the three dots to find a list of options. Then select **View run history**. This action opens a fly-out on the right side of your screen with all your recent runs and run statuses.



My Lambda Functions		Execution results for Pipeline			
	Name	Last Run	Duration	State	Logs
0	MyLambdaFunction	1 hour ago	00:00:00.000	Success	
1	MyLambdaFunction2	1 hour ago	00:00:00.000	Success	
2	MyLambdaFunction3	1 hour ago	00:00:00.000	Success	
3	MyLambdaFunction4	1 hour ago	00:00:00.000	Success	
4	MyLambdaFunction5	1 hour ago	00:00:00.000	Success	

3. Select **Go to monitoring hub** from the prior screenshot to view more details and filter results. Use the filter to find specific data pipeline runs based on several criteria.

Monitoring Hub					
Search for the name of the pipeline or run ID, or use the dropdown to select a specific run.					
Filter	Actions	Run ID	Run Type	Run Status	Run Details
Run ID	Run ID	11111111111111111111111111111111	Batch	Success	
Run ID	Run ID	22222222222222222222222222222222	Batch	Success	
Run ID	Run ID	33333333333333333333333333333333	Batch	Success	
Run ID	Run ID	44444444444444444444444444444444	Batch	Success	
Run ID	Run ID	55555555555555555555555555555555	Batch	Success	

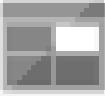
4. Select one of your pipeline runs to view detailed information. You'll be able to view what your pipeline looks like and view more properties like Run ID or errors if your pipeline run failed.



5. To find additional information on your pipeline runs **Input** and **Output**, select the input or output links to the right of the relevant row in the Activity Runs.
6. You can select **Update pipeline** to make changes to your pipeline from this screen. This selection will take you back in the pipeline canvas.
7. You can also **Rerun** your data pipeline. You can choose to rerun the entire pipeline or only rerun the pipeline from the failed activity.
8. To view performance details, select an activity from the list of **Activity Runs**. Performance details will pop up.

Copy data details

Copy detail

Source	Destination
 Azure Blob Storage	 Tabular
Data read: 11	Data written: 11
Files read: 10	File written: 1
Blocks read: 93,557	Row written: 60,000

Status:  Succeeded
Start time: 10/27/2023, 3:21:34 PM

Close

More details can be found under **Duration breakdown** and **Advanced**.

Copy data details

 Azure Blob Storage →  Lakehouse

Data read:	504,515 KB	Data written:	2,053,011 KB
File read:	1	File written:	1
Rows read:	69,557	Rows written:	69,557

Status:	Success
Start time:	10/27/2023, 02:11:14 PM

Activity run ID:	09e794D3-e147-4f6c-95ca-041E0d924C09
Throughput:	65,765 KB/s
Total duration:	00:00:15

Duration breakdown

Start time:	10/27/2023, 02:11:14 PM
Optimized throughput:	Standard
Used parallel copies:	1

Copy: 

Transfer: 

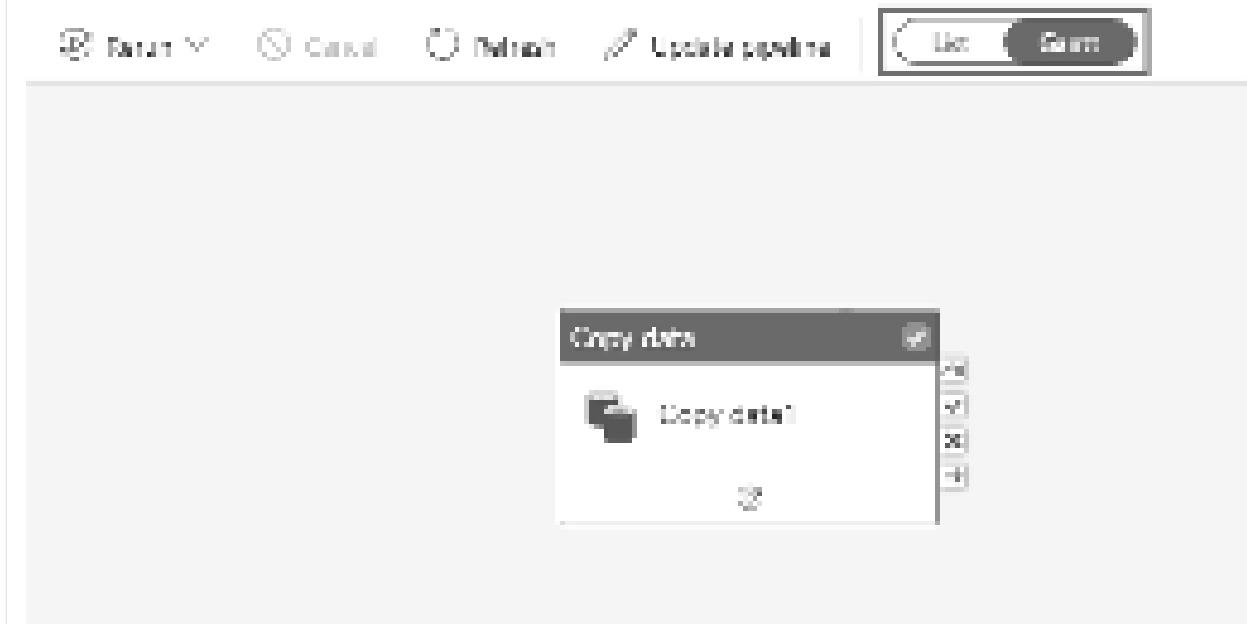
> Advanced

Close

Gantt view

A Gantt chart is a view that lets you see the run history over a time range. If you switch to a Gantt view, all pipeline runs will be grouped by name, displayed as bars relative to how long the run took.

Monitoring hub > MyDemoPipeline



The length of the bar relates to the duration of the pipeline. You can select the bar to view more details.

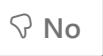


Next steps

- Quickstart: Create your first data pipeline to copy data
 - Quickstart: Create your first Dataflow Gen2 to get and transform data
-

Feedback

Was this page helpful?



Provide product feedback [↗](#) | Ask the community [↗](#)

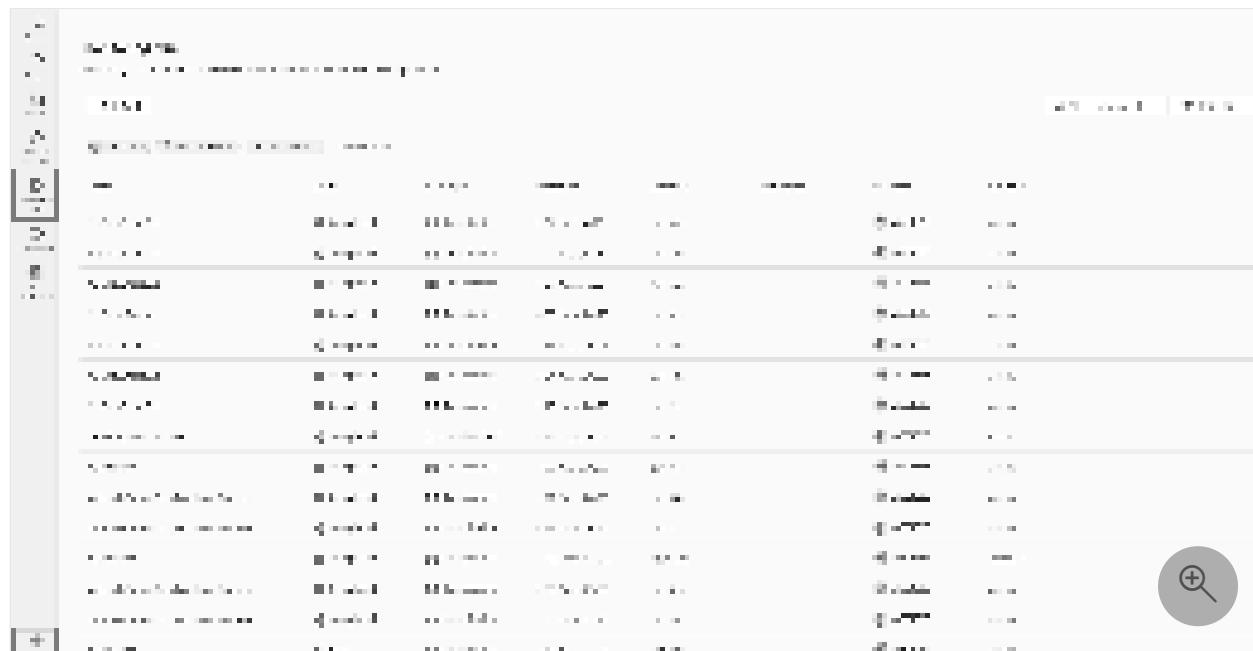
Browse data pipeline runs in the Monitoring hub

Article • 11/15/2023

The Monitoring hub serves as a centralized portal for browsing data pipeline runs across items, when you are in the **Data Factory** or **Data Engineering** experience.

Access the monitoring hub

In the **Data Factory** or **Data Engineering** experience, you can access the Monitoring hub to view various data pipeline runs by selecting **Monitoring hub** in the left-side navigation links.



Sort, search and filter data pipeline runs

For better usability and discoverability, you can sort the data pipeline runs by selecting different columns in the UI. You can also filter the pipeline runs based on different columns and search for specific pipeline runs.

Sort data pipeline runs

To sort data pipeline runs, you can select on each column header, such as **Name**, **Status**, **Item type**, **Start time**, **Location**, **Run kind**, and so on.

The screenshot shows a table of data pipeline runs with the following columns: Run ID, Item Type, Status, Start Time, End Time, Duration, Submitter, and Location. The table contains 15 rows of data. A magnifying glass icon is located in the bottom right corner of the table area.

Run ID	Item Type	Status	Start Time	End Time	Duration	Submitter	Location	Action
run-12345	File upload	Completed	2023-01-01T00:00:00Z	2023-01-01T00:05:00Z	5 minutes	User-A	New York	
run-67890	Image analysis	Completed	2023-01-01T00:10:00Z	2023-01-01T00:15:00Z	5 minutes	User-B	London	
run-12345	File upload	Completed	2023-01-01T00:20:00Z	2023-01-01T00:25:00Z	5 minutes	User-C	Paris	
run-67890	Image analysis	Completed	2023-01-01T00:30:00Z	2023-01-01T00:35:00Z	5 minutes	User-D	London	
run-12345	File upload	Completed	2023-01-01T00:40:00Z	2023-01-01T00:45:00Z	5 minutes	User-E	New York	
run-67890	Image analysis	Completed	2023-01-01T00:50:00Z	2023-01-01T00:55:00Z	5 minutes	User-F	London	
run-12345	File upload	Completed	2023-01-01T01:00:00Z	2023-01-01T01:05:00Z	5 minutes	User-G	New York	
run-67890	Image analysis	Completed	2023-01-01T01:10:00Z	2023-01-01T01:15:00Z	5 minutes	User-H	London	
run-12345	File upload	Completed	2023-01-01T01:20:00Z	2023-01-01T01:25:00Z	5 minutes	User-I	New York	
run-67890	Image analysis	Completed	2023-01-01T01:30:00Z	2023-01-01T01:35:00Z	5 minutes	User-J	London	
run-12345	File upload	Completed	2023-01-01T01:40:00Z	2023-01-01T01:45:00Z	5 minutes	User-K	New York	
run-67890	Image analysis	Completed	2023-01-01T01:50:00Z	2023-01-01T01:55:00Z	5 minutes	User-L	London	
run-12345	File upload	Completed	2023-01-01T02:00:00Z	2023-01-01T02:05:00Z	5 minutes	User-M	New York	
run-67890	Image analysis	Completed	2023-01-01T02:10:00Z	2023-01-01T02:15:00Z	5 minutes	User-N	London	
run-12345	File upload	Completed	2023-01-01T02:20:00Z	2023-01-01T02:25:00Z	5 minutes	User-O	New York	
run-67890	Image analysis	Completed	2023-01-01T02:30:00Z	2023-01-01T02:35:00Z	5 minutes	User-P	London	

Filter data pipeline runs

You can filter data pipeline runs by **Status**, **Item Type**, **Start Time**, **Submitter**, and **Location** using the Filter pane in the upper-right corner.

The screenshot shows a table of data pipeline runs with the following columns: Run ID, Item Type, Status, Start Time, End Time, Duration, Submitter, and Location. The table contains 20 rows of data. A magnifying glass icon is located in the bottom right corner of the table area.

Run ID	Item Type	Status	Start Time	End Time	Duration	Submitter	Location	Action
run-12345	File upload	Completed	2023-01-01T00:00:00Z	2023-01-01T00:05:00Z	5 minutes	User-A	New York	
run-67890	Image analysis	Completed	2023-01-01T00:10:00Z	2023-01-01T00:15:00Z	5 minutes	User-B	London	
run-12345	File upload	Completed	2023-01-01T00:20:00Z	2023-01-01T00:25:00Z	5 minutes	User-C	Paris	
run-67890	Image analysis	Completed	2023-01-01T00:30:00Z	2023-01-01T00:35:00Z	5 minutes	User-D	London	
run-12345	File upload	Completed	2023-01-01T00:40:00Z	2023-01-01T00:45:00Z	5 minutes	User-E	New York	
run-67890	Image analysis	Completed	2023-01-01T00:50:00Z	2023-01-01T00:55:00Z	5 minutes	User-F	London	
run-12345	File upload	Completed	2023-01-01T01:00:00Z	2023-01-01T01:05:00Z	5 minutes	User-G	New York	
run-67890	Image analysis	Completed	2023-01-01T01:10:00Z	2023-01-01T01:15:00Z	5 minutes	User-H	London	
run-12345	File upload	Completed	2023-01-01T01:20:00Z	2023-01-01T01:25:00Z	5 minutes	User-I	New York	
run-67890	Image analysis	Completed	2023-01-01T01:30:00Z	2023-01-01T01:35:00Z	5 minutes	User-J	London	
run-12345	File upload	Completed	2023-01-01T01:40:00Z	2023-01-01T01:45:00Z	5 minutes	User-K	New York	
run-67890	Image analysis	Completed	2023-01-01T01:50:00Z	2023-01-01T01:55:00Z	5 minutes	User-L	London	
run-12345	File upload	Completed	2023-01-01T02:00:00Z	2023-01-01T02:05:00Z	5 minutes	User-M	New York	
run-67890	Image analysis	Completed	2023-01-01T02:10:00Z	2023-01-01T02:15:00Z	5 minutes	User-N	London	
run-12345	File upload	Completed	2023-01-01T02:20:00Z	2023-01-01T02:25:00Z	5 minutes	User-O	New York	
run-67890	Image analysis	Completed	2023-01-01T02:30:00Z	2023-01-01T02:35:00Z	5 minutes	User-P	London	
run-12345	File upload	Completed	2023-01-01T02:40:00Z	2023-01-01T02:45:00Z	5 minutes	User-Q	New York	
run-67890	Image analysis	Completed	2023-01-01T02:50:00Z	2023-01-01T02:55:00Z	5 minutes	User-R	London	
run-12345	File upload	Completed	2023-01-01T03:00:00Z	2023-01-01T03:05:00Z	5 minutes	User-S	New York	
run-67890	Image analysis	Completed	2023-01-01T03:10:00Z	2023-01-01T03:15:00Z	5 minutes	User-T	London	
run-12345	File upload	Completed	2023-01-01T03:20:00Z	2023-01-01T03:25:00Z	5 minutes	User-U	New York	
run-67890	Image analysis	Completed	2023-01-01T03:30:00Z	2023-01-01T03:35:00Z	5 minutes	User-V	London	
run-12345	File upload	Completed	2023-01-01T03:40:00Z	2023-01-01T03:45:00Z	5 minutes	User-W	New York	
run-67890	Image analysis	Completed	2023-01-01T03:50:00Z	2023-01-01T03:55:00Z	5 minutes	User-X	London	
run-12345	File upload	Completed	2023-01-01T04:00:00Z	2023-01-01T04:05:00Z	5 minutes	User-Y	New York	
run-67890	Image analysis	Completed	2023-01-01T04:10:00Z	2023-01-01T04:15:00Z	5 minutes	User-Z	London	

Search data pipeline runs

To search for specific data pipelines, you can enter certain keywords in the search box located in the upper-right corner.



Manage a data pipeline run

When you hover over a data pipeline run row, you can see various row-level actions that enable you to manage a particular data pipeline run.

View data pipeline run detail pane

You can hover over a data pipeline run row and click the **View detail** icon to open the **Detail** pane and view more details about a data pipeline run.



Retry a data pipeline run

If you need to retry a completed data pipeline run, hover over its row and click the Retry icon.



Navigate to data pipeline run detail view

If you need more information about detail activity runs of the data pipeline run, you can click on the name of a data pipeline run to navigate to its corresponding data pipeline run detail page.



Next steps

- Quickstart: Create your first data pipeline to copy data
 - Quickstart: Create your first Dataflow Gen2 to get and transform data
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback [↗](#) | Ask the community [↗](#)

Run a Script activity and send an output summary by email

Article • 11/15/2023

This example shows how to run a script on an Azure SQL instance and send an output summary by email.

Prerequisites

To get started, you must complete the following prerequisites:

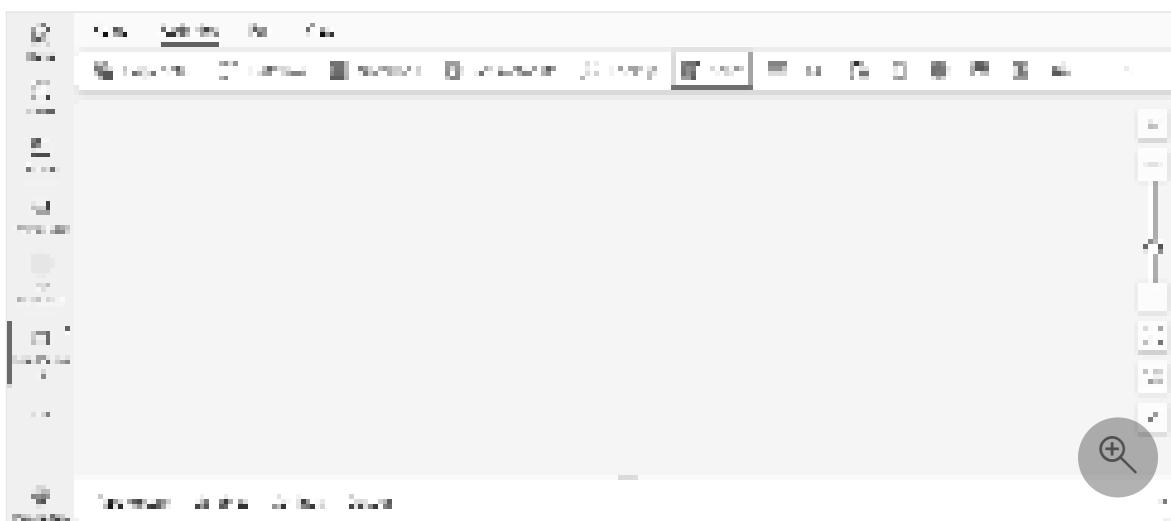
- A tenant account with an active subscription. Create an account for free.
- A workspace is created.

Create a script activity and configure its output to be sent by email

In this example, you create a Script activity to execute a script on an Azure SQL instance, and then send an output summary by email.

Create the Script activity

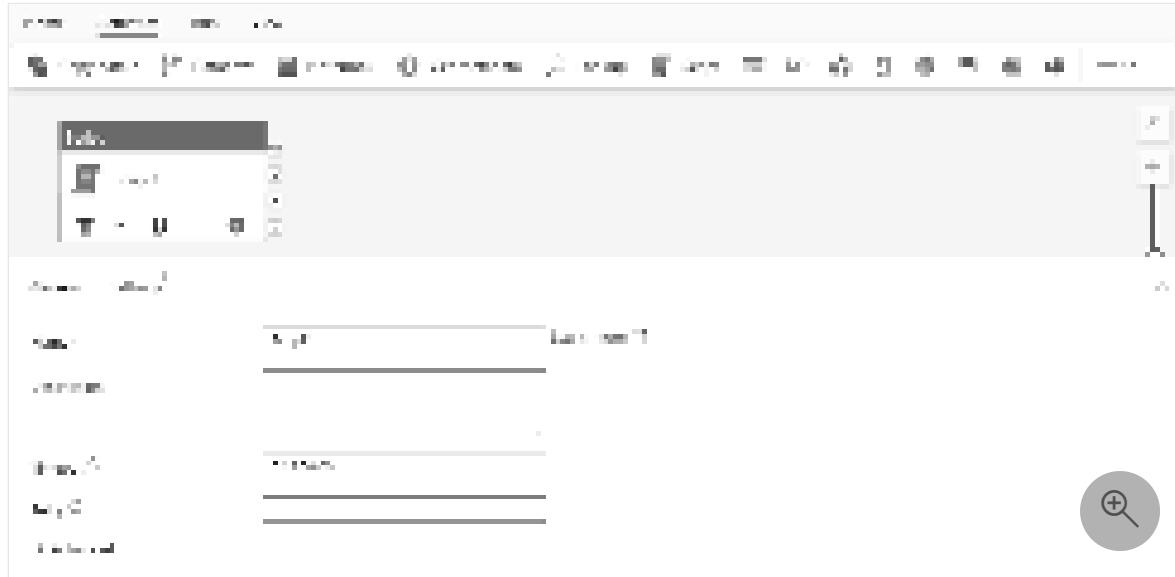
1. Create a new pipeline in your workspace.
2. Search for Script activity in the pipeline **Activities** pane, and select it to add it to the pipeline canvas.



ⓘ Note

Depending on the size of your browser window, the word **Script** may be hidden, showing only the icon.

3. Select the new Script activity on the canvas if it isn't already selected.



Refer to the General settings guidance to configure the **General** settings tab.

Create the connection and specify a SQL script

1. Select the **Settings** tab, and then select an existing Azure SQL or Azure SQL Managed Instance connection from the **Connection** dropdown, or use the **+ New** button to create a new connection, and specify its configuration details.
2. Specify a script. You can execute any query, or non-query scripts that perform operations on the database as well, in the **Script** textbox on the Script activity settings page. For this example, you can use this simple SQL script:

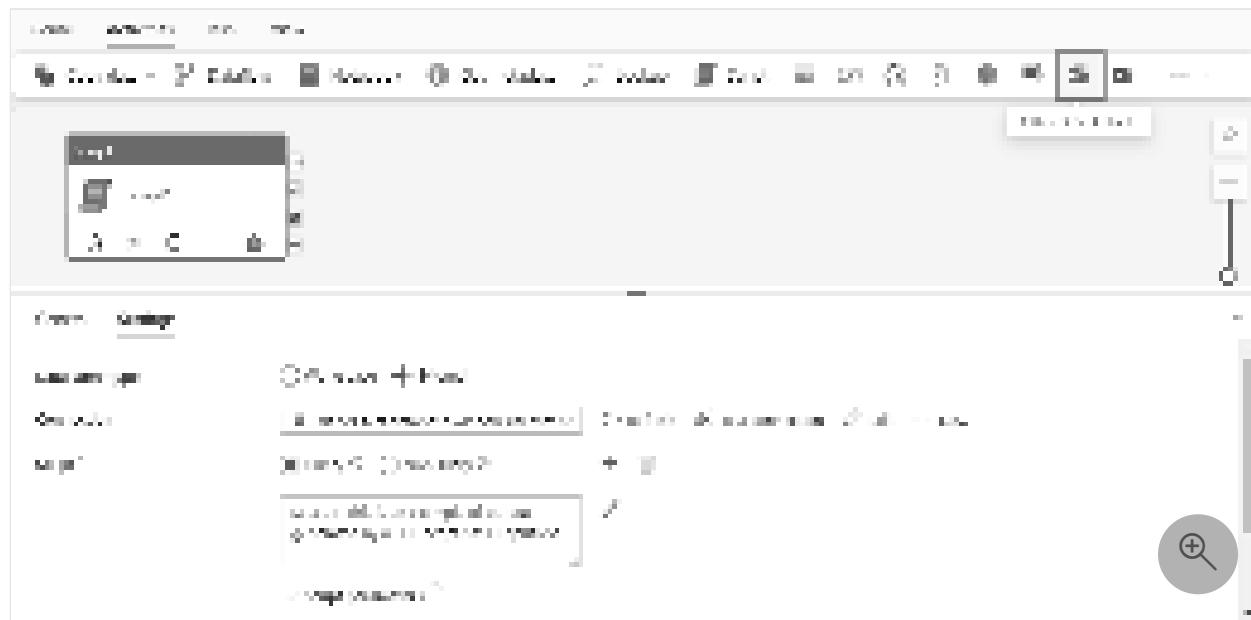
SQL

```
SELECT 'This is an example of output generated by a SQL script.' as OutputText
```



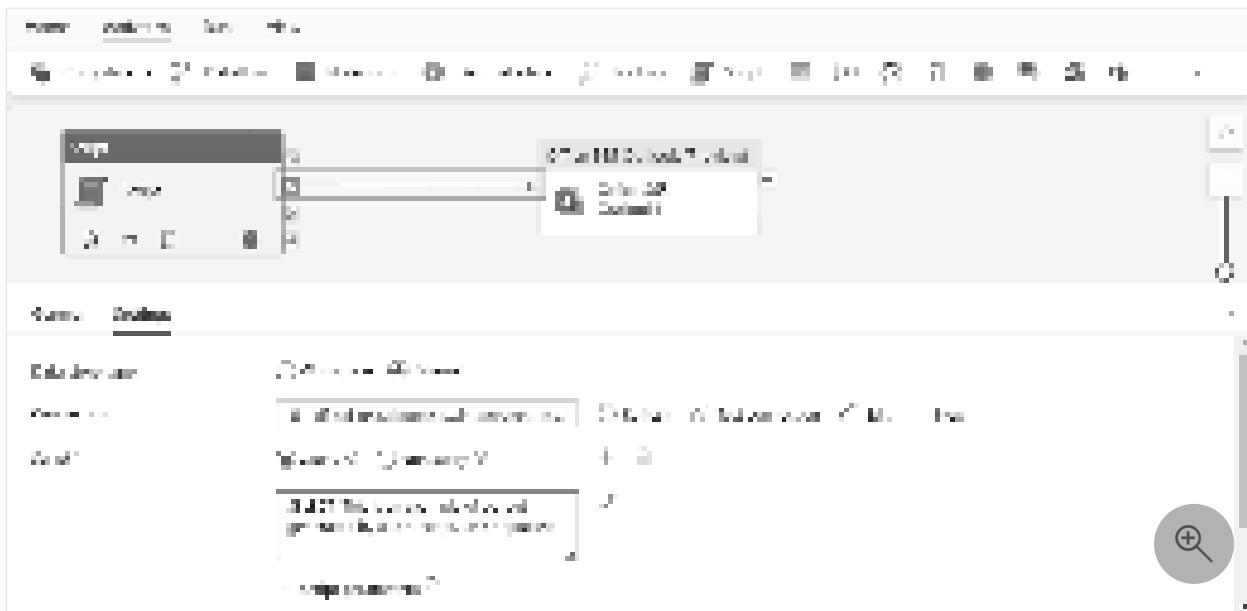
Create an Office 365 Outlook activity

Search for the **Office 365 Outlook** activity on the **Activities** toolbar. The text description may not be displayed if the window size isn't large enough, so you can look for the icon or expand the window if necessary.



Link the activities

Select the Script activity, and then drag its **On success** output (a green checkmark on the right side of the activity in the editor pane) to the Office 365 Outlook activity.



Configure the Office 365 Outlook activity

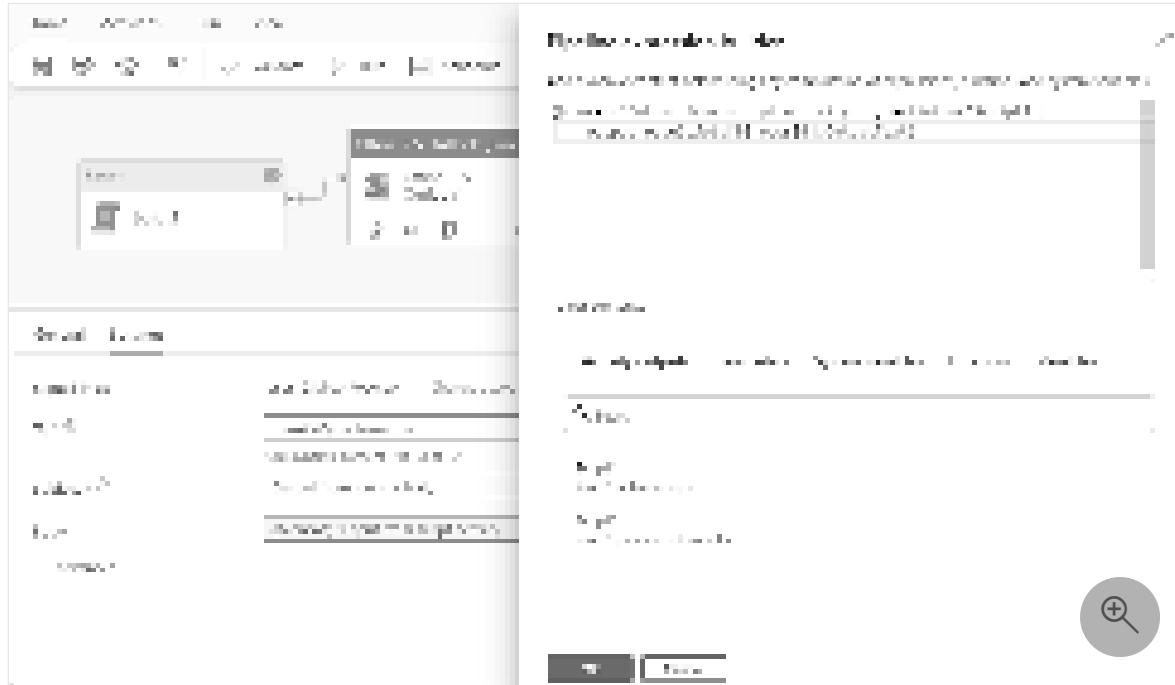
1. Select the Office 365 Outlook activity in the pipeline editor, and then select its **Settings** tab, and **Sign in** to your Office 365 Outlook account that will send the email.



2. After you sign in, you see the outgoing email template. Provide a list of emails for the email to go to in the **To** text box, and a **Subject** in that text box. Note that all fields in the template support dynamic content. Select the **Body** text area and then select **Add dynamic content**, to customize what we will add. Select **Activity outputs** if it isn't already selected, and then select the output of the activity. Select **OK** to use this dynamic content as the **Body** of the email. You can use any combination of outputs to generate emails of any level of complexity you need. In

In this case we use the following expression to output the returned value from the SQL script:

```
@concat('Output from script activity: ',  
activity('Script1').output.resultSets[0].rows[0].OutputText)
```

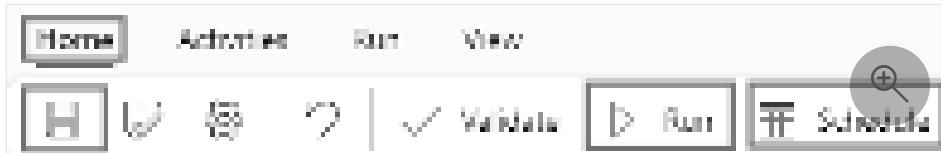


3. You can also specify advanced settings for the email if you wish, including an alternate **From (Send as)** value, **CC**, **BCC**, **Sensitivity**, **Reply To**, or **Importance** fields:



Save and run or schedule the pipeline

Switch to the **Home** tab at the top of the pipeline editor, and select the save button to save your pipeline. Select **Run** to run it directly, or **Schedule** to schedule it. You can also view the run history here or configure other settings.



Confirm success

After the pipeline runs successfully, check the account(s) you specified in the **To** box of the Office 365 Outlook activity settings to validate the output.



Next steps

How to monitor pipeline runs

Feedback

Was this page helpful?

Yes

No

Provide product feedback | Ask the community

Copy activity performance and scalability guide

Article • 11/15/2023

Sometimes you want to perform a large-scale data migration from data lake or enterprise data warehouse (EDW), to Fabric OneLake. Other times you want to ingest large amounts of data, from different sources into Fabric OneLake, for big data analytics. In each case, it is critical to achieve optimal performance and scalability.

Data Factory pipelines provide a mechanism to ingest data, with the following advantages:

- Handles large amounts of data
- Is highly performant
- Is cost-effective

These advantages are an excellent fit for data engineers who want to build scalable data ingestion pipelines that are highly performant.

After reading this article, you will be able to answer the following questions:

- What level of performance and scalability can I achieve using copy activity for data migration and data ingestion scenarios?
- What steps should I take to tune the performance of the copy activity?
- What performance optimizations can I utilize for a single copy activity run?
- What other external factors to consider when optimizing copy performance?

Copy performance and scalability achievable using Data Factory pipelines

Data Factory pipelines offer a serverless architecture that allows parallelism at different levels.

This architecture allows you to develop pipelines that maximize data movement throughput for your environment. These pipelines fully utilize the following resources:

- Network bandwidth between the source and destination data stores
- Source or destination data store input/output operations per second (IOPS) and bandwidth

This full utilization means you can estimate the overall throughput by measuring the minimum throughput available with the following resources:

- Source data store
- Destination data store
- Network bandwidth in between the source and destination data stores

Copy is scalable at different levels:

- **Control flow** can start multiple copy activities in parallel, for example using **For Each loop**.
- A single copy activity can take advantage of **scalable compute resources**.
 - You can specify the Intelligent throughput optimization to maximum for each copy activity, in a serverless manner.
- A single copy activity reads from and writes to the data store using **multiple threads** in parallel.

Copy performance optimization features

The service provides the following performance optimization features:

- Intelligent throughput optimization
- Parallel copy

Intelligent throughput optimization

Intelligent throughput optimization allows the service to optimize the throughput intelligently by combining the factors of CPU, memory, and network resource allocation and expected cost of running a single copy activity. The allowed options to empower a copy activity run intelligently are **Auto**, **Standard**, **Balanced**, **Maximum**. You can also specify the value **between 4 and 256**.

The following table lists the recommended value in different copy scenarios:

Value	Description
Auto	Allow the service to dynamically apply the optimal throughput optimization based on your source-destination pair and data pattern.
Standard	Allow the service to dynamically apply the throughput optimization under standard compute resources based on your source-destination pair and data pattern.
Balanced	Allow the service to dynamically apply throughput optimization which balances the throughput and available compute resources based on your source-destination pair

Value	Description
	and data pattern.
Maximum	Allow the service to dynamically apply the throughput optimization by utilizing the maximum available compute resources based on your source-destination pair and data pattern.

Parallel copy

You can set the 'Degree of copy parallelism' setting in the Settings tab of the Copy activity to indicate the parallelism you want the copy activity to use. Think of this property as the maximum number of threads within the copy activity. The threads operate in parallel. The threads either read from your source or write to your destination data stores.

The parallel copy is orthogonal to the intelligent throughput optimization setting. For each copy activity run, by default the service dynamically applies the optimal parallel copy setting based on your source-destination pair and data pattern.

To control the load on machines that host your data stores, or to tune copy performance, you can override the default value and specify a value for the Degree of copy parallelism. The value must be an integer greater than or equal to 1. At run time, for the best performance, the copy activity uses a value that is less than or equal to the value that you set.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Copy from Azure Blob Storage to Lakehouse

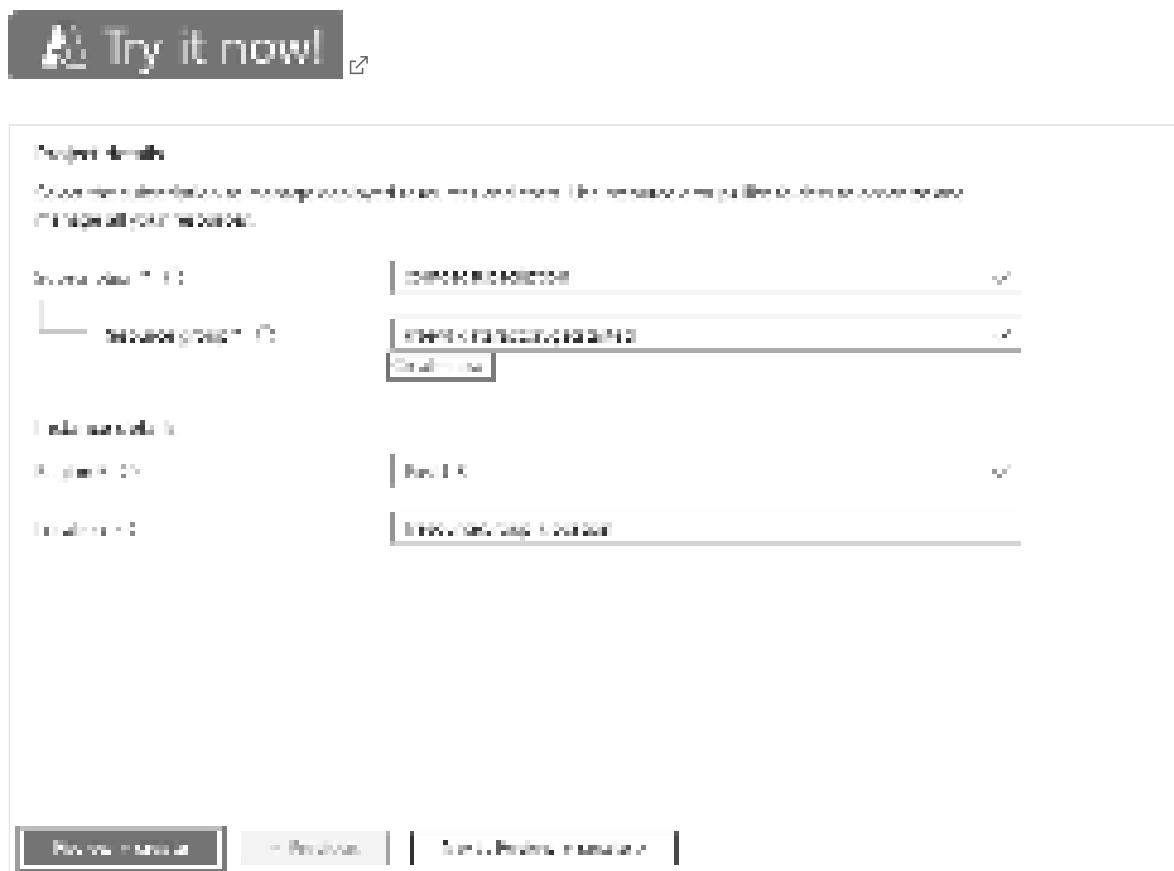
Article • 11/15/2023

In this tutorial, you'll build a data pipeline to move a CSV file from an input folder of an Azure Blob Storage source to a Lakehouse destination.

Prerequisites

To get started, you must complete the following prerequisites:

- Make sure you have a Project Microsoft Fabric enabled Workspace: Create a workspace.
- Select the **Try it now!** button to prepare the Azure Blob Storage data source of the Copy. Create a new resource group for this Azure Blob Storage and select **Review + Create > Create**.

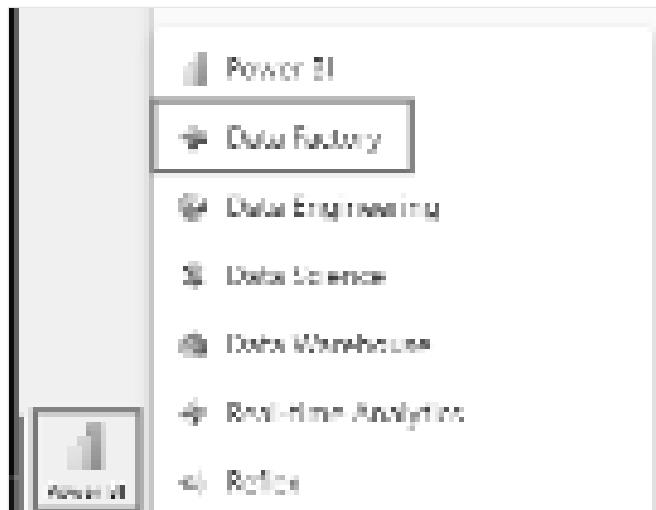


Then an Azure Blob Storage will be created and [moviesDB2.csv](#) will be uploaded to the input folder of the created Azure Blob Storage.

Name	Type
<input type="text"/> storagewyjfc03l002q	Storage account

Create a data pipeline

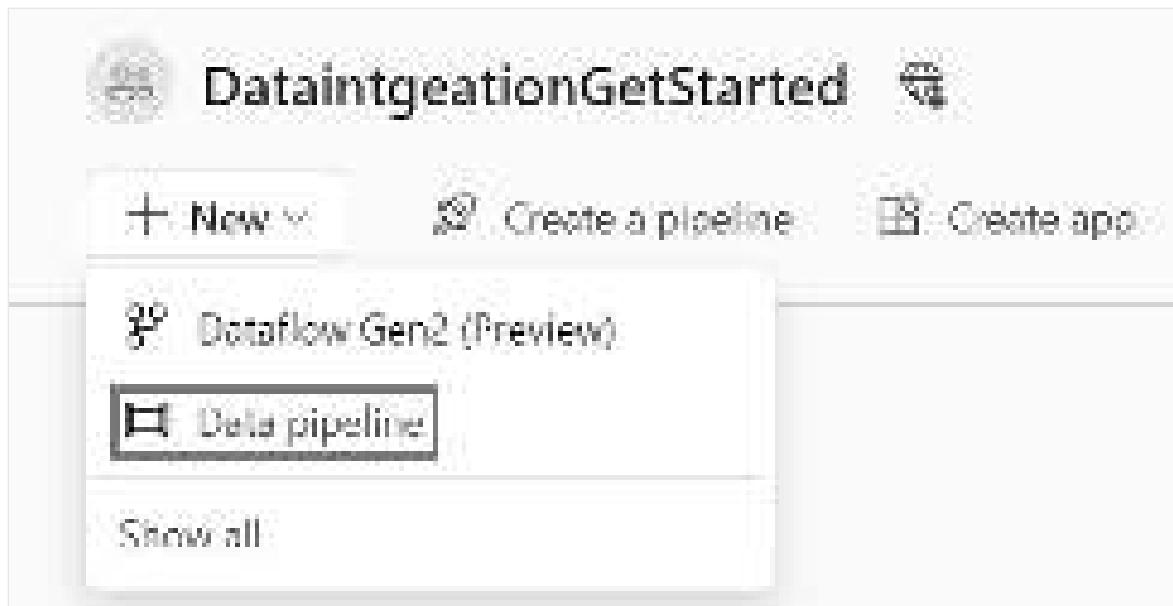
1. Switch to **Data factory** on the app.powerbi.com page.



2. Create a new workspace for this demo.



3. Select **New**, and then select **Data Pipeline**.

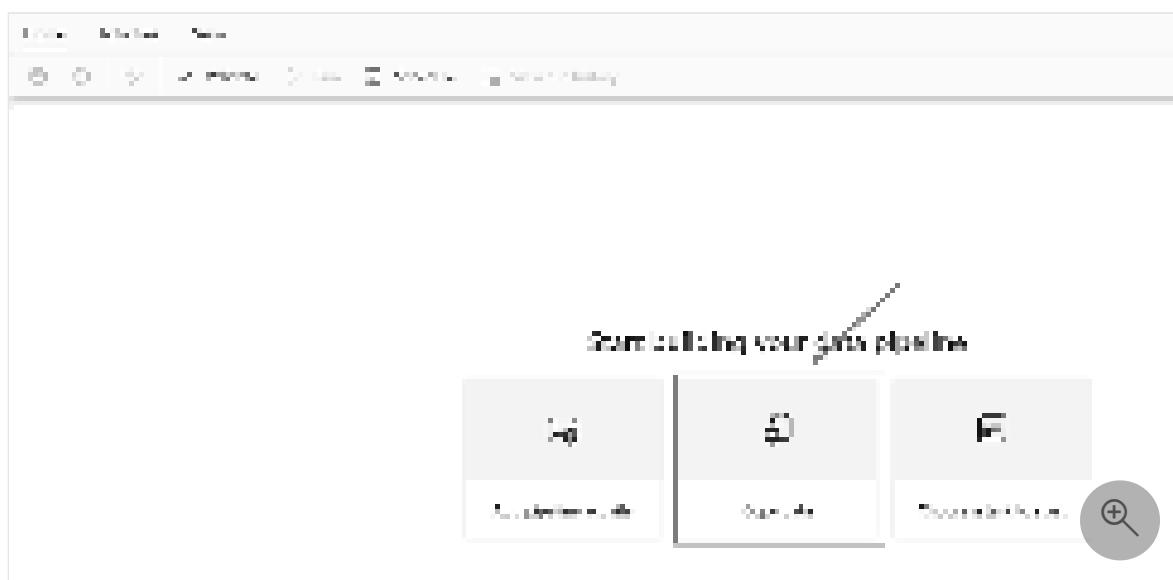


Copy data using the Copy Assistant

In this session, you'll start to build a data pipeline by using the following steps. These steps copy a CSV file from an input folder of an Azure Blob Storage to a Lakehouse destination using the copy assistant.

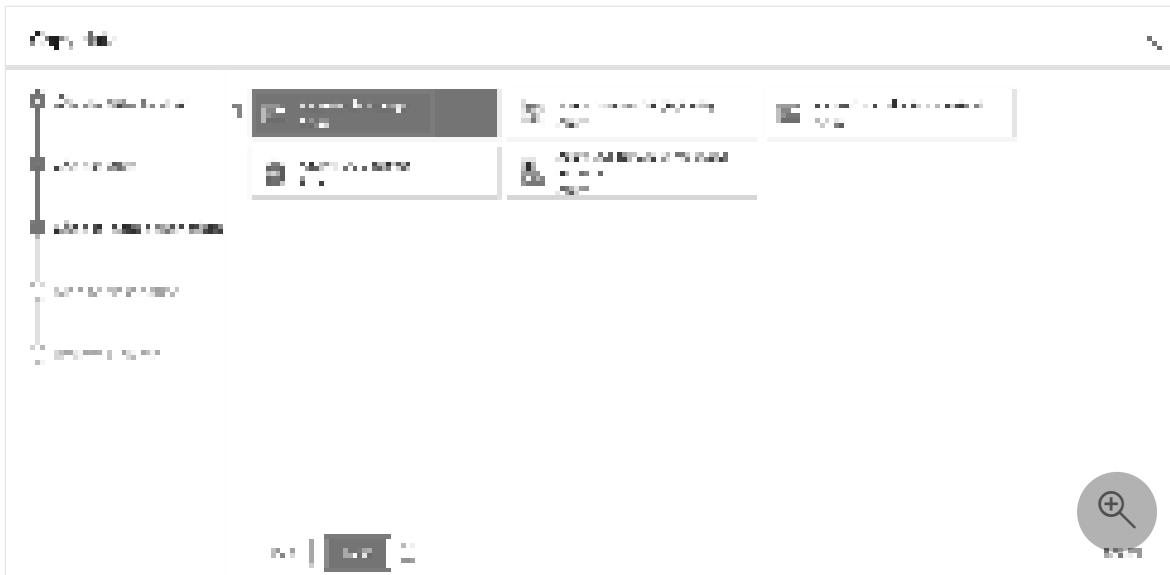
Step 1: Start with copy assistant

1. Select **Copy data** on the canvas to open the **copy assistant** tool to get started. Or Select **Use copy assistant** from the **Copy data** drop down list under the **Activities** tab on the ribbon.

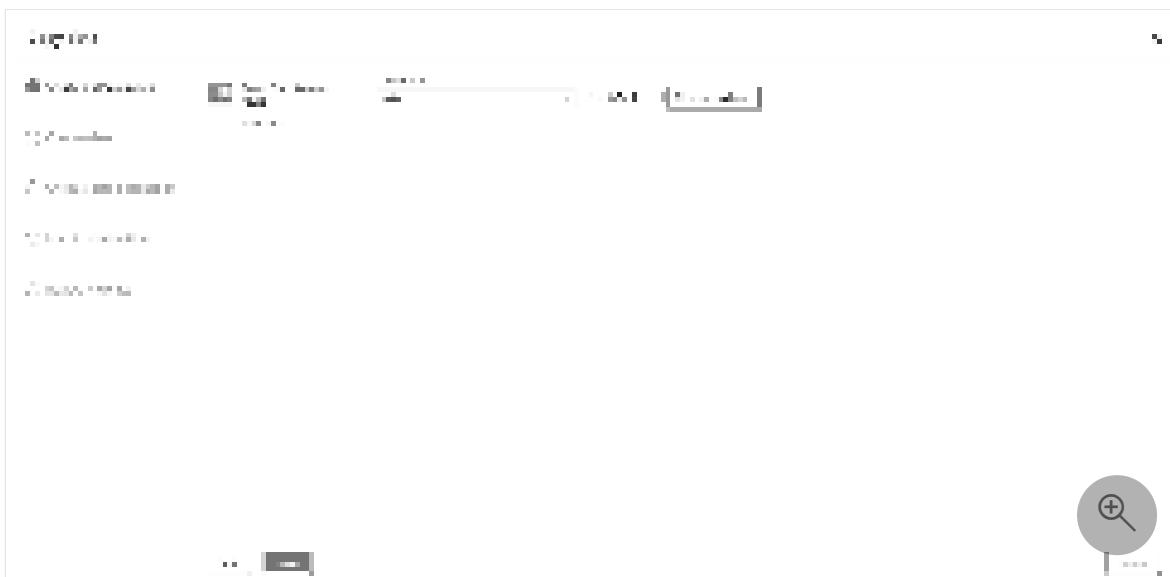


Step 2: Configure your source

1. Select **Azure Blob Storage**, and then select **Next**.



2. Create a connection to your data source by selecting **New connection**.



a. After selecting **Create new connection**, you only need to fill in **Account name or URL**, and **Authentication kind**. If you input **Account name or URL** using your Azure Blob Storage account name, the connection will be auto filled. In this demo, we will choose **Account key** but you can choose other Authentication kind regarding your preference. After selecting **Sign in**, you only need to log in to one account that having this blob storage permission.



- b. Once your connection is created successfully, you only need to select Next to Connect to data source.
3. Choose the file moviesDB2.csv in the source configuration to preview, and then select **Next**.



Step 3: Configure your destination

1. Select **Lakehouse** and then **Next**.



2. Create a new Lakehouse and input the Lakehouse name. Then select **Next**.



3. Configure and map your source data to your destination; then select **Next** to finish your destination configurations.

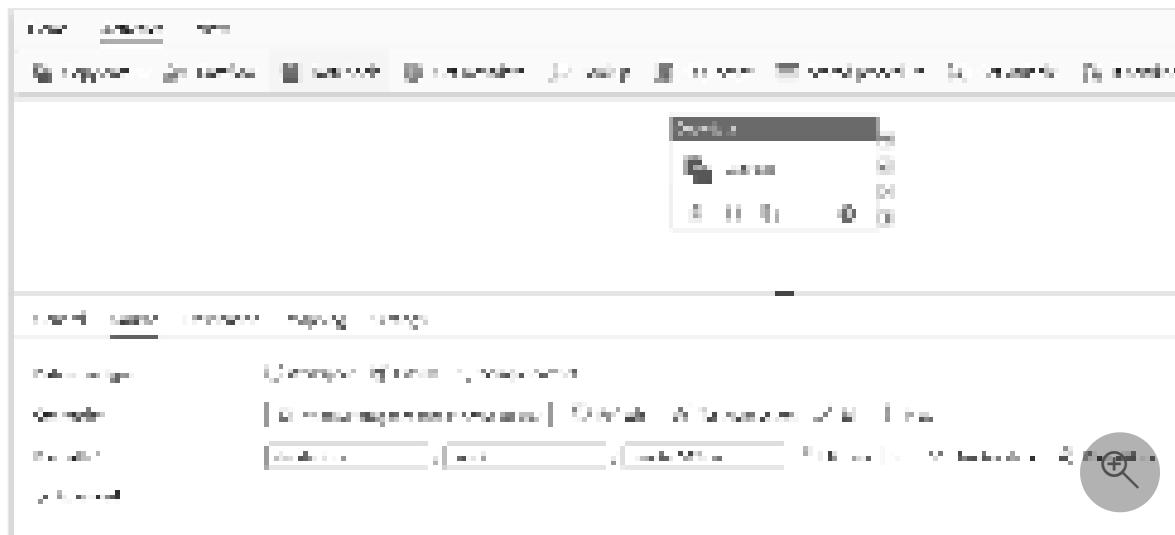


Step 4: Review and create your copy activity

1. Review your copy activity settings in the previous steps and select OK to finish. Or you can go back to the previous steps to edit your settings if needed in the tool.

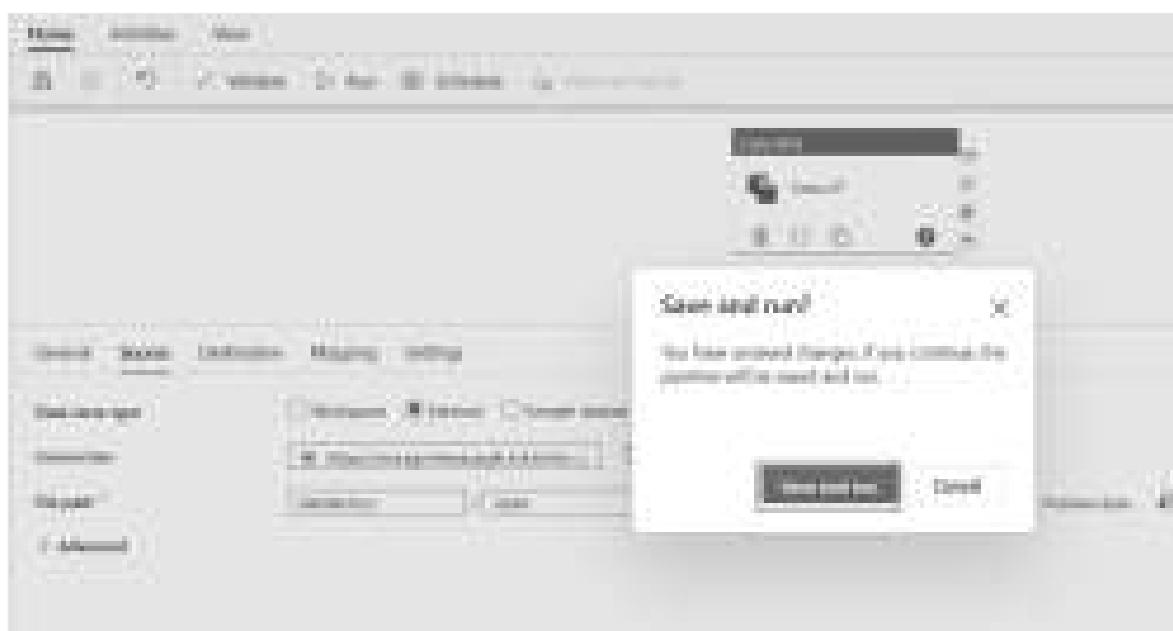
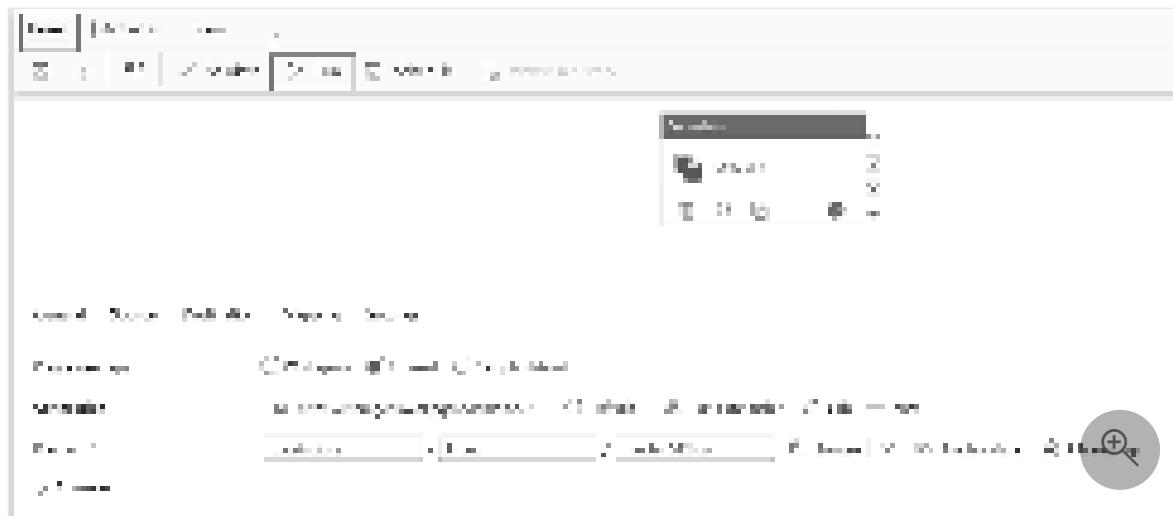


2. Once finished, the copy activity will then be added to your data pipeline canvas. All settings including advanced settings to this copy activity are available under the tabs below when it's selected.

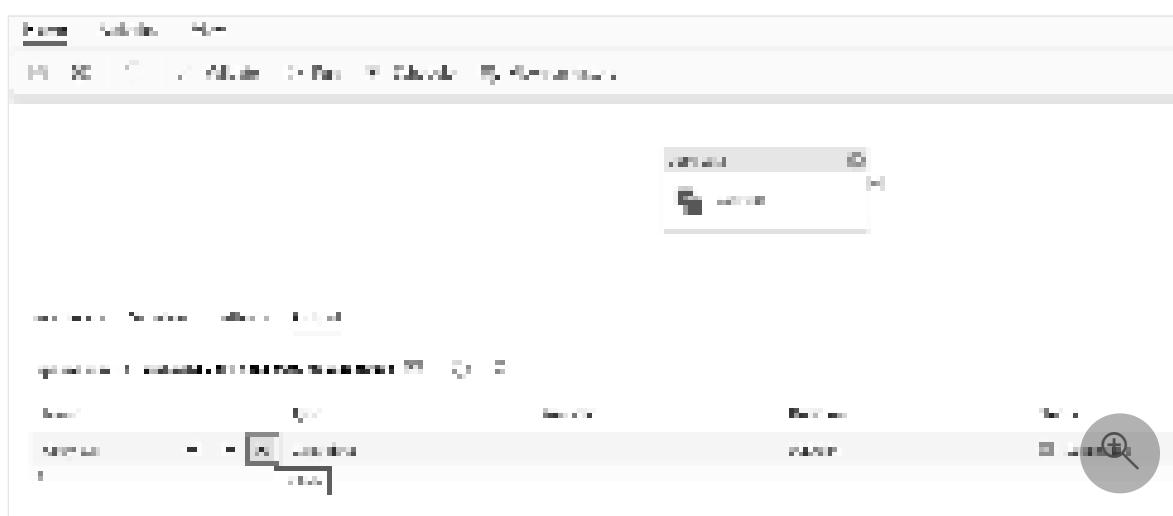


Run and schedule your data pipeline

1. Switch to **Home** tab and select **Run**. Then select **Save and Run**.



2. Select the **Details** button to monitor progress and check the results of the run.



3. The **Copy data details** dialog displays results of the run including status, volume of data read and written, start and stop times, and duration.

Copy data details

Source	Destination
 Azure Blob Storage	 Lakehouse
Data read: 450,221 KB	Data written: 154,466 KB
Files read: 1	Files written: 1
Rows read: 2,125	Rows written: 2,125

Status:  Succeeded

Start time: 2023-07-18T17:17:30Z

Pipeline run activity ID: a5d1f6c8-411c-4171-97d8-11c7ffbb3e00

Throughput: 75,097 KB/s

Total duration: 00:00:11

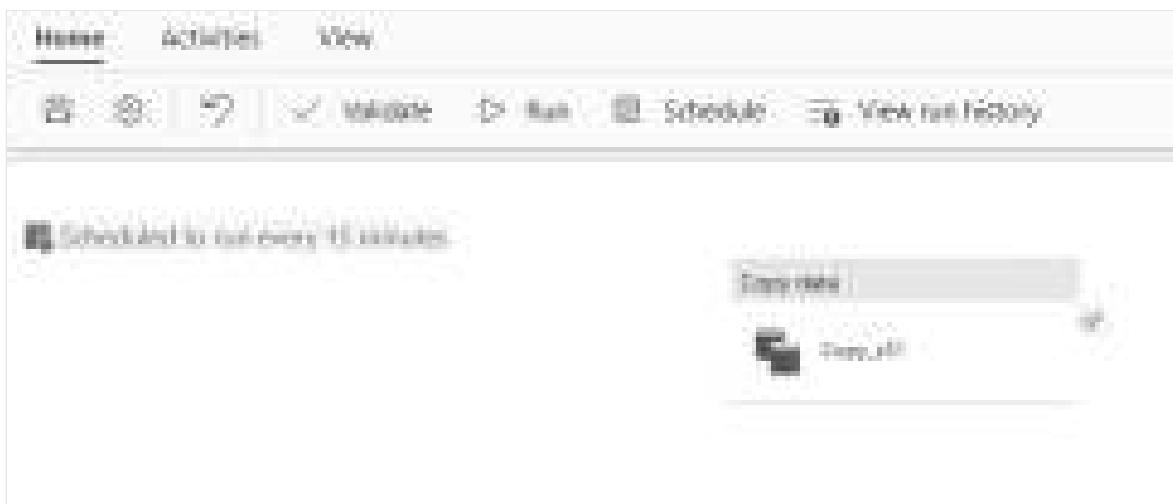
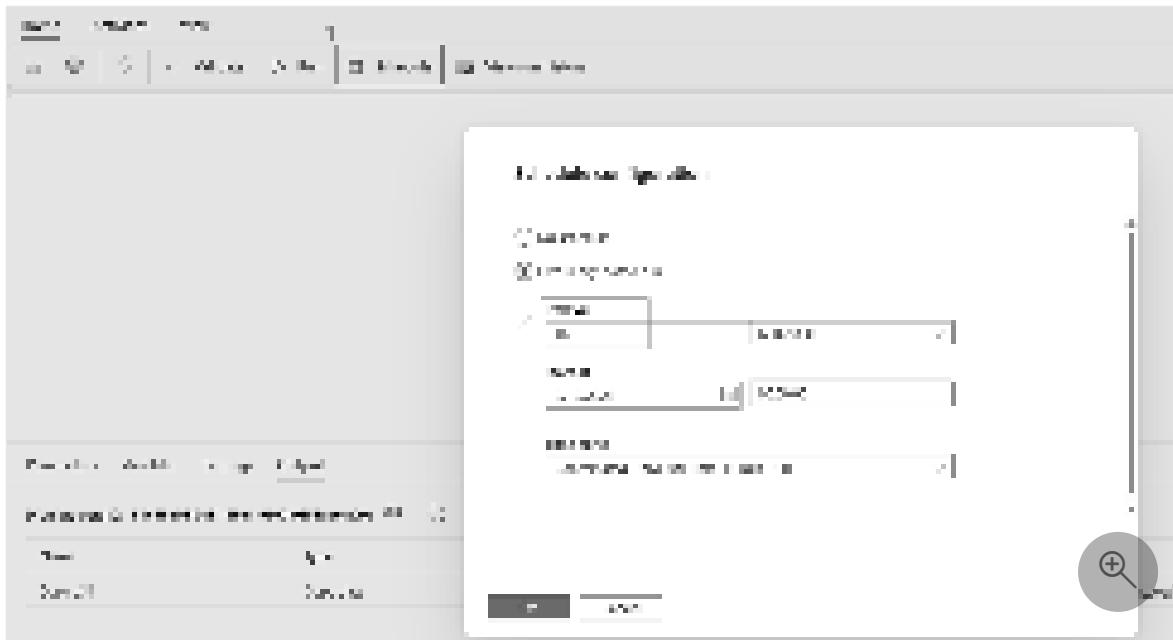
 **Duration breakdown**

Start time	End time
2023-07-18T17:17:30Z	2023-07-18T17:17:41Z

 Used parallel copies: 1



4. You can also schedule the pipeline to run with a specific frequency as required.
Below is the sample to schedule the pipeline to run every 15 minutes.



Next steps

The pipeline in this sample shows you how to copy data from Azure Blob Storage to Lakehouse. You learned how to:

- ✓ Create a data pipeline.
- ✓ Copy data with the Copy Assistant.
- ✓ Run and schedule your data pipeline.

Next, advance to learn more about monitoring your pipeline runs.

[How to monitor pipeline runs in Microsoft Fabric](#)

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Copy sample data into Lakehouse and transform with a dataflow with Data Factory in Microsoft Fabric

Article • 11/15/2023

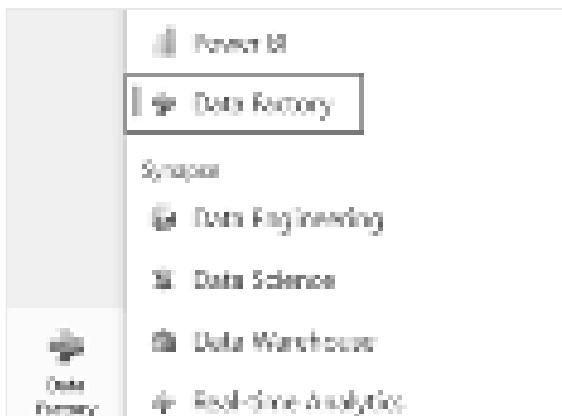
In this tutorial, we provide end-to-end steps to a common scenario that uses the pipeline to load source data into Lakehouse at high performance copy and then transform the data by dataflow to make users can easily load and transform data.

Prerequisites

A Microsoft Fabric enabled workspace. If you don't already have one, refer to the article [Create a workspace](#).

Create a data pipeline

1. Switch to the **Data Factory** experience.



2. Select **New** and then **Data pipeline**, and then input a name for your pipeline.

Data Factory - DF Getting Started

DF Getting Started

+ New

- [Data pipeline] **Data pipeline**
- [Data flow]
- [Event hub]
- [Federated dataset]
- [KQL Database]
- [KQL Query]
- [Lake house]
- [Machine learning]
- [Notebook]
- [Power BI Preview]
- [Report]
- [Schedule Definition]
- [Workflow]

More options...

Import from:

Import notebook

The screenshot shows the 'DF Getting Started' blade in the Azure Data Factory interface. On the left, there's a vertical sidebar with icons for Home, + New, Data pipeline, Data flow, Event hub, Federated dataset, KQL Database, KQL Query, Lake house, Machine learning, Notebook, Power BI Preview, Report, Schedule Definition, Workflow, More options, Import from, and Import notebook. The 'Data pipeline' icon is highlighted with a red box. At the top, there's a search bar with placeholder text 'Search' and a magnifying glass icon. Below the search bar, there are three buttons: '+ New' (highlighted with a red box), 'View all...', and 'Create'.

New pipeline

X

Name

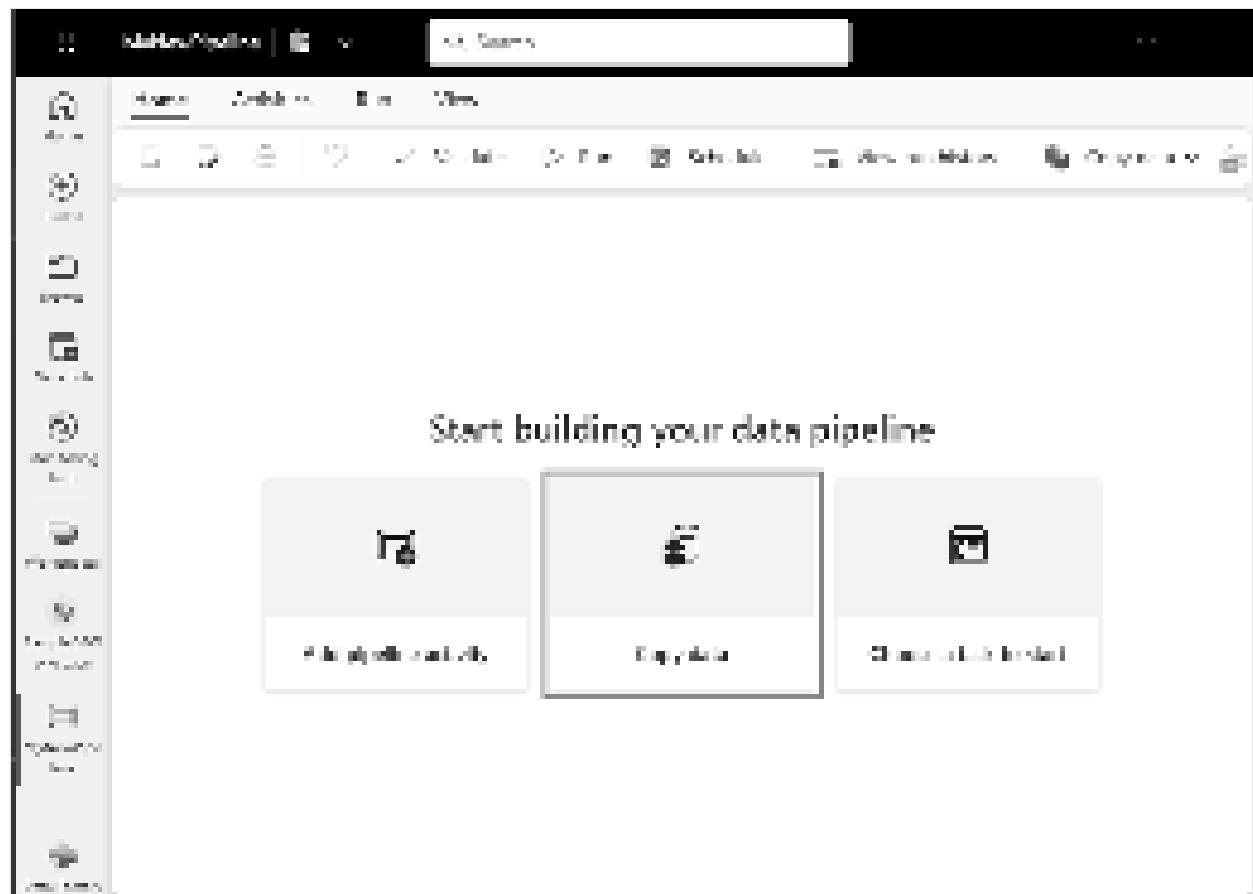
Create Cancel

Use a pipeline to load sample data into Lakehouse

Use the following steps to load sample data into Lakehouse.

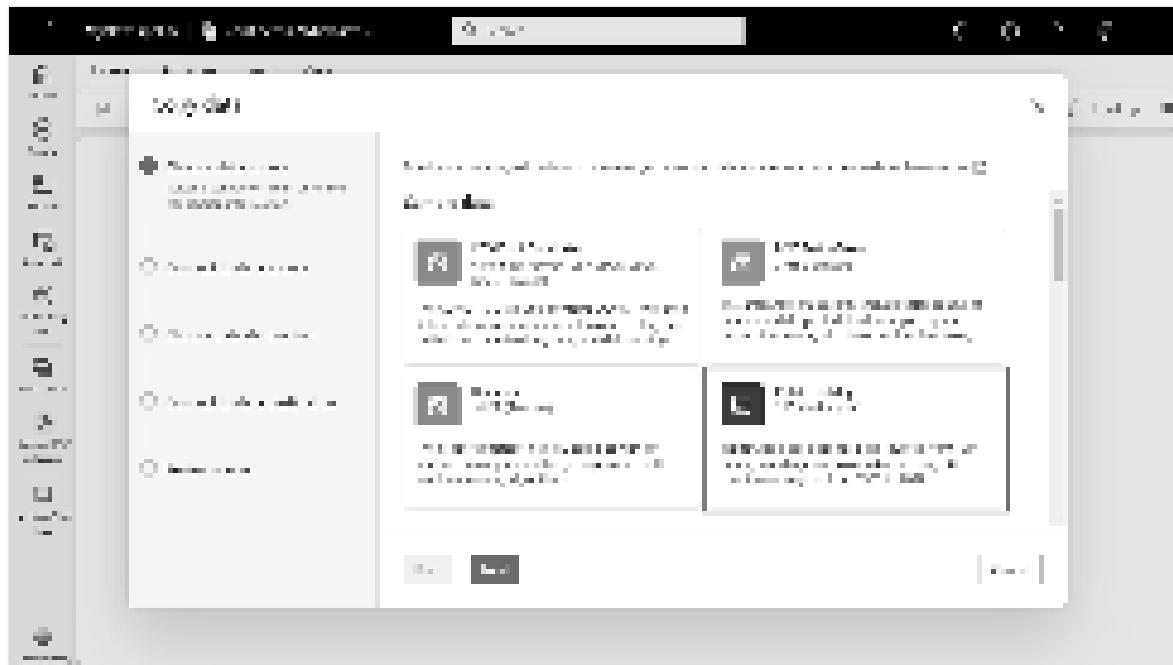
Step 1: Start with the Copy assistant

Select **Copy Data** on the canvas, to open the **Copy assistant** tool to get started.

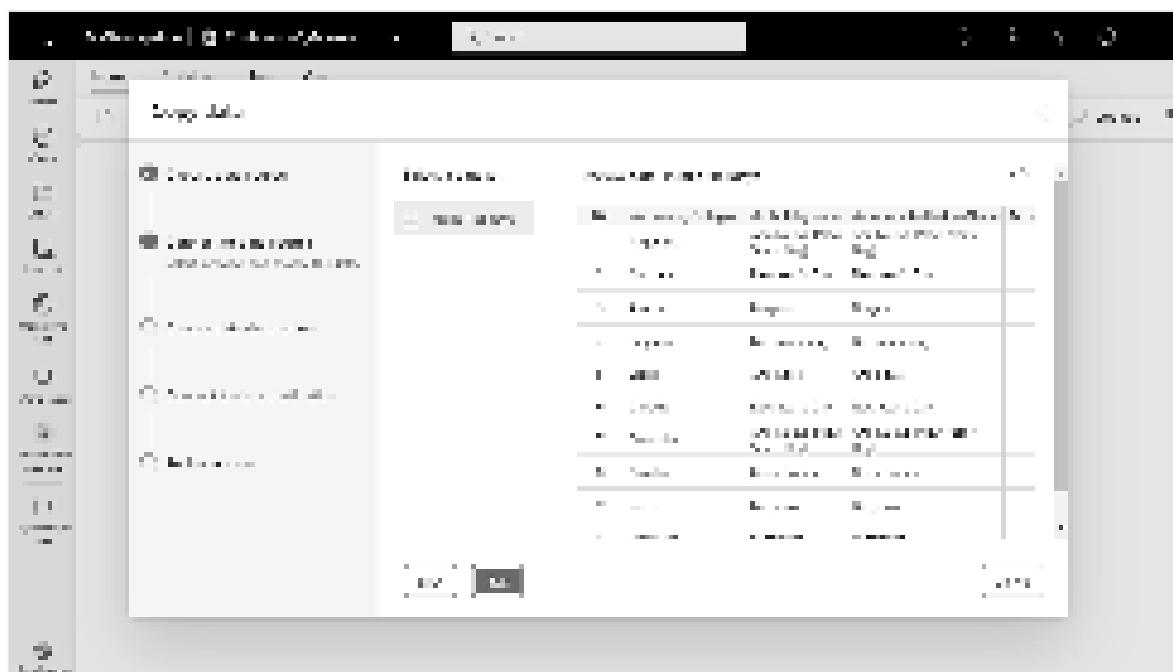


Step 2: Configure your source

1. Choose the **Public Holidays** from the **Sample data** options for your data source, and then select **Next**.

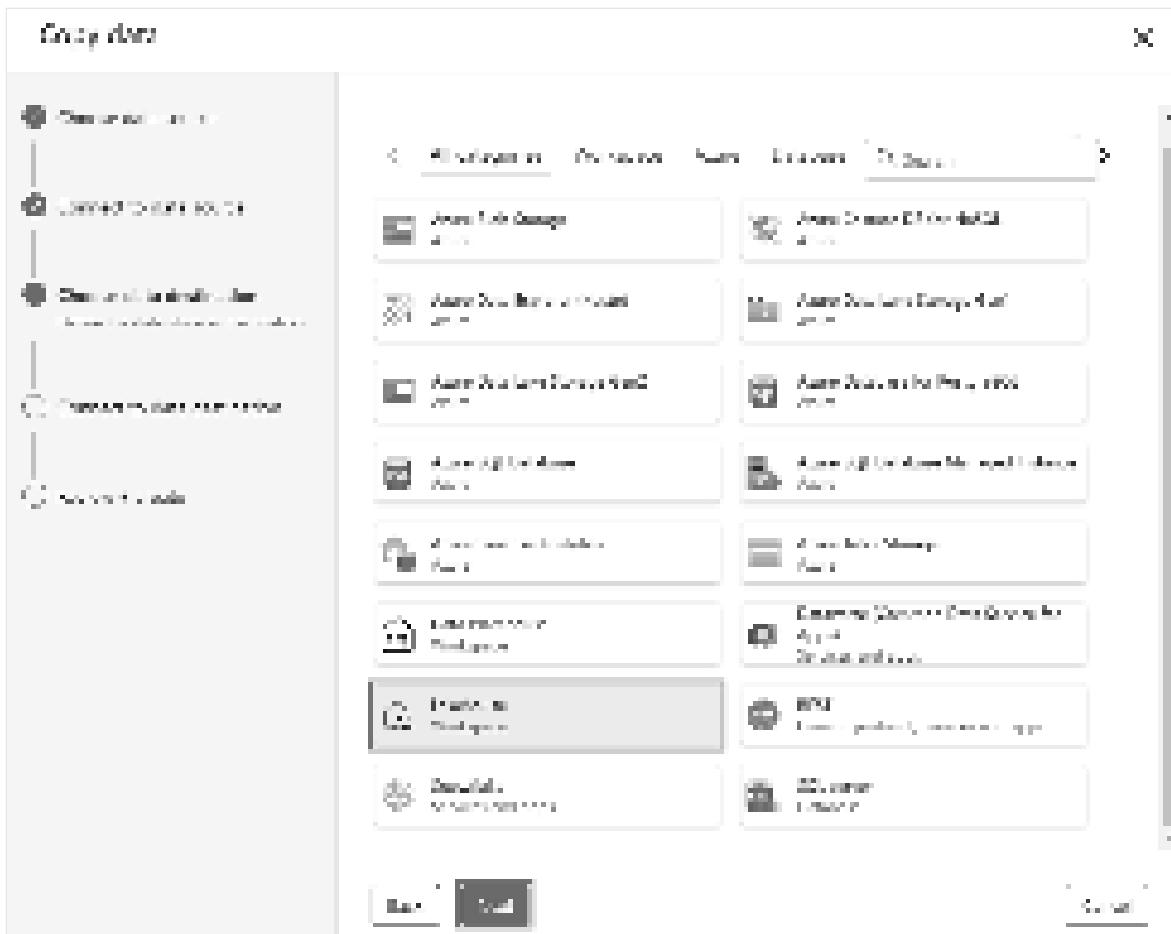


2. In the **Connect to data source** section of the **Copy data** assistant, a preview of the sample data is displayed. Select **Next** to move on to the data destination.



Step 3: Configure your destination

1. Select the **Workspace** tab and choose **Lakehouse**. Then select **Next**.



2. Select **Create new Lakehouse** and enter **LHDemo** for the name, then select **Next**.



3. Configure and map your source data to the destination Lakehouse table by entering **Table name**, then select **Next** one more time.

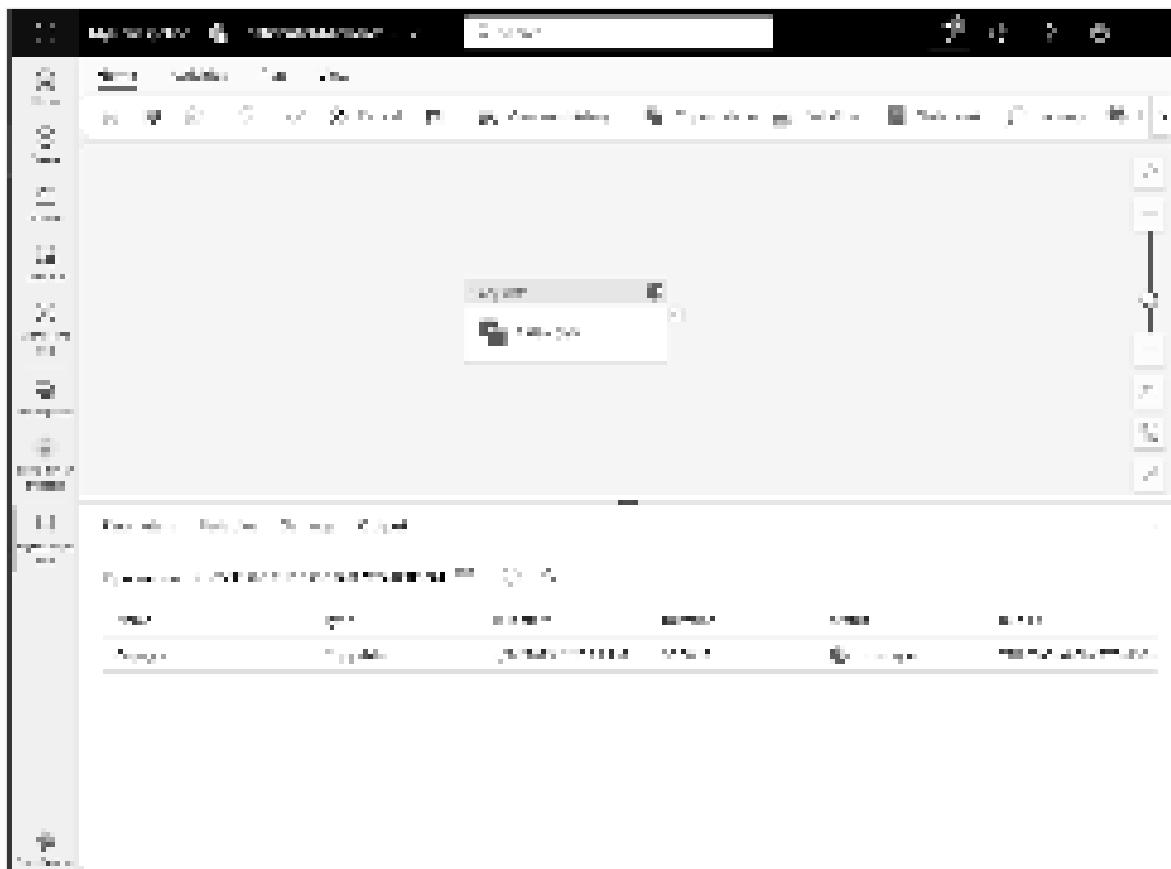


Step 4: Review and create your copy activity

1. Review your copy activity settings in the previous steps and select **Start data transfer immediately**. Then select **Save + Run** to run the new pipeline.



2. Once finished, the copy activity is added to your new data pipeline canvas, and the pipeline automatically runs to load data into Lakehouse.



3. You can monitor the running process and check the results on the **Output** tab below the pipeline canvas. Hover over the name in the output row to see the **Run details** button (an icon of a pair of glasses, highlighted) to view the run details.

Run ID	Run	Run Date	Run Time	Run Status	Run Details
1	Run 1	2023-10-12	14:30:00	Completed	Run 1

4. The run details show 69,557 rows were read and written, and various other details about the run, including a breakdown of the duration.

Copy data details

Copy_1774

Source	Destination
 Azure Blob Storage	 Lakehouse
Data read: 669,359 KB	Data written: 295,632 KB
File read: 1	File written: 1
Row read: 60,557	Row written: 60,557

Status:  Successful

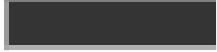
Start time: 4/14/2023, 1:20:25 PM

Pipeline run activity ID: 77180545-8C1-4870-510c-0b3557291fca

Throughput: 50,306 KB/s

Total duration: 00:00:15

Duration breakdown:

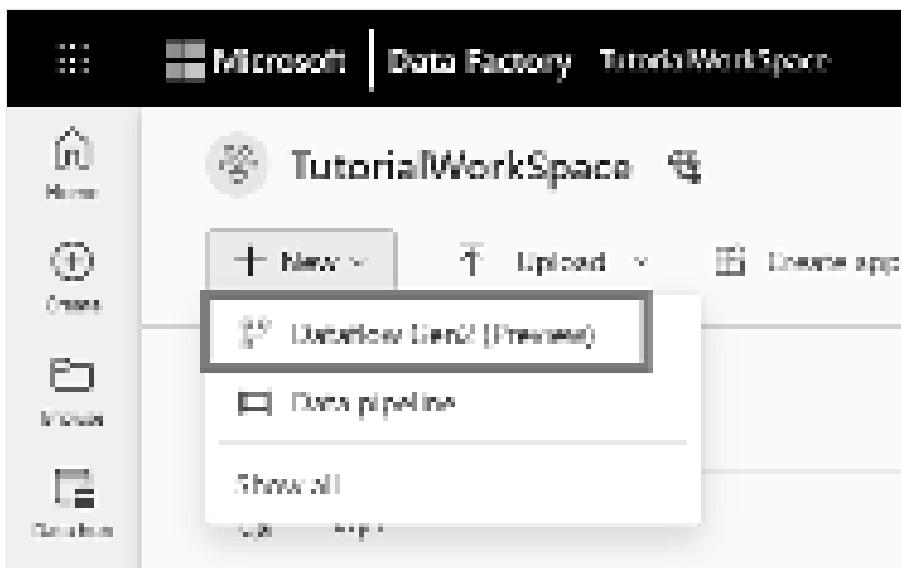
Start time:	4/14/2023, 1:20:24 PM
Used parallel copies:	1
From:	
To:	

Close

Use a dataflow gen2 to transform data in the Lakehouse

You now have a Lakehouse with sample data loaded. Next, you'll use a dataflow to transform the data. Dataflows are a code-free way to transform data at scale.

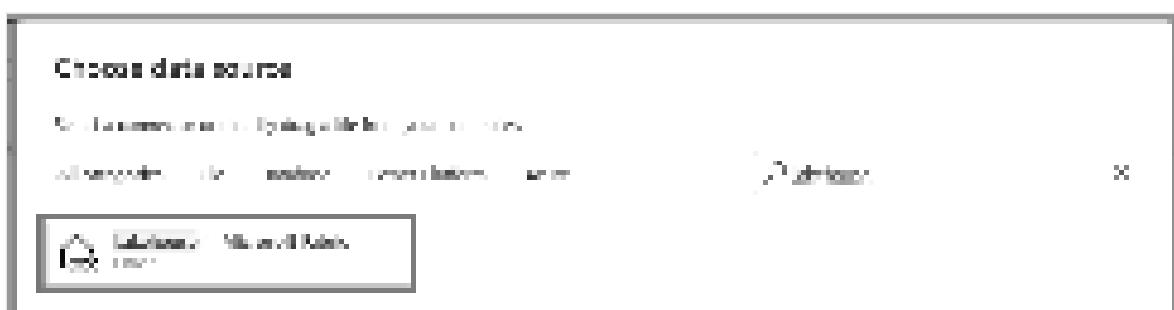
1. Select New and then Dataflow Gen2.



2. Click on get data dropdown and select More....



3. Search for Lakehouse and select Lakehouse in Microsoft Fabric.



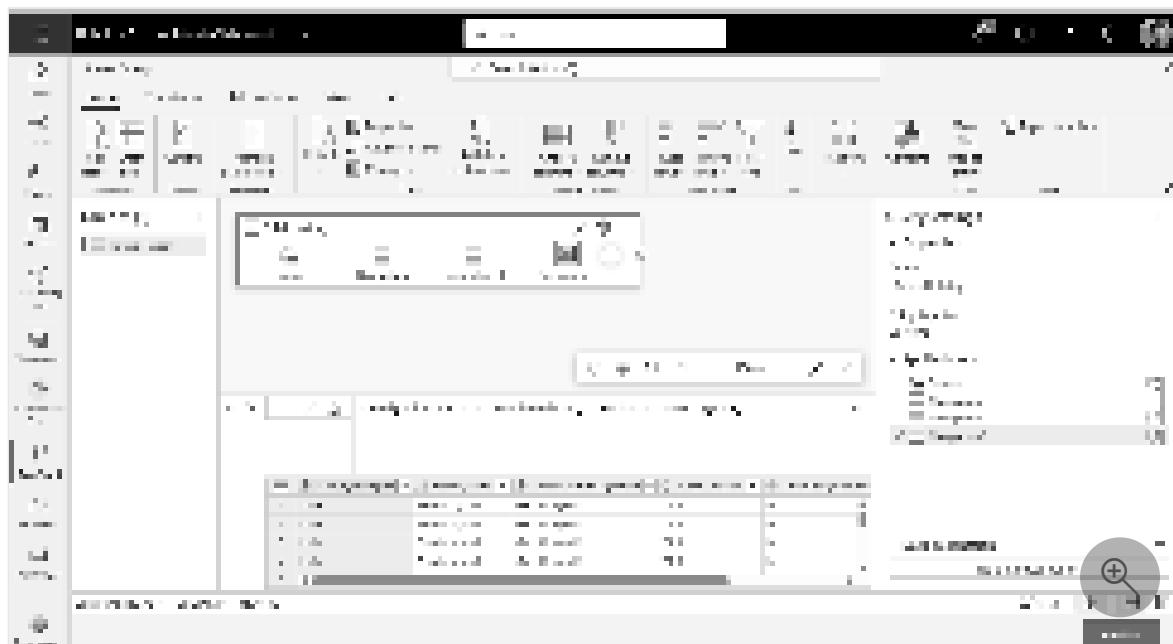
4. Sign-in and click Next to continue.



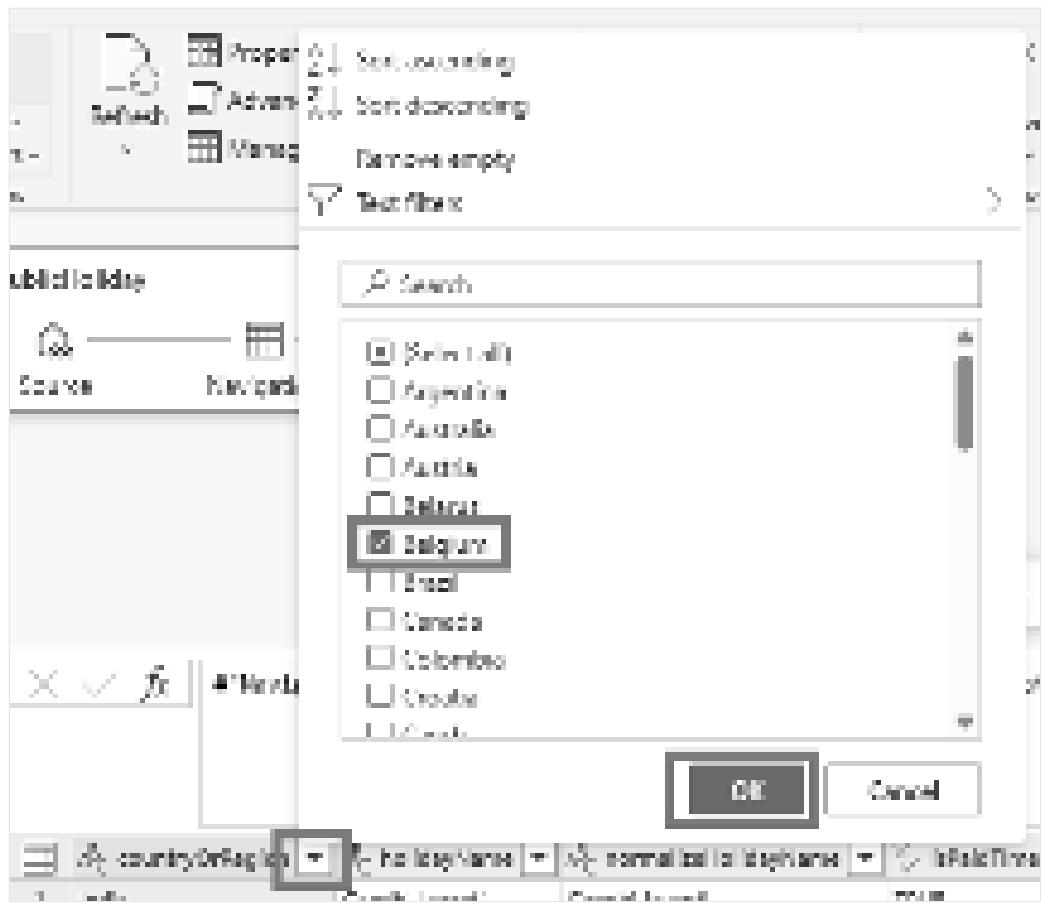
5. Select the table you created in the previous step and click **Create**.



6. Review the data preview in the editor.



7. Apply a filter to the dataflow to only include rows where the **Countryorregion** column is equal to **Belgium**.



8. Add a data destination to the query by selecting **Add data destination** and then **Lakehouse in Microsoft Fabric**.



9. Sign-in and click **Next** to continue.



10. Create a new table called **BelgiumPublicHolidays** and click **Next**.



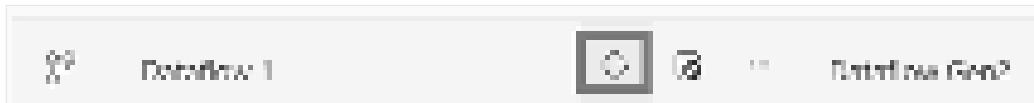
11. Review the settings and click **Save settings**.



12. Publish the dataflow by clicking **Publish**.



13. After the dataflow is published, click **Refresh now** to run the dataflow.



After the refresh is complete, you can view the data in the Lakehouse table. You can also use this data now to create reports, dashboards, and more.

Next steps

This sample shows you how to copy sample data to Lakehouse and transform the data with a dataflow using Data Factory in Microsoft Fabric. You learned how to:

- ✓ Create a data pipeline.
- ✓ Use the pipeline to load sample data into Lakehouse.
- ✓ Use dataflow to transform data in the Lakehouse.

Next, advance to learn more about monitoring your pipeline runs.

[How to monitor pipeline runs in Microsoft Fabric](#)

Feedback

Was this page helpful?

Yes

No

[Provide product feedback](#) | [Ask the community](#)

Preprocess data with a stored procedure before loading into Lakehouse

Article • 11/15/2023

In this tutorial, we show you how to use a pipeline Script activity to run a stored procedure to create a table and preprocess the data in a Synapse Data Warehouse. After that, we load the preprocessed table into Lakehouse.

Prerequisites

- A Microsoft Fabric enabled workspace. If you don't already have one, refer to the article [Create a workspace](#).
- Prepare a stored procedure in your Azure Synapse Data Warehouse. Create the following stored procedure in advance:

SQL

```
CREATE PROCEDURE spM_add_names
AS
--Create initial table
IF EXISTS (SELECT * FROM sys.objects
WHERE object_id = OBJECT_ID(N'[dbo].[names]') AND TYPE IN (N'U'))
BEGIN
DROP TABLE names
END;

CREATE TABLE names
(id INT,fullname VARCHAR(50));

--Populate data
INSERT INTO names VALUES (1,'John Smith');
INSERT INTO names VALUES (2,'James Dean');

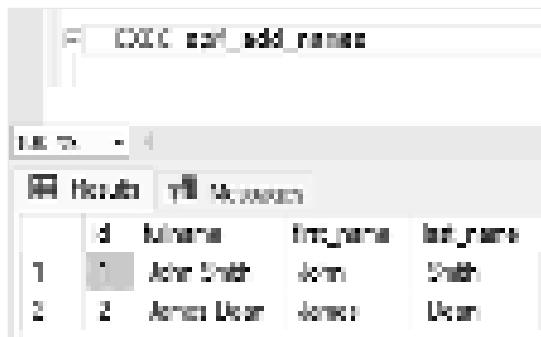
--Alter table for new columns
ALTER TABLE names
ADD first_name VARCHAR(50) NULL;

ALTER TABLE names
ADD last_name VARCHAR(50) NULL;

--Update table
UPDATE names
SET first_name = SUBSTRING(fullname, 1, CHARINDEX(' ', fullname)-1);

UPDATE names
SET last_name = SUBSTRING(fullname, CHARINDEX(' ', fullname)+1,
```

```
LEN(fullname)-CHARINDEX(' ', fullname));  
  
--View Result  
SELECT * FROM names;
```



	id	fname	mname	lname
1	1	John	Smith	Smith
2	2	James	Dean	Dean

Create a pipeline Script activity to run the stored procedure

In this section, we use a Script activity to run the stored procedure created in the prerequisites.

1. Choose Script activity and then select **New** to connect to your Azure Synapse Data Warehouse.



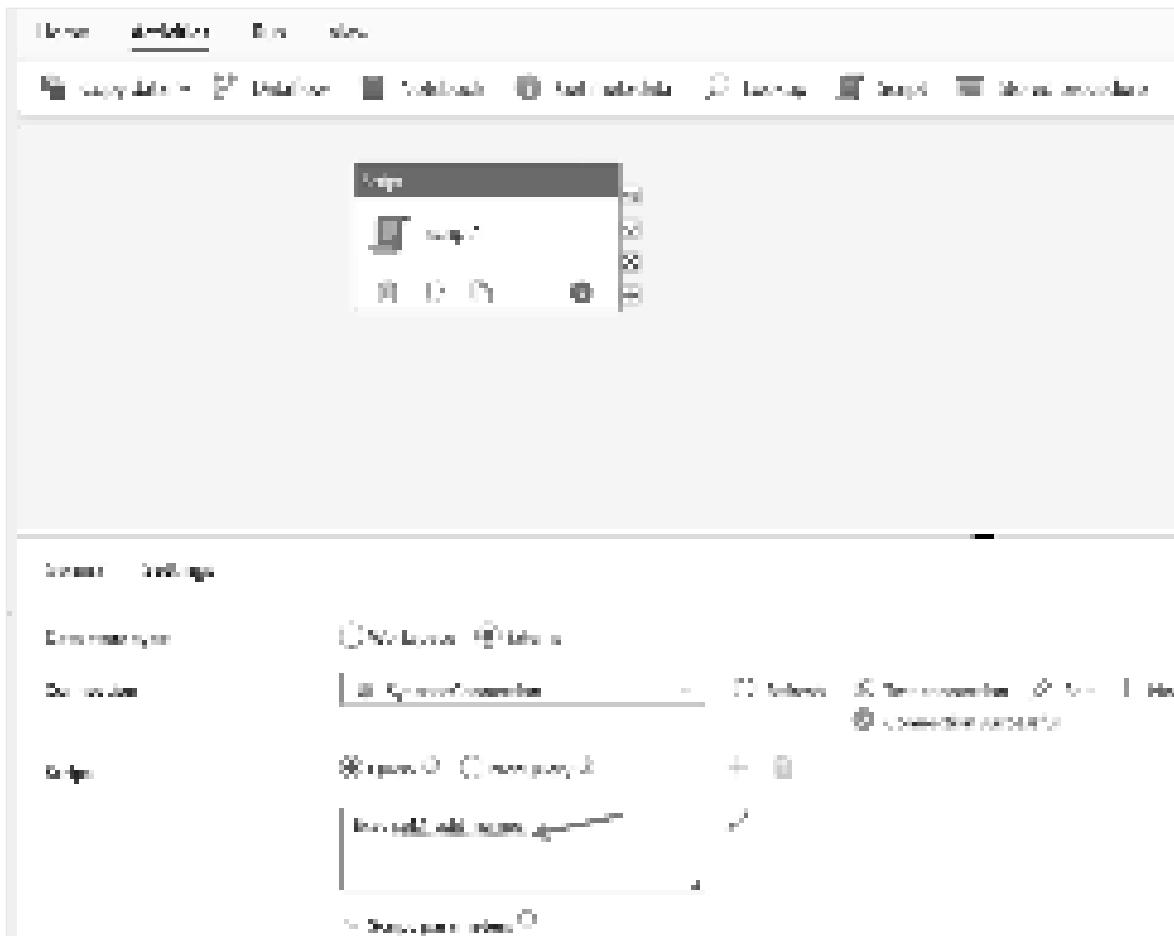
2. Select Azure Synapse Analytics and then **Continue**.



3. Provide your **Server**, **Database**, and **Username** and **Password** fields for **Basic authentication**, and enter **SynapseConnection** for the **Connection name**. Then select **Create** to create the new connection.



4. Input **EXEC spM_add_names** to run the stored procedure. It creates a new table **dbo.name** and preprocess the data with a simple transformation to change the **fullname** field into two fields, **first_name** and **last_name**.



Use a pipeline activity to load preprocessed table data into Lakehouse

1. Select Copy data and then select Use copy assistant.



2. Select Azure Synapse Analytics for the data source, and then select Next.



3. Choose the existing connection **SynapseConnection** that you created previously.



4. Choose the table **dbo.names** that was created and preprocessed by the stored procedure. Then select **Next**.



5. Select **Lakehouse** under the **Workspace** tab as the destination, and then select **Next** again.



6. Choose an existing or create a new Lakehouse, then select **Next**.



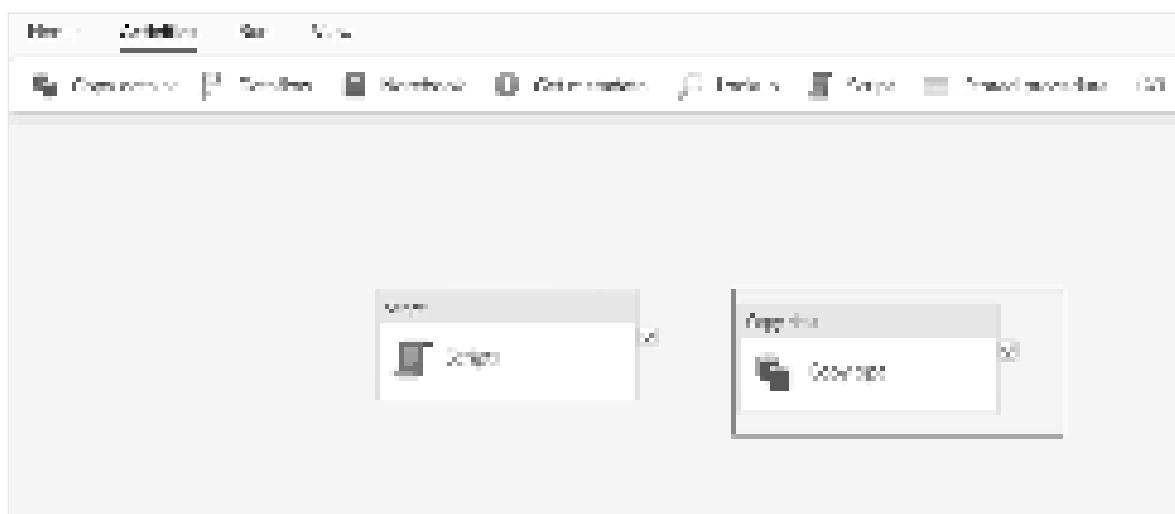
7. Input a destination table name for the data to be copied into for the Lakehouse destination and select **Next**.



8. Review the summary on the final page of the Copy assistant and then select **OK**.



9. After you select **OK**, the new Copy activity will be added onto the pipeline canvas.

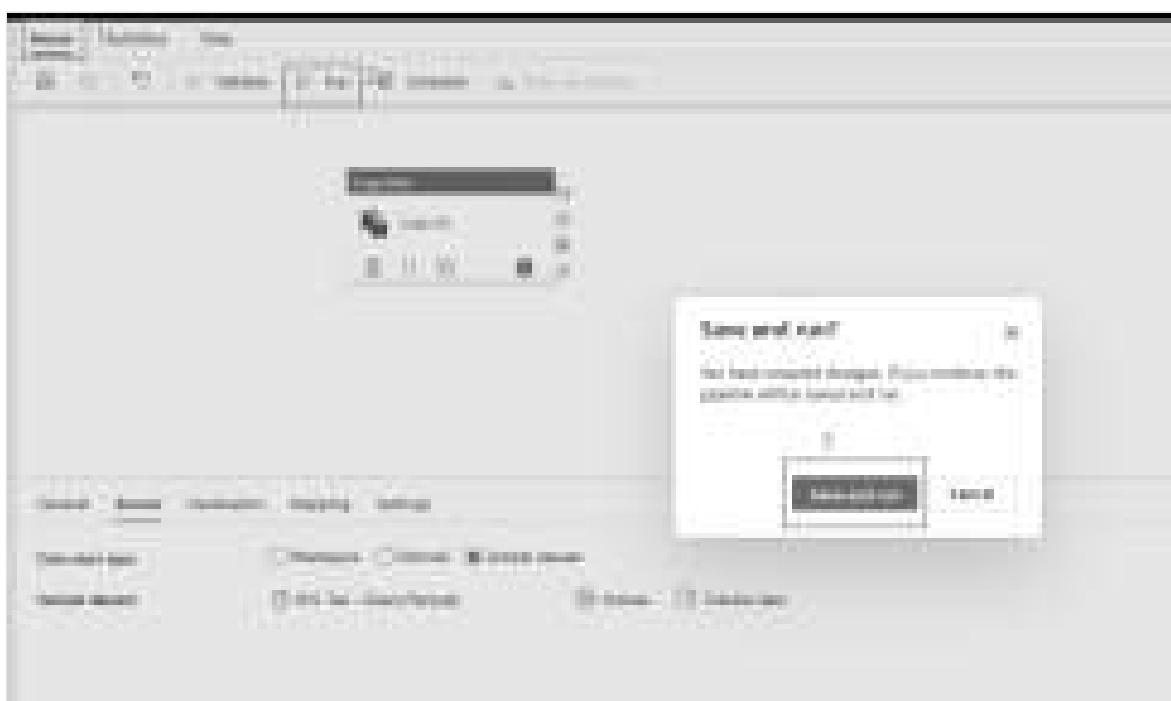


Execute the two pipeline activities to load the data

1. Connect the Script and Copy data activities by **On success** from the Script activity.



2. Select **Run** and then **Save and run** to run the two activities in the pipeline.



3. After the pipeline successfully runs, you can view the details for more information.

The screenshot shows the Azure Synapse Analytics workspace interface. At the top, there's a navigation bar with 'Home', 'Pipelines', 'Jobs', 'Pins', and 'Help'. Below it, a search bar and a 'New' button are visible. The main area displays a pipeline named 'Copy_spa' with two stages: 'Copy' and 'sink'. The 'Copy' stage has a 'Source' input and an 'Output' output. The 'sink' stage has an 'Input' input and a 'Sink' output. The pipeline is currently in the 'Running' state.

Copy data details

Copy_spa

This dialog provides detailed information about the completed pipeline run 'Copy_spa'.

Source	Destination
Azure Synapse Analytics	Lakehouse
Data read: 84 bytes	Data written: 1.119 KB
Rows read: 2	Files written: 1
	Rows written: 2

Status: Succeeded
Start time: 4/25/2023, 1:25:12 AM

Pipeline run activity ID: 7644d7ef-2a21-46c0-bb76-7aff74e1c4d6
Throughput: 10 bytes/s
Total duration: 00:00:15

Duration breakdown:

Start time	End time
4/25/2023, 1:25:12 AM	4/25/2023, 1:25:13 AM

Close

4. Switch to the workspace and select the Lakehouse to check the results.

The screenshot shows the Microsoft Fabric Data Catalog interface. At the top, there's a navigation bar with 'TutorialWorkSpace' and icons for 'New', 'Upload', 'Create deployment pipeline', and 'Get'. Below the navigation is a table listing datasets:

	Name	Type
1	HILearn	Dataset (default)
2	HILearn	SQL endpoint
3	HILearn	Table

5. Select the table SynapseNamesTable to view the data loaded into Lakehouse.

The screenshot shows the Microsoft Fabric Data Catalog interface. On the left, the 'Select new endpoint' sidebar is open, showing a list of endpoints. The 'SynapseNamesTable' endpoint is highlighted with a gray box. On the right, the 'Compare two tables' section displays the table structure:

	Name	Schema	Last name	First name
1	Patricia	1000	Patricia	Patricia
2	Patricia	1000	Patricia	Patricia

Next steps

This sample shows you how to preprocess data with a stored procedure before loading the results into Lakehouse. You learned how to:

- ✓ Create a data pipeline with a Script activity to run a stored procedure.
- ✓ Use a pipeline activity to load the preprocessed table data into Lakehouse.
- ✓ Execute the pipeline activities to load the data.

Next, advance to learn more about monitoring your pipeline runs.

[How to monitor pipeline runs in Microsoft Fabric](#)

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Move data from Azure SQL DB into Lakehouse via copy assistant

Article • 11/15/2023

This tutorial describes the steps to move data into Lakehouse.

Two approaches are provided using the copy assistant:

1. The first approach moves source data into destination tables with delta format.
2. The second approach moves source data into destination files.

Prerequisites

To get started, you must complete the following prerequisites:

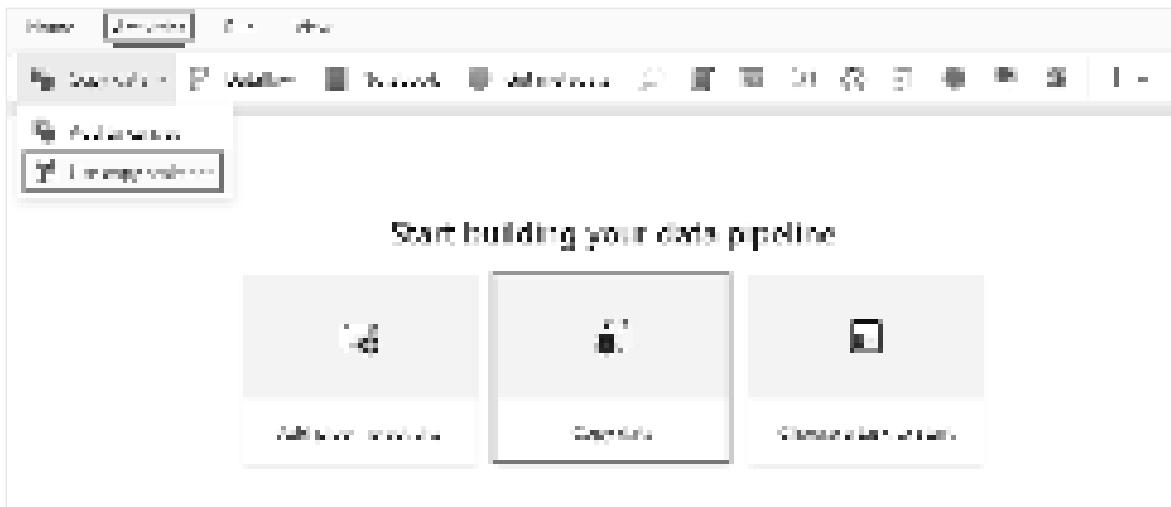
- A tenant account with an active subscription. Create an account for free.
- A workspace is created.
- A Lakehouse is created in your workspace.

Move files into Lakehouse as tables in delta format via copy assistant

Follow these steps to set up your copy activity.

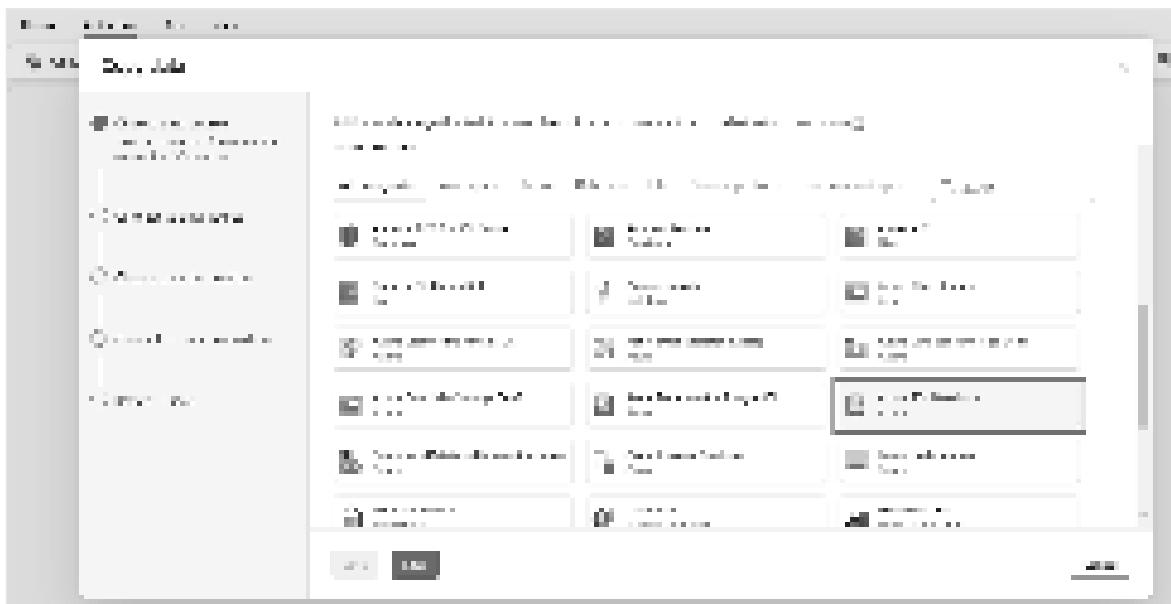
Step 1: Start with copy assistant

1. Open an existing data pipeline or create a new data pipeline.
2. Select **Copy data** on the canvas to open the **Copy Assistant** tool to get started. Or select **Use copy assistant** from the **Copy data** drop down list under **Activities** tab on the ribbon.

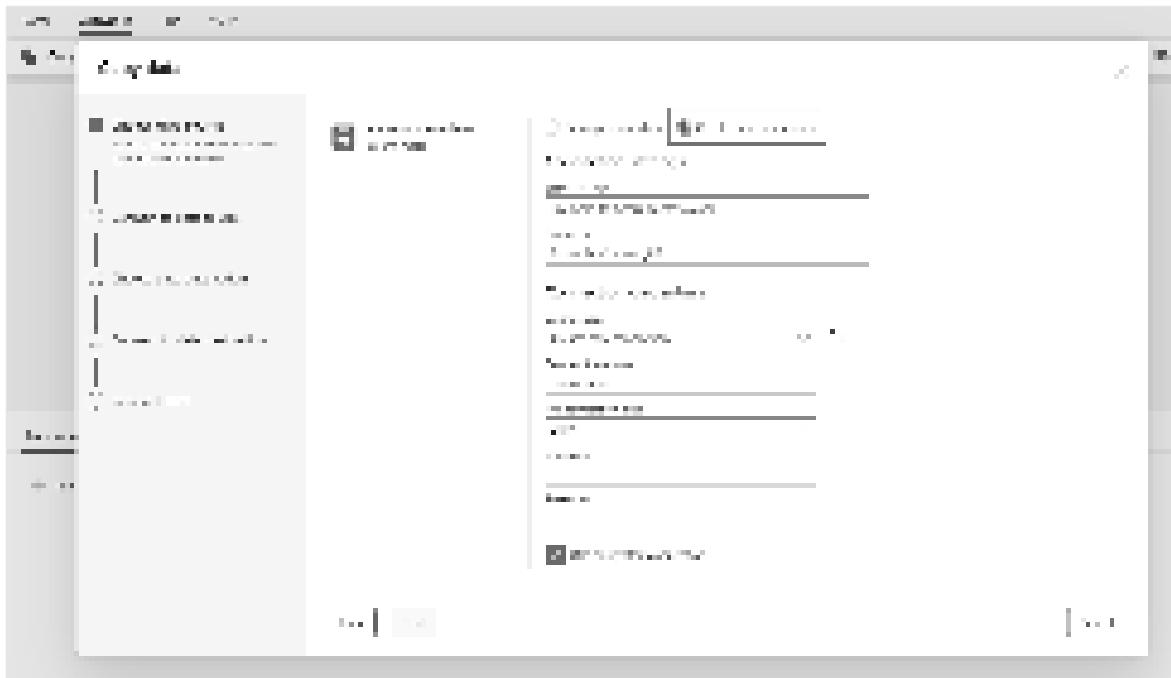


Step 2: Configure your source

1. Choose your data source by choosing a data source type. In this tutorial, we'll use Azure SQL Database as an example. Scroll down on the **Choose data source** screen to find and select **Azure SQL Database** and then select **Next**.

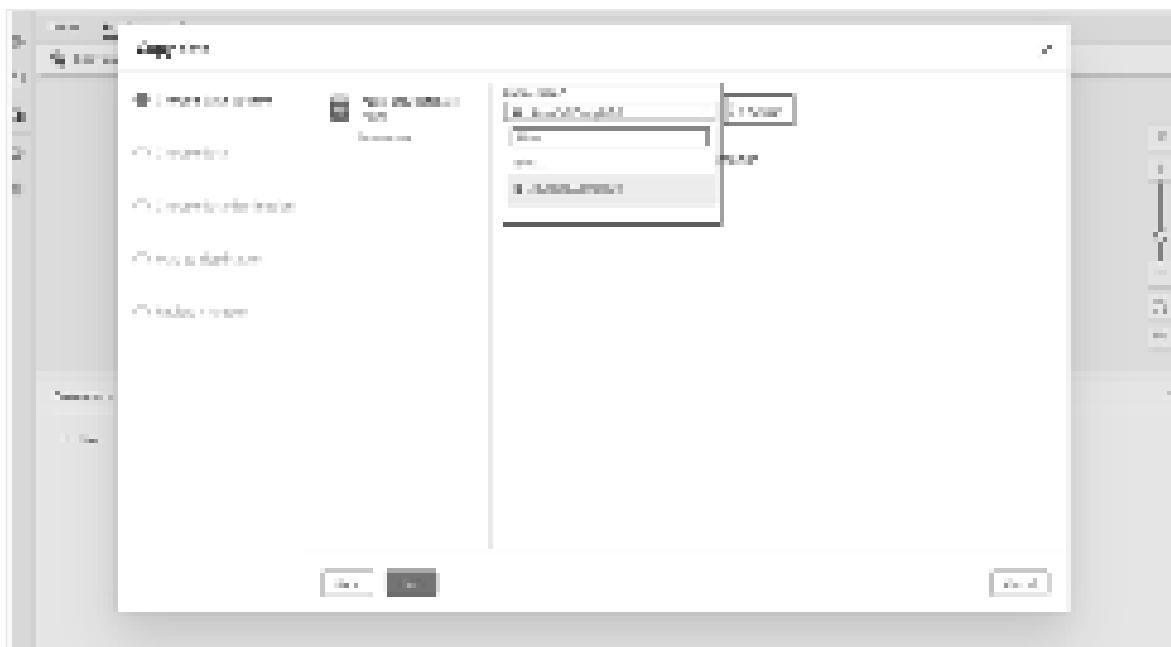


2. Create a connection to your data source by selecting **New Connection**, and filling in the required connection information on the panel.



After you fill in the required connection information on the panel, select **Next**.

Once your connection is created successfully, you will see a list of tables you can select.



3. Select the table(s) that is to be moved. Then, select **Next**.



Step 3: Configure your destination

1. Choose **Lakehouse** as your destination and then go to next.



Select your existing Lakehouse from your current workspace directly and then go to next.



2. Configure your table settings in Lakehouse. Select **Tables** under Root folder and specify the **table name**. Select **Next** to continue.

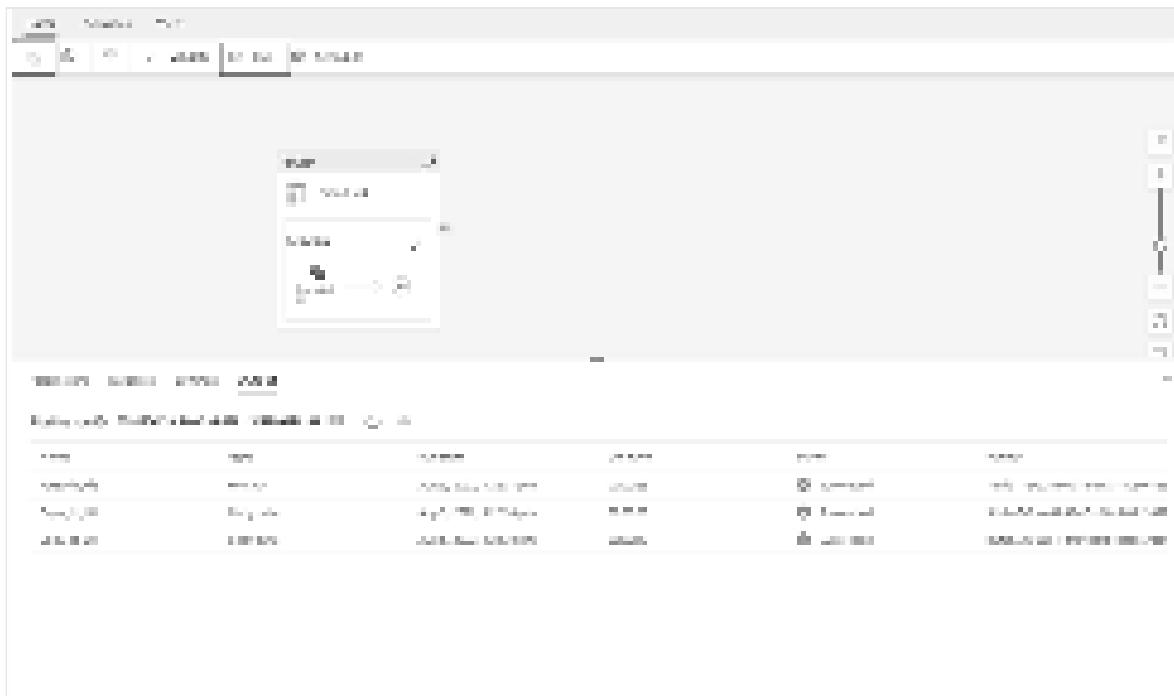


3. Select **OK** to finish the assistant experience.



Step 4: Save your data pipeline and run it to load data

1. Select **Save**. Then, select **Run**

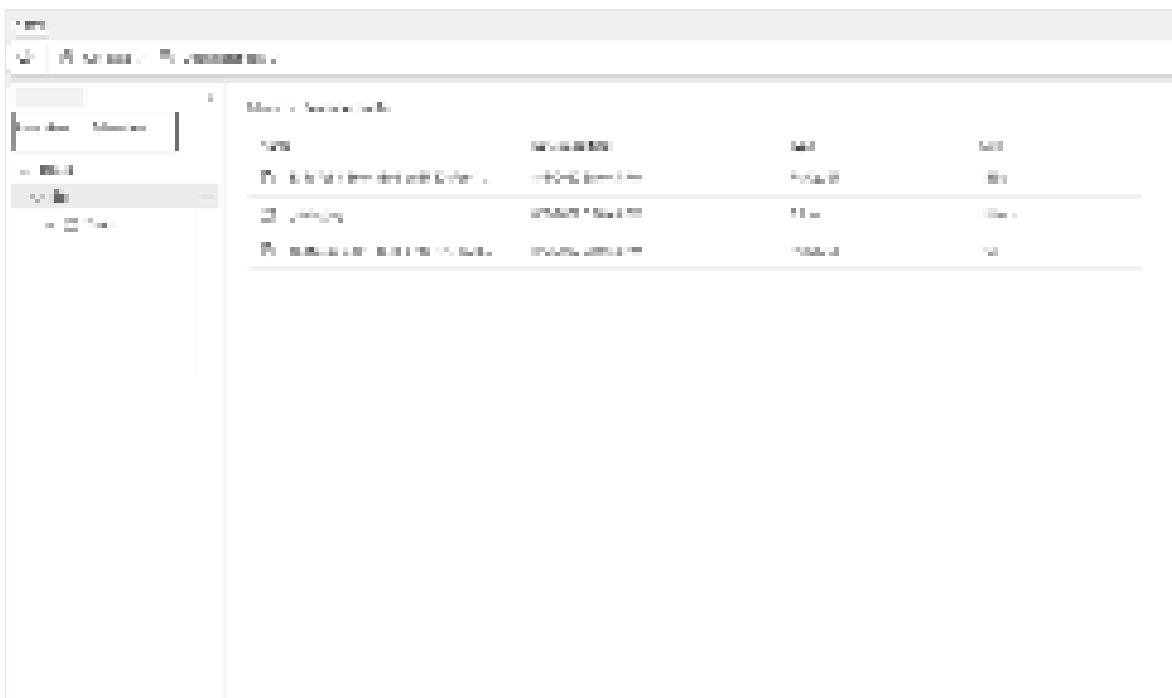


2. Select the **glasses** icon to view the details for each copy activity run:



Step 5: View your tables from Lakehouse

1. Go to your Lakehouse and refresh your **Lake view** to see the latest data ingested.
2. Switch to **Table view** to view the data in table.



(!) Note

Currently data lands into Lakehouse Tables folders (a managed area) in Delta format only. Those files will be automatically registered as a table and be visible under Table view from Lakehouse portal. Only the first layer folders under Tables will be registered as delta table. Browsing or Previewing from Lakehouse Table isn't supported yet. Data that gets loaded into the same table will be appended. Delete or Update to tables isn't supported yet.

Next steps

This sample shows you how to move data from Azure SQL DB into Lakehouse with the Copy Assistant in Data Factory for Microsoft Fabric. You learned how to:

- ✓ Move files into Lakehouse as tables in delta format with the Copy Assistant.

Next, advance to learn more about monitoring your pipeline runs.

[How to monitor pipeline runs in Microsoft Fabric](#)

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#) | [Ask the community](#)

Load Sample data to Data Warehouse

Article • 11/15/2023

In this tutorial, you build a data pipeline to move a Sample dataset to the Data Warehouse. This experience shows you a quick demo about how to use pipeline copy activity and how to load data into Data Warehouse.

Prerequisites

To get started, you must complete the following prerequisites:

- A Microsoft Fabric tenant account with an active subscription. Create an account for free.
- Make sure you have a Microsoft Fabric enabled Workspace: Create a workspace.
- Make sure you have already created a Data Warehouse. To create it, refer to Create a Data Warehouse

Create a data pipeline

1. Navigate to Power BI .
2. Select the Power BI icon in the bottom left of the screen, then select **Data factory** to open homepage of Data Factory.



3. Navigate to your Microsoft Fabric workspace. If you created a new workspace in the prior Prerequisites section, use this one.

The screenshot shows the Power BI Data Flow interface. On the left, there's a sidebar with icons for Home, Data, Reports, and Components. The main area has tabs at the top: 'Data Flow' (selected), 'Report', 'Visuals', and 'Components'. Below the tabs, there's a 'Recent' section with items like 'Newest', 'Last used', and 'Recently used'. To the right, there's a table-like view with columns 'Name', 'Type', and 'Last updated'. It lists three pipelines: 'Data Pipeline 1' (Type: Data pipeline, Last updated: 7 days ago), 'Data Pipeline 2' (Type: Data pipeline, Last updated: 7 days ago), and 'Data Pipeline 3' (Type: Data pipeline, Last updated: 7 days ago). At the bottom, there's a 'Create new Data Flow' button.

4. Select Data pipeline and then input a pipeline name to create a new pipeline.

The screenshot shows the Data Factory 'DF Getting Started' interface. On the left, there's a sidebar with icons for Home, Data, Reports, and Components. The main area has tabs at the top: 'DF Getting Started' (selected), 'Data Flow', 'Report', and 'Components'. Below the tabs, there's a 'Create' section with buttons for '+ New', 'Upload...', and 'Create from'. A dropdown menu is open, showing options: 'Data pipeline' (selected), 'Data Flow', 'Event hubs', 'Experiment', 'KQL Query', 'Lakehouse', 'Machine learning', 'Notebook', 'Refine (Preview)', 'Report', 'SSIS Job Definition', and 'Variables'. At the bottom, there's a 'Import options' section with a 'Import notebook' button.

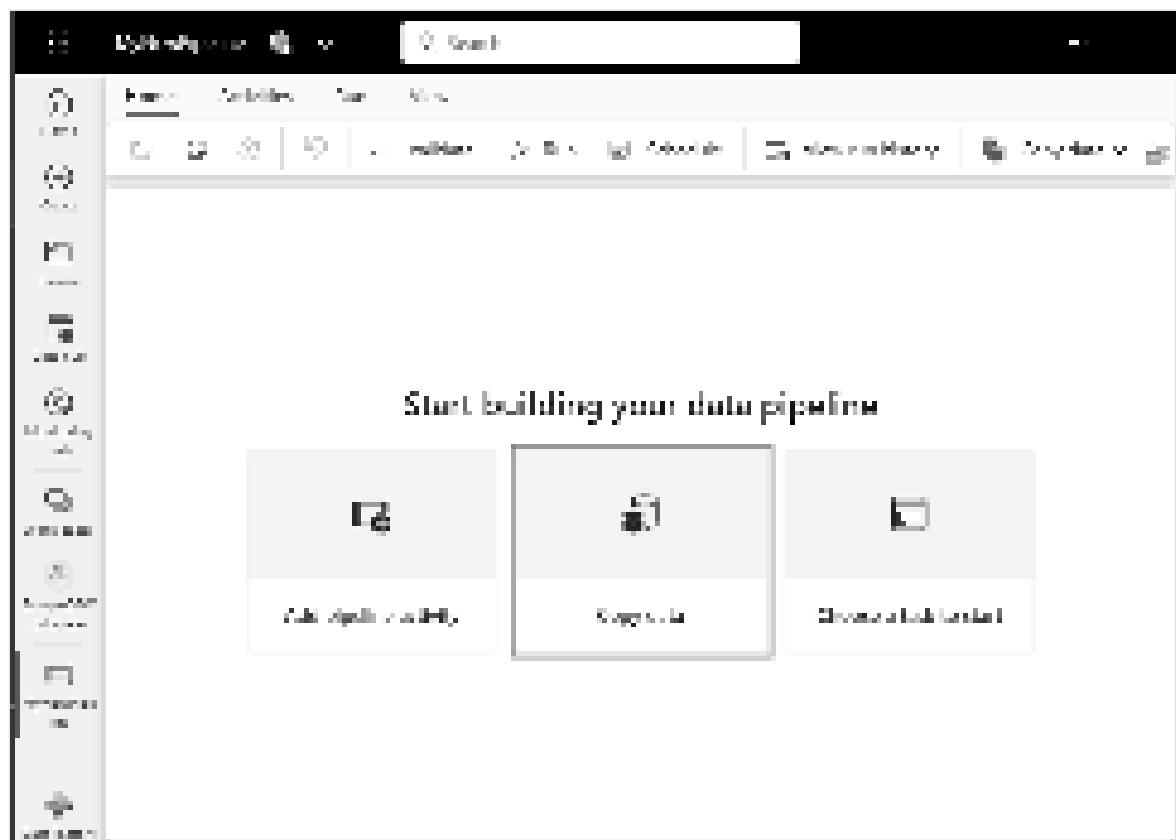


Copy data using pipeline

In this session, you start to build your pipeline by following below steps about copying from a sample dataset provided by pipeline into Data Warehouse.

Step 1: Start with the Copy assistant

1. After selecting **Copy data** on the canvas, the **Copy assistant** tool will be opened to get started.

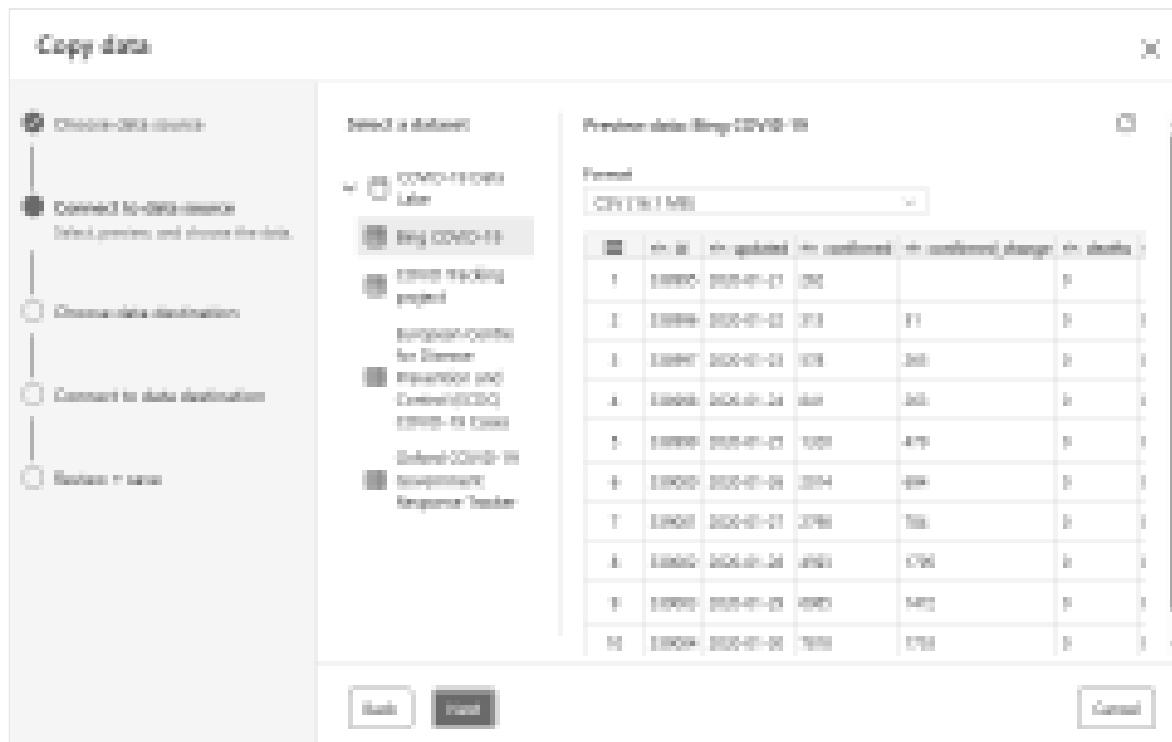


Step 2: Configure your source

1. Choose the **COVID-19 Data Lake** from the **Sample data** options for your data source, and then select **Next**.

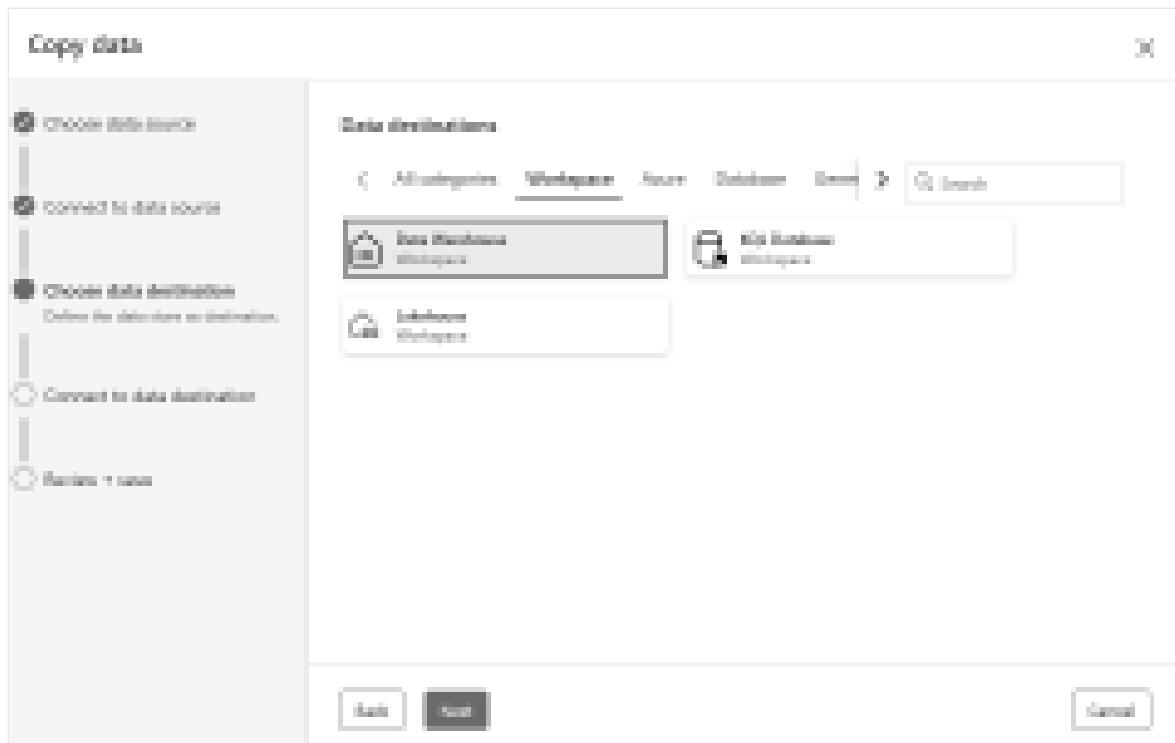


2. In the **Connect to data source** section of the **Copy data** assistant, a preview of the sample data **Bing COVID-19** is displayed. Select **Next** to move on to the data destination.

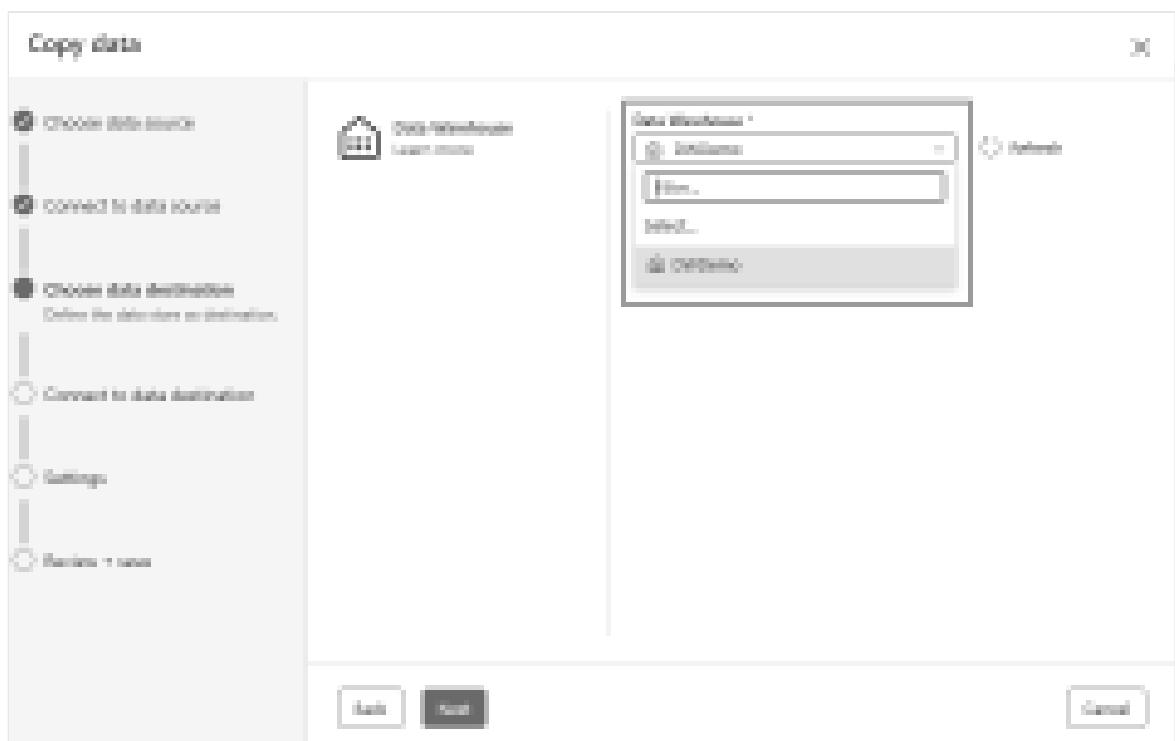


Step 3: Configure your destination

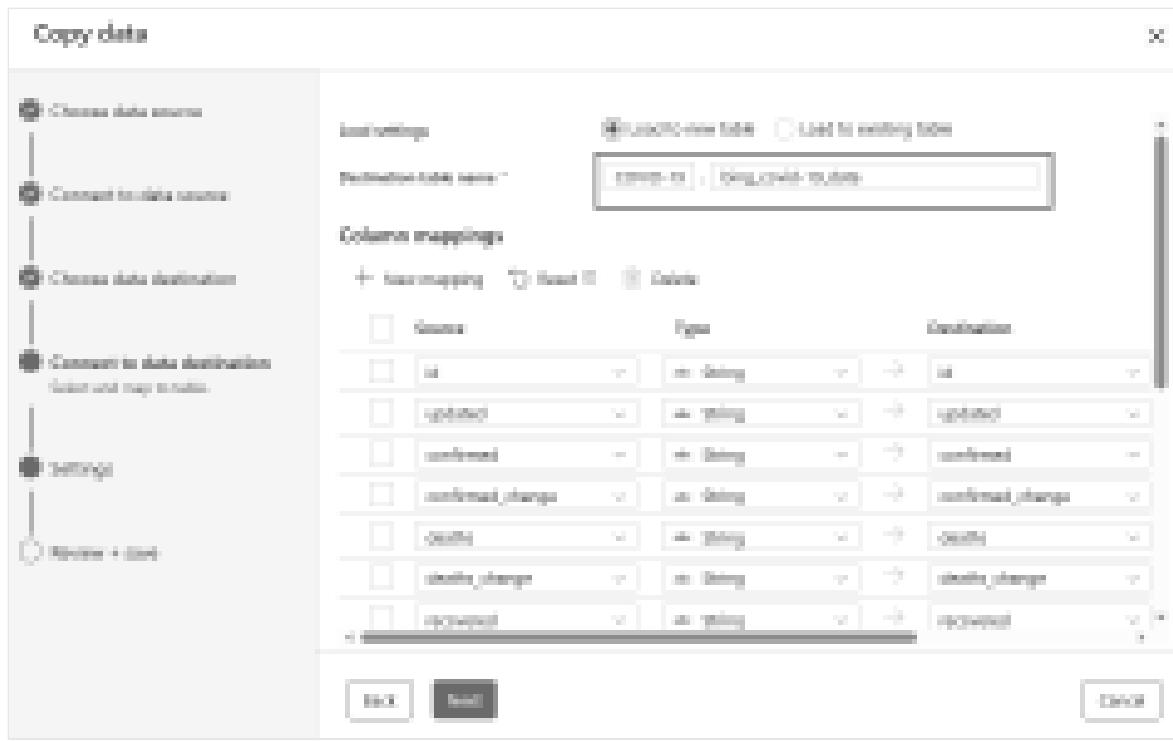
1. Select the **Workspace tab** and choose **Data warehouse**. Then select **Next**.



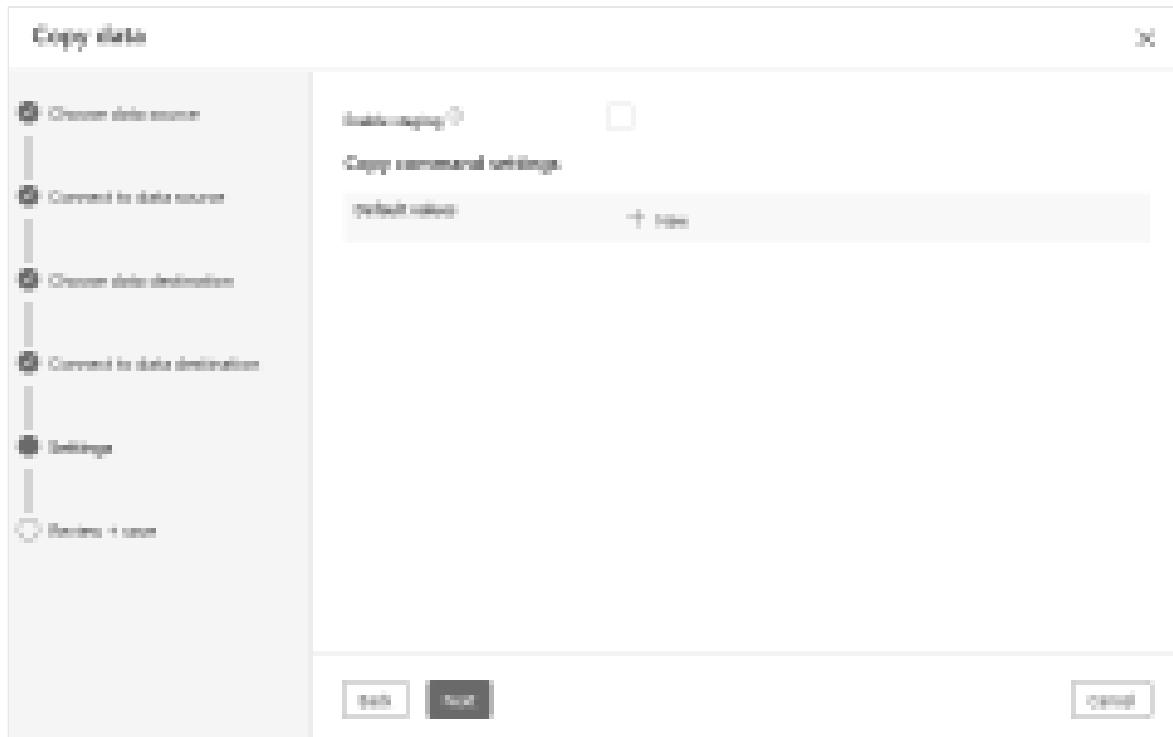
2. Select your Data Warehouse from the drop-down list, then select **Next**.



3. Configure and map your source data to the destination Data Warehouse table by entering **Destination table name**, then select **Next** one more time.

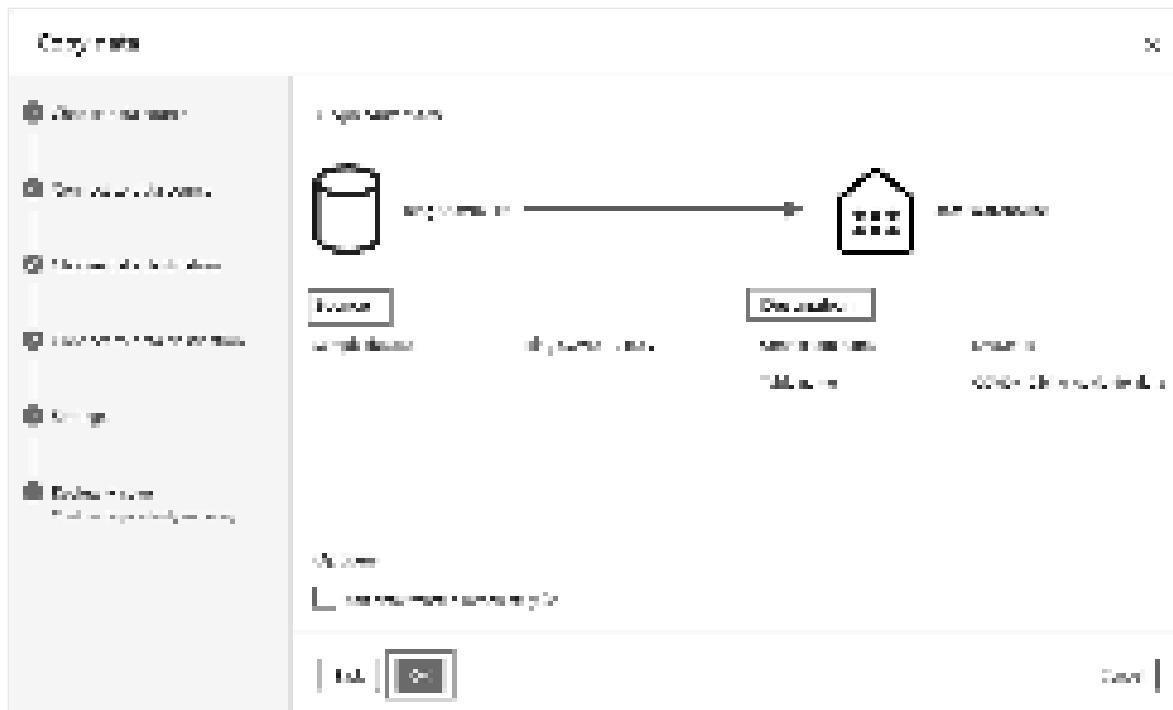


4. Configure other settings on **Settings** page. In this tutorial, select **Next** directly since you don't need to use staging and copy command.

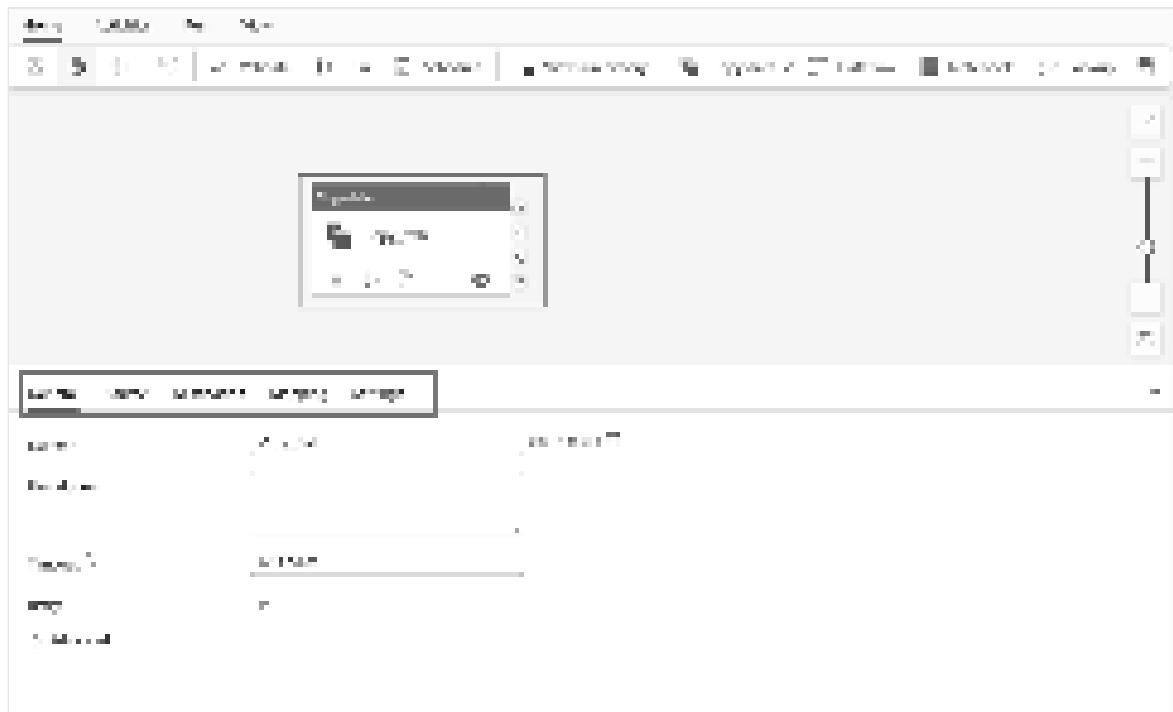


Step 4: Review and create your copy activity

1. Review your copy activity settings in the previous steps and select **OK** to finish. Or you can revisit the previous steps in the tool to edit your settings, if needed.

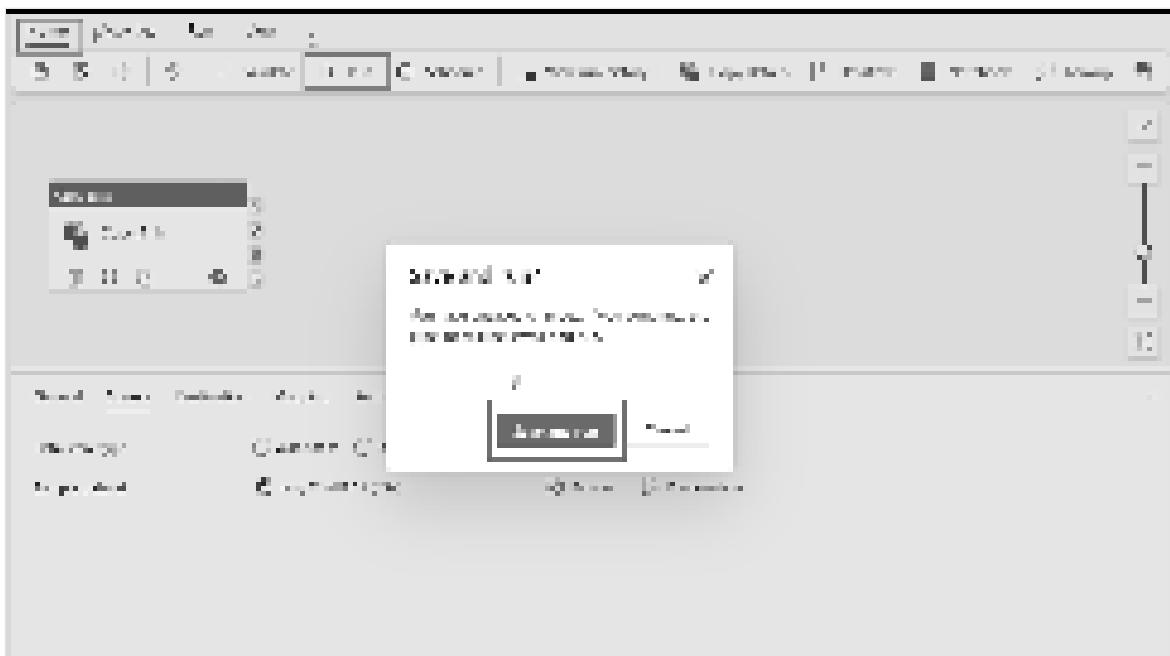


2. The **Copy activity** is added to your new data pipeline canvas. All settings including advanced settings for the activity are available in the tabs below the pipeline canvas when the created **Copy data** activity is selected.

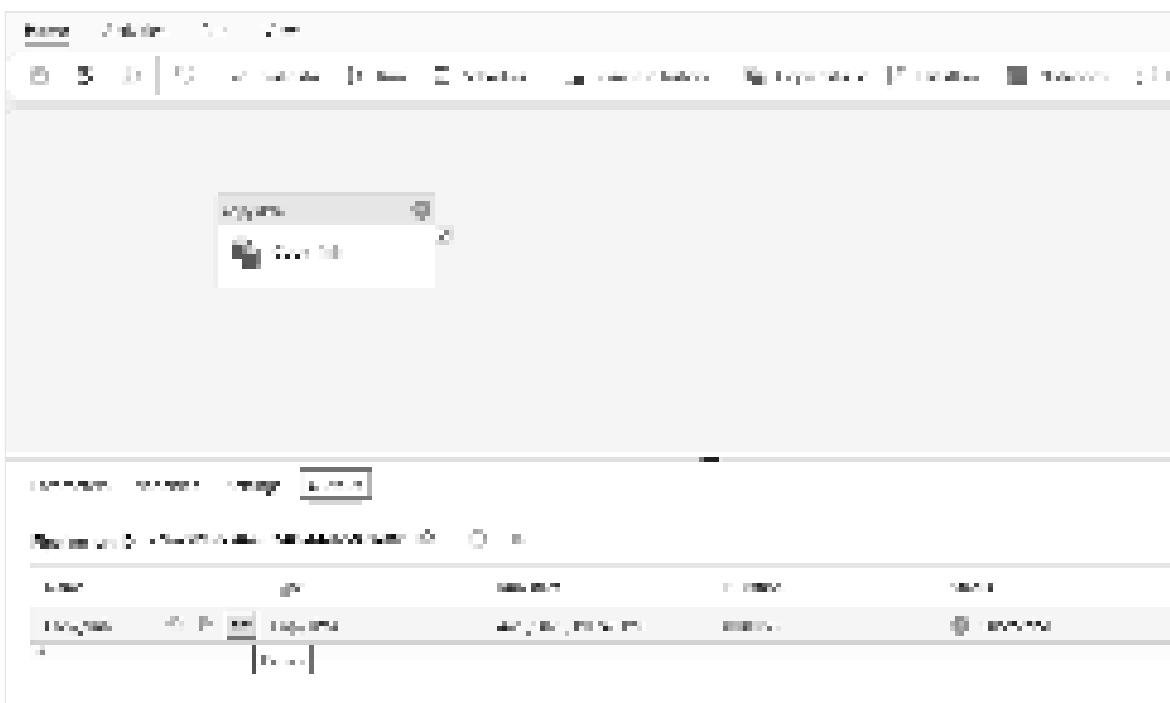


Run and schedule your data pipeline

1. Switch to the **Home** tab and select **Run**. A confirmation dialog is displayed. Then select **Save and run** to start the activity.



2. You can monitor the running process and check the results on the **Output** tab below the pipeline canvas. Select the run details button (with the glasses icon highlighted) to view the run details.

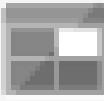


3. The run details show how much data was read and written and various other details about the run.

Copy data details

Copy_1st

Source

 Azure Blob Storage

Data reads: 571,757
MB

Block count: 1

Destination

 Data Warehouse

Data writes: 571,757
MB

Row writes: 4,705,716

Status

 Completed

Start time

4/24/2025, 10:50:11 AM

Pipeline run activity ID

c0de3be1-0340-45ca-9e63-131761c1ec12

Throughput

12.75 MB/s

Total duration

0h20m11s

[Duration breakdown](#)

[Advanced](#)

[Close](#)

4. You can also schedule the pipeline to run with a specific frequency as required.

Below is an example scheduling the pipeline to run every 15 minutes. You can also specify the **Start** time and **End** time for your schedule. If you don't specify a start time, the start time is the time your schedule applies. If you don't specify an end time, your pipeline run will keep recurring every 15 minutes.



Next steps

This sample shows you how to load sample data into a Data Warehouse using Data Factory in Microsoft Fabric. You learned how to:

- ✓ Create a data pipeline.
- ✓ Copy data using your pipeline.
- ✓ Run and schedule your data pipeline.

Next, advance to learn more about monitoring your pipeline runs.

[How to monitor pipeline runs in Microsoft Fabric](#)

Feedback

Was this page helpful?

Yes

No

Provide product feedback | Ask the community

Incrementally load data from Data Warehouse to Lakehouse

Article • 11/15/2023

In this tutorial, you learn how to incrementally load data from Data Warehouse to Lakehouse.

Overview

Here is the high-level solution diagram:



Here are the important steps to create this solution:

1. **Select the watermark column.** Select one column in the source data table, which can be used to slice the new or updated records for every run. Normally, the data in this selected column (for example, `last_modify_time` or `ID`) keeps increasing when rows are created or updated. The maximum value in this column is used as a watermark.
2. **Prepare a table to store the last watermark value in your Data Warehouse.**
3. **Create a pipeline with the following workflow:**

The pipeline in this solution has the following activities:

- Create two lookup activities. Use the first lookup activity to retrieve the last watermark value. Use the second lookup activity to retrieve the new watermark value. These watermark values are passed to the copy activity.
- Create a copy activity that copies rows from the source data table with the value of the watermark column greater than the old watermark value and less than the new watermark value. Then, it copies the data from the Data Warehouse to Lakehouse as a new file.
- Create a stored procedure activity that updates the last watermark value for the next pipeline run.

Prerequisites

- **Data Warehouse.** You use the Data Warehouse as the source data store. If you don't have it, see [Create a Data Warehouse](#) for steps to create one.
- **Lakehouse.** You use the Lakehouse as the destination data store. If you don't have it, see [Create a Lakehouse](#) for steps to create one. Create a folder named *IncrementalCopy* to store the copied data.

Preparing your source

Here are some tables and stored procedure that you need to prepare in your source Data Warehouse before configuring the incremental copy pipeline.

1. Create a data source table in your Data Warehouse

Run the following SQL command in your Data Warehouse to create a table named *data_source_table* as the data source table. In this tutorial, you'll use it as the sample data to do the incremental copy.

SQL

```
create table data_source_table
(
    PersonID int,
    Name varchar(255),
    LastModifytime DATETIME2(6)
);

INSERT INTO data_source_table
    (PersonID, Name, LastModifytime)
VALUES
    (1, 'aaaa', '9/1/2017 12:56:00 AM'),
    (2, 'bbbb', '9/2/2017 5:23:00 AM'),
    (3, 'cccc', '9/3/2017 2:36:00 AM'),
    (4, 'dddd', '9/4/2017 3:21:00 AM'),
    (5, 'eeee', '9/5/2017 8:06:00 AM');
```

The data in the data source table is shown below:

PersonID	Name	LastModifytime
1	aaaa	2017-09-01 00:56:00.000
2	bbbb	2017-09-02 05:23:00.000
3	cccc	2017-09-03 02:36:00.000
4	dddd	2017-09-04 03:21:00.000
5	eeee	2017-09-05 08:06:00.000

In this tutorial, you use *LastModifytime* as the watermark column.

2. Create another table in your Data Warehouse to store the last watermark value

1. Run the following SQL command in your Data Warehouse to create a table named *watermarktable* to store the last watermark value:

SQL

```
create table watermarktable
(
    TableName varchar(255),
```

```
WatermarkValue DATETIME2(6),  
);
```

2. Set the default value of the last watermark with the table name of source data table. In this tutorial, the table name is *data_source_table*, and the default value is `1/1/2010 12:00:00 AM`.

SQL

```
INSERT INTO watermarktable  
VALUES ('data_source_table','1/1/2010 12:00:00 AM')
```

3. Review the data in the table *watermarktable*.

SQL

```
Select * from watermarktable
```

Output:

TableName	WatermarkValue
----- -----	
data_source_table	2010-01-01 00:00:00.000

3. Create a stored procedure in your Data Warehouse

Run the following command to create a stored procedure in your Data Warehouse. This stored procedure is used to help updates the last watermark value after last pipeline run.

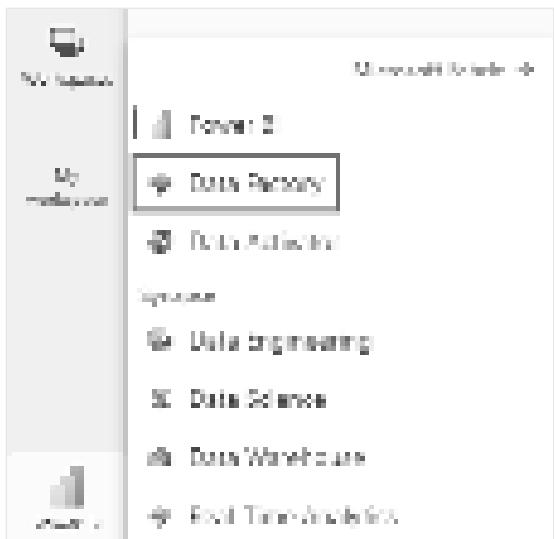
SQL

```
CREATE PROCEDURE usp_write_watermark @LastModifiedtime datetime, @TableName varchar(50)  
AS  
  
BEGIN  
  
UPDATE watermarktable  
SET [WatermarkValue] = @LastModifiedtime  
WHERE [TableName] = @TableName  
  
END
```

Configure a pipeline for incremental copy

Step 1: Create a pipeline

1. Navigate to Power BI .
2. Select the Power BI icon in the bottom left of the screen, then select **Data factory** to open homepage of Data Factory.



3. Navigate to your Microsoft Fabric workspace.

4. Select **Data pipeline** and then input a pipeline name to create a new pipeline.

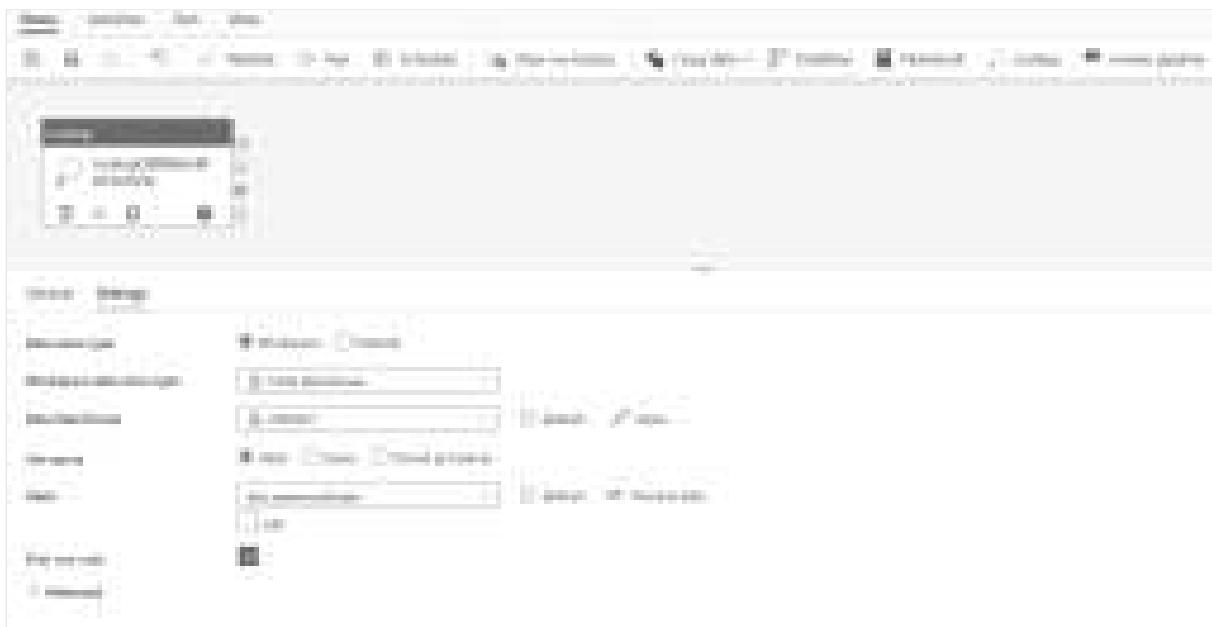




Step 2: Add a lookup activity for the last watermark

In this step, you create a lookup activity to get the last watermark value. The default value `1/1/2010 12:00:00 AM` set before will be obtained.

1. Select **Add pipeline activity** and select **Lookup** from the drop-down list.
2. Under **General** tab, rename this activity to **LookupOldWaterMarkActivity**.
3. Under **Settings** tab, perform the following configuration:
 - **Data store type:** Select **Workspace**.
 - **Workspace data store type:** Select **Data Warehouse**.
 - **Data Warehouse:** Select your Data Warehouse.
 - **Use query:** Choose **Table**.
 - **Table:** Choose *dbo.watermarktable*.
 - **First row only:** Selected.



Step 3: Add a lookup activity for the new watermark

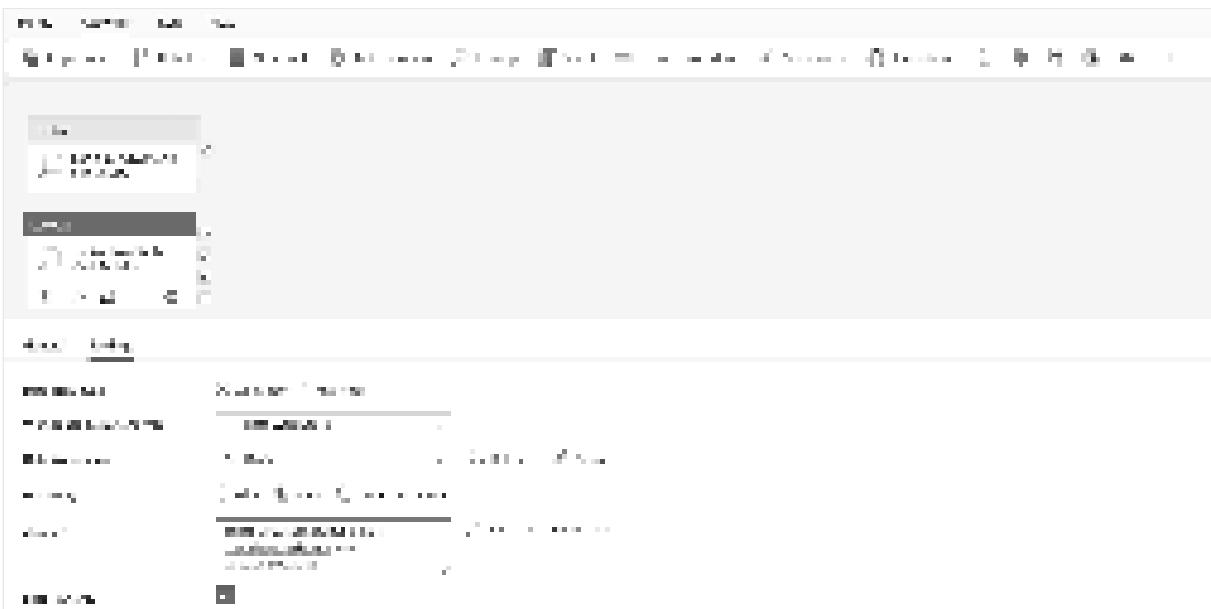
In this step, you create a lookup activity to get the new watermark value. You'll use a query to obtain the new watermark from your source data table. The maximum value in *LastModifytime* column in

data_source_table will be obtained.

1. On the top bar, select **Lookup** under **Activities** tab to add the second lookup activity.
2. Under **General** tab, rename this activity to **LookupNewWaterMarkActivity**.
3. Under **Settings** tab, perform the following configuration:
 - **Data store type:** Select **Workspace**.
 - **Workspace data store type:** Select **Data Warehouse**.
 - **Data Warehouse:** Select your Data Warehouse.
 - **Use query:** Choose **Query**.
 - **Query:** Enter the following query to pick the maximum last modified time as the new watermark:

```
SQL
select MAX(LastModifytime) as NewWatermarkvalue from data_source_table
```

- **First row only:** Selected.



Step 4: Add the copy activity to copy incremental data

In this step, you add a copy activity to copy the incremental data between the last watermark and new watermark from Data Warehouse to Lakehouse.

1. Select **Activities** on the top bar and select **Copy data** -> **Add to canvas** to get the copy activity.
2. Under **General** tab, rename this activity to **IncrementalCopyActivity**.
3. Connect both Lookup activities to the copy activity by dragging the green button (On success) attached to the lookup activities to the copy activity. Release the mouse button when you see the border color of the copy activity changes to green.

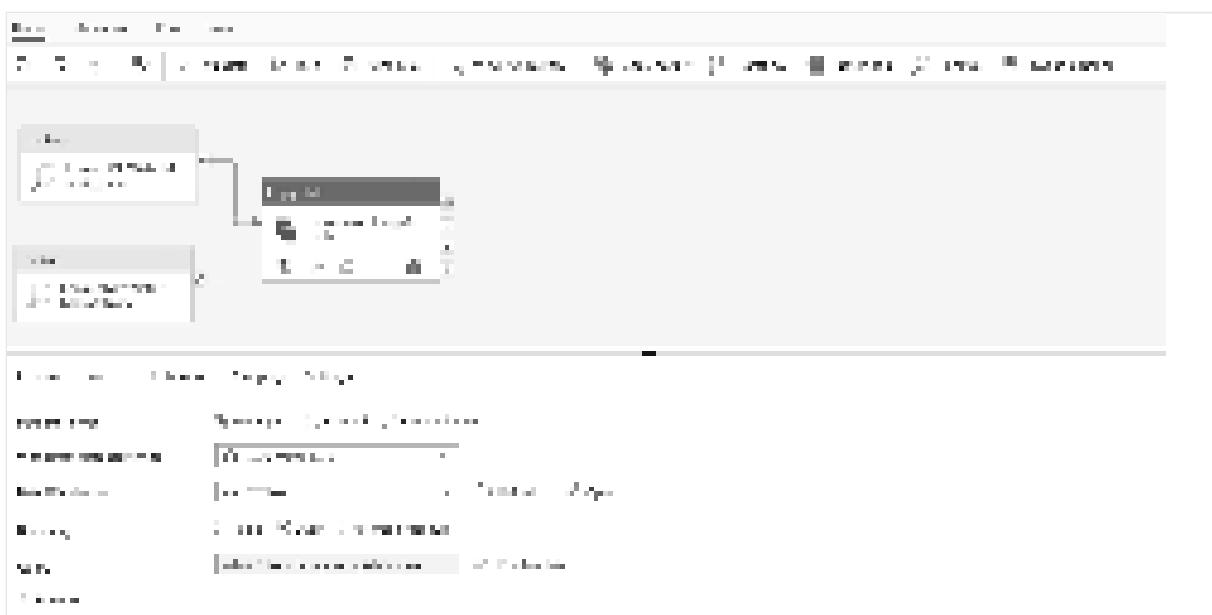


4. Under **Source** tab, perform the following configuration:

- **Data store type:** Select **Workspace**.
- **Workspace data store type:** Select **Data Warehouse**.
- **Data Warehouse:** Select your Data Warehouse.
- **Use query:** Choose **Query**.
- **Query:** Enter the following query to copy incremental data between last watermark and new watermark.

```
SQL

select * from data_source_table where LastModifytime >
'{activity('LookupOldWaterMarkActivity').output.firstRow.WatermarkValue}' and
LastModifytime <=
'{activity('LookupNewWaterMarkActivity').output.firstRow.NewWatermarkvalue}'
```



5. Under **Destination** tab, perform the following configuration:

- **Data store type:** Select **Workspace**.
- **Workspace data store type:** Select **Lakehouse**.

- **Lakehouse:** Select your Lakehouse.
- **Root folder:** Choose Files.
- **File path:** Specify the folder that you want to store your copied data. Select **Browse** to select your folder. For the file name, open **Add dynamic content** and enter `@CONCAT('Incremental-', pipeline().RunId, '.txt')` in the opened window to create file names for your copied data file in Lakehouse.
- **File format:** Select the format type of your data.



Step 5: Add a stored procedure activity

In this step, you add a stored procedure activity to update the last watermark value for the next pipeline run.

1. Select **Activities** on the top bar and select **Stored procedure** to add a stored procedure activity.
2. Under **General** tab, rename this activity to **StoredProceduretoWriteWatermarkActivity**.
3. Connect the green (On success) output of the copy activity to the stored procedure activity.
4. Under **Settings** tab, perform the following configuration:
 - **Data store type:** Select **Workspace**.
 - **Data Warehouse:** Select your Data Warehouse.
 - **Stored procedure name:** Specify the stored procedure that you created in your Data Warehouse: `[dbo].[usp_write_watermark]`.
 - Expand **Stored procedure parameters**. To specify values for the stored procedure parameters, select **Import**, and enter following values for the parameters:

Name	Type	Value
LastModifiedtime	DateTime	@{activity('LookupNewWaterMarkActivity').output.firstRow.NewWatermarkvalue}
TableName	String	@{activity('LookupOldWaterMarkActivity').output.firstRow.TableName}

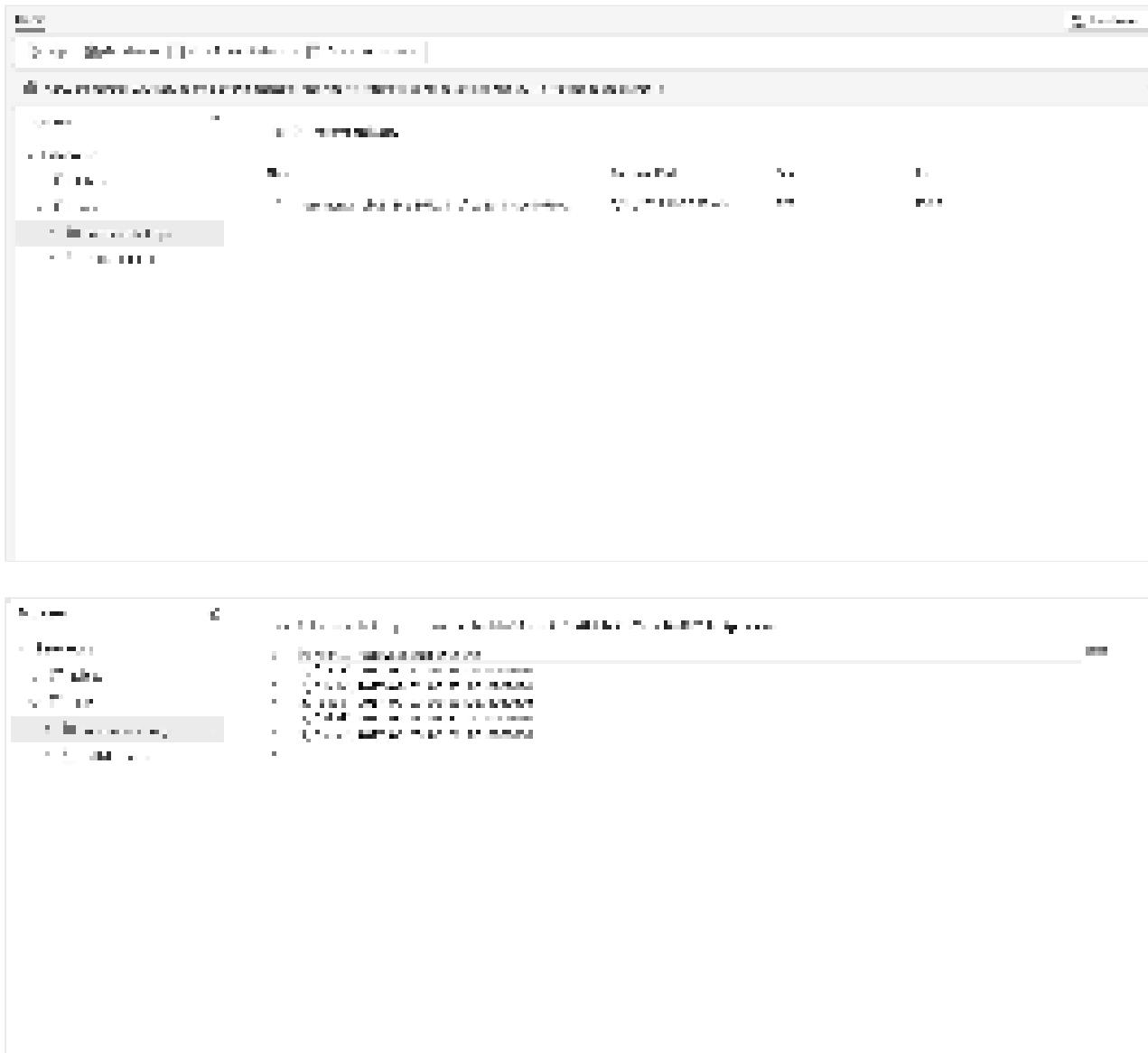


Step 6: Run the pipeline and monitor the result

On the top bar, select **Run** under **Home** tab. Then select **Save and run**. The pipeline starts run and you can monitor the pipeline under **Output** tab.



Go to your Lakehouse, you find the data file is under the folder that you specified and you can select the file to preview the copied data.



Add more data to see the incremental copy results

After you finish the first pipeline run, let's try to add more data in your Data Warehouse source table to see if this pipeline can copy your incremental data.

Step 1: Add more data to source

Insert new data into your Data Warehouse by running the following query:

SQL

```
INSERT INTO data_source_table
VALUES (6, 'newdata','9/6/2017 2:23:00 AM')

INSERT INTO data_source_table
VALUES (7, 'newdata','9/7/2017 9:01:00 AM')
```

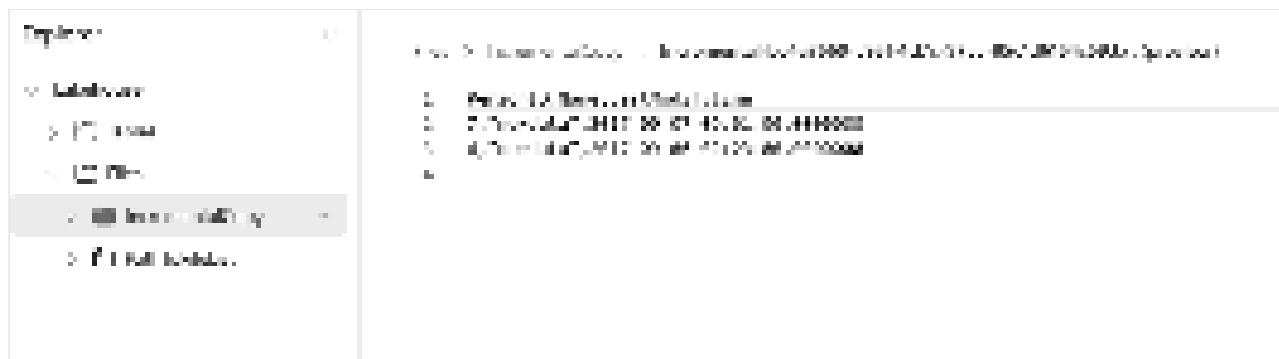
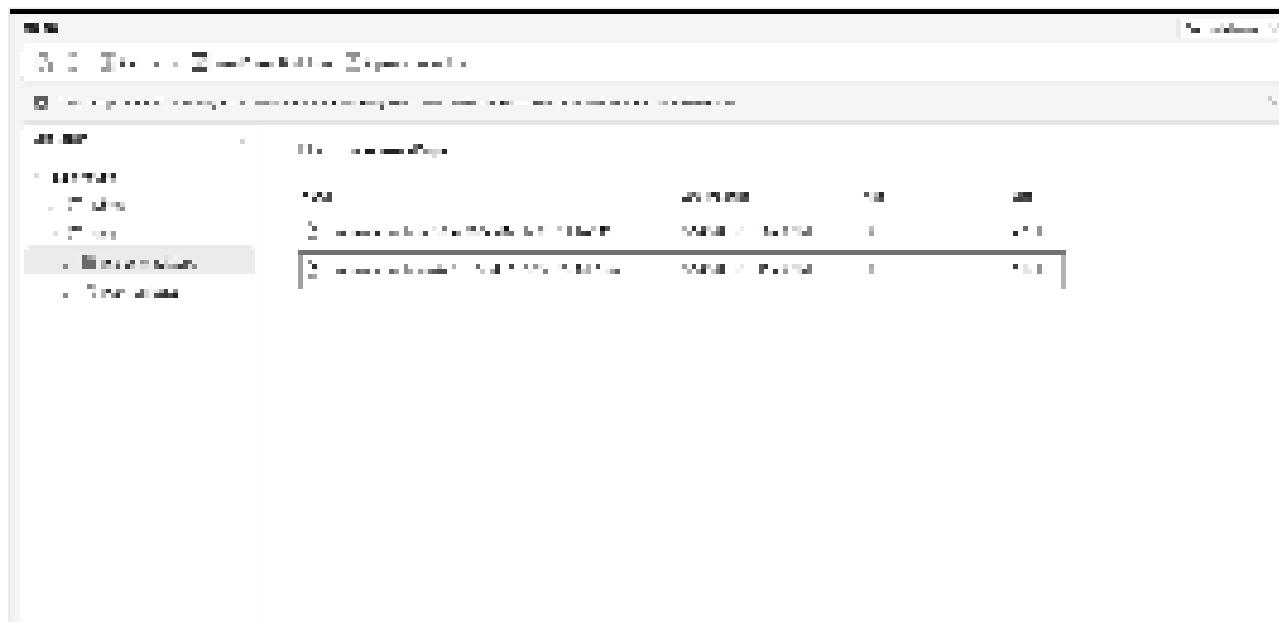
The updated data for *data_source_table* is:

PersonID	Name	LastModifytime
1	aaaa	2017-09-01 00:56:00.000
2	bbbb	2017-09-02 05:23:00.000
3	cccc	2017-09-03 02:36:00.000
4	dddd	2017-09-04 03:21:00.000
5	eeee	2017-09-05 08:06:00.000
6	newdata	2017-09-06 02:23:00.000
7	newdata	2017-09-07 09:01:00.000

Step 2: Trigger another pipeline run and monitor the result

Go back to your pipeline page. On the top bar, select **Run** under **Home** tab again. The pipeline starts run and you can monitor the pipeline under **Output**.

Go to your Lakehouse, you find the new copied data file is under the folder that you specified, and you can select the file to preview the copied data. You see your incremental data are shown in this file.



Next steps

Next, advance to learn more about copy from Azure Blob Storage to Lakehouse.

[Copy from Azure Blob Storage to Lakehouse](#)

Feedback

Was this page helpful?

Provide product feedback  | Ask the community 

Data source management

Article • 11/15/2023

Microsoft Fabric supports many on-premises and cloud data sources, and each source has its own requirements. To learn how to add and manage an on-premises data source, go to Add or remove a gateway data source. In this article, you'll learn how to add an Azure SQL Server as a cloud data source. The steps are similar for other data sources.

ⓘ Note

Currently, these cloud connections are only supported for data pipelines and Kusto. In the future, other items can also make use of the cloud connections. To create personal cloud connections in datasets, datamarts, and dataflows, use the Power Query Online experience in "get data".

Add a data source

1. From the page header in the Microsoft Fabric service, select the **Settings** icon, and then select **Manage connections and gateways**.



Settings

Settings

X

Preferences

General →

Notifications →

Item settings →

Developer settings →

Resources and extensions

Manage group storage →

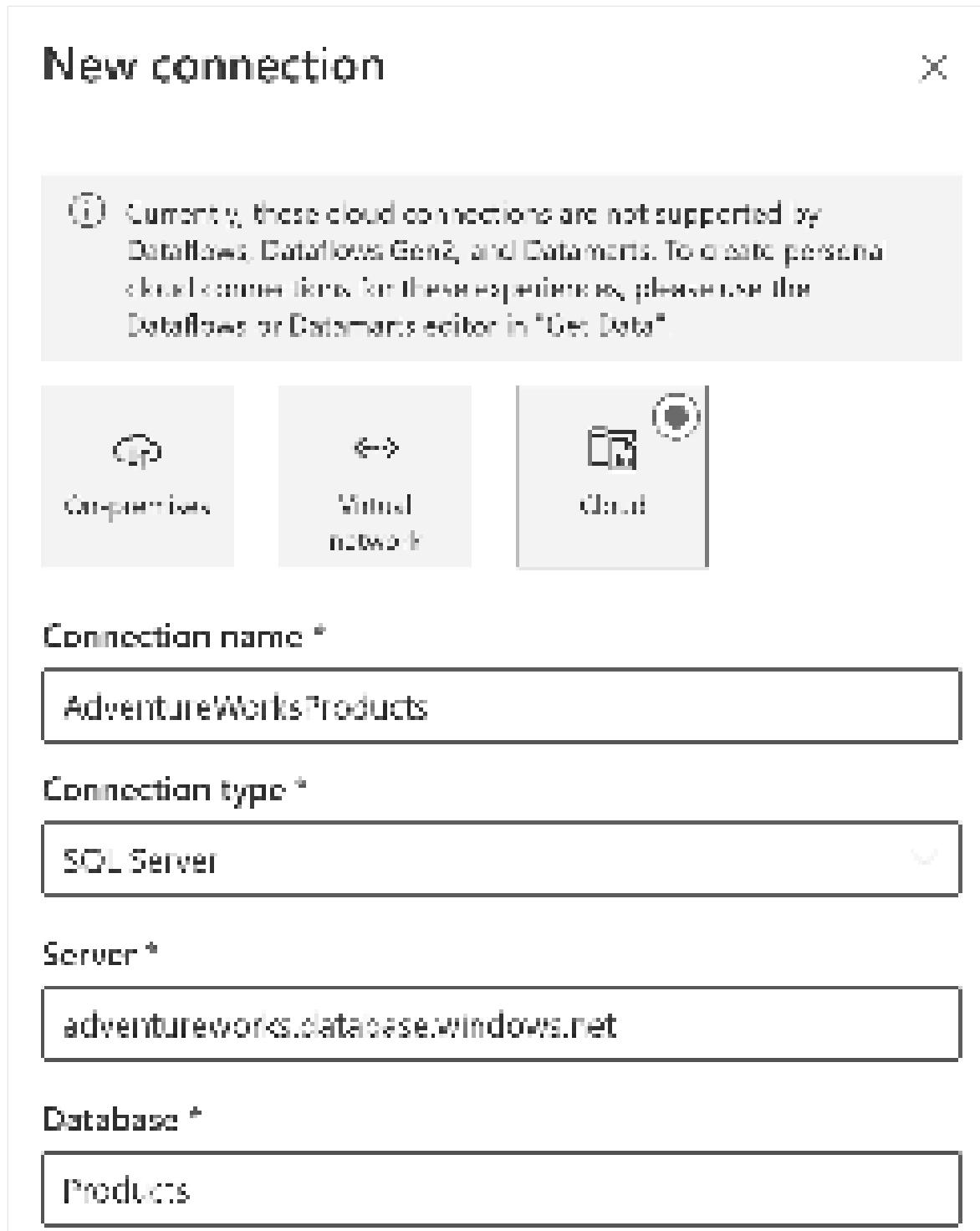
Power BI settings →

Manage connections and gateways →

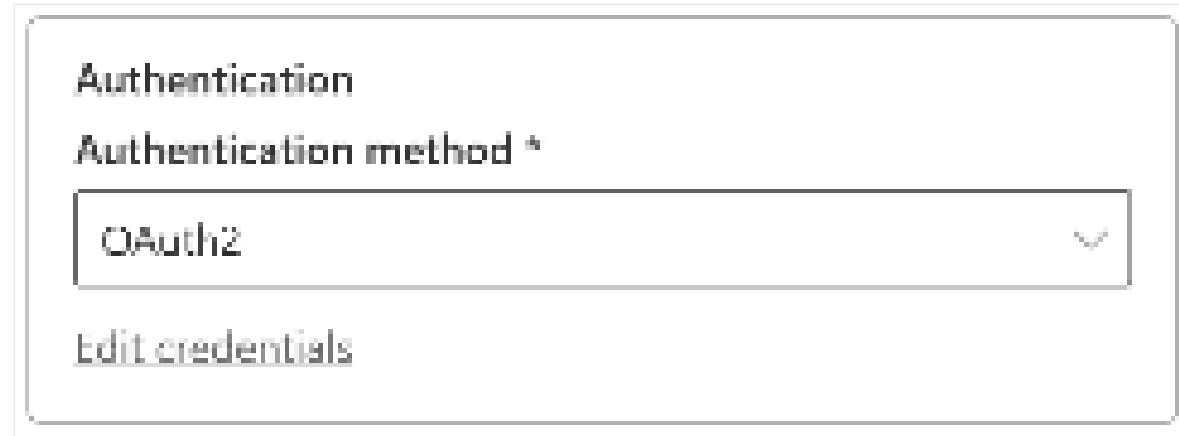
Manage embed codes →

2. Select the **Connections** tab, then select **New** at the top of the screen to add a new data source.

3. In the New connection screen, select **Cloud**, provide a **Connection name**, and select the **Connection Type**. For this example, choose **SQL server**.
4. Enter information about the data source. For SQL server, provide the **Server** and **Database**.



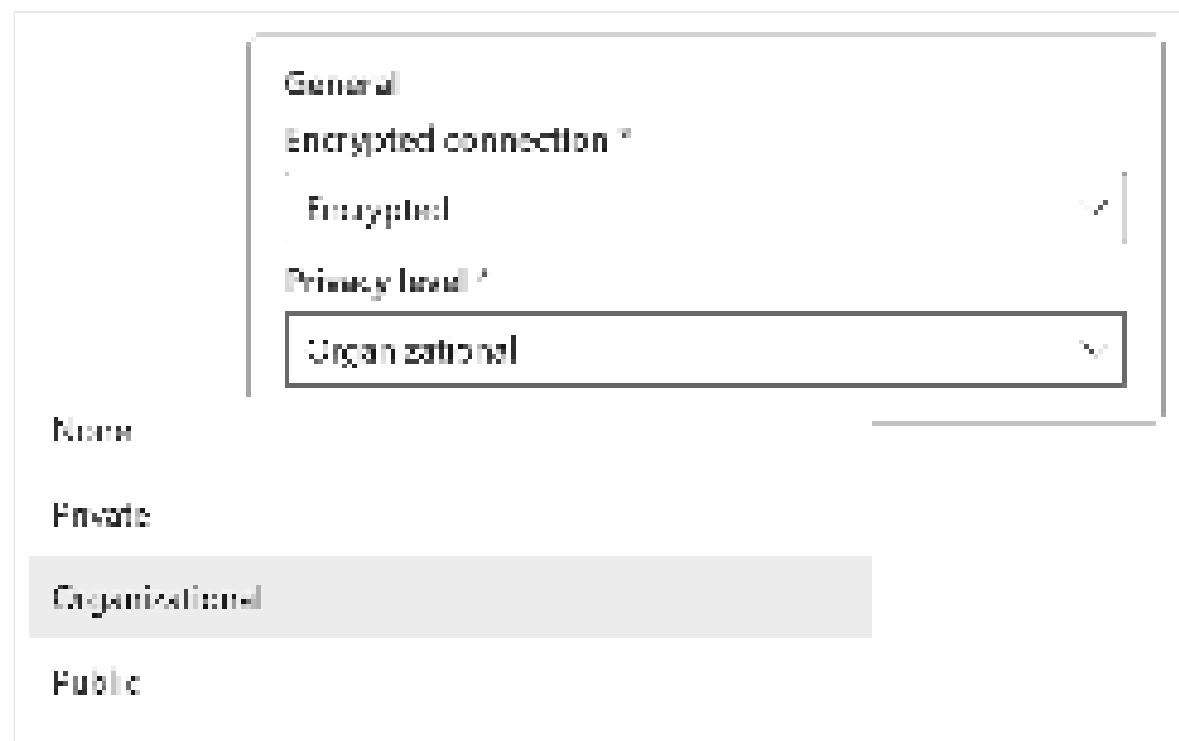
5. Select an **Authentication method** to use when connecting to the data source, either **Basic**, **OAuth2**, or **Service Principal**. For example, choose **OAuth2** and sign in with your account.



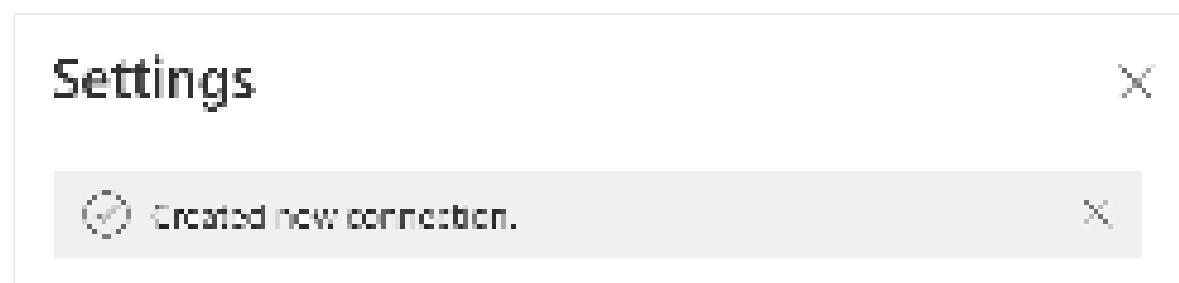
If you selected the **OAuth2** authentication method:

- Any query that runs longer than the OAuth token expiration policy might fail.
- Cross-tenant Azure Active Directory (Azure AD) accounts aren't supported.

6. Under **General > Privacy level**, optionally configure a privacy level for your data source. This setting doesn't apply to DirectQuery.



7. Select **Create**. Under **Settings**, you see **Created new connection** if the process succeeds.

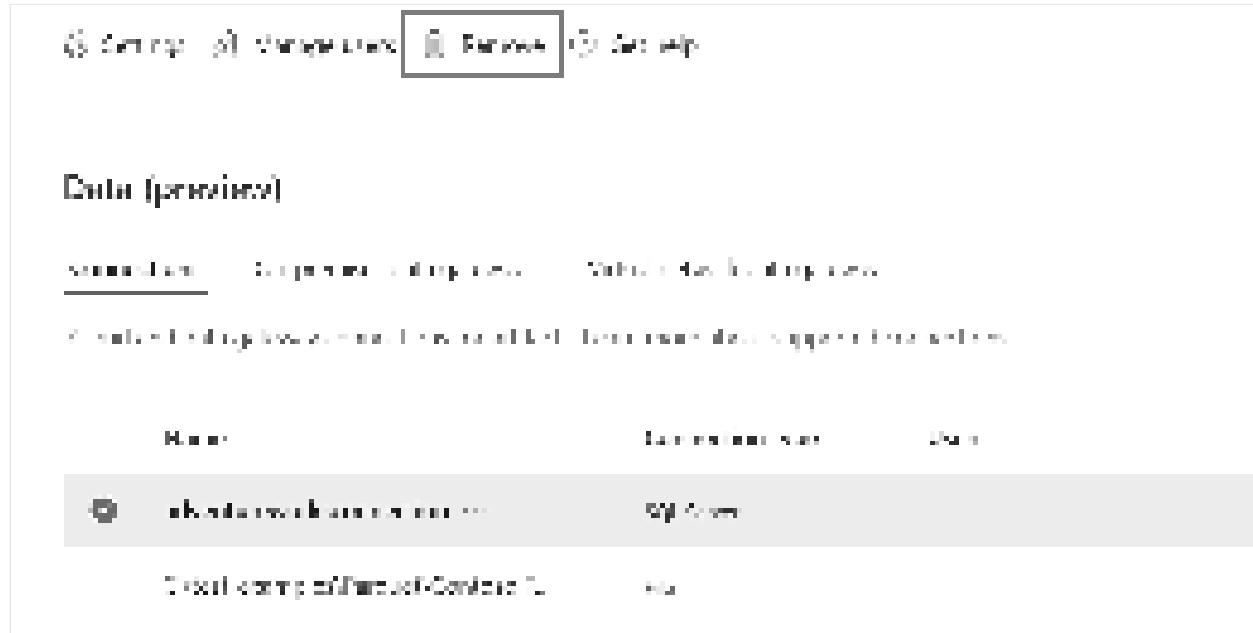


You can now use this data source to include data from Azure SQL in the supported Microsoft Fabric items.

Remove a data source

You can remove a data source if you no longer use it. If you remove a data source, any items that rely on that data source no longer work.

To remove a data source, select the data source from the **Data** screen in **Manage connections and gateways**, and then select **Remove** from the top ribbon.



Manage users

After you add a cloud data source, you give users and security groups access to the specific data source. The access list for the data source controls only who is allowed to use the data source in items that include data from the data source.

Add users to a data source

1. From the page header in the Power BI service, select the **Settings** icon, and then select **Manage connections and gateways**.
2. Select the data source where you want to add users. To easily find all cloud connections, use the top ribbon to filter or search.



3. Select **Manage users** from the top ribbon.
4. In the **Manage users** screen, enter the users and/or security groups from your organization who can access the selected data source.
5. Select the new user name, and select the role to assign, either **User**, **User with resharing**, or **Owner**.
6. Select **Share**. The added member's name is added to the list of people who can publish reports that use this data source.

A screenshot of the 'Manage users' screen. At the top, there's a message about sharing data sources. Below it, a section for 'Enter a name or email address' has a placeholder 'Search users'. A list of users is shown, with three items highlighted:

- User** allows the user to view and interact with reports
- User with resharing** allows the user to view and interact with reports, and to share them with others
- Owner** allows the user to manage the data source, including changing settings and adding individuals

The bottom of the screen shows a toolbar with icons for back, forward, and search.

Remember that you need to add users to each data source that you want to grant access to. Each data source has a separate list of users. Add users to each data source separately.

Remove users from a data source

On the **Manage Users** tab for the data source, you can remove users and security groups that use this data source.

Next steps

[Connectors overview](#)

Feedback

Was this page helpful?



[Provide product feedback](#) | [Ask the community](#)

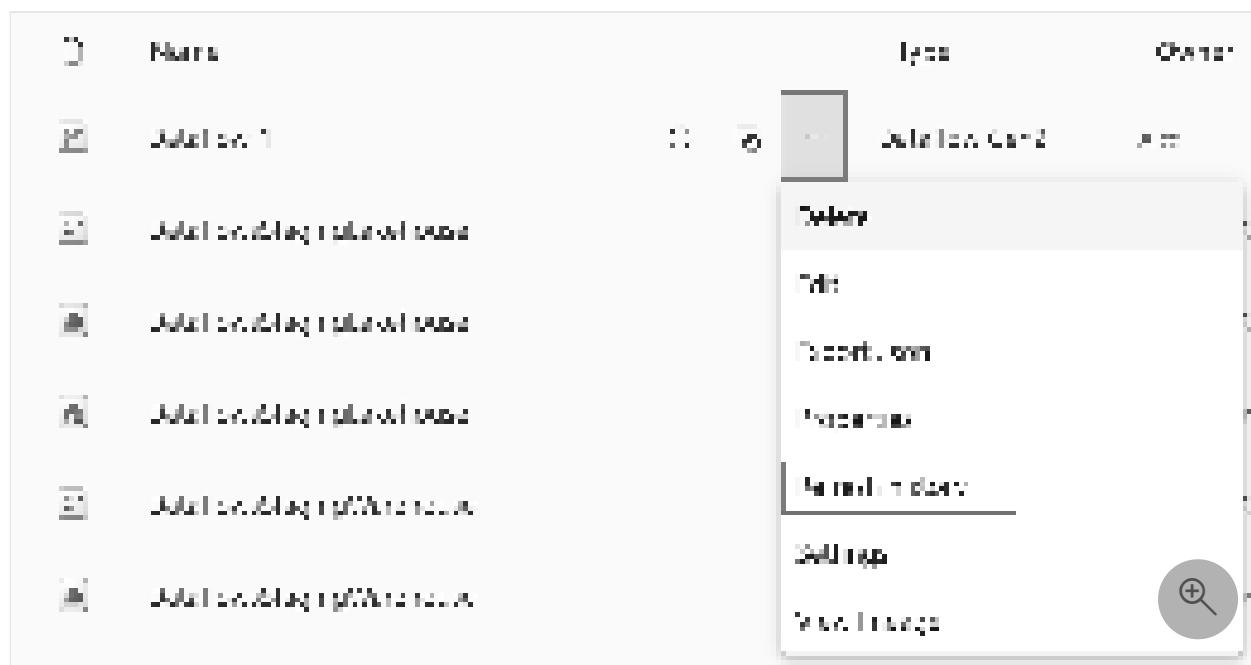
View refresh history and monitor your dataflows

Article • 11/15/2023

Monitoring your dataflow refreshes is key in ensuring that your dataflows are running as expected. Refresh history and monitoring hub allows you to evaluate in detail what happened during the refresh of your dataflow. This article provides you with an overview of the features that are available in the refresh history and monitoring hub. We also provide you with some guidance on how to use these features.

Refresh history

Refresh history is available using the drop-down menu in your workspace. You can access it by selecting the **Refresh History** button.



When you open your dataflow refresh history, you first notice a list of all your data refreshes. This first screen provides you with information about:

- Start time
- Status
- Duration
- Type

Refresh history

Dataflow 1

Refresh now Schedule refresh Download as CSV

Run time	Status	Duration	Type
11/17/2021, 12:17:59 AM	Succeeded	00:00:11	On-demand
11/16/2021, 2:50:00 AM	Succeeded	00:00:01	On-demand

Edit dataflow

You can take some actions right away from this page like start a new refresh, schedule a refresh, or edit the dataflow.

Download a CSV file of the refresh

Some times you might need to get a CSV file of your refresh. To get this file, take the following steps:

1. Open the refresh history of the dataflow.
2. Select the run you want to get a CSV file from.
3. Download the CSV.

Refresh history

Dataflow 1

Refresh now Schedule refresh Download as CSV

Run time	Status	Duration	Type
11/17/2021, 12:17:59 AM	Succeeded	00:00:11	On-demand
11/16/2021, 2:50:00 AM	Succeeded	00:00:01	On-demand

Edit dataflow

Reviewing your dataflow refresh from the UI

Once you've determined which dataflow you want to investigate, you can drill down into one of the refreshes by selecting the **Start time** field. This screen provides you with more information about the refresh that was performed. This includes general information about the refresh and a list of tables and activities.

The screenshot shows a detailed view of a dataflow refresh. At the top, there's a header with the refresh ID '1234567890' and a status bar indicating 'Last Refreshed: 1 hour ago'. Below this, the 'Refresh' section displays the refresh type as 'On demand', the start time as '2023-10-01T12:00:00Z', and the duration as '1 hour'. A progress bar shows the refresh is complete. The 'Tables' section lists several tables: 'Customer', 'Order', 'Product', 'Region', and 'Supplier'. Each table entry includes a 'Details' button. The 'Activities' section lists 'Get Data' and 'Transform' steps, each with a 'Details' button. A large circular button in the bottom right corner contains a magnifying glass icon, likely for search.

In short, this overview provides you:

- Status of the dataflow
- Type of refresh
- Start and End time
- Duration
- Request ID
- Session ID
- Dataflow ID

The **Tables** section reflects all the entities you've enabled load for in your dataflow. Meaning that those tables shown here are being loaded into the staging area of your dataflow. These tables are the entities you can access using the Power Query Dataflow connector in Power BI, Excel, or dataflows. You can select any of the listed table names to view the details of this specific table. Once you select the name, you arrive at the following **Details** screen:

The screenshot shows the 'Activities' section of the Refresh Details screen. It lists several activities with their status, start time, end time, duration, and connector name. The activities listed are:

Activity	Status	Start Time	End Time	Duration	Connector
Load Data	Success	2023-10-01T10:00:00Z	2023-10-01T10:05:00Z	5 minutes	File Connector
Process Data	Success	2023-10-01T10:05:00Z	2023-10-01T10:10:00Z	5 minutes	File Connector
Send Data	Success	2023-10-01T10:10:00Z	2023-10-01T10:15:00Z	5 minutes	File Connector
Finalize Refresh	Success	2023-10-01T10:15:00Z	2023-10-01T10:20:00Z	5 minutes	File Connector

A large green button labeled 'View Details' is located at the bottom right of the activities table.

The **Activities** section reflects all the actions that have taken place during the refresh, for example loading data to your output destination. This table also allows you to dive deeper into the details of the specific activity. By selecting the name of the activity, you arrive at the following **Details** screen:

The screenshot shows the 'Load Data' activity details screen. It provides a detailed view of the activity, including its status, start and end times, duration, and connector information. The activity details are as follows:

Activity	Status	Start Time	End Time	Duration	Connector
Load Data	Success	2023-10-01T10:00:00Z	2023-10-01T10:05:00Z	5 minutes	File Connector

The screen also includes a 'Logs' section with a table showing log entries, and a 'Metrics' section with a table showing performance metrics. A large green button labeled 'View Details' is located at the bottom right of the activity details table.

This screen gives you more clarity in what happened during the activity. For example, for output destinations the activity screen provides you with the:

- Status of the Activity
- Start and End time
- Duration
- Activity statistics:
 - Output destinations:
 - Endpoints contacted
 - Volume processed by the connector

To investigate what happened, you can drill down into an activity or table. The following screen provides you with general information about the refresh and errors. If you're drilling into an activity, you are presented with how much data got processed and sent to your output destination.

Monitoring hub

The monitoring hub is available using the side menu. You can access it by selecting the **Monitoring hub** button.



The monitoring hub provides you with a dashboard that gives you an overview of the status of your dataflows.

A screenshot of a web-based monitoring dashboard titled "Monitoring hub". The dashboard features a header with a search bar and navigation links. Below the header is a table with two rows of data. The columns represent different metrics: Name, Status, Refreshed At, Refresh Duration, Submitter, and Workspace. The first row shows "Dataflow 1" with status "Success", refreshed at "2023-01-14 01", duration "00:00:00", submitter "User123", and workspace "My Workspace". The second row shows "Dataflow 2" with similar values. A large circular button with a plus sign is located in the bottom right corner of the dashboard area.

This dashboard provides you with the following information:

- Status of your dataflows
- Start time of the refresh
- Refresh duration
- Submitter of the refresh
- Workspace name
- Fabric capacity used for the refresh of your dataflow
- Average refresh duration
- Number of refreshes per day

- Type of refresh

Next steps

- Compare differences between Dataflow Gen1 and Gen2 in Data Factory
 - Dataflows save as draft
-

Feedback

Was this page helpful?



Provide product feedback [↗](#) | Ask the community [↗](#)

Save a draft of your dataflow

Article • 11/15/2023

With Dataflow Gen2, we changed how saving a dataflow works. We wanted to improve the experience and resiliency of Dataflow Gen1 by:

1. Automatically saving to the cloud any change made to a dataflow. This saved change is called the draft version of the dataflow.
2. Deferring long running validation required to guarantee a dataflow can refresh, to the background. The version of the dataflow that passed validation and is ready to refresh is called the published version.

This powerful feature allows you to make changes to your dataflow without immediately publishing them to your workspace. Instead, all your changes are automatically saved as a draft, which you can review a later time, and then publish when you're ready. With this feature, you don't have to worry about losing your work if you want to resume it at a later time, if your dataflow fails validation, or if your editing session abruptly ends. In this article, you learn how to use the new Dataflow Gen2 auto-save and publish feature and how it can benefit your dataflow development and management.

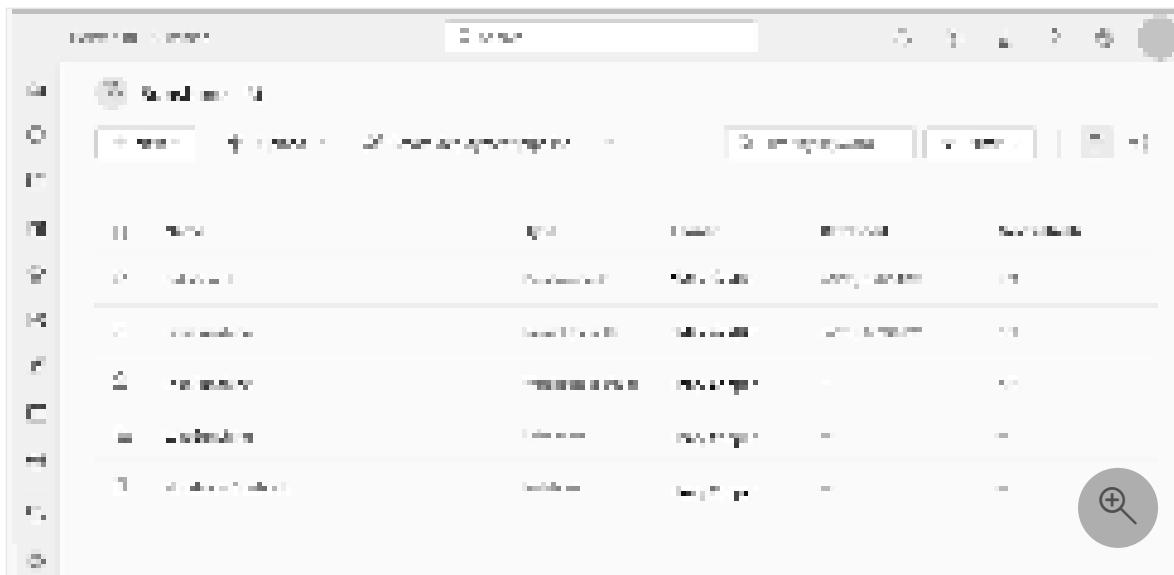
How to save a draft version of your dataflow

Saving a draft of your dataflow is as easy as just closing your browser, closing the dataflows editors, or navigating to another workspace. Anytime you add a new Power Query step to your dataflow, a draft of your changes is saved to the workspace.

How to publish your draft dataflow

To publish the changes you made in your draft, you take the following steps:

1. Navigate to your workspace.



2. Open the draft dataflow that you recently saved changes to.
3. Review all the changes you made last time.
4. Publish the dataflow with the button on the bottom of the page



Your dataflow starts the publishing process in the background, which is indicated by a spinner next to the dataflow name. Once the spinner completes, the dataflow can be refreshed. If there are any publishing related errors, an indication is visible next to the dataflow name. Selecting the indication reveals the publishing errors and allows you to edit the dataflow from your last saved version.

Next steps

- Compare differences between Dataflow Gen1 and Gen2 in Data Factory

- Dataflows refresh history and monitoring
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Data obfuscation in Data Factory with Delphix Compliance Services

Article • 11/15/2023

The following how-to outlines the use of Delphix Compliance Services (DCS) in Data Factory in Microsoft Fabric dataflows to mask sensitive data prior to delivery.

DCS is a highly scalable masking API service that automatically masks personally identifiable information (PII), supplanting manual processes for delivering compliant data. Its out-of-the-box and configurable algorithms replace sensitive data values with fictitious yet realistic ones, so teams mitigate risk while ensuring end-users can easily consume the right data.

Masked data maintains multicloud referential integrity, is production-like in quality, and remains fully functional for accurate analysis or testing. Note that a DCS account needs to be created prior to use, and you can sign up for a free trial ↗.

What is the challenge?

The cloud is filled with personally identifiable information (PII), fueling privacy and security risk. PII from production apps needs to flow to downstream systems for analytics, exposing organizations to risks or creating data silos. Power Query and DCS automate data compliance and security to unblock data movement.

Breaking down data silos is difficult:

- Data must be manipulated to fit a common format. ETL pipelines must be adapted to each system of record and must scale to support the massive data sets of modern enterprises.
- Compliance with regulations regarding sensitive information must be maintained when data is moved from systems of record. Customer content and other sensitive elements must be obscured without impacting the business value of the data set.

How do DCS and Data Factory solve automating compliant data?

The movement of secure data is a challenge for all organizations. Delphix makes achieving consistent data compliance easy, while Data Factory enables connecting and

moving data seamlessly. Together Delphix and Data Factory make the delivery of on-demand, compliant data easy.

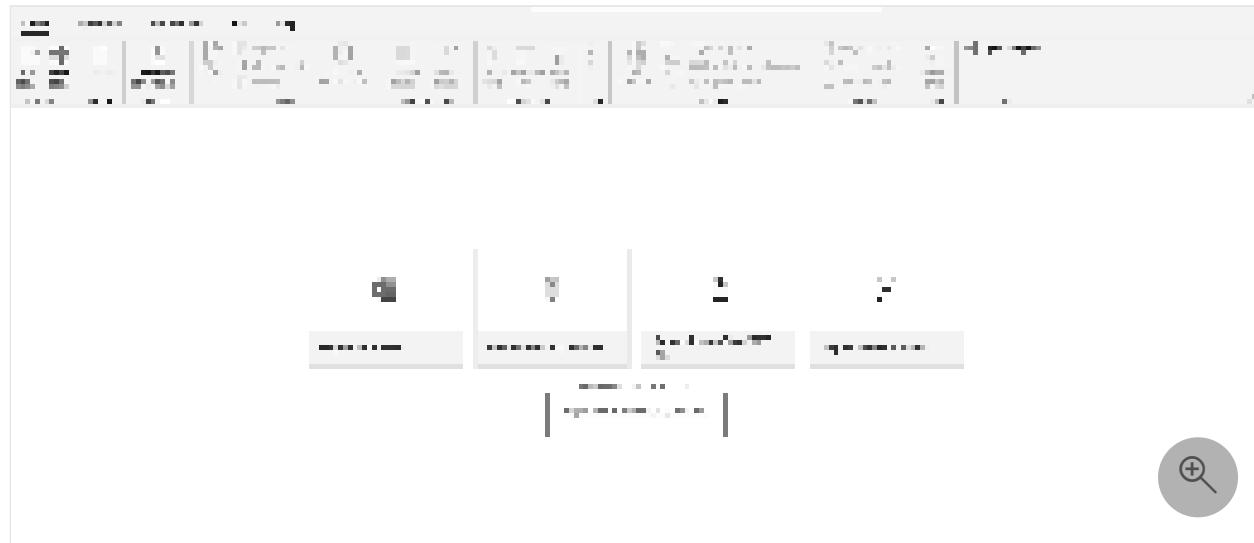
Using Data Factory data flows, you can create a workflow that automates the following steps:

- Read data from the desired source.
- Map sensitive fields to appropriate masking algorithms (and manage as a central configuration table).
- Call DCS masking APIs to replace sensitive data elements with similar but fictitious values.
- Load the compliant data to a desired target.

How to get started

Go to the [Delphix free preview page](#) to request a free trial of DCS. The Delphix team then contacts you for access and provides the template that is used in the example setup scenario described in this article.

In Power Query, upload the provided template by selecting **Import from a Power Query Template**, and then select the Power Query template file to import. This selection loads a set of queries.

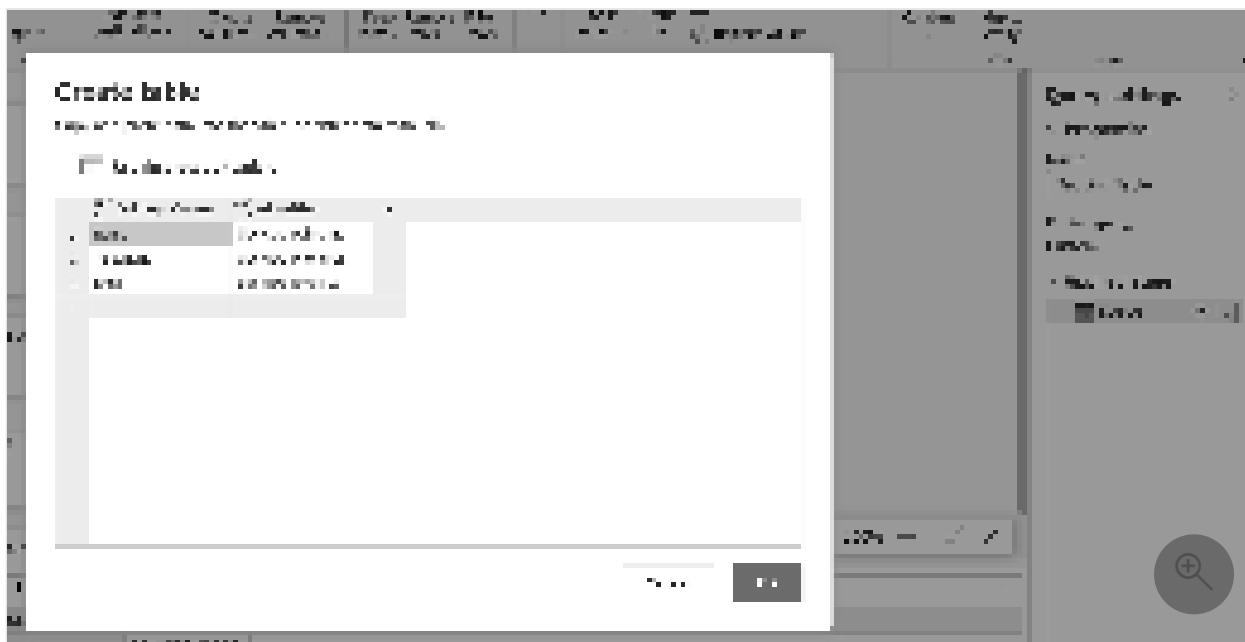


Import the data source that contains sensitive data that you would like masked.

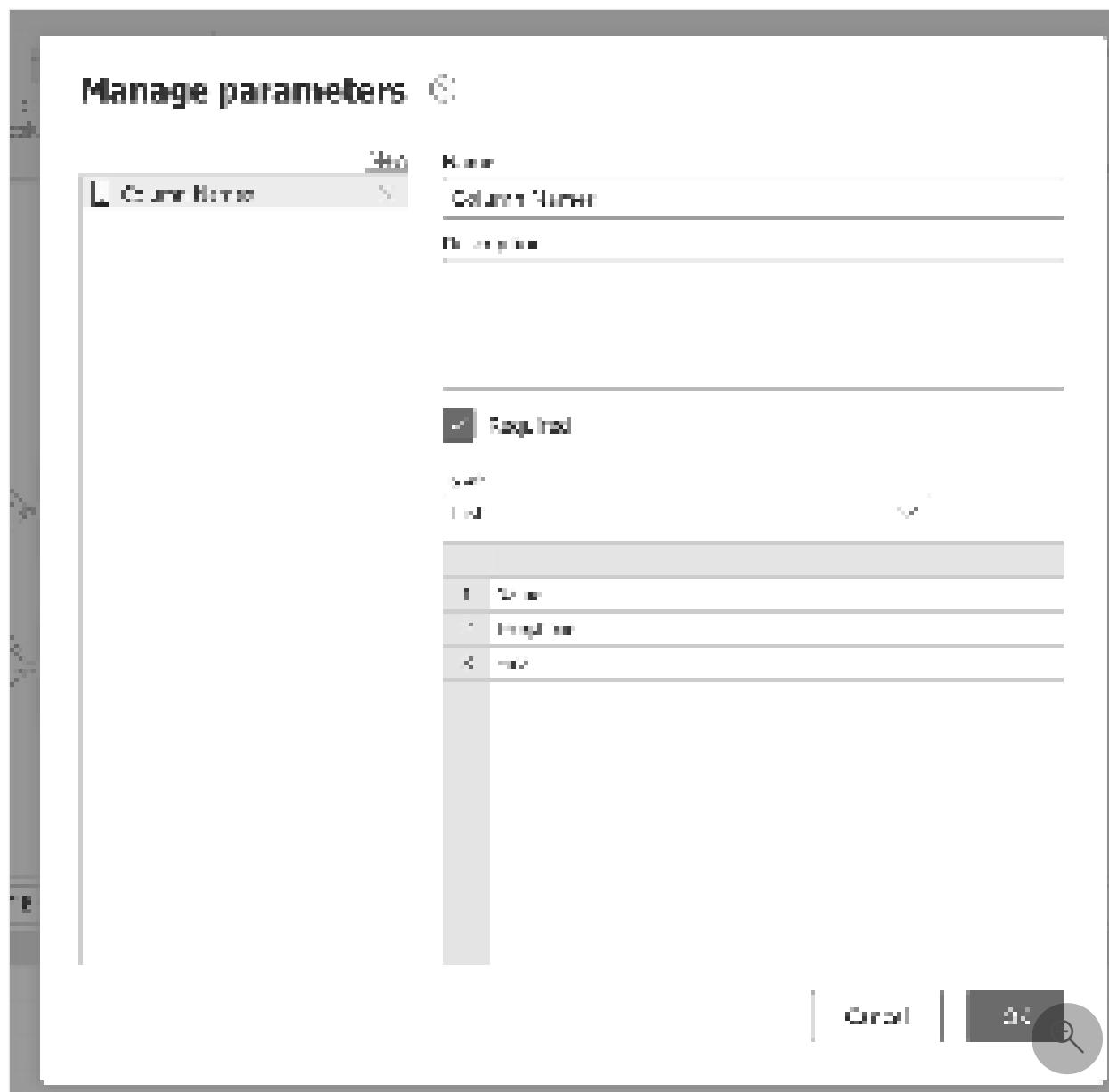
Original Column	Algorithm	Masked Column
Original Column 1	Algorithm 1	Masked Column 1
Original Column 2	Algorithm 2	Masked Column 2
Original Column 3	Algorithm 3	Masked Column 3
Original Column 4	Algorithm 4	Masked Column 4
Original Column 5	Algorithm 5	Masked Column 5
Original Column 6	Algorithm 6	Masked Column 6
Original Column 7	Algorithm 7	Masked Column 7
Original Column 8	Algorithm 8	Masked Column 8
Original Column 9	Algorithm 9	Masked Column 9
Original Column 10	Algorithm 10	Masked Column 10

The mapping table is where you configure what fields to mask, and which Delphix masking algorithms to use. In the **Query Settings** pane, right-click on the gear icon. Enter the column names where sensitive data resides in **Original Column**. Enter the corresponding Delphix algorithm in **Algorithm**. Details on available algorithms can be found in the Delphix documentation [↗](#).

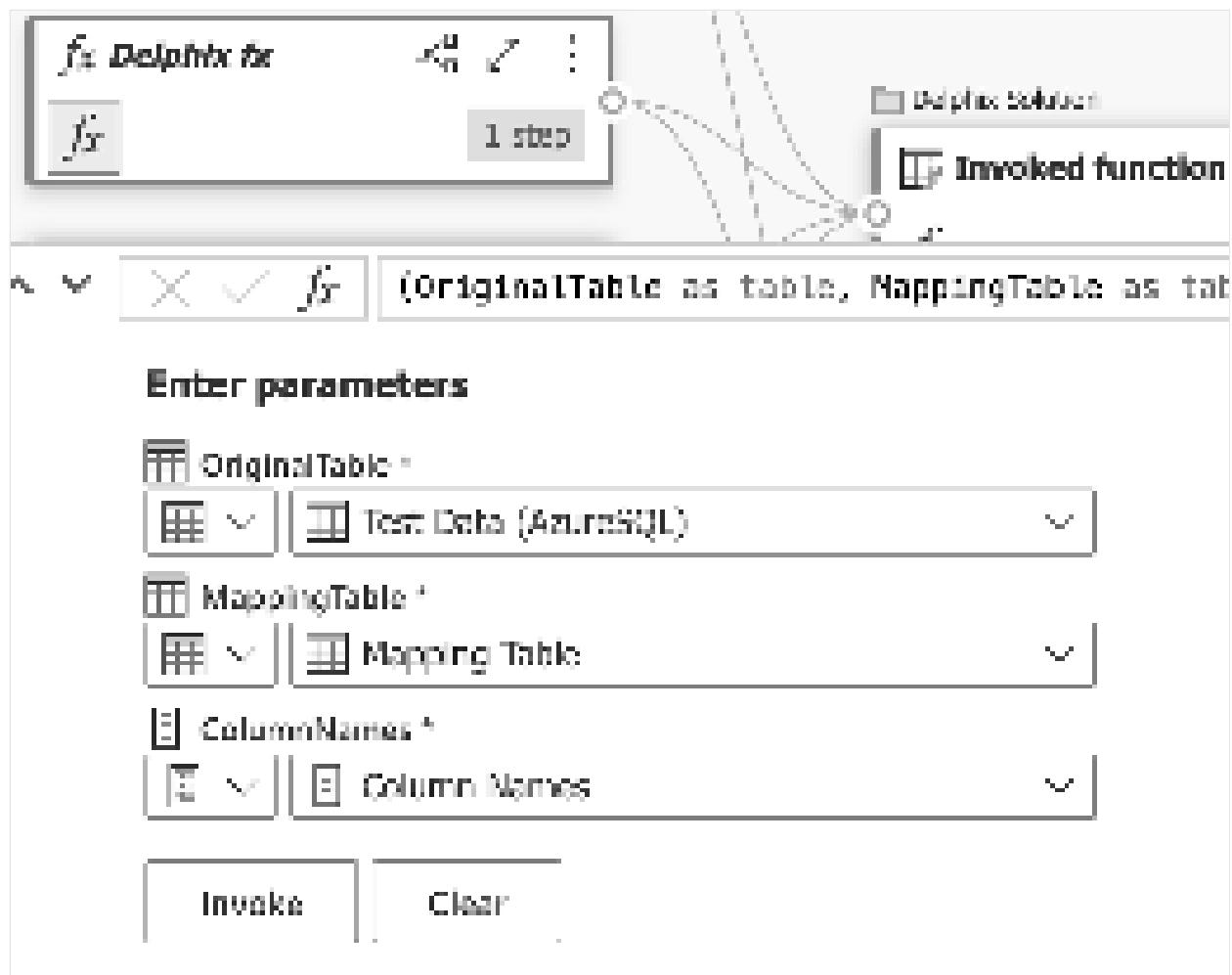
Delphix's out-of-the-box masking algorithms can be customized, or new algorithms can be defined if needed. All Delphix masking algorithms replace sensitive data with fictitious, yet realistic values, and do so consistently across data sets.



This mapping table can be a global configuration across any tables you would like to mask. Should you want to leave any columns unmasked for a given table, the **Column Names** parameter serves as a filter. Copy and paste the list of **Original Columns** (from the mapping table) into the **Column Names** parameter, and delete any column names that you would like to leave unmasked.



You're now ready to mask your data. Select **Delphix fx** and enter parameters as displayed in the following image (with the `OriginalTable` field as the data source that contains sensitive data).



Once this change is complete, select **Invoke** to run the data flow. This selection automatically calls the DCS masking API service to mask data prior to delivery to the destination of your choice.

Your data is now ready to be used safely by end users. The data is masked consistently, ensuring that references remain intact across data sets. As an example, George Smith becomes Edward Robinson regardless of data source or destination, ensuring it's still valuable for integrated analytics scenarios.

Next steps

- [Delphix free preview page ↗](#)
- [Delphix documentation ↗](#)

Feedback

Was this page helpful?

Yes

No

Provide product feedback ↗ | Ask the community ↗

Use a dataflow in a pipeline

Article • 11/15/2023

In this tutorial, you build a data pipeline to move OData from a Northwind source to a lakehouse destination and send an email notification when the pipeline is completed.

Prerequisites

To get started, you must complete the following prerequisites:

- Make sure you have a Microsoft Fabric enabled Workspace that isn't the default My Workspace.

Create a Lakehouse

To start, you first need to create a lakehouse. A lakehouse is a data lake that is optimized for analytics. In this tutorial, you create a lakehouse that's used as a destination for the dataflow.

1. Switch to the **Data Engineering** experience.

Microsoft Fabric →

 Power BI

 Data Factory

 Data Activator

Synapse

 Data Engineering

 Data Science

 Data Warehouse

 Real-Time Analytics



Data
Factory

2. Go to your Fabric enabled workspace.



Home



Create



Storage

OneLake
data hubMonitoring
hub

Workspaces

Workspaces

Search

My workspace

All

Sunshine ▾

Sales team ▾

Marketing workspace ▾

3. Select **Lakehouse** in the create menu.

 New

 Upload



Data pipeline (Preview)



Dataflow Gen2 (Preview)



Eventstream (Preview)



Experiment (Preview)



KQL Database (Preview)



KQL Queryset (Preview)



Lakehouse (Preview)



Model (Preview)



... (truncated)

4. Enter a **Name** for the lakehouse.

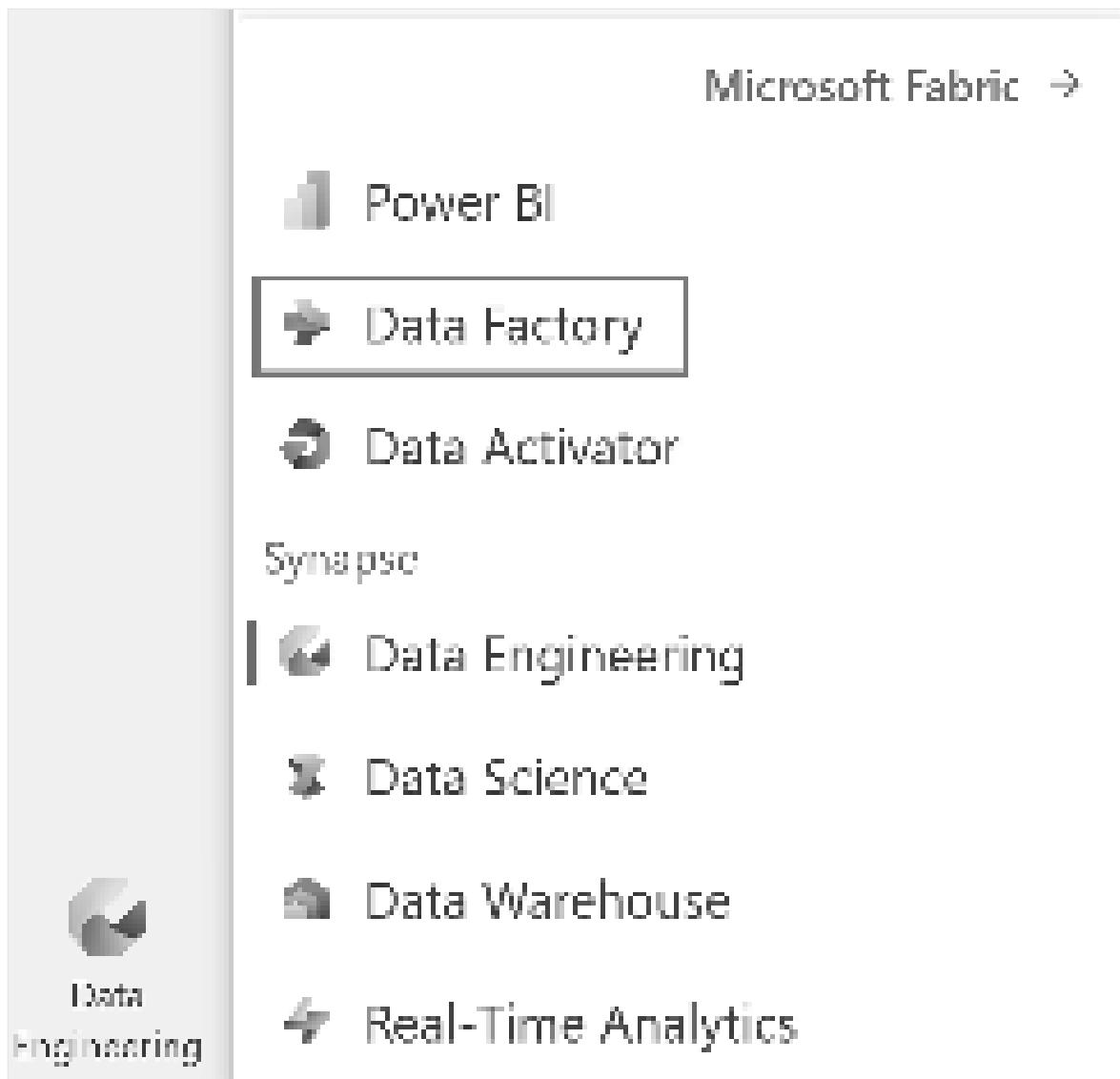
5. Select **Create**.

Now you've created a lakehouse and you can now set up the dataflow.

Create a dataflow

A dataflow is a reusable data transformation that can be used in a pipeline. In this tutorial, you create a dataflow that gets data from an OData source and writes the data to a lakehouse destination.

1. Switch to the Data Factory experience.



2. Go to your Fabric enabled workspace.

The screenshot shows the Data Factory Home interface. On the left, there is a vertical sidebar with icons and labels: Home (house icon), Create (plus sign icon), Store (folder icon), OData hub (globe icon), Monitoring hub (gauge icon), and Workspaces (TV icon). The 'Workspaces' item is currently selected. The main area is titled 'Workspaces' and contains a search bar labeled 'Search'. Below the search bar, there is a section titled 'My workspace' with an icon of a person. Underneath it, there is a list of workspaces: 'All', 'Sunshine' (selected, highlighted with a gray box), 'Sales team', and 'Marketing workspace'.

3. Select Dataflow Gen2 in the create menu.

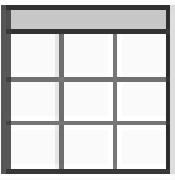
The screenshot shows the 'Create' menu with several options: 'New' (selected and highlighted with a gray box), 'Upload', 'Data pipeline (Preview)', 'Dataflow Gen2 (Preview)' (selected and highlighted with a gray box), and 'Eventstream (Preview)'. There are also other options like 'Container' and 'Dataset' at the bottom.

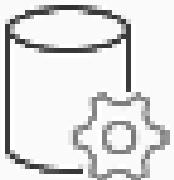
4. Ingest the data from the OData source.

a. Select Get data, and then select More.

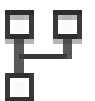
Home Transform

 Get data

 Enter data

 Manage connections

 Excel workbook

 Dataflows

 SQL Server database

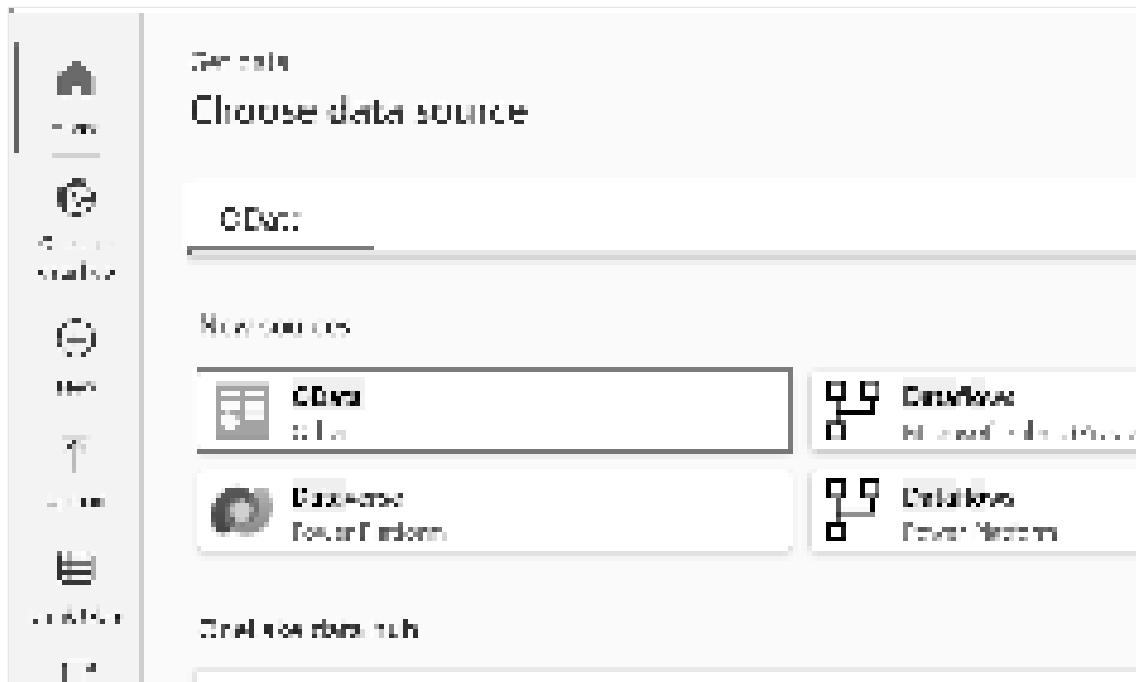
 Text/CSV

 Web page

 Blank query

 More...

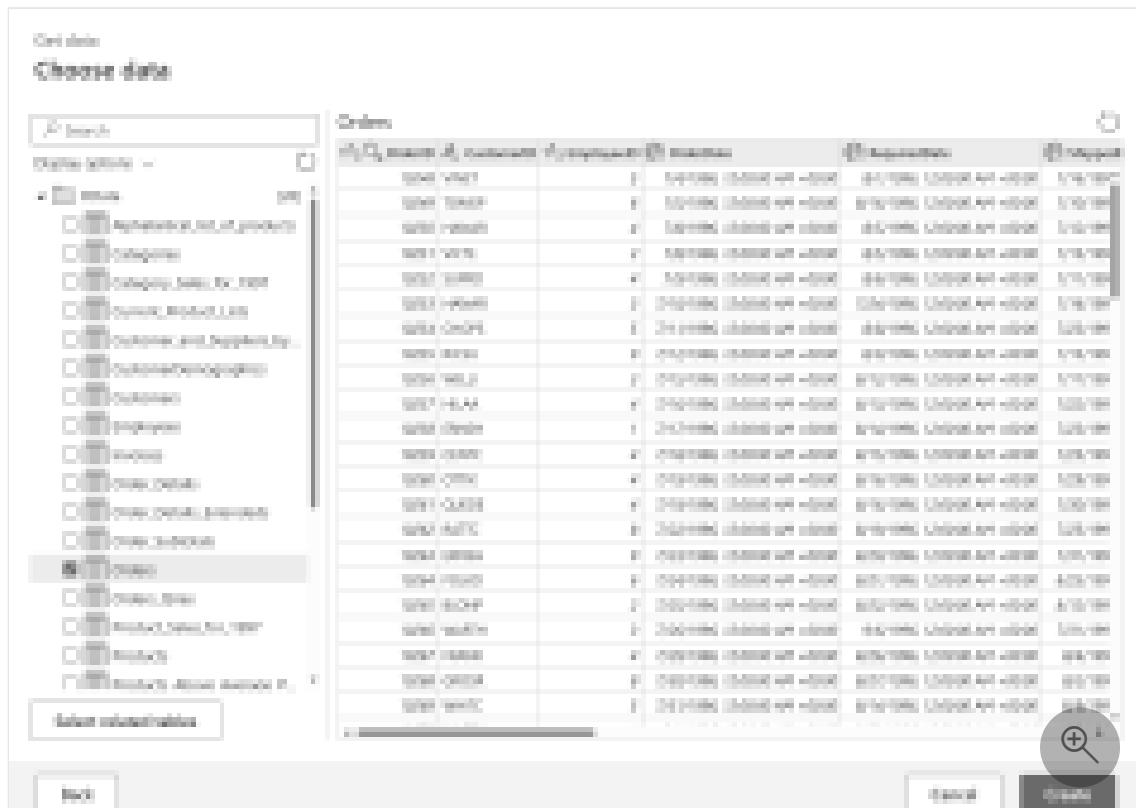
- b. From **Choose data source**, search for **OData**, and then select the OData connector.



c. Enter the **URL** of the OData source. For this tutorial, use the OData sample service ↗.

d. Select **Next**.

e. Select the **Entity** that you want to ingest. In this tutorial, use the **Orders** entity.



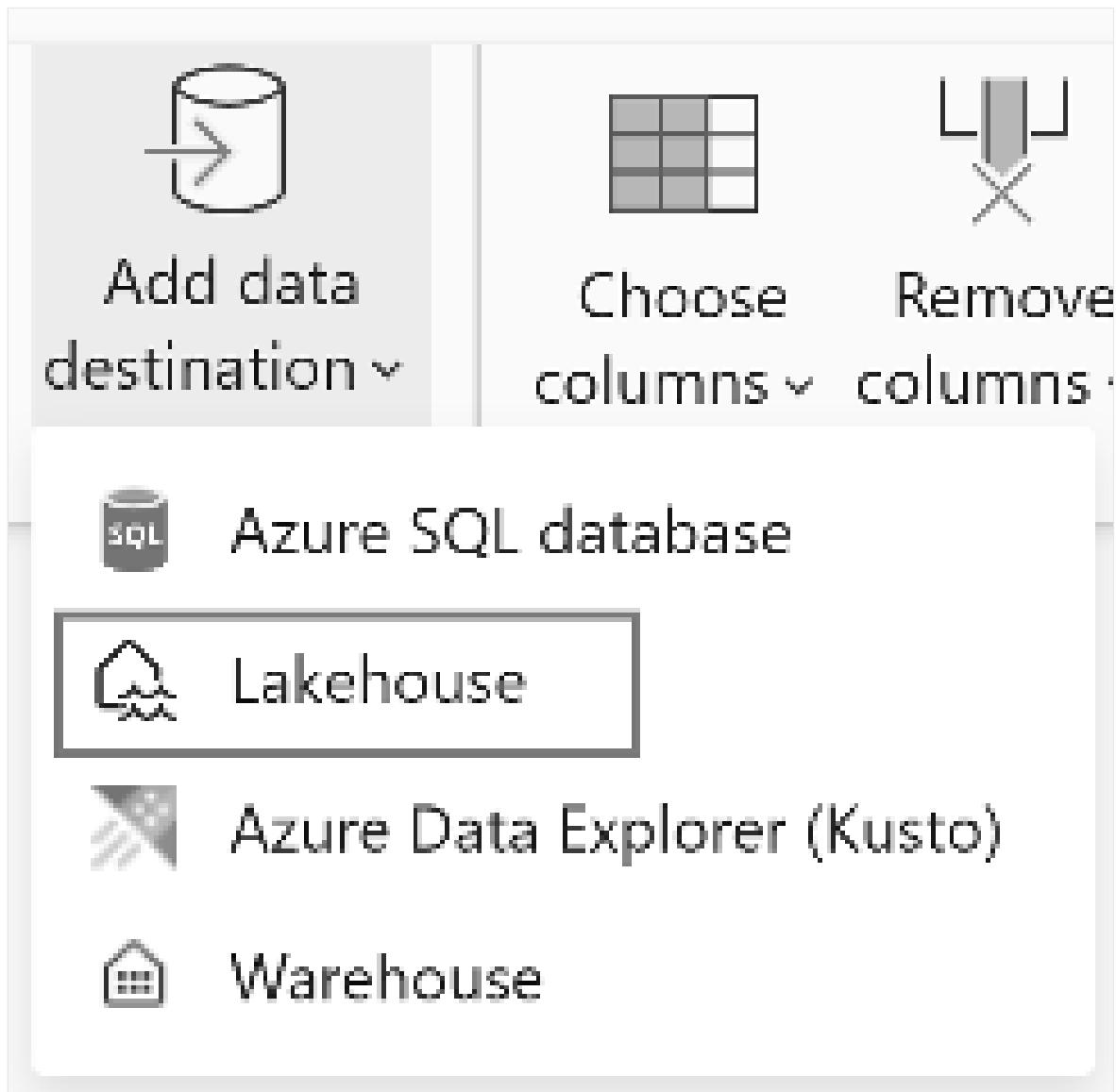
f. Select **Create**.

Now that you've ingested the data from the OData source, you can set up the lakehouse destination.

To ingest the data to the lakehouse destination:

1. Select **Add data destination**.

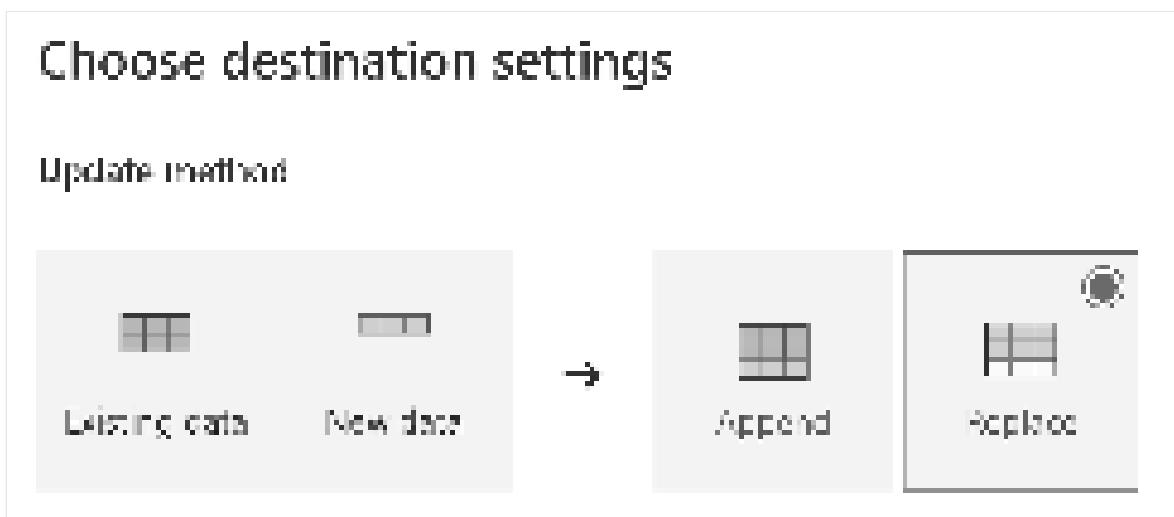
2. Select **Lakehouse**.



3. Configure the connection you want to use to connect to the lakehouse. The default settings are fine.
4. Select **Next**.
5. Navigate to the workspace where you created the lakehouse.
6. Select the lakehouse that you created in the previous steps.



7. Confirm the table name.
8. Select **Next**.
9. Confirm the update method and select **Save settings**.



10. Publish the dataflow.

(i) Important

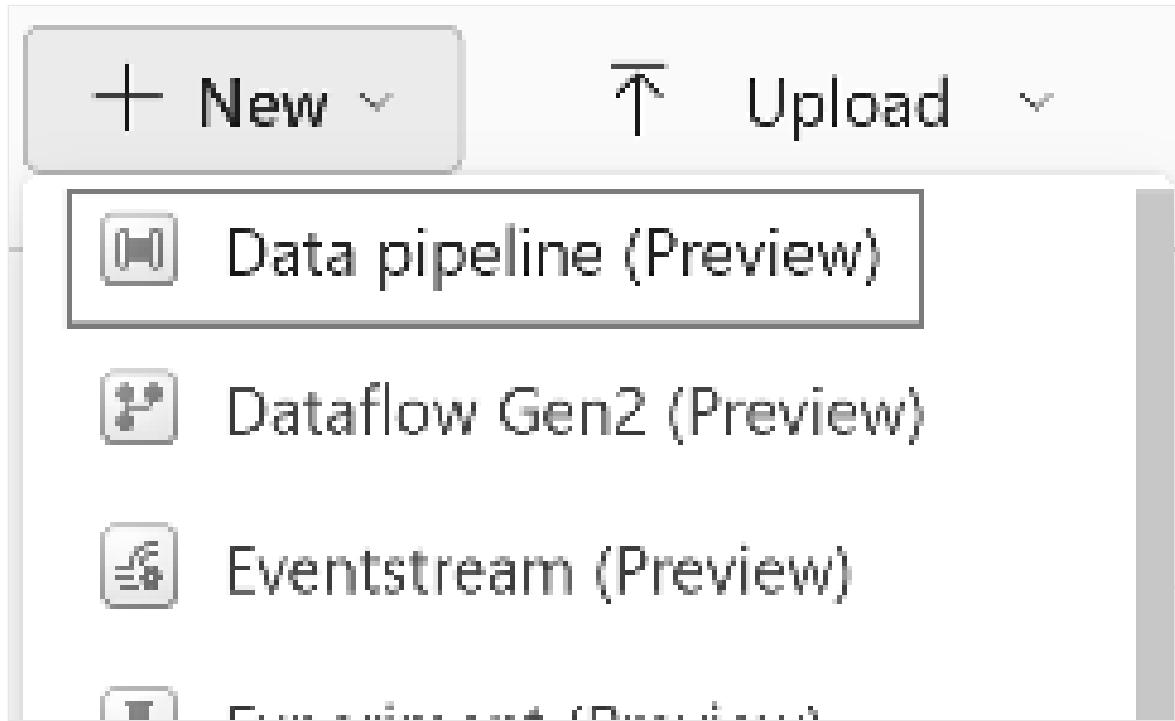
When the first Dataflow Gen2 is created in a workspace, Lakehouse and Warehouse items are provisioned along with their related SQL analytics endpoint and semantic models. These items are shared by all dataflows in the workspace and are required for Dataflow Gen2 to operate, shouldn't be deleted, and aren't intended to be used directly by users. The items are an implementation detail of Dataflow Gen2. The items aren't visible in the workspace, but might be accessible in other experiences such as the Notebook, SQL-endpoint, Lakehouse, and Warehouse experiences. You can recognize the items by their prefix in the name. The prefix of the items is 'DataflowsStaging'.

Now that you've ingested the data to the lakehouse destination, you can set up your data pipeline.

Create a data pipeline

A data pipeline is a workflow that can be used to automate data processing. In this tutorial, you create a data pipeline that runs the Dataflow Gen2 that you created in the previous procedure.

1. Navigate back to the workspace overview page and select **Data Pipelines** in the create menu.

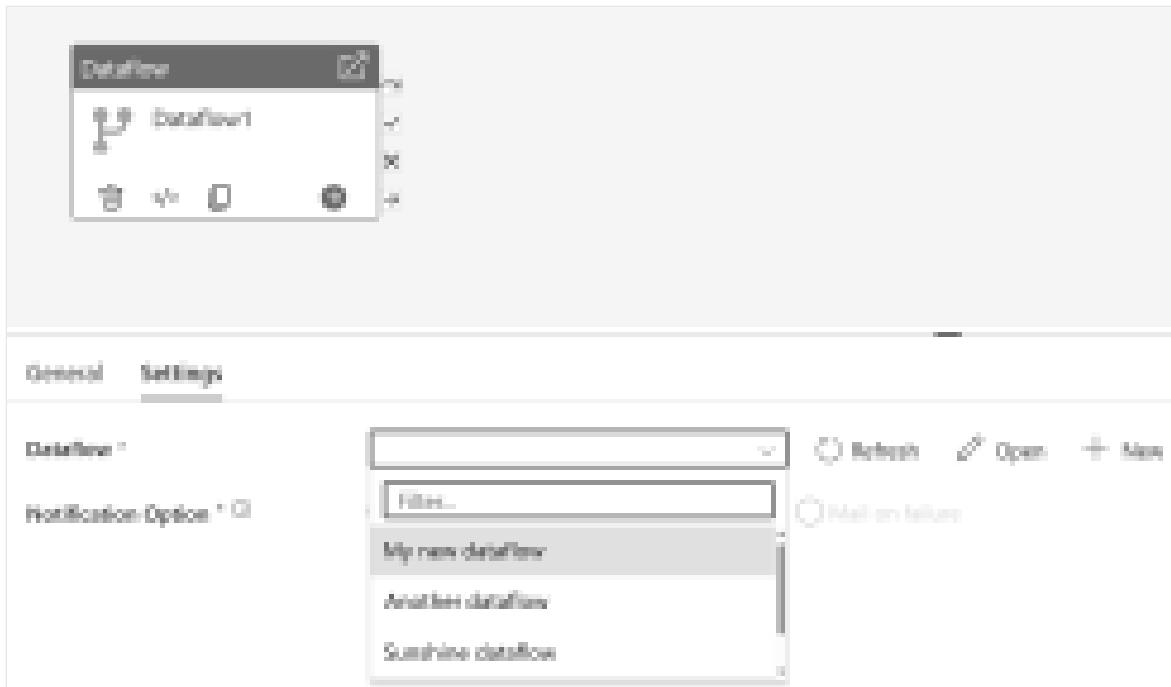


2. Provide a **Name** for the data pipeline.

3. Select the **Dataflow** activity.



4. Select the **Dataflow** that you created in the previous procedure in the **Dataflow** dropdown list under **Settings**.

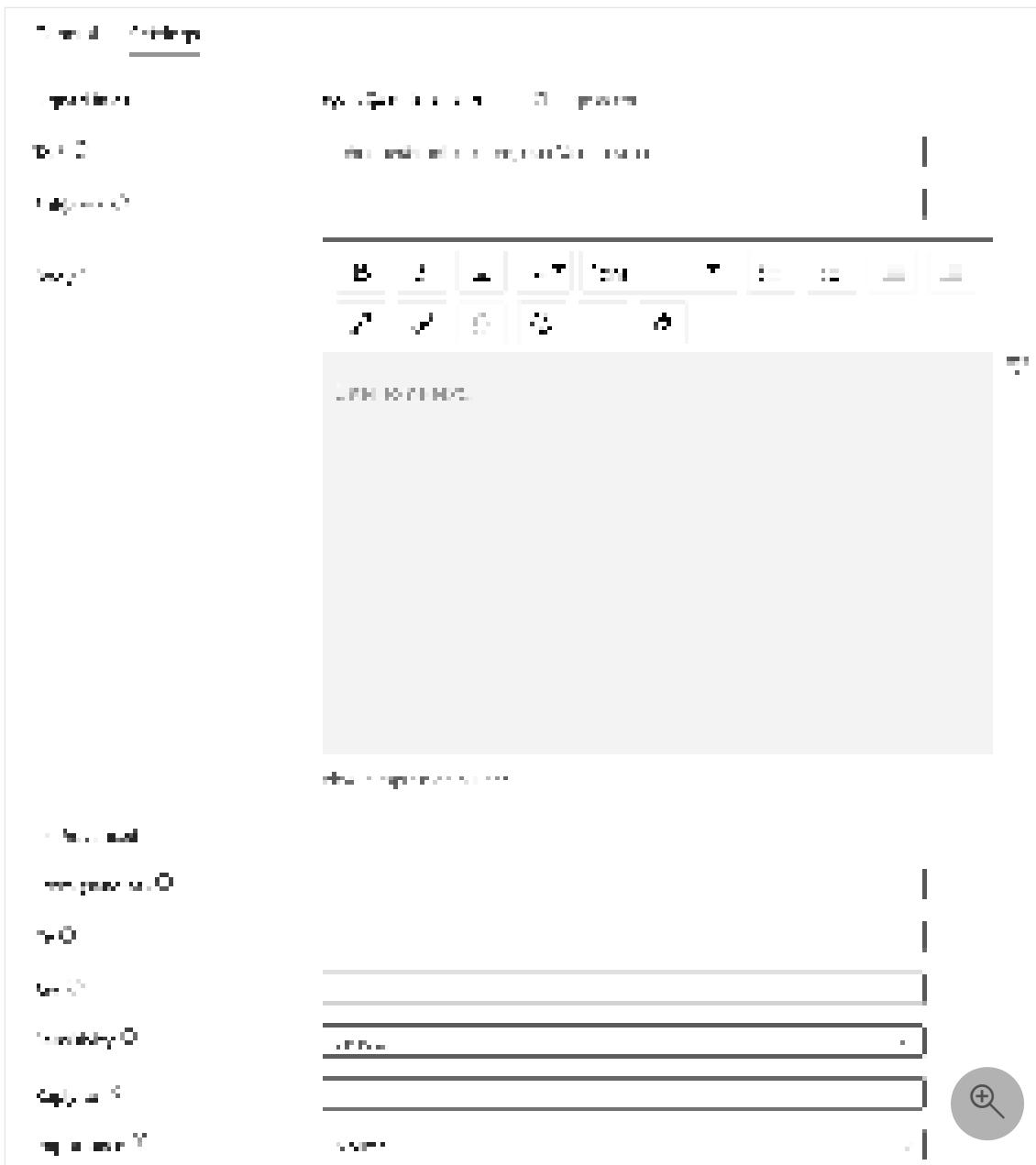


5. Add an **Email notification** activity.



6. Configure the **Email notification** activity.

- a. Authenticate with your Office 365 account.
- b. Select the **Email address** that you want to send the notification to.
- c. Enter a **Subject** for the email.
- d. Enter a **Body** for the email.



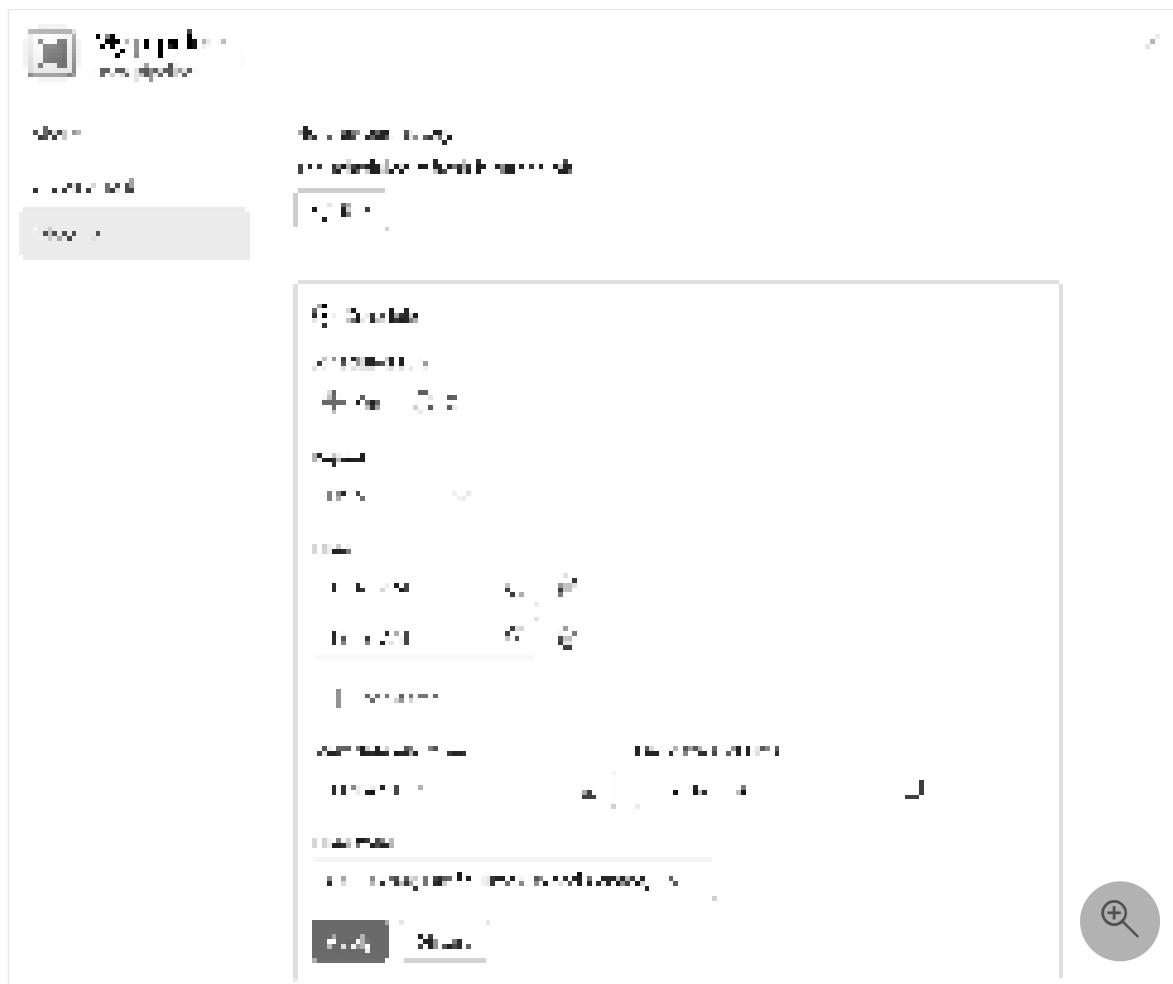
Run and schedule the data pipeline

In this section, you run and schedule the data pipeline. This schedule allows you to run the data pipeline on a schedule.

1. Go to your workspace.
2. Open the dropdown menu of the data pipeline that you created in the previous procedure, and then select **Schedule**.



3. In Scheduled run, select On.



4. Provide the schedule you want to use to run the data pipeline.

- Repeat, for example, every Day or every Minute.
- When selected Daily, you can also select the Time.
- Start On a specific Date.
- End On a specific Date.
- Select the Timezone.

5. Select **Apply** to apply the changes.

You've now created a data pipeline that runs on a schedule, refreshes the data in the lakehouse, and sends you an email notification. You can check the status of the data pipeline by going to the **Monitor Hub**. You can also check the status of the data pipeline by going to **Data Pipeline** and selecting the **Run history** tab in the dropdown menu.

Next steps

This sample shows you how to use a dataflow in a pipeline with Data Factory in Microsoft Fabric. You learned how to:

- ✓ Create a dataflow.
- ✓ Create a pipeline invoking your dataflow.
- ✓ Run and schedule your data pipeline.

Next, advance to learn more about monitoring your pipeline runs.

[How to monitor pipeline runs in Microsoft Fabric](#)

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#) | [Ask the community](#)

Pattern to incrementally amass data with Dataflow Gen2

Article • 11/15/2023

ⓘ Important

This is a pattern to incrementally amass data with Dataflow Gen2. This isn't the same as incremental refresh. Incremental refresh is a feature that's currently in development. This feature is one of the top voted ideas on our ideas website. You can vote for this feature on the Fabric Ideas site [↗](#).

This tutorial takes 15 minutes, and describes how to incrementally amass data into a lakehouse using Dataflow Gen2.

Incrementally amassing data in a data destination requires a technique to load only new or updated data into your data destination. This technique can be done by using a query to filter the data based on the data destination. This tutorial shows how to create a dataflow to load data from an OData source into a lakehouse and how to add a query to the dataflow to filter the data based on the data destination.

The high-level steps in this tutorial are as follows:

- Create a dataflow to load data from an OData source into a lakehouse.
- Add a query to the dataflow to filter the data based on the data destination.
- (Optional) reload data using notebooks and pipelines.

Prerequisites

You must have a Microsoft Fabric enabled workspace. If you don't already have one, refer to [Create a workspace](#). Also, the tutorial assumes you are using the diagram view in Dataflow Gen2. To check if you are using the diagram view, in the top ribbon go to [View](#) and make sure [Diagram view](#) is selected.

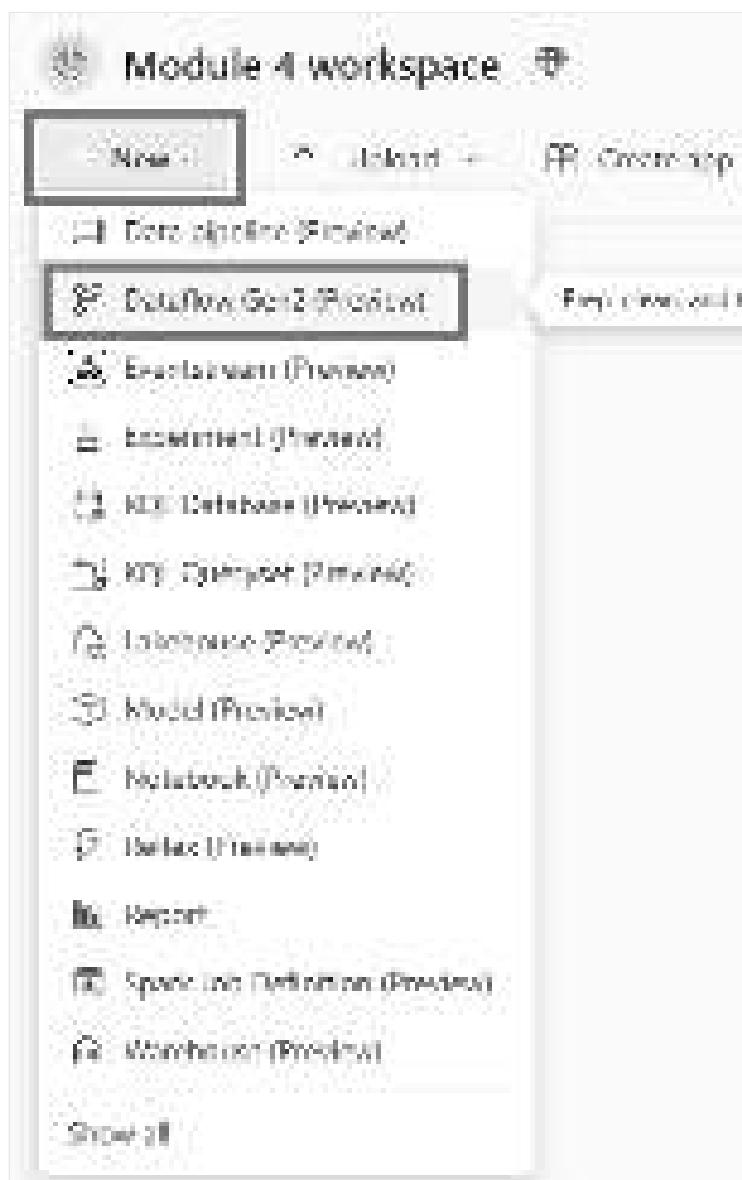
Create a dataflow to load data from an OData source into a lakehouse

In this section, you create a dataflow to load data from an OData source into a lakehouse.

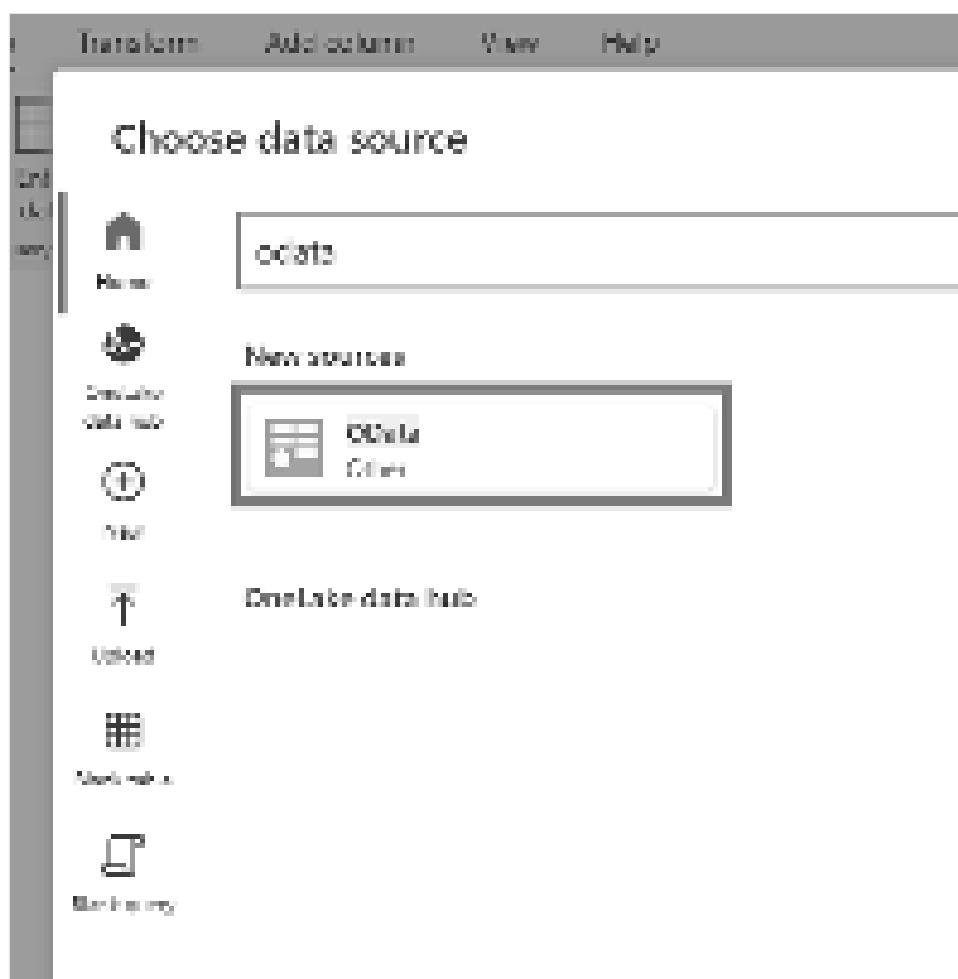
1. Create a new lakehouse in your workspace.

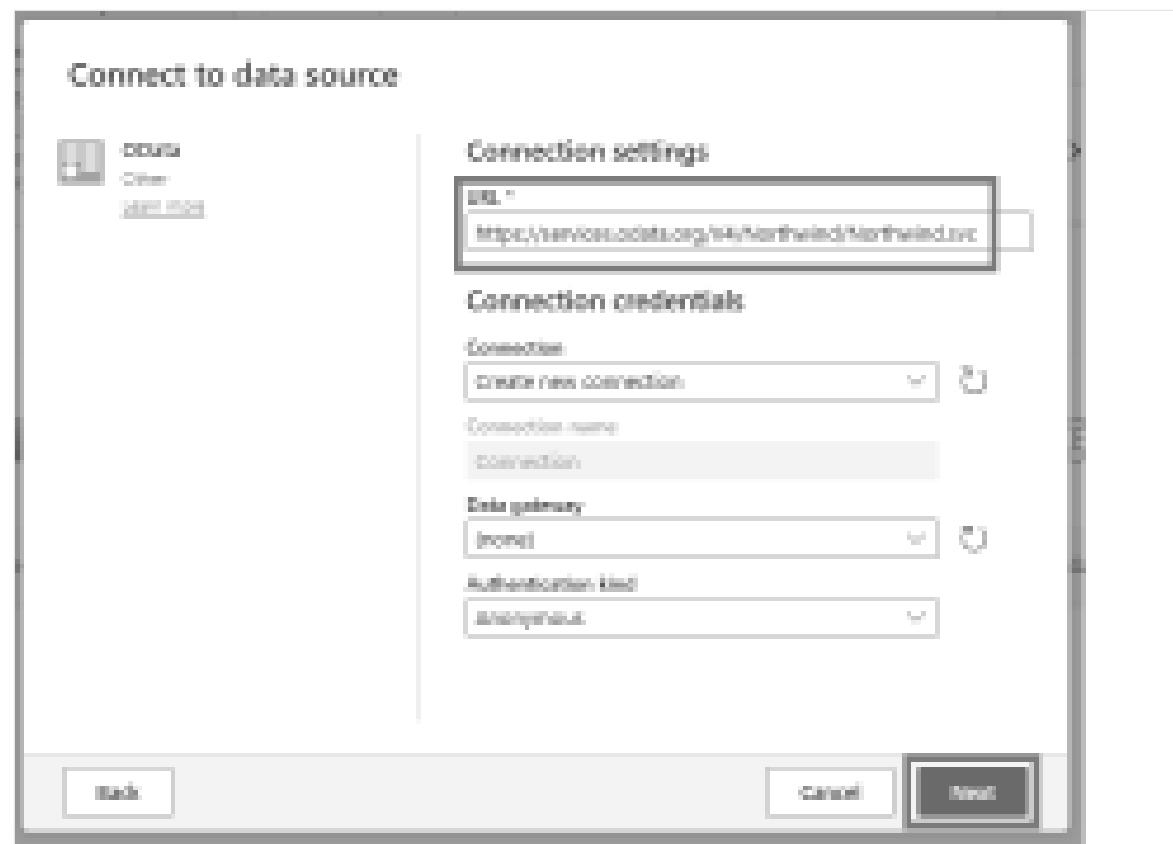


2. Create a new Dataflow Gen2 in your workspace.

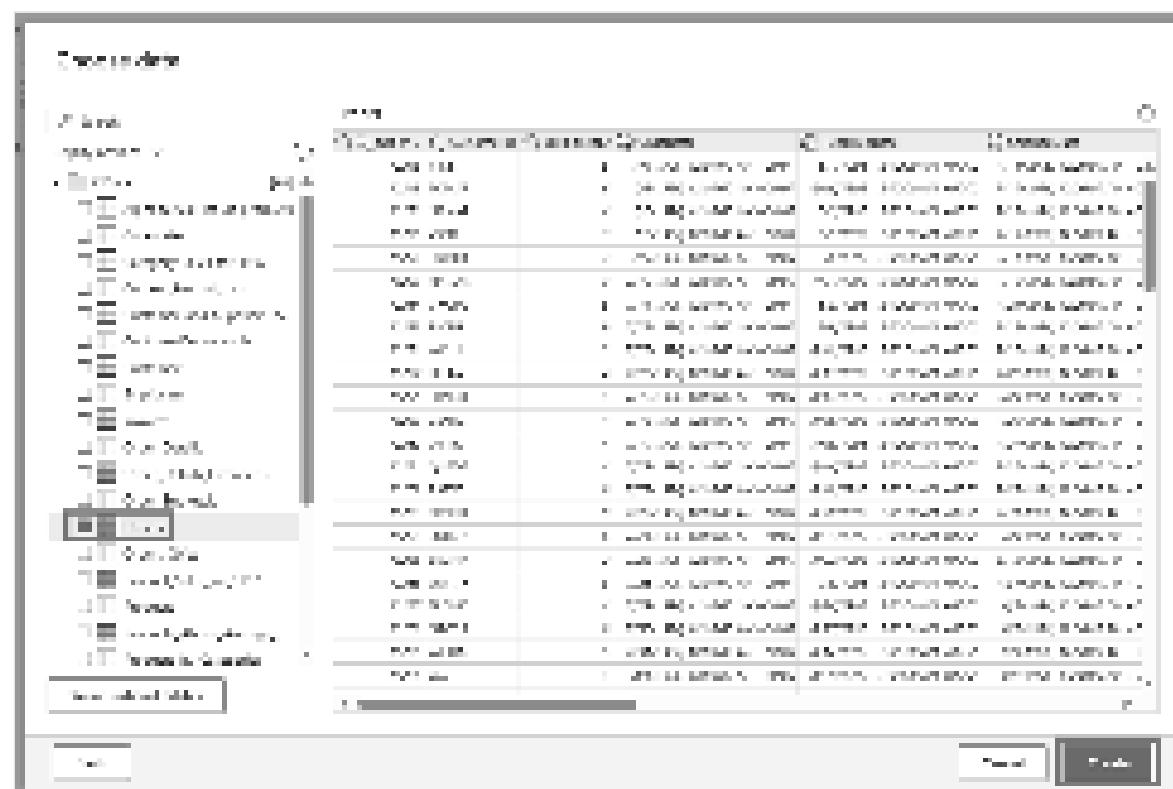


3. Add a new source to the dataflow. Select the OData source and enter the following URL: <https://services.OData.org/V4/Northwind/Northwind.svc>





4. Select the Orders table and select Next.



5. Select the following columns to keep:

- OrderID
- CustomerID
- EmployeeID

- OrderDate
- RequiredDate
- ShippedDate
- ShipVia
- Freight
- ShipName
- ShipAddress
- ShipCity
- ShipRegion
- ShipPostalCode
- ShipCountry



6. Change datatype of OrderDate, RequiredDate, and ShippedDate to datetime.

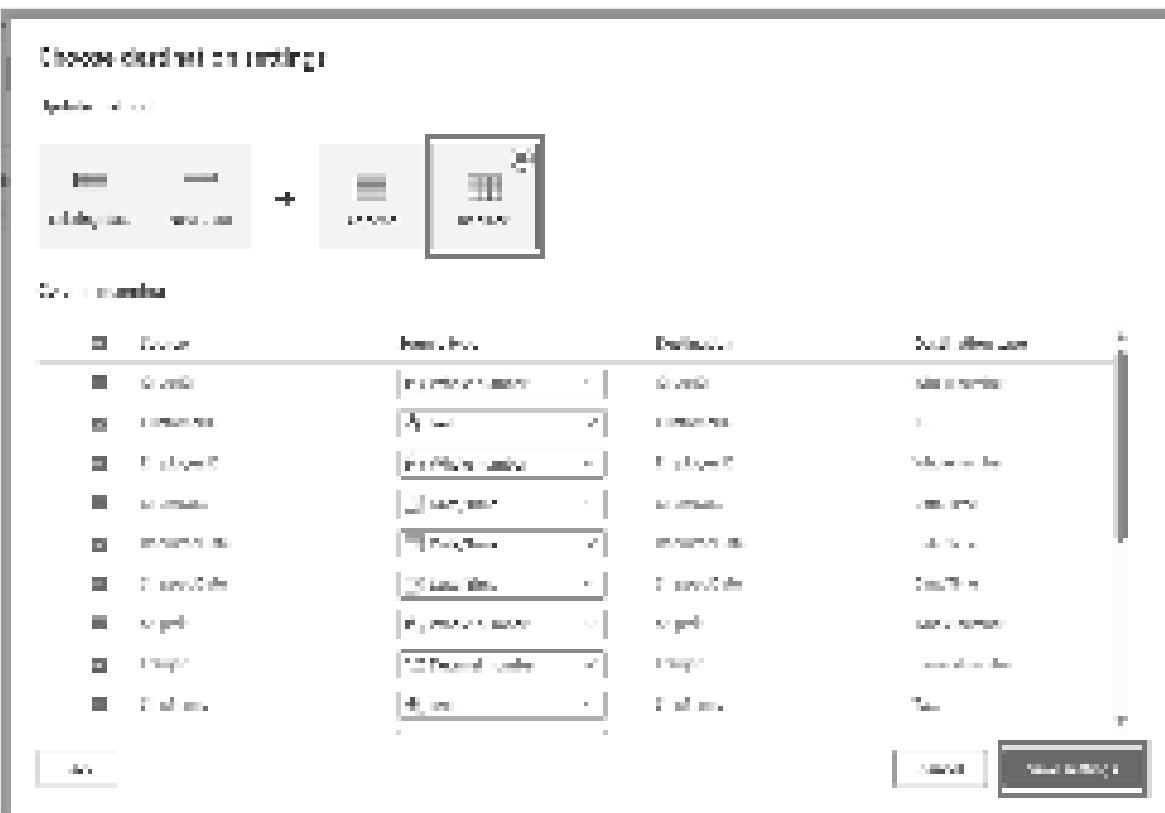
A screenshot of a database management tool showing a table named 'Orders'. The table has approximately 20 columns and over 100 rows. A context menu is open over the table, and the 'Copy' option is highlighted with a red box.

7. Set up the data destination to your lakehouse using the following settings:

- Data destination: `Lakehouse`
- Lakehouse: Select the lakehouse you created in step 1.
- New table name: `Orders`
- Update method: `Replace`

A screenshot of a data pipeline configuration screen. The 'Lakehouse' tab is selected. The 'Add data destination' button is highlighted with a red box. Below it, the 'Orders' table is listed under 'Selected tables'.

A screenshot of a 'Choose destination target' dialog. It shows two radio button options: 'New table' (selected) and 'Existing table'. On the left, there's a search bar and a sidebar with 'Display workspace' and 'My workspace' sections. On the right, there's a search bar and a table listing with one entry: 'Orders'.



8. select **Next** and publish the dataflow.



You have now created a dataflow to load data from an OData source into a lakehouse. This dataflow is used in the next section to add a query to the dataflow to filter the data based on the data destination. After that, you can use the dataflow to reload data using notebooks and pipelines.

Add a query to the dataflow to filter the data based on the data destination

This section adds a query to the dataflow to filter the data based on the data in the destination lakehouse. The query gets the maximum `OrderID` in the lakehouse at the beginning of the dataflow refresh and uses the maximum `OrderID` to only get the orders with a higher `OrderID` from source to append to your data destination. This assumes that orders are added to the source in ascending order of `OrderID`. If this isn't the case, you can use a different column to filter the data. For example, you can use the `OrderDate` column to filter the data.

1. After the dataflow refreshes, reopen the dataflow you created in the previous section.

	Name	Type
1	Dataflow 1	④ Dataflow Gen2
2	"DataflowStaging lakehouse1"	Update Edit
3	"DataflowStaging lakehouse1"	Export JSON

2. Create a new query named `IncrementalOrderID` and get data from the Orders table in the lakehouse you created in the previous section.



Choose data source

lakehouse

New sources

Lakehouse Microsoft Fabric (Preview)

Create data hub

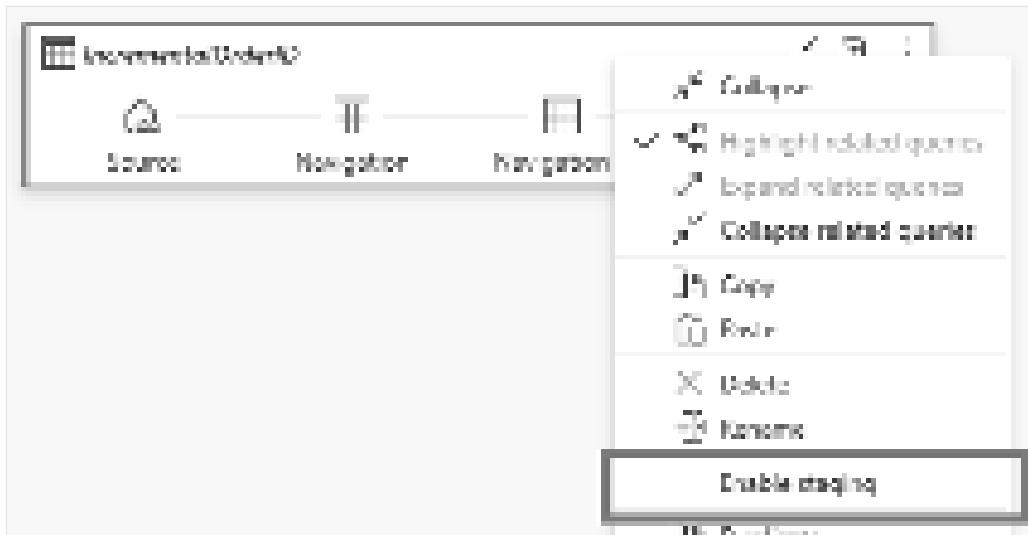
My data

Name	Type
MyExternalDatabase	SQL Endpoint
MyDataLakehouse	Lakehouse

The screenshot shows the SAP Fiori 'Choose data' interface. On the left, a navigation tree displays 'MySAPcloudapp' as the selected workspace, with 'Orders' highlighted. A context menu is open over the second row of the 'Orders' table, listing options: Copy, Paste, Delete, and Rename. The 'Rename' option is highlighted with a red box.

OrderID	CustomerID	EmployeeID	OrderDate	RequiredDate
10248	ALFKI	2	1997-02-12 00:00:00	1997-02-22 00:00:00
10249	ALFKI	2	1997-02-13 00:00:00	1997-03-02 00:00:00
10250	ANATC	2	1997-02-13 00:00:00	1997-03-02 00:00:00
10251	TBLON	2	1997-02-13 00:00:00	1997-03-02 00:00:00
10252	TRADL	2	1997-02-13 00:00:00	1997-03-02 00:00:00
10253	ANATC	2	1997-02-13 00:00:00	1997-03-02 00:00:00
10254	ANATC	2	1997-02-13 00:00:00	1997-03-02 00:00:00
10255	ANATC	2	1997-02-13 00:00:00	1997-03-02 00:00:00
10256	LEHLS	2	1997-02-13 00:00:00	1997-03-02 00:00:00
10257	LAUREL	2	1997-02-13 00:00:00	1997-03-02 00:00:00
10258	LAUREL	2	1997-02-13 00:00:00	1997-03-02 00:00:00
10259	KOSTA	2	1997-02-13 00:00:00	1997-03-02 00:00:00

3. Disable staging of this query.



4. In the data preview, right-click on the OrderID column and select Drill Down.

A screenshot of the Power BI Data Preview interface. The 'Drill Down' option in the context menu is highlighted with a red box. Below the menu, a table is displayed with four rows of data. The columns are labeled 'OrderID', 'CustomerID', and 'EmployeeID'. The data is as follows:

1	10234	SCARL
2	10235	SCAUR
3	10236	SCAUR
4	10237	SCAUR

5. From the ribbon, select List Tools -> Statistics -> Maximum.

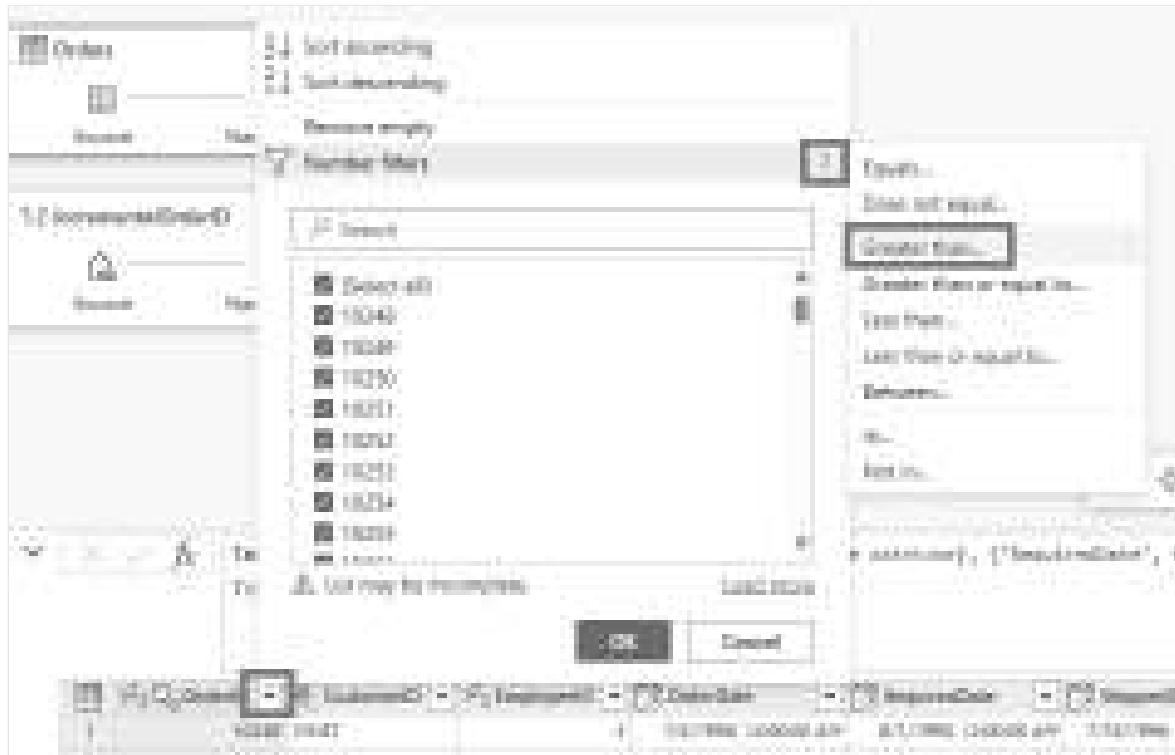


You now have a query that returns the maximum OrderID in the lakehouse. This query is used to filter the data from the OData source. The next section adds a query to the

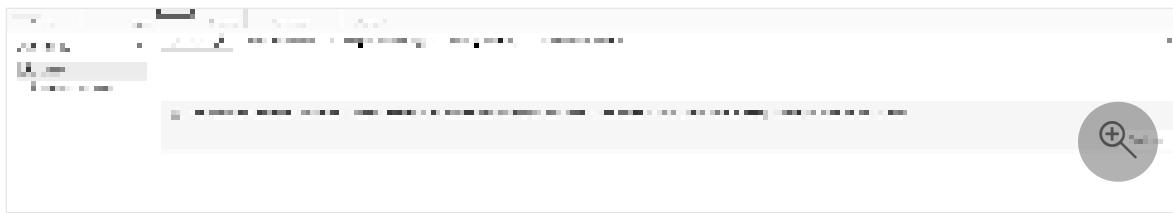
dataflow to filter the data from the OData source based on the maximum OrderID in the lakehouse.

1. Go back to the Orders query and add a new step to filter the data. Use the following settings:

- Column: OrderID
- Operation: Greater than
- Value: parameter IncrementalOrderID

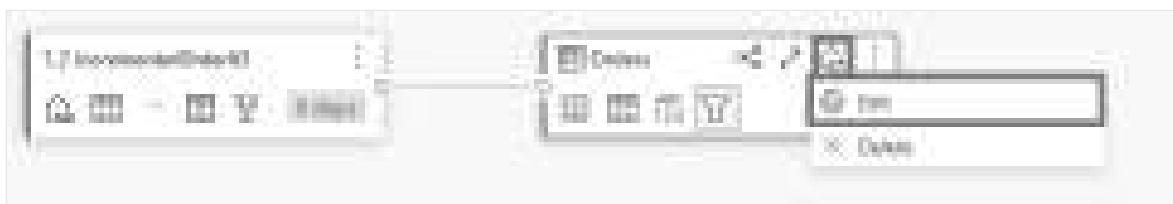


2. Allow combining the data from the OData source and the lakehouse by confirming the following dialog:



3. Update the data destination to use the following settings:

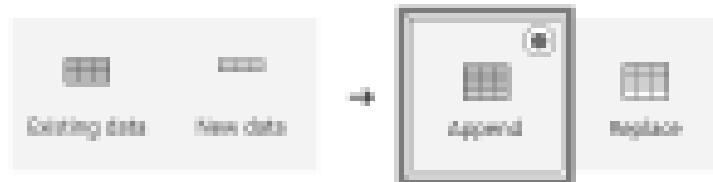
- Update method: Append



This is a continuation of the previous screenshot, showing the 'Choose destination target' dialog box again. The 'Append' update method is selected. The 'Tables' section shows a table named 'M1_IncomeAndHealth' with 14 columns. The 'Destination' section shows a database connection named 'Data'.

Choose destination settings

Update method



Column mapping

⚠️ Values changed since you last ran the output settings. Column mappings have been reset to their default.

Source	Destination	Type
OrderID	OrderID	Whole number
CustomerID	CustomerID	Text
EmployeeID	EmployeeID	Whole number
OrderDate	OrderDate	Date/Time
RequiredDate	RequiredDate	Date/Time
ShippedDate	ShippedDate	Date/Time
ShipVia	ShipVia	Whole number

4. Publish the dataflow.

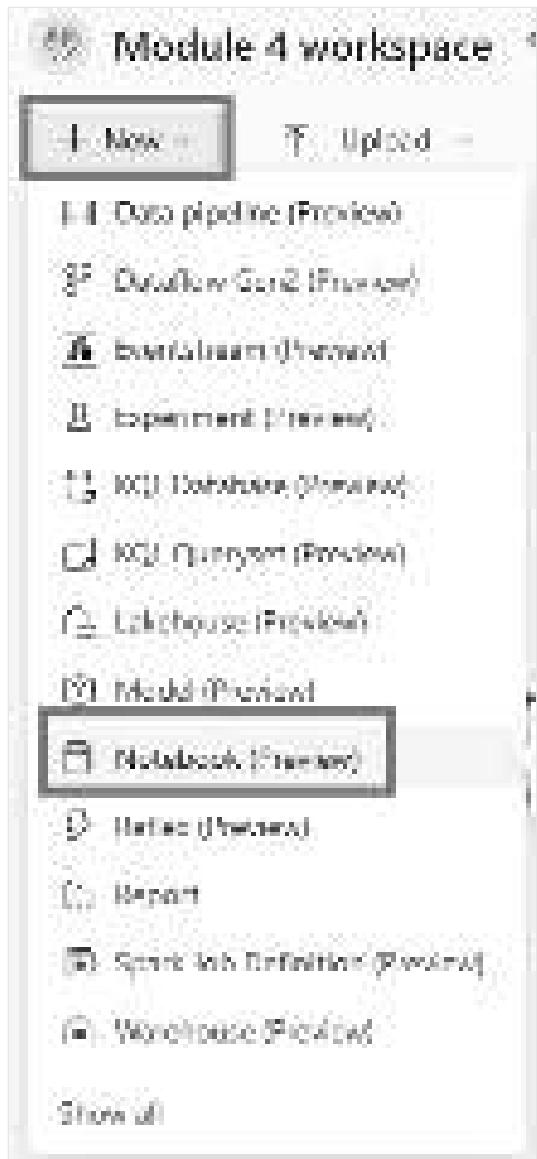


Your dataflow now contains a query that filters the data from the OData source based on the maximum OrderID in the lakehouse. This means that only new or updated data is loaded into the lakehouse. The next section uses the dataflow to reload data using notebooks and pipelines.

(Optional) reload data using notebooks and pipelines

Optionally, you can reload specific data using notebooks and pipelines. With custom python code in the notebook, you remove the old data from the lakehouse. By then creating a pipeline in which you first run the notebook and sequentially run the dataflow, you reload the data from the OData source into the lakehouse. Notebooks support multiple languages, but this tutorial uses PySpark. Pyspark is a Python API for Spark and is used in this tutorial to run Spark SQL queries.

1. Create a new notebook in your workspace.



2. Add the following PySpark code to your notebook:

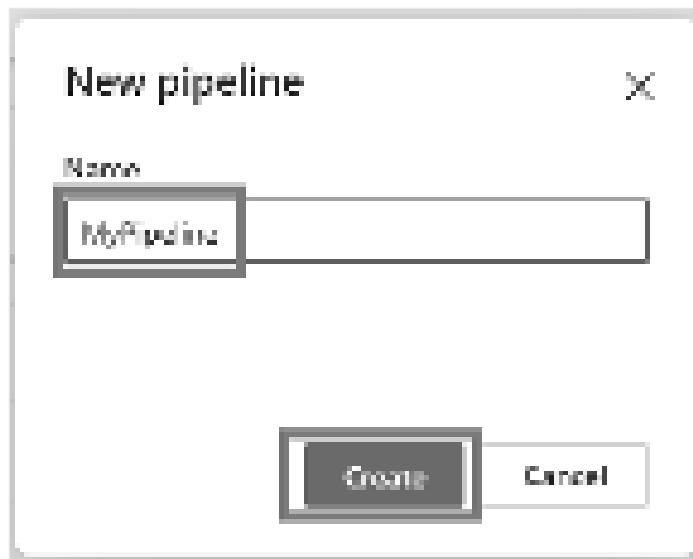
```
Python

### Variables
LakehouseName = "YOURLAKEHOUSE"
TableName = "Orders"
ColName = "OrderID"
NumberOfOrdersToRemove = "10"

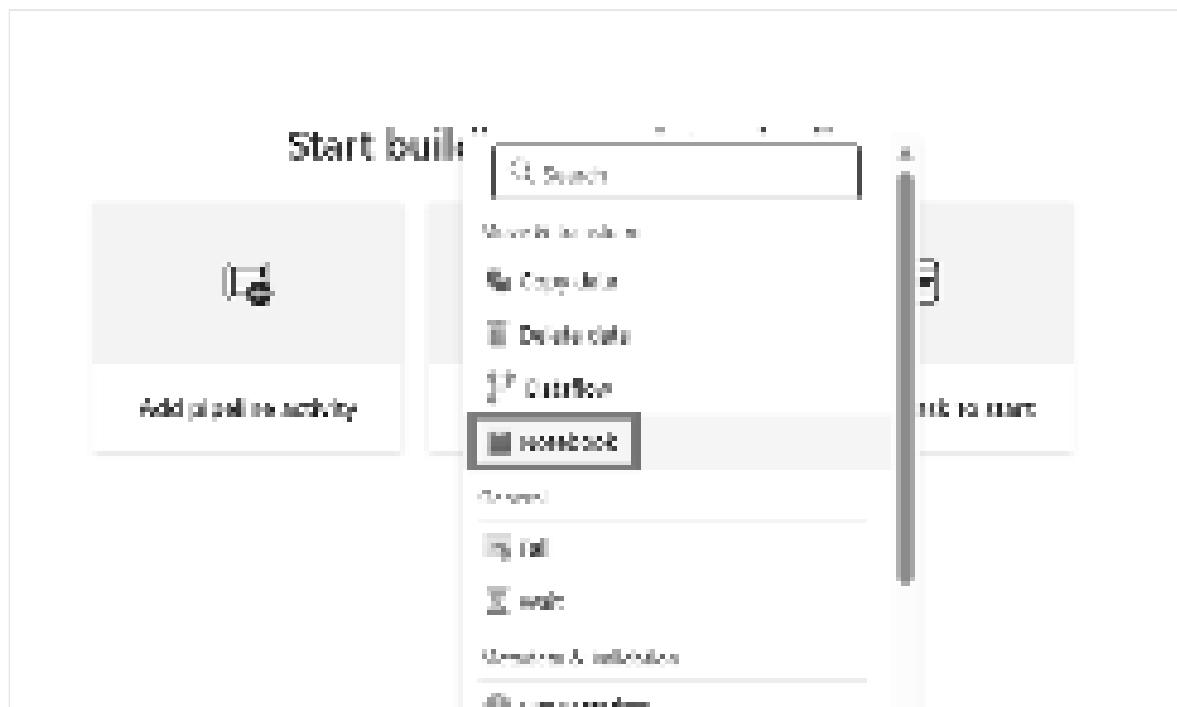
### Remove Old Orders
Reload = spark.sql("SELECT Max({0})-{1} as ReLoadValue FROM {2}.
{3}".format(ColName,NumberOfOrdersToRemove,LakehouseName,TableName)).co
llect()
Reload = Reload[0].ReLoadValue
spark.sql("Delete from {0}.{1} where {2} > {3}".format(LakehouseName,
TableName, ColName, Reload))
```

3. Run the notebook to verify that the data is removed from the lakehouse.

4. Create a new pipeline in your workspace.



5. Add a new notebook activity to the pipeline and select the notebook you created in the previous step.

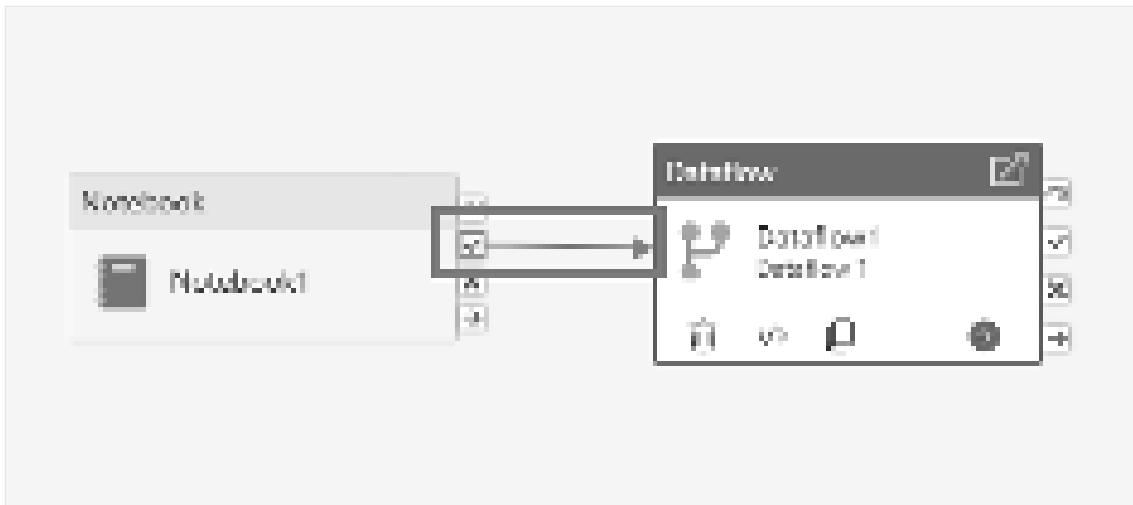




6. Add a new dataflow activity to the pipeline and select the dataflow you created in the previous section.



7. Link the notebook activity to the dataflow activity with a success trigger.



8. Save and run the pipeline.



You now have a pipeline that removes old data from the lakehouse and reloads the data from the OData source into the lakehouse. With this setup, you can reload the data from the OData source into the lakehouse on a regular basis.

Feedback

Was this page helpful?

Yes

No

Provide product feedback [↗](#) | Ask the community [↗](#)

A guide to learning Dataflows for Mapping Data Flow users

Article • 11/15/2023

Microsoft Fabric's Data Factory experience provides an intuitive and user-friendly interface using Power Query Online that can help you streamline your data transformation workflows when authoring Dataflow Gen2. If you're a developer with a background in Azure Data Factory and Mapping Data Flows, you'll find this guide helpful in mapping your existing Mapping Data Flow transformations to the Dataflow Gen2 Power Query user interface.

Global search box

When first starting to author Dataflows, you can also leverage the Global search box (shortcut: **Alt + Q**) found in the middle of the editor to search for connectors, transformations (actions), queries, and more while you learn your way around the Power Query Online interface.



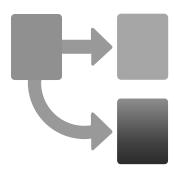
① Note

Learn more about [Global search box](#).

Transformations

When authoring Dataflow Gen2 using the Power Query Online interface, you can use the following table to understand how the current Mapping Data Flows capabilities translate and their equivalent toolbar navigation location in Dataflow Gen2 where applicable.

Multiple inputs/outputs

Mapping Data Flow	Dataflow Gen2
New branch	Reference
	 Important: Right click a query and select Reference.
Join	Merge queries
	 Merge queries ▾  Merge queries  Merge queries as new
<p>Navigation: Home > Merge queries</p> <p>Important: Merge queries – Merge this query into another query in this project. Merge queries as new – Merge this query into another query in this project to create a new query.</p>	
Conditional Split	Reference

Mapping Data**Dataflow Gen2****Flow****Reference**

Important: Right click a query and select Reference, you can then apply additional transformation logic to limit specific rows manually.

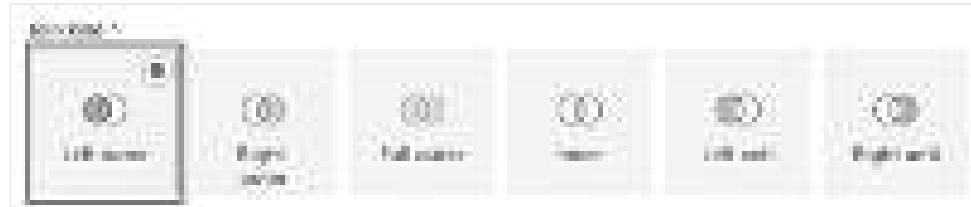
Union**Append queries****Append queries****Append queries****Append queries as new****Navigation:**

Home > Append queries

Important:

Append queries – Append this query into another query in this project.

Append queries as new – Append this query into another query in this project to create a new query.

Lookup**Merge queries****Merge queries****Merge queries****Merge queries as new****Navigation:**

Mapping Data Flow **Dataflow Gen2**

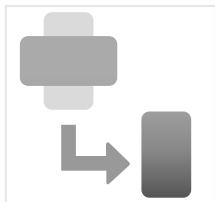
Home > Merge queries

Important: Select Left outer from the Join kind options.

Schema modifier

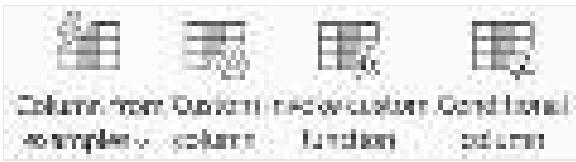
Mapping Data Flow

Derived Column



Dataflow Gen2

Custom column



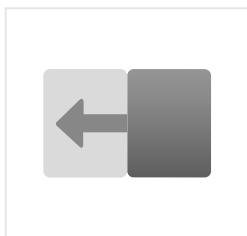
Navigation:

Add column > Custom column

Important:

Column from examples - Use examples to create a new column in this table. (Ctrl + E)
Custom column - Create a new column based on other columns, or a custom expression.
Invoke custom function - Invoke a custom function for each row of this table.
Conditional column - Create a new column that conditionally adds the values in the currently selected column.

Select



Choose columns



Note: Includes the ability to select, drop and rename (Name as) columns and whether to drop or pass on duplicate columns

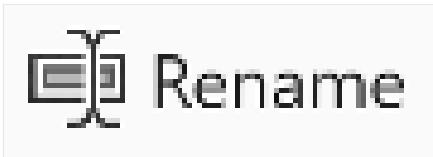
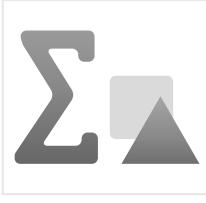
Navigation:

Home > Choose columns

(Remove Mapping)

Remove columns

Mapping Data Flow**Dataflow Gen2**

	 Remove columns
	<p>Navigation: Home > Remove columns</p>
(Name as)	Rename
	 Rename
	<p>Navigation: Transform > Rename</p>
Aggregate	Group by
	 Group by
	<p>Navigation: Transform > Group by</p>
Surrogate Key	Index column

Mapping Data Flow

Dataflow Gen2



Index column ▾

From 0

From 1

Custom...

Navigation:

Add column > Index column

Important:

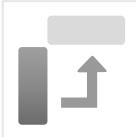
From 0 - Create a new column with an index starting at 0.

From 1 - Create a new column with an index starting at 1.

Custom... - Create a new column with an index starting at a specified value and a specified increment.

Pivot

Pivot column



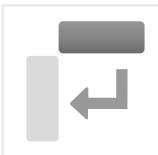
Pivot column

Navigation:

Transform > Pivot column

Unpivot

Unpivot columns



Unpivot columns ▾

Unpivot columns

Unpivot other columns

Unpivot only selected columns

Mapping Data Flow**Dataflow Gen2****Navigation:**

Transform > Unpivot columns

Important:

Unpivot columns - Translate all but the currently unselected columns into attribute-value pairs.

Unpivot other columns - Select all but the currently selected columns into attribute-value pairs.

Unpivot only selected columns - Translate only the currently selected columns into attribute-value pairs.

Rank**Rank column****Navigation:**

Add column > Rank column

External Call**Custom column****Navigation:**

Add column > Custom column

Expression:

Web.Contents

Cast**Data type**

Mapping Data Flow



Dataflow Gen2

The screenshot shows a dropdown menu titled "Data type" with the following options:

- 1.2 Decimal number
- \$ Currency
- 123 Whole number
- % Percentage
- 2023-01-01 Date/Time
- 2023-01-01 Date
- 00:00:00 Time
- 2023-01-01T00:00:00Z Date/Time/DateTime
- 00:00:00 Duration
- A Text
- True/False
- 0/1 Binary
- Using layout...

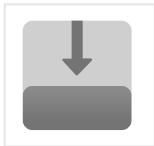
Navigation:

Transform > Data type

Formatters

Mapping Data Flow

Flatten



Dataflow Gen2

(Expand column)



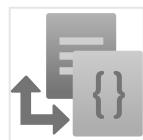
Important:

The expand option is available when structured data types exist in your column.

Mapping Data Flow**Dataflow Gen2**

Parse

Parse

**Navigation:**

Transform > Text column > Parse

Row modifier**Mapping Data Flow****Dataflow Gen2**

Filter

Filter rows

**Navigation:**

Home > Filter rows

Sort

Sort

Mapping Data Flow

Dataflow Gen2

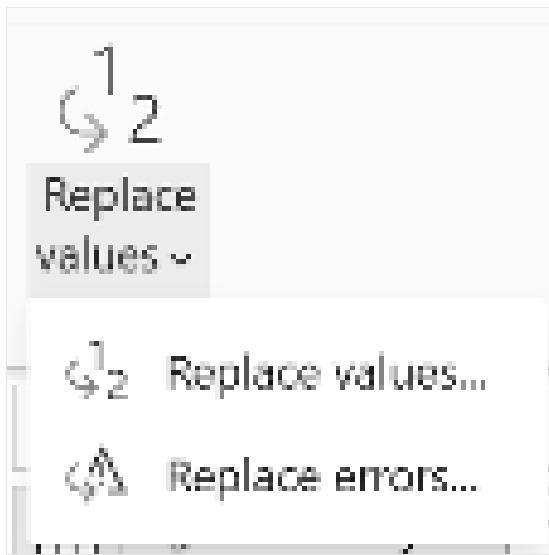
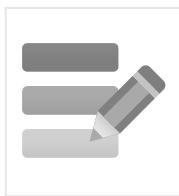


Navigation:

Home > Sort

Alter row

Replace values



Navigation:

Transform > Replace values

Flowlets

Mapping Data Flow

Dataflow Gen2

Flowlet

(Custom functions)

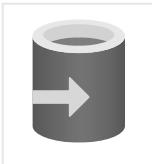


Learn more about how to Create a custom function

Destination

Mapping Data Flow	Dataflow Gen2
Sink	Add data destination

Sink



Add data destination



Navigation:
Home > Add data destination

Considerations and limitations

The following mapping data flow transformations are not supported in Dataflow Gen2. To vote for support of these operations, please visit Fabric ideas [↗](#).

- Assert
- Stringify
- Window

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback [↗](#) | Ask the community [↗](#)

Data Factory in Microsoft Fabric FAQ

FAQ

This article provides answers to frequently asked questions about Data Factory in Microsoft Fabric.

What is the future of Azure Data Factory (ADF) and Synapse Pipelines?

Azure Data Factory (ADF) and Azure Synapse pipelines maintain a separate Platform as a Service (PaaS) roadmaps. These two solutions continue to coexist alongside Fabric Data Factory, which serves as the Software as a Service (SaaS) offering. ADF and Synapse pipelines remain fully supported, and there are no plans for depreciation. It's important to highlight that, for any upcoming projects, our suggestion is to initiate them using Fabric Data Factory. Additionally, we have strategies in place to facilitate the transition of ADF and Synapse pipelines to Fabric Data Factory, enabling them to take advantage of new Fabric functionalities.

Given the functionality gaps in Data Factory for Fabric, what are the reasons for choosing it over ADF / Synapse pipelines?

As we strive to bridge functionality gaps and incorporate the robust data pipeline orchestration and workflow capabilities found in ADF / Azure Synapse pipelines into Fabric Data Factory, we acknowledge that certain features present in ADF / Synapse pipelines might be essential for your needs. While you're encouraged to continue utilizing ADF / Synapse pipelines if these features are necessary, we encourage you to first explore your new data integration possibilities in Fabric. Your feedback on which features are pivotal for your success is invaluable. To facilitate this, we're actively working on introducing a new capability, enabling the migration of your existing data factories from Azure into Fabric workspaces as well.

Are new features in Fabric Data Factory also available in ADF/Synapse?

We don't backport new features from Fabric pipelines into ADF / Synapse pipelines. We maintain two separate roadmaps for Fabric Data Factory and ADF/ Synapse. We evaluate backport requests in response to incoming feedback.

Is Fabric Pipeline same as Azure Synapse Pipeline?

The main function of Fabric pipeline is similar to Azure Synapse pipeline, but by using Fabric pipeline, users can apply all the data analytics capabilities in the Fabric platform. Notable differences and feature mappings between Fabric pipeline and Azure Synapse pipeline can be found here: Differences between Data Factory in Fabric and Azure.

What is the difference between data factory and data engineering tab in Fabric?

Data Factory helps you solve complex data integration and ETL scenarios with cloud-scale data movement and data transformation services while data engineering helps you create lake house, use Apache Spark to transform and prepare your data. Differences between each of the Fabric terminologies / experiences are available under Microsoft Fabric terminology.

Where can I find monthly updates available in Fabric?

Fabric monthly updates are available at the Microsoft Fabric Blog [↗](#).

How do I migrate existing pipelines from Azure Data Factory (or) Azure

Synapse workspace to Fabric Data Factory?

Currently, the only available method is to recreate the pipelines in Fabric Data Factory. We're diligently developing a new feature that empowers users to effectively oversee and manage both Fabric and ADF pipelines within the Fabric platform. This innovative new capability not only guarantees the seamless preservation of product continuity but also grants users the opportunity to immerse themselves in the enhanced functionalities offered by Fabric's data integration capabilities.

How do I track and monitor the capacity of Fabric used with the pipelines?

Microsoft Fabric capacity admins can use Microsoft Fabric Capacity Metrics app, also known as the metrics app, to gain visibility into capacity resources. This app enables admins to see how much CPU utilization, processing time, and memory are utilized by data pipelines, dataflows, and other items in their Fabric capacity-enabled workspaces. Gain visibility into overload causes, peak demand times, resource consumption and more and easily identify the most demanding or most popular items.

Is Fabric Dataflow Gen2 similar to Power Query embedded in Azure Data Factory?

The Power Query activity within ADF shares similarities with Dataflow Gen2, but it has extra features that enable actions like writing to specific data destinations etc. This comparison aligns more fairly with Dataflow Gen1 (Power BI dataflows or Power Apps dataflows). Have a look here for more details: Differences between Dataflow Gen1 and Dataflow Gen2.

How can I connect to on-premises data sources in Fabric Data Factory?

Our current focus involves the active development of Fabric pipeline support within the on-premises data gateway. This forthcoming capability empowers you to seamlessly harness Fabric pipelines for direct on-premises data access. Until this feature is available,

a viable workaround is possible: you can utilize Fabric dataflow to transfer data to cloud storage and then employ Fabric pipeline to facilitate the movement of data to your desired destination. This ensures a smooth transition until the on-premises data gateway integration is available.

Is it possible to connect to existing Private Endpoint (PE) enabled resources in Fabric Data Factory?

Presently, the VNet gateway offers an injective method to seamlessly integrate into your virtual network, providing a robust avenue for using private endpoints to establish secure connections to your data stores. It's important to note that the VNet gateway only accommodates Fabric dataflows at this moment. However, our upcoming initiatives encompass expanding its capabilities to encompass Fabric pipelines.

How fast can I ingest data in Fabric Data Pipelines?

Fabric Data Factory allows you to develop pipelines that maximize data movement throughput for your environment. These pipelines fully utilize the following resources:

- Network bandwidth between the source and destination data stores
- Source or destination data store input/output operations per second (IOPS) and bandwidth. This full utilization means you can estimate the overall throughput by measuring the minimum throughput available with the following resources:
 - Source data store
 - Destination data store
 - Network bandwidth in between the source and destination data stores
- Meanwhile, we continuously work on innovations to boost the best possible throughput you can achieve. Today, the service can move 1 TB TPC-DI dataset (parquet files) into both Fabric Lakehouse table and Data Warehouse within 5 minutes - moving 1B rows under 1 min; Please note that this performance is only a reference by running the above testing dataset. The actual throughput still depends on the factors listed previously. In addition, you can always multiply your throughput by running multiple copy activities in parallel. For example, using ForEach loop.

What approach is recommended for assigning roles within Fabric Data Factory?

You can separate the different workloads between workspaces and use the roles like member and viewer to have a workspace for data engineering that preps data for a workspace that is used for report or AI training. With the viewer role, you can then consume data from the data engineering workspace.

Next steps

[Introduction - Data Factory tutorial](#)

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#) | [Ask the community](#)

On-premises data gateway considerations for data destinations in Dataflow Gen2

Article • 11/15/2023

This article tries to list the limitations and considerations when using the Data Gateway with data destinations scenarios in Dataflow Gen2.

Evaluation time outs

Dataflows that use a Gateway and the data destination feature are limited to an evaluation or refresh time of one hour.

Learn more about this limitation from the article on the [Troubleshoot the on-premises data gateway](#) article.

Network issues with port 1433

When using Microsoft Fabric Dataflow Gen2 with an on-premises data gateway, you might encounter issues with the dataflow refresh process. The underlying problem occurs when the gateway is unable to connect to the dataflow staging Lakehouse in order to read the data before copying it to the desired data destination. This issue can occur regardless of the type of data destination being used.

During the overall dataflow refresh, the tables refresh can show as "Succeeded," but the activities section shows as "*Failed*". The error details for the activity

`WriteToDatabaseTableFrom_...` indicate the following error:

```
Mashup Exception Error: Couldn't refresh the entity because of an issue with the
mashup document MashupException.Error: Microsoft SQL: A network-related or
instance-specific error occurred while establishing a connection to SQL Server. The
server was not found or was not accessible. Verify that the instance name is
correct and that SQL Server is configured to allow remote connections. (provider:
TCP Provider, error: 0 - An attempt was made to access a socket in a way forbidden
by its access permissions.) Details: DataSourceKind = Lakehouse;DataSourcePath =
Lakehouse;Message = A network-related or instance-specific error occurred while
establishing a connection to SQL Server. The server was not found or was not
accessible. Verify that the instance name is correct and that SQL Server is
```

```
configured to allow remote connections. (provider: TCP Provider, error: 0 - An attempt was made to access a socket in a way forbidden by its access permissions.);ErrorCode = -2146232060;Number = 10013
```

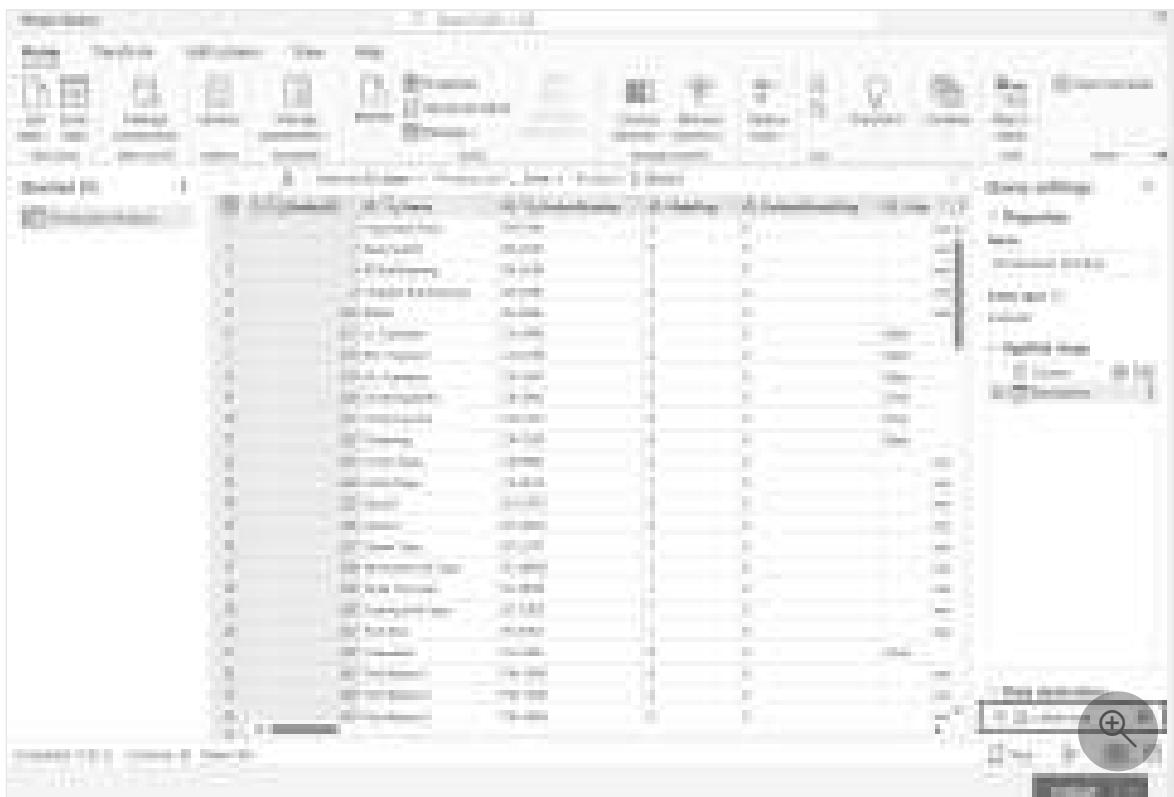
Note

From an architectural perspective, the dataflow engine uses an outbound HTTPS (port 443) endpoint to write data into a Lakehouse. However, reading data from the Lakehouse requires the use of the TDS protocol (TCP over port 1433). This protocol is utilized to copy the data from the staging lakehouse to the data destination. This explains why the Tables Load step succeeds while the data destination activity fails, even when both lakehouses are in the same OneLake instance.

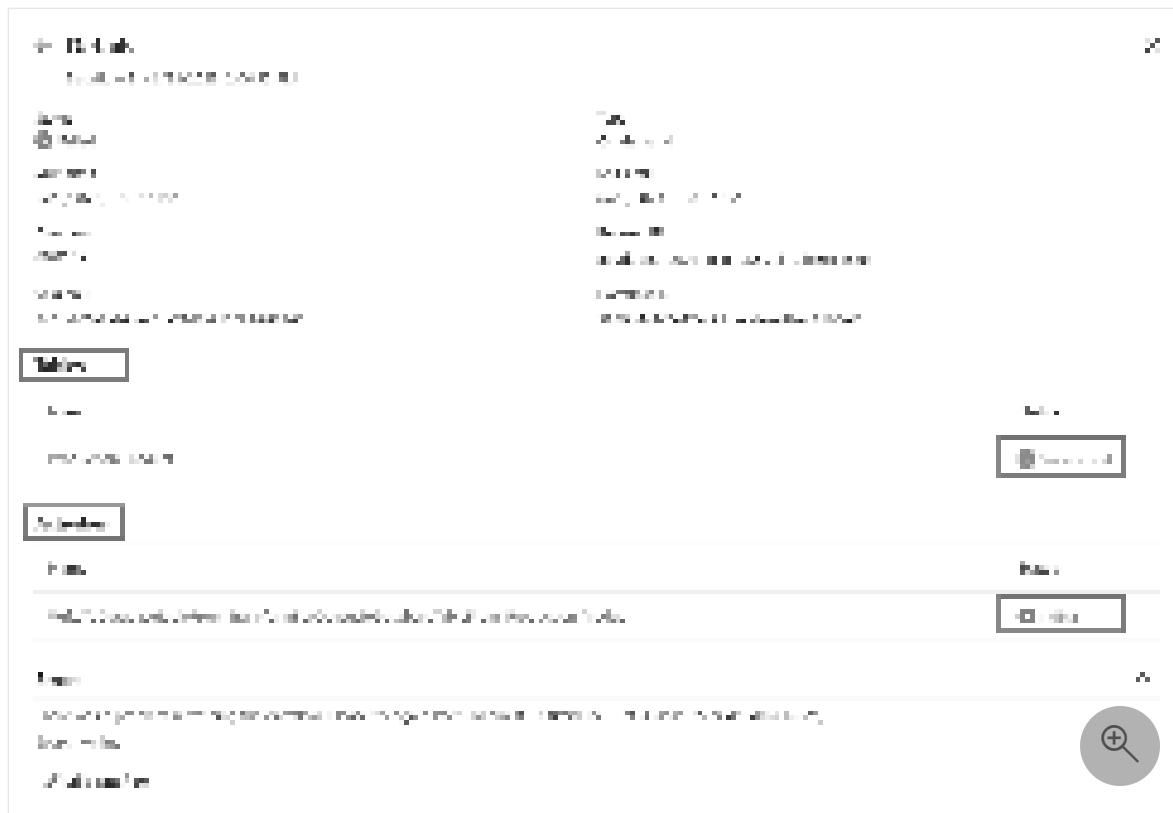
Troubleshooting

To troubleshoot the issue, follow these steps:

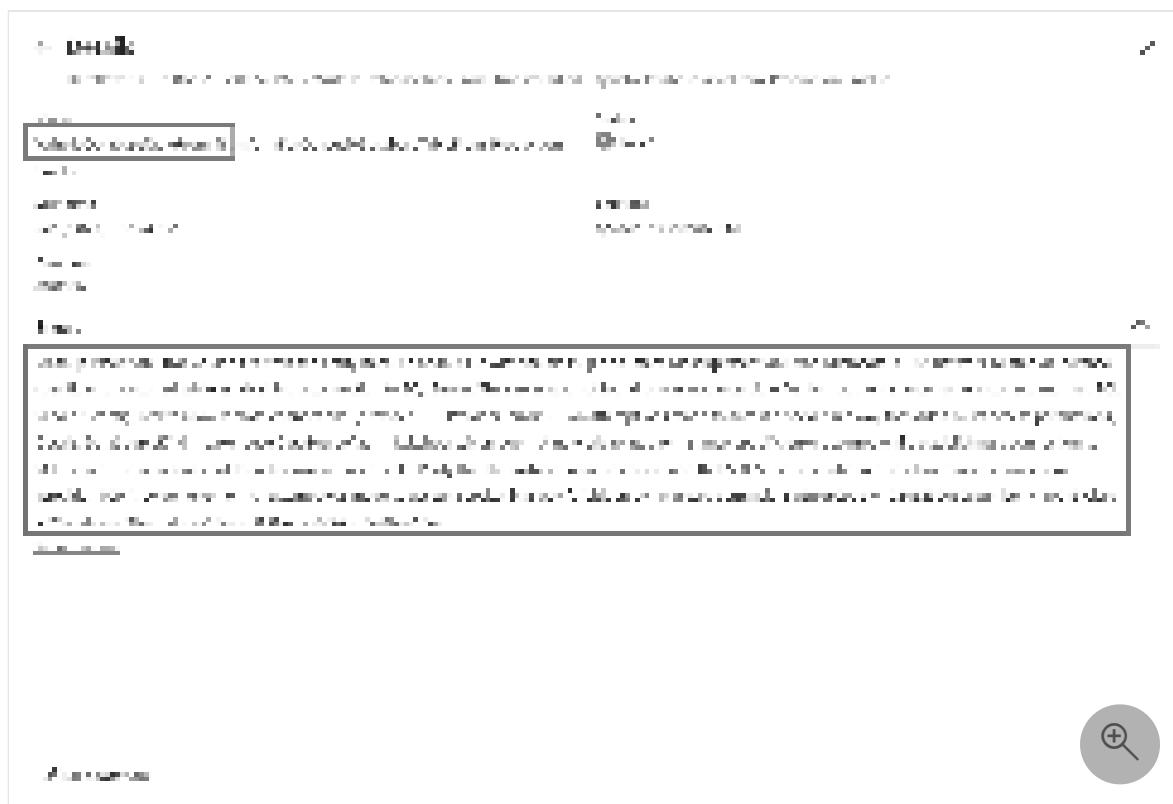
1. Confirm that the dataflow is configured with a data destination.



2. Verify that the dataflow refresh fails, with tables refresh showing as "*Succeeded*" and activities showing as "*Failed*".



3. Review the error details for the Activity `WriteToDatabaseTableFrom_...`, which provides information about the encountered error.

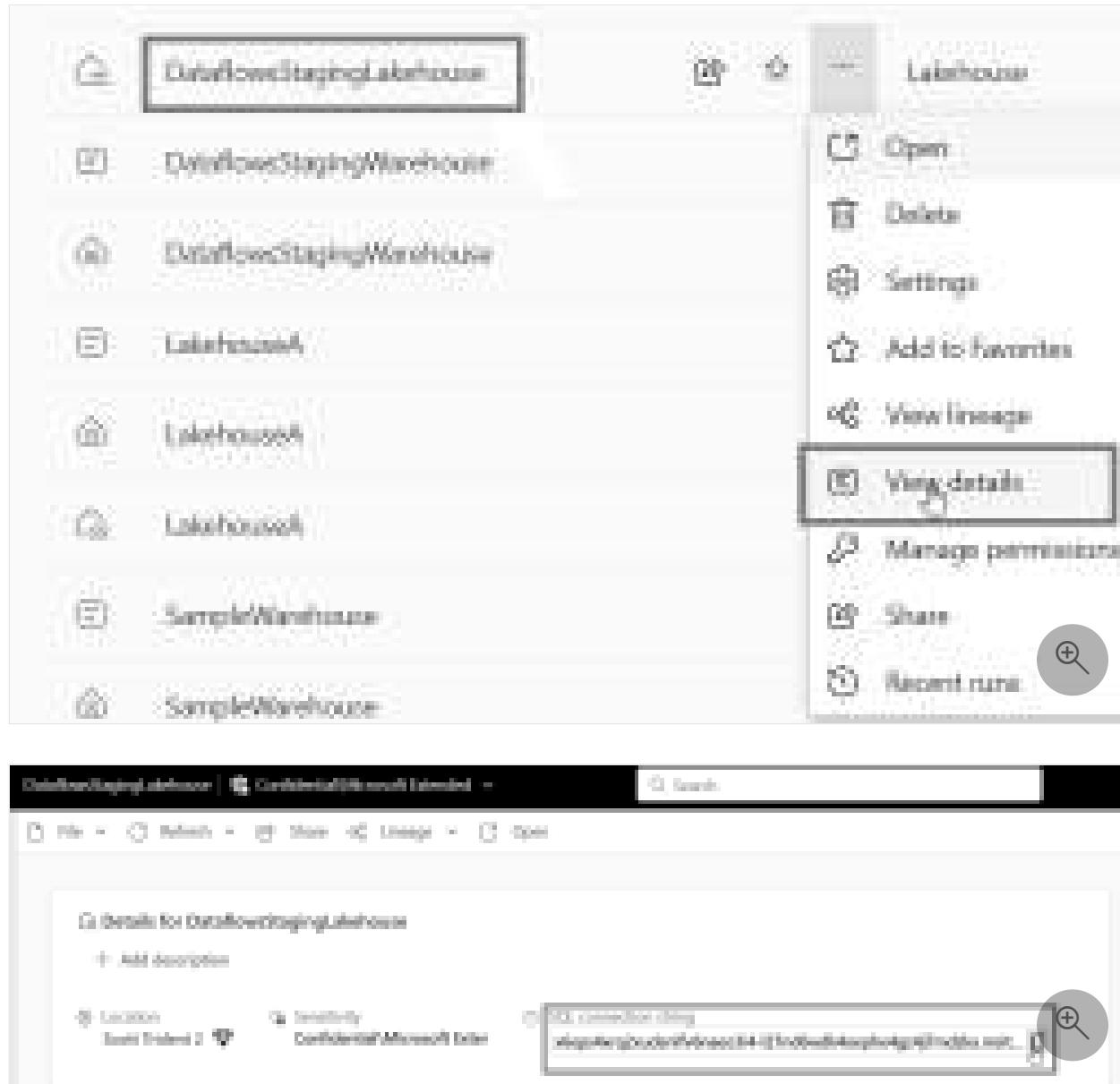


Solution: Set new firewall rules on server running the gateway

The firewall rules on the gateway server and/or customer's proxy servers need to be updated to allow outbound traffic from the gateway server to the following:

- **Protocol:** TCP
- **Endpoint:** *.datawarehouse.pbid.edicated.windows.net
- **Port:** 1433

If you want to narrow down the scope of the endpoint to the actual OneLake instance in a workspace (instead of the wildcard *.datawarehouse.pbid.edicated.windows.net), that URL can be found by navigating to the Fabric workspace, locating `DataflowsStagingLakehouse`, and selecting **View Details**. Then, copy and paste the SQL connection string.



The entire endpoint name looks similar to the following example:

x6eps4xrq2xuden1fv6naeo3i4-

127nd6wdk4oephe4gz4j7mdzka.datawarehouse.pbid.edicated.windows.net

Workaround: Split dataflow in a separate ingest and load dataflow

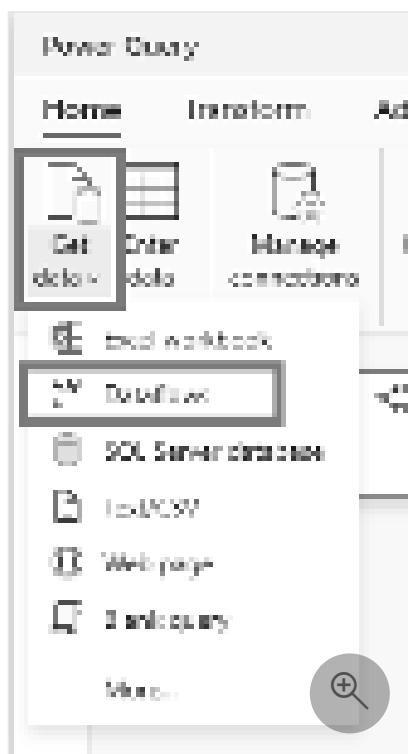
If you're unable to update the firewall rules, you can split the dataflow into two separate dataflows. The first dataflow is responsible for ingesting the data into the staging lakehouse. The second dataflow is responsible for loading the data from the staging lakehouse into the data destination. This workaround isn't ideal, as it requires the use of two separate dataflows, but it can be used as a temporary solution until the firewall rules can be updated.

To implement this workaround, follow these steps:

1. Remove the data destination from your current dataflow that ingests data via your gateway.



2. Create a new dataflow that uses the dataflow connector to connect to the ingest dataflow. This dataflow is responsible for ingesting the data from staging into the data destination.





3. Set the data destination to be the data destination of your choice for this new dataflow.



4. Optionally, you can disable staging for this new dataflow. This change prevents the data from being copied to the staging lakehouse again and instead copies the data directly from the ingest dataflow to the data destination.



Feedback

Was this page helpful?

 Yes

No

Provide product feedback ↗ | Ask the community ↗

Getting from Azure Data Factory to Data Factory in Microsoft Fabric

Article • 11/15/2023

Data Factory in Microsoft Fabric is the next generation of Azure Data Factory which provides cloud-scale data movement and data transformation services that allow you to solve the most complex ETL scenarios. It's intended to make your experience easy to use, powerful, and truly enterprise-grade. This article compares the differences between Azure Data Factory and Data Factory in Microsoft Fabric.

Feature mapping

In the modern experience of Data Factory in Fabric, there are some different features concepts compared to Azure Data Factory. Detail features mapping is presented as the table below.

Azure Data Factory	Data Factory in Fabric	Description
Pipeline	Data pipeline	Data pipeline in Fabric is better integrated with the unified data platform including Lakehouse, Datawarehouse, and more.
Mapping dataflow	Dataflow Gen2	Dataflow Gen2 provides easier experience to build transformation. We are in progress of letting more functions of mapping dataflow supported in Dataflow Gen2
Activities	Activities	We are in progress to make more activities of ADF supported in Data Factory in Fabric. Data Factory in Fabric also has some newly attracted activities like Office 365 Outlook activity. Details are in Activity overview.
Dataset	Not Applicable	Data Factory in Fabric doesn't have dataset concepts. Connection will be used for connecting each data source and pull data.
Linked Service	Connections	Connections have similar functionality as linked service, but connections in Fabric have more intuitive way to create.
Triggers	Schedules (other triggers are in	Fabric can use the schedule to automatically run pipeline. We are adding more triggers supported by

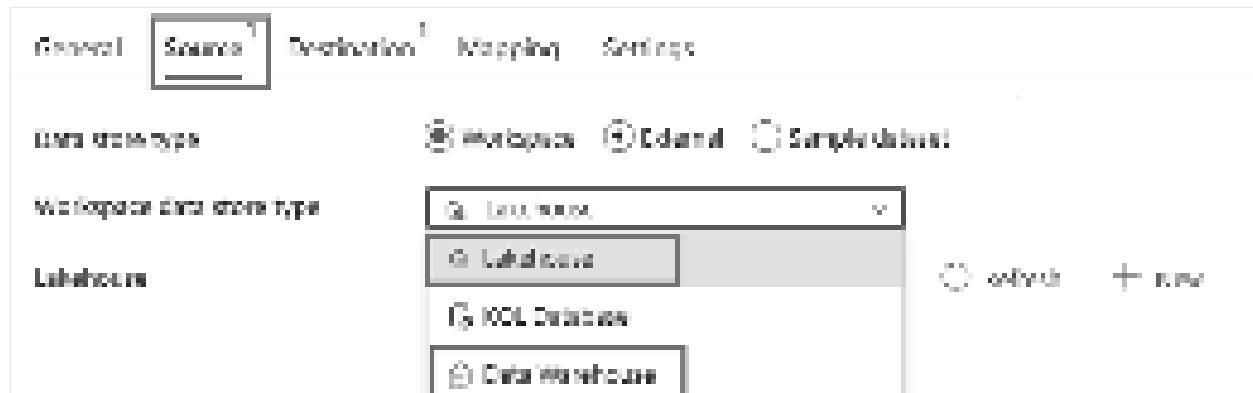
Azure Data Factory	Data Factory in Fabric	Description
	progress)	ADF in Microsoft Fabric.
Publish	Save, Run	For pipeline in Fabric, you don't need to publish to save the content. Instead, you can use Save button to save the content directly. When you click Run button, it will save the content before running pipeline.
Autoresolve and Azure Integration runtime	Not Applicable	In Fabric, we don't have the concept of Integration runtime.
Self-hosted integration runtimes	On-premises Data Gateway(in design)	The capability in Fabric is still in progress of design.
Azure-SSIS integration runtimes	To be determined	The capability in Fabric hasn't confirmed the roadmap and design.
MVNet and Private End Point	To be determined	The capability in Fabric hasn't confirmed the roadmap and design.
Expression language	Expression language	Expression language is similar in ADF and Fabric.
Authentication type in linked service	Authentication kind in connection	Authentication kind in Fabric pipeline already supported popular authentication types in ADF, and more authentication kinds will be added.
CI/CD	CI/CD	CI/CD capability in Fabric Data Factory will be coming soon.
Export and Import ARM	Save as	Save as is available in Fabric pipeline to duplicate a pipeline.
Monitoring	Monitoring, Run history	The monitoring hub in Fabric has more advanced functions and modern experience like monitoring across different workspaces for better insights.

Data pipeline of data factory in Microsoft Fabric

There are many exciting features of data pipeline in Data Factory of Microsoft Fabric. Leveraging these features, you can feel the power of pipeline in Fabric.

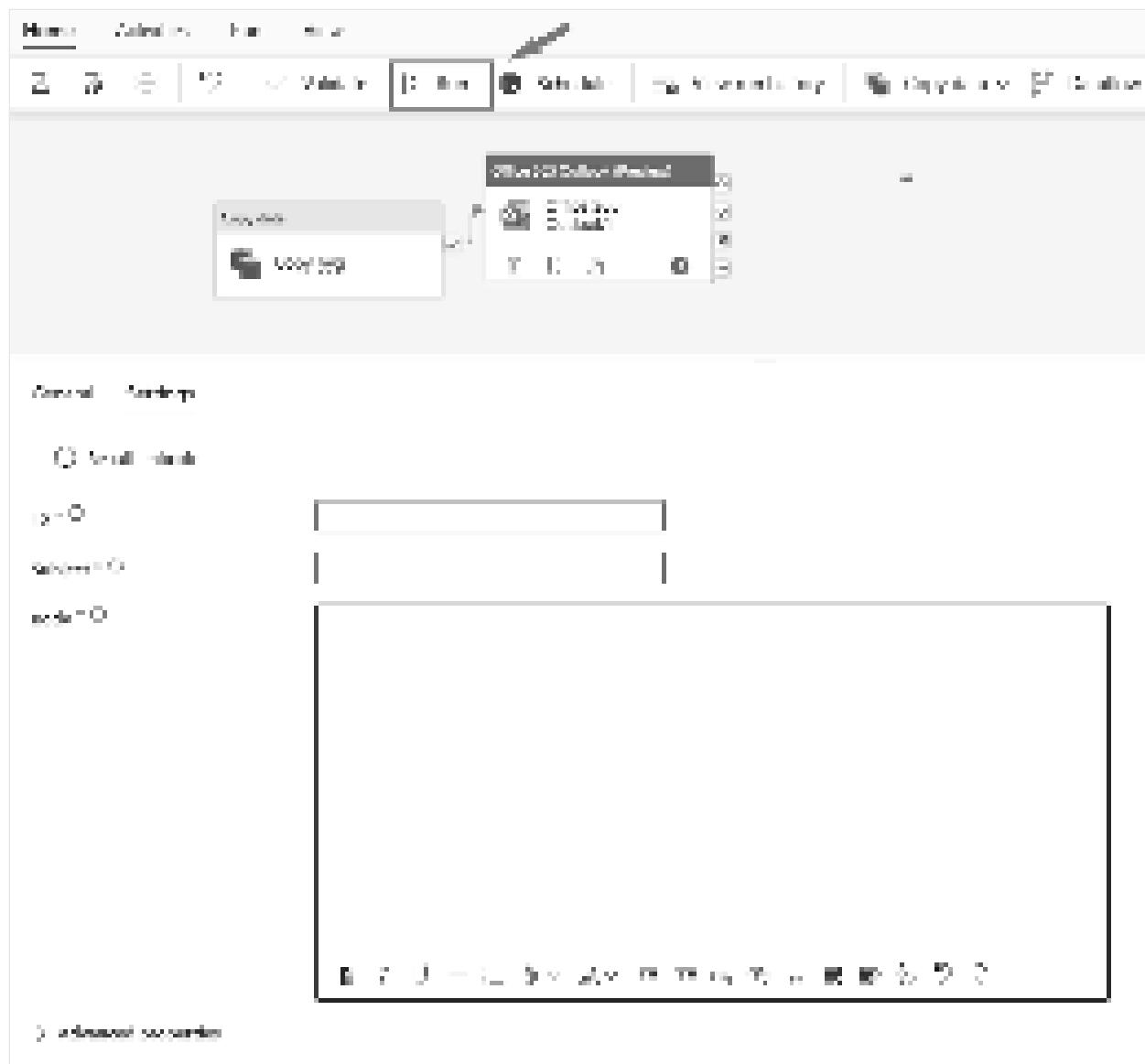
Lakehouse/Datawarehouse integration

Lakehouse and Data Warehouse are available as source and destination in Pipeline of Fabric, so it's extremely convenient for you to build your own projects integrated with Lakehouse and Datawarehouse.



Office 365 outlook activity

Office 365 outlook activity provides an intuitive and simple way to send customized email notification about info of pipeline and activity, and output of pipeline by easy configuration.



Get data experience

A modern and easy Get Data experience is provided in Data Factory in Fabric, so it's super-fast for you to set up your copy pipeline and create a new connection.

The screenshot shows the 'Copy data' interface in the Azure portal. On the left, there's a sidebar with navigation links: 'Copy data home', 'Data flows', 'Data pipelines', 'Data integration', and 'Data factories'. The main area is titled 'Data sources' and contains a grid of 12 items, each with a preview icon and a name. The items are: 'Azure Blob Storage (classic)', 'Azure Blob Storage (File)', 'Azure Data Lake Storage Gen1', 'Azure Data Lake Storage Gen2', 'Azure Data Lake Storage (File)', 'Azure Data Lake Storage (File) (classic)', 'Azure Data Lake Storage (File) (classic) (File)', 'Azure Data Lake Storage (File) (classic) (File) (classic)', 'Azure Data Lake Storage (File) (classic) (File) (classic) (File)', 'Azure Data Lake Storage (File) (classic) (File) (classic) (File) (classic)', 'Azure Data Lake Storage (File) (classic) (File) (classic) (File) (classic) (File)', and 'Azure Data Lake Storage (File) (classic) (File) (classic) (File) (classic) (File) (classic) (File)'. Below the grid are two buttons: 'Create' and 'Import'.

This screenshot shows a detailed view of a data source within the 'Copy data' interface. The left sidebar shows 'Copy data home', 'Data flows', 'Data pipelines', 'Data integration', and 'Data factories'. The main area has a title 'Copy data' and a subtitle 'Copy data from Azure Blob Storage (classic) to Azure Data Lake Storage (classic)'. It includes sections for 'Source settings', 'Sink settings', and 'Transformation'. A large 'Configure' button is at the bottom. On the right, there's a detailed configuration pane for 'Source settings' which lists 'Source type', 'Container', 'File path', 'File type', 'File format', and 'File encoding'. The 'File path' dropdown is expanded, showing a list of paths: 'Root folder', 'Container', 'File path', 'File name', 'File type', 'File format', 'File encoding', 'File separator', 'File quote', 'File escape', 'File ignore first n lines', 'File ignore last n lines', 'File ignore first n bytes', 'File ignore last n bytes', and 'File ignore first n chars'. Below this is a 'File separator' dropdown with options like 'Comma', 'Tab', 'Semicolon', and 'Space'.

Modern monitoring experience

With the combined capabilities of the monitoring hub and the items of Data Factory, such as data flows and data pipelines, we can get a full view of all the workloads and drill into any activity within a data factory experience. It's also convenient for you to do the cross-workspace analysis through monitoring hub.

The screenshot shows the 'Monitoring' tab in the Azure Data Factory portal. It displays a table of copy activity runs. One row in the table has a tooltip labeled 'Run details' with a glasses icon, indicating that clicking it will provide more detailed information about that specific run.

The pipeline copy monitoring results provides breakdown detail of copy activity. By selecting the run details button (with the **glasses icon** highlighted) to view the run details. Expand the **Duration breakdown**, you can know the time duration of each stage in copy activity.

This screenshot shows the 'Run details' modal window for a specific copy activity. The window title is 'Run details' and it contains the activity name 'Copy Data' and the source 'Azure Blob Storage'. The 'Duration breakdown' section is expanded, showing the time spent on 'Copy' and 'Compute' tasks. The total duration is listed as 1.007 ms.

Copy data details

Start time 4/26/2023, 4:22:58 PM

Pipeline run activity ID < Pipeline run activity ID >

Throughput < Throughput >

Total duration 00:00:56

< Duration breakdown > ↗

Start time 4/26/2023, 4:22:59 PM

Used DIUs ⓘ 4

Used parallel copies ⓘ 1

Queue

Transit

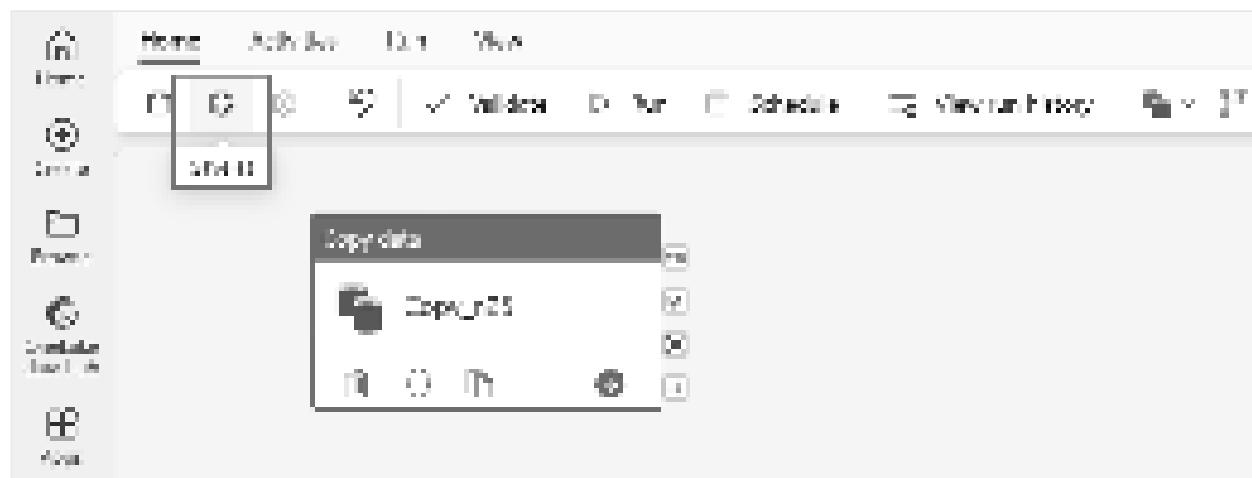
Reading from...

Writing to sink

> Advanced

Save as

Save as in Fabric pipeline provides a convenient way for you to duplicate an existing pipeline for other development purposes.



Next steps

- Differences between Dataflow Gen1 Gen2
 - Build your first data integration
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Getting from Dataflow Generation 1 to Dataflow Generation 2

Article • 11/15/2023

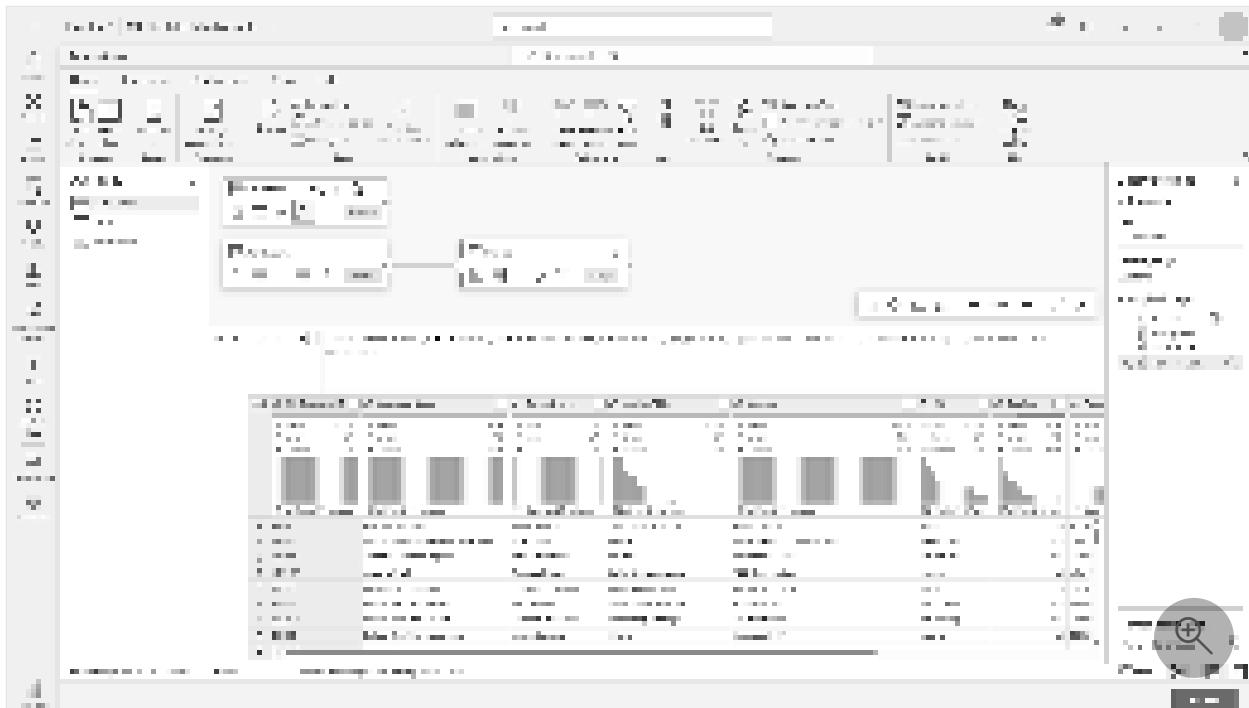
Dataflow Gen2 is the new generation of dataflows. The new generation of dataflows resides alongside the Power BI Dataflow (Gen1) and brings new features and improved experiences. The following section provides a comparison between Dataflow Gen1 and Dataflow Gen2.

Feature overview

Feature	Dataflow Gen2	Dataflow Gen1
Author dataflows with Power Query	✓	✓
Shorter authoring flow	✓	
Auto-Save and background publishing	✓	
Data destinations	✓	
Improved monitoring and refresh history	✓	
Integration with data pipelines	✓	
High-scale compute	✓	
Get Data via Dataflows connector	✓	✓
Direct Query via Dataflows connector		✓
Incremental refresh		✓
AI Insights support		✓

Shorter authoring experience

Working with Dataflow Gen2 feels like coming home. We kept the full Power Query experience you're used to in Power BI dataflows. When you enter the experience, you're guided step-by-step for getting the data into your dataflow. We also shorten the authoring experience to reduce the number of steps required to create dataflows, and added a few new features to make your experience even better.



New dataflow save experience

With Dataflow Gen2, we changed how saving a dataflow works. Any changes made to a dataflow are autosaved to the cloud. So you can exit the authoring experience at any point and continue from where you left off at a later time. Once you're done authoring your dataflow, you publish your changes and those changes are used when the dataflow refreshes. In addition, publishing the dataflow saves your changes and runs validations that must be performed in the background. This feature lets you save your dataflow without having to wait for validation to finish.

To learn more about the new save experience, go to [Save a draft of your dataflow](#).

Data destinations

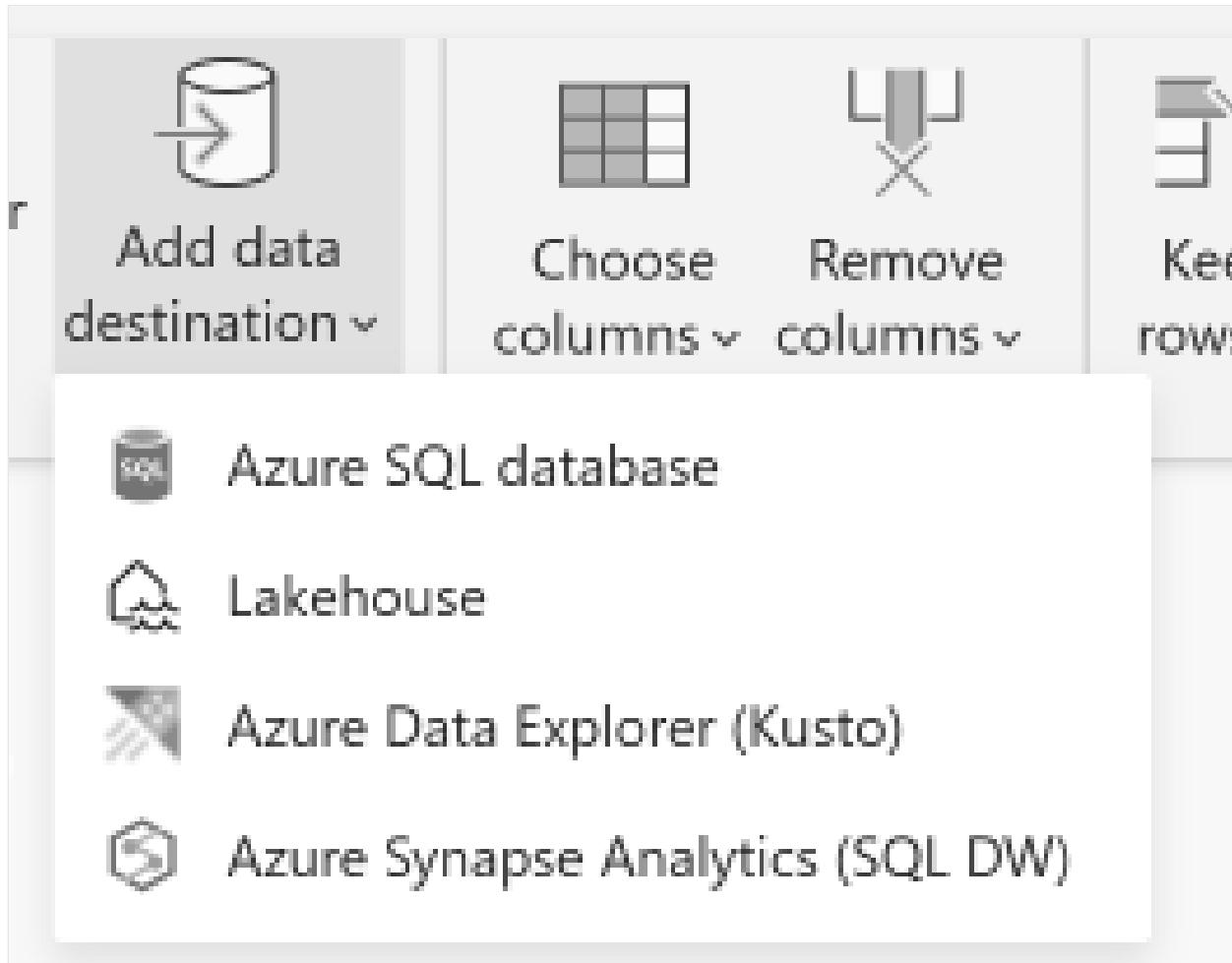
Similar to Dataflow Gen1, Dataflow Gen2 allows you to transform your data into dataflow's internal/staging storage where it can be accessed using the Dataflow connector. Dataflow Gen2 also allows you to specify a data destination for your data. Using this feature, you can now separate your ETL logic and destination storage. This feature benefits you in many ways. For example, you can now use a dataflow to load data into a lakehouse and then use a notebook to analyze the data. Or you can use a dataflow to load data into an Azure SQL database and then use a data pipeline to load the data into a data warehouse.

In Dataflow Gen2, we added support for the following destinations and many more are coming soon:

- Fabric Lakehouse
- Azure Data Explorer (Kusto)
- Azure Synapse Analytics (SQL DW)
- Azure SQL Database

① Note

To load your data to the Fabric Warehouse, you can use the Azure Synapse Analytics (SQL DW) connector by retrieving the SQL connection string. More information: [Connectivity to data warehousing in Microsoft Fabric](#)



New refresh history and monitoring

With Dataflow Gen2, we introduce a new way for you to monitor your dataflow refreshes. We integrate support for Monitoring Hub and give our Refresh History experience a major upgrade.



Integration with data pipelines

Data pipelines allow you to group activities that together perform a task. An activity is a unit of work that can be executed. For example, an activity can copy data from one location to another, run a SQL query, execute a stored procedure, or run a Python notebook.

A pipeline can contain one or more activities that are connected by dependencies. For example, you can use a pipeline to ingest and clean data from an Azure blob, and then kick off a Dataflow Gen2 to analyze the log data. You can also use a pipeline to copy data from an Azure blob to an Azure SQL database, and then run a stored procedure on the database.



Save as draft

With Dataflow Gen2, we introduce a worry free experience by removing the need for publishing to save your changes. With save as draft functionality, we store a draft version of your dataflow every time you make a change. Did you lose internet connectivity? Did you accidentally close your browser? No worries; we got your back. Once you return to your dataflow, your recent changes are still there and you can continue where you left off. This is a seamless experience and doesn't require any input from you. This allows you to work on your dataflow without having to worry about losing your changes or having to fix all the query errors before you can save your changes. To learn more about this feature, go to [Save a draft of your dataflow](#).

High scale compute

Similar to Dataflow Gen1, Dataflow Gen2 also features an enhanced compute engine to improve performance of both transformations of referenced queries and get data scenarios. To achieve this, Dataflow Gen2 creates both Lakehouse and Warehouse items in your workspace, and uses them to store and access data to improve performance for all your dataflows.

Licensing Dataflow Gen1 vs Gen2

Dataflow Gen2 is the new generation of dataflows that resides alongside the Power BI dataflow (Gen1) and brings new features and improved experiences. To understand better how licensing works for dataflows you can read the following article: Microsoft Fabric concepts and licenses

Try out Dataflow Gen2 by reusing your queries from Dataflow Gen1

You probably have many Dataflow Gen1 queries and you're wondering how you can try them out in Dataflow Gen2. We have a few options for you to recreate your Gen1 dataflows as Dataflow Gen2.

- Export your Dataflow Gen1 queries and import them into Dataflow Gen2

You can now export queries in both the Dataflow Gen1 and Gen2 authoring experiences and save them to a PQT file you can then import into Dataflow Gen2. For more information, go to [Use the export template feature](#).
- Copy and paste in Power Query

If you have a dataflow in Power BI or Power Apps, you can copy your queries and paste them in the editor of your Dataflow Gen2. This functionality allows you to migrate your dataflow to Gen2 without having to rewrite your queries. For more information, go to [Copy and paste existing Dataflow Gen1 queries](#).

Next steps

- Dataflows refresh history and monitoring
- Dataflows save as draft
- Move queries from Dataflow Gen1 to Dataflow Gen2

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Troubleshoot pipelines for Data Factory in Microsoft Fabric

Article • 11/17/2023

This article explores common troubleshooting methods for external control activities for Data Factory in Microsoft Fabric.

Connector and copy activity

For connector issues such as an encounter error using the copy activity, refer to the Troubleshoot Connectors article.

Azure Databricks

Error code: 3200

- **Message:** Error 403.
- **Cause:** The Databricks access token has expired.
- **Recommendation:** By default, the Azure Databricks access token is valid for 90 days. Create a new token and update the connection.

Error code: 3201

- **Message:** Missing required field: settings.task.notebook_task.notebook_path.
 - **Cause:** Bad authoring: Notebook path not specified correctly.
 - **Recommendation:** Specify the notebook path in the Databricks activity.
-
- **Message:** Cluster... does not exist.
 - **Cause:** Authoring error: Databricks cluster does not exist or has been deleted.
 - **Recommendation:** Verify that the Databricks cluster exists.

- **Message:** Invalid Python file URI... Please visit Databricks user guide for supported URI schemes.
 - **Cause:** Bad authoring.
 - **Recommendation:** Specify either absolute paths for workspace-addressing schemes, or dbfs:/folder/subfolder/foo.py for files stored in the Databricks File System (DFS).
-
- **Message:** {0} LinkedService should have domain and accessToken as required properties.
 - **Cause:** Bad authoring.
 - **Recommendation:** Verify the connection definition.
-
- **Message:** {0} LinkedService should specify either existing cluster ID or new cluster information for creation.
 - **Cause:** Bad authoring.
 - **Recommendation:** Verify the [connection definition]/azure/data-factory/compute-linked-services#azure-databricks-linked-service).
-
- **Message:** Node type Standard_D16S_v3 is not supported. Supported node types:
Standard_DS3_v2, Standard_DS4_v2, Standard_DS5_v2, Standard_D8s_v3,
Standard_D16s_v3, Standard_D32s_v3, Standard_D64s_v3, Standard_D3_v2,
Standard_D8_v3, Standard_D16_v3, Standard_D32_v3, Standard_D64_v3,
Standard_D12_v2, Standard_D13_v2, Standard_D14_v2, Standard_D15_v2,
Standard_DS12_v2, Standard_DS13_v2, Standard_DS14_v2, Standard_DS15_v2,
Standard_E8s_v3, Standard_E16s_v3, Standard_E32s_v3, Standard_E64s_v3,
Standard_L4s, Standard_L8s, Standard_L16s, Standard_L32s, Standard_F4s,
Standard_F8s, Standard_F16s, Standard_H16, Standard_F4s_v2, Standard_F8s_v2,
Standard_F16s_v2, Standard_F32s_v2, Standard_F64s_v2, Standard_F72s_v2,
Standard_NC12, Standard_NC24, Standard_NC6s_v3, Standard_NC12s_v3,

```
Standard_NC24s_v3, Standard_L8s_v2, Standard_L16s_v2, Standard_L32s_v2,  
Standard_L64s_v2, Standard_L80s_v2.
```

- **Cause:** Bad authoring.
- **Recommendation:** Refer to the error message.

Error code: 3202

- **Message:** There were already 1000 jobs created in past 3600 seconds, exceeding rate limit: 1000 job creations per 3600 seconds.
- **Cause:** Too many Databricks runs in an hour.
- **Recommendation:** Check all pipelines that use this Databricks workspace for their job creation rate. If pipelines launched too many Databricks runs in aggregate, migrate some pipelines to a new workspace.
- **Message:** Could not parse request object: Expected 'key' and 'value' to be set for JSON map field base_parameters, got 'key: "..."' instead.
- **Cause:** Authoring error: No value provided for the parameter.
- **Recommendation:** Inspect the pipeline JSON and ensure all parameters in the baseParameters notebook specify a nonempty value.
- **Message:** User: SimpleUserContext{userId=..., name=user@company.com, orgId=...} is not authorized to access cluster.
- **Cause:** The user who generated the access token isn't allowed to access the Databricks cluster specified in the connection.
- **Recommendation:** Ensure the user has the required permissions in the workspace.
- **Message:** Job is not fully initialized yet. Please retry later.
- **Cause:** The job hasn't initialized.

- **Recommendation:** Wait and try again later.

Error code: 3203

- **Message:** The cluster is in Terminated state, not available to receive jobs.
Please fix the cluster or retry later.
- **Cause:** The cluster was terminated. For interactive clusters, this issue might be a race condition.
- **Recommendation:** To avoid this error, use job clusters.

Error code: 3204

- **Message:** Job execution failed.
- **Cause:** Error messages indicate various issues, such as an unexpected cluster state or a specific activity. Often, no error message appears.
- **Recommendation:** N/A

Error code: 3208

- **Message:** An error occurred while sending the request.
- **Cause:** The network connection to the Databricks service was interrupted.
- **Recommendation:** If you're using a self-hosted Data Factory runtime, make sure that the network connection is reliable from the Data Factory runtime nodes. If you're using Azure Data Factory runtime, retry usually works.

The Boolean run output starts coming as string instead of expected int

- **Symptoms:** Your Boolean run output starts coming as string (for example, "0" or "1") instead of expected int (for example, 0 or 1).



You noticed this change on September 28, 2021 at around 9 AM IST when your pipeline relying on this output started failing. No change was made on the pipeline, and the Boolean output data arrived as expected prior to the failure.



- **Cause:** This issue is caused by a recent change, which is by design. After the change, if the result is a number that starts with zero, Data Factory will convert the number to the octal value, which is a bug. This number is always 0 or 1, which never caused issues before the change. So to fix the octal conversion, the string output is passed from the Notebook run as is.
- **Recommendation:** Change the `if` condition to something like `if(value=="0")`.

Functions

Error code: 3602

- **Message:** `Invalid HttpMethod: '%method;'`.
- **Cause:** The `Httpmethod` specified in the activity payload isn't supported by Azure Function Activity.
- **Recommendation:** The supported `Httpmethods` are: PUT, POST, GET, DELETE, OPTIONS, HEAD, and TRACE.

Error code: 3603

- **Message:** `Response Content is not a valid JObject.`

- **Cause:** The Azure function that was called didn't return a JSON Payload in the response. Data Factory and Synapse pipeline Azure function activity only support JSON response content.
- **Recommendation:** Update the Azure function to return a valid JSON Payload such as a C# function may return `(ActionResult)new OkObjectResult("{"Id": "123"}");`

Error code: 3606

- **Message:** Azure function activity missing function key.
- **Cause:** The Azure function activity definition isn't complete.
- **Recommendation:** Check that the input Azure function activity JSON definition has a property named `functionKey`.

Error code: 3607

- **Message:** Azure function activity missing function name.
- **Cause:** The Azure function activity definition isn't complete.
- **Recommendation:** Check that the input Azure function activity JSON definition has a property named `functionName`.

Error code: 3608

- **Message:** Call to provided Azure function '%FunctionName;' failed with status-
'%statusCode;' and message - '%message;' .
- **Cause:** The Azure function details in the activity definition may be incorrect.
- **Recommendation:** Fix the Azure function details and try again.

Error code: 3609

- **Message:** Azure function activity missing functionAppUrl.
- **Cause:** The Azure function activity definition isn't complete.
- **Recommendation:** Check that the input Azure Function activity JSON definition has a property named `functionAppUrl`.

Error code: 3610

- **Message:** There was an error while calling endpoint.
- **Cause:** The function URL may be incorrect.
- **Recommendation:** Verify that the value for `functionAppUrl` in the activity JSON is correct and try again.

Error code: 3611

- **Message:** Azure function activity missing Method in JSON.
- **Cause:** The Azure function activity definition isn't complete.
- **Recommendation:** Check that the input Azure function activity JSON definition has a property named `method`.

Error code: 3612

- **Message:** Azure function activity missing LinkedService definition in JSON.
- **Cause:** The Azure function activity definition isn't complete.
- **Recommendation:** Check that the input Azure function activity JSON definition has connection details.

Azure Machine Learning

Error code: 4101

- **Message:** AzureMLExecutePipeline activity '%activityName;' has invalid value for property '%propertyName;' .
- **Cause:** Bad format or missing definition of property `%propertyName;` .
- **Recommendation:** Check if the activity `%activityName;` has the property `%propertyName;` defined with correct data.

Error code: 4110

- **Message:** AzureMLExecutePipeline activity missing LinkedService definition in JSON.
- **Cause:** The AzureMLExecutePipeline activity definition isn't complete.
- **Recommendation:** Check that the input AzureMLExecutePipeline activity JSON definition has correctly connection details.

Error code: 4111

- **Message:** AzureMLExecutePipeline activity has wrong LinkedService type in JSON. Expected LinkedService type: '%expectedLinkedServiceType;', current LinkedService type: Expected LinkedService type: '%currentLinkedServiceType;' .
- **Cause:** Incorrect activity definition.
- **Recommendation:** Check that the input AzureMLExecutePipeline activity JSON definition has correctly connection details.

Error code: 4112

- **Message:** AzureMLService connection has invalid value for property '%propertyName; '.
- **Cause:** Bad format or missing definition of property '%propertyName; '.
- **Recommendation:** Check if the connection has the property %propertyName; defined with correct data.

Error code: 4121

- **Message:** Request sent to Azure Machine Learning for operation '%operation;' failed with http status code '%statusCode; '. Error message from Azure Machine Learning: '%externalMessage; '.
- **Cause:** The Credential used to access Azure Machine Learning has expired.
- **Recommendation:** Verify that the credential is valid and retry.

Error code: 4122

- **Message:** Request sent to Azure Machine Learning for operation '%operation;' failed with http status code '%statusCode;'. Error message from Azure Machine Learning: '%externalMessage;' .
- **Cause:** The credential provided in Azure Machine Learning connection is invalid, or doesn't have permission for the operation.
- **Recommendation:** Verify that the credential in connection is valid, and has permission to access Azure Machine Learning.

Error code: 4123

- **Message:** Request sent to Azure Machine Learning for operation '%operation;' failed with http status code '%statusCode;'. Error message from Azure Machine Learning: '%externalMessage;' .
- **Cause:** The properties of the activity such as `pipelineParameters` are invalid for the Azure Machine Learning (ML) pipeline.
- **Recommendation:** Check that the value of activity properties matches the expected payload of the published Azure Machine Learning pipeline specified in connection.

Error code: 4124

- **Message:** Request sent to Azure Machine Learning for operation '%operation;' failed with http status code '%statusCode;'. Error message from Azure Machine Learning: '%externalMessage;' .
- **Cause:** The published Azure Machine Learning pipeline endpoint doesn't exist.
- **Recommendation:** Verify that the published Azure Machine Learning pipeline endpoint specified in connection exists in Azure Machine Learning.

Error code: 4125

- **Message:** Request sent to Azure Machine Learning for operation '%operation;' failed with http status code '%statusCode;'. Error message from Azure Machine Learning: '%externalMessage;' .
- **Cause:** There is a server error on Azure Machine Learning.

- **Recommendation:** Retry later. Contact the Azure Machine Learning team for help if the issue continues.

Error code: 4126

- **Message:** Azure ML pipeline run failed with status: '%amlPipelineRunStatus;'.
Azure ML pipeline run Id: '%amlPipelineRunId;'. Please check in Azure Machine Learning for more error logs.
- **Cause:** The Azure Machine Learning pipeline run failed.
- **Recommendation:** Check Azure Machine Learning for more error logs, then fix the ML pipeline.

Common

Error code: 2103

- **Message:** Please provide value for the required property '%propertyName;' .
- **Cause:** The required value for the property has not been provided.
- **Recommendation:** Provide the value from the message and try again.

Error code: 2104

- **Message:** The type of the property '%propertyName;' is incorrect.
- **Cause:** The provided property type isn't correct.
- **Recommendation:** Fix the type of the property and try again.

Error code: 2105

- **Message:** An invalid json is provided for property '%propertyName;'.
Encountered an error while trying to parse: '%message;' .
- **Cause:** The value for the property is invalid or isn't in the expected format.
- **Recommendation:** Refer to the documentation for the property and verify that the value provided includes the correct format and type.

Error code: 2106

- **Message:** The storage connection string is invalid. %errorMessage;
- **Cause:** The connection string for the storage is invalid or has incorrect format.
- **Recommendation:** Go to the Azure portal and find your storage, then copy-and-paste the connection string into your connection and try again.

Error code: 2110

- **Message:** The connection type '%linkedServiceType;' is not supported for '%executorType;' activities.
- **Cause:** The connection specified in the activity is incorrect.
- **Recommendation:** Verify that the connection type is one of the supported types for the activity.

Error code: 2111

- **Message:** The type of the property '%propertyName;' is incorrect. The expected type is %expectedType;.
- **Cause:** The type of the provided property isn't correct.
- **Recommendation:** Fix the property type and try again.

Error code: 2112

- **Message:** The cloud type is unsupported or could not be determined for storage from the EndpointSuffix '%endpointSuffix;'.
- **Cause:** The cloud type is unsupported or couldn't be determined for storage from the EndpointSuffix.
- **Recommendation:** Use storage in another cloud and try again.

Azure Batch

The following table applies to Azure Batch.

Error code: 2500

- **Message:** Hit unexpected exception and execution failed.
- **Cause:** Can't launch command, or the program returned an error code.
- **Recommendation:** Ensure that the executable file exists. If the program started, verify that *stdout.txt* and *stderr.txt* were uploaded to the storage account. It's a good practice to include logs in your code for debugging.

Error code: 2501

- **Message:** Cannot access user batch account; please check batch account settings.
- **Cause:** Incorrect Batch access key or pool name.
- **Recommendation:** Verify the pool name and the Batch access key in the connection.

Error code: 2502

- **Message:** Cannot access user storage account; please check storage account settings.
- **Cause:** Incorrect storage account name or access key.
- **Recommendation:** Verify the storage account name and the access key in the connection.

Error code: 2504

- **Message:** Operation returned an invalid status code 'BadRequest'.
- **Cause:** Too many files in the `FolderPath` of the Azure activity. The total size of `resourceFiles` can't be more than 32,768 characters.
- **Recommendation:** Remove unnecessary files, or Zip them and add an unzip command to extract them.

For example, use `powershell.exe -nologo -noprofile -command "& { Add-Type -A 'System.IO.Compression.FileSystem';`

```
[IO.Compression.ZipFile]::ExtractToDirectory($zipFile, $folder); }" ;  
$folder\yourProgram.exe
```

Error code: 2505

- **Message:** Cannot create Shared Access Signature unless Account Key credentials are used.
- **Cause:** Azure Batch activities support only storage accounts that use an access key.
- **Recommendation:** Refer to the error description.

Error code: 2507

- **Message:** The folder path does not exist or is empty: ...
- **Cause:** No files are in the storage account at the specified path.
- **Recommendation:** The folder path must contain the executable files you want to run.

Error code: 2508

- **Message:** There are duplicate files in the resource folder.
- **Cause:** Multiple files of the same name are in different subfolders of folderPath.
- **Recommendation:** Azure Batch activities flatten folder structure under folderPath. If you need to preserve the folder structure, zip the files and extract them in Azure Batch by using an unzip command.

For example, use `powershell.exe -nologo -noprofile -command "& { Add-Type -A 'System.IO.Compression.FileSystem'; [IO.Compression.ZipFile]::ExtractToDirectory($zipFile, $folder); }" ; $folder\yourProgram.exe`

Error code: 2509

- **Message:** Batch url ... is invalid; it must be in Uri format.
- **Cause:** Batch URLs must be similar to
`https://mybatchaccount.eastus.batch.azure.com`

- **Recommendation:** Refer to the error description.

Error code: 2510

- **Message:** An error occurred while sending the request.
- **Cause:** The batch URL is invalid.
- **Recommendation:** Verify the batch URL.

Web activity

Error Code: 2001

- **Message:** The length of execution output is over limit (around 4MB currently).
- **Cause:** The execution output is greater than 4 MB in size but the maximum supported output response payload size is 4 MB.
- **Recommendation:** Make sure the execution output size does not exceed 4 MB. For more information, see How to scale out the size of data moving using Data Factory.

Error Code: 2002

- **Message:** The payload including configurations on activity/data/connection is too large. Please check if you have settings with very large value and try to reduce its size.
- **Cause:** The payload you are attempting to send is too large.
- **Recommendation:** Refer to Payload is too large.

Error Code: 2003

- **Message:** There are substantial concurrent external activity executions which is causing failures due to throttling under subscription <subscription id>, region <region code> and limitation <current limit>. Please reduce the concurrent executions. For limits, refer <https://aka.ms/adflimits>.

- **Cause:** Too many activities are running concurrently. This can happen when too many pipelines are triggered at once.
- **Recommendation:** Reduce pipeline concurrency. You might have to distribute the trigger time of your pipelines.

Error Code: 2105

- **Message:** The value type '<provided data type>', in key '<key name>' is not expected type '<expected data type>'
- **Cause:** Data generated in the dynamic content expression doesn't match with the key and causes JSON parsing failure.
- **Recommendation:** Look at the key field and fix the dynamic content definition.

Error code: 2108

- **Message:** Error calling the endpoint '<URL>'. Response status code: 'NA - Unknown'. More details: Exception message: 'NA - Unknown [ClientSideException] Invalid Url: <URL>. Please verify Url or Data Factory runtime is valid and retry. Localhost URLs are allowed only with SelfHosted Data Factory runtime'
- **Cause:** Unable to reach the URL provided. This can occur because there was a network connection issue, the URL was unresolvable, or a localhost URL was being used on an Azure Data Factory runtime.
- **Recommendation:** Verify that the provided URL is accessible.

- **Message:** Error calling the endpoint '%url;'. Response status code: '%code;'
- **Cause:** The request failed due to an underlying issue such as network connectivity, a DNS failure, a server certificate validation, or a timeout.
- **Recommendation:** Use Fiddler/Postman/Netmon/Wireshark to validate the request.

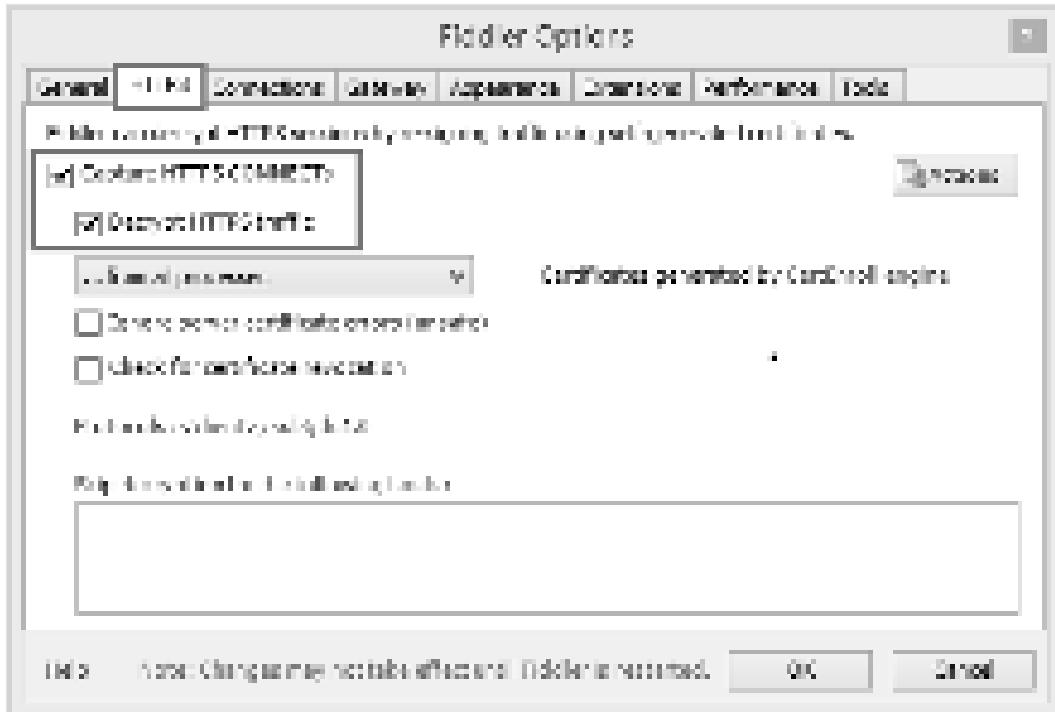
Using Fiddler

To use Fiddler to create an HTTP session of the monitored web application:

1. Download, install, and open Fiddler ↗.

2. If your web application uses HTTPS, go to **Tools > Fiddler Options > HTTPS**.

- In the HTTPS tab, select both **Capture HTTPS CONNECTs** and **Decrypt HTTPS traffic**.



3. If your application uses TLS/SSL certificates, add the Fiddler certificate to your device.

Go to: **Tools > Fiddler Options > HTTPS > Actions > Export Root Certificate to Desktop**.

4. Turn off capturing by going to **File > Capture Traffic**. Or press **F12**.

5. Clear your browser's cache so that all cached items are removed and must be downloaded again.

6. Create a request:

7. Select the **Composer** tab.

- Set the HTTP method and URL.

- If needed, add headers and a request body.

- Select **Execute**.

8. Turn on traffic capturing again, and complete the problematic transaction on your page.

9. Go to: **File > Save > All Sessions**.

For more information, see [Getting started with Fiddler](#).

Error Code: 2113

- **Message:** ExtractAuthorizationCertificate: Unable to generate a certificate from a Base64 string/password combination
- **Cause:** Unable to generate certificate from Base64 string/password combination.
- **Recommendation:** Verify that the Base64 encoded PFX certificate and password combination you are using are correctly entered.

Error Code: 2403

- **Message:** Get access token from MSI failed for Datafactory <DF mname>, region <region code>. Please verify resource url is valid and retry.
- **Cause:** Unable to acquire an access token from the resource URL provided.
- **Recommendation:** Verify that you have provided the correct resource URL for your managed identity.

General

REST continuation token NULL error

Error message: {"token":null,"range":{ "min":..}}

Cause: When querying across multiple partitions/pages, backend service returns continuation token in JObject format with 3 properties: **token, min and max key ranges**, for instance, {"token":null,"range": {""min": "05C1E9AB0DAD76","max": "05C1E9CD673398"}). Depending on source data, querying can result 0 indicating missing token though there is more data to fetch.

Recommendation: When the continuationToken is non-null, as the string {"token":null,"range":{ "min": "05C1E9AB0DAD76","max": "05C1E9CD673398"}}, it is required to call queryActivityRuns API again with the continuation token from the previous response. You need to pass the full string for the query API again. The activities will be returned in the subsequent pages for the query result. You should ignore that there is empty array in this page, as long as the full continuationToken value != null, you

need continue querying. For more details, please refer to REST api for pipeline run query.

Activity stuck issue

When you observe that the activity is running much longer than your normal runs with barely no progress, it may happen to be stuck. You can try canceling it and retry to see if it helps. If it's a copy activity, you can learn about the performance monitoring and troubleshooting from Troubleshoot copy activity performance; if it's a data flow, learn from Mapping data flows performance and tuning guide.

Payload is too large

Error message: The payload including configurations on activity/data/connection is too large. Please check if you have settings with very large value and try to reduce its size.

Cause: The payload for each activity run includes the activity configuration, the associated data(s), and connection(s) configurations if any, and a small portion of system properties generated per activity type. The limit of such payload size is 896 KB as mentioned in the Azure limits documentation for Data Factory and Azure Synapse Analytics.

Recommendation: You hit this limit likely because you pass in one or more large parameter values from either upstream activity output or external, especially if you pass actual data across activities in control flow. Check if you can reduce the size of large parameter values, or tune your pipeline logic to avoid passing such values across activities and handle it inside the activity instead.

Unsupported compression causes files to be corrupted

Symptoms: You try to unzip a file that is stored in a blob container. A single copy activity in a pipeline has a source with the compression type set to "deflate64" (or any unsupported type). This activity runs successfully and produces the text file contained in the zip file. However, there is a problem with the text in the file, and this file appears corrupted. When this file is unzipped locally, it is fine.

Cause: Your zip file is compressed by the algorithm of "deflate64", while the internal zip library of Data Factory only supports "deflate". If the zip file is compressed by the Windows system and the overall file size exceeds a certain number, Windows will use "deflate64" by default, which is not supported in Data Factory. On the other hand, if the

file size is smaller or you use some third party zip tools that support specifying the compress algorithm, Windows will use "deflate" by default.

Tip

Actually, both Binary format in Data Factory and Synapse Analytics and Delimited text format in Data Factory and Azure Synapse Analytics clearly state that the "deflate64" format is not supported in Data Factory.

Execute Pipeline passes array parameter as string to the child pipeline

Error message: Operation on target ForEach1 failed: The execution of template action 'MainForEach1' failed: the result of the evaluation of 'foreach' expression '@pipeline().parameters.<parameterName>' is of type 'String'. The result must be a valid array.

Cause: Even if in the Execute Pipeline you create the parameter of type array, as shown in the below image, the pipeline will fail.



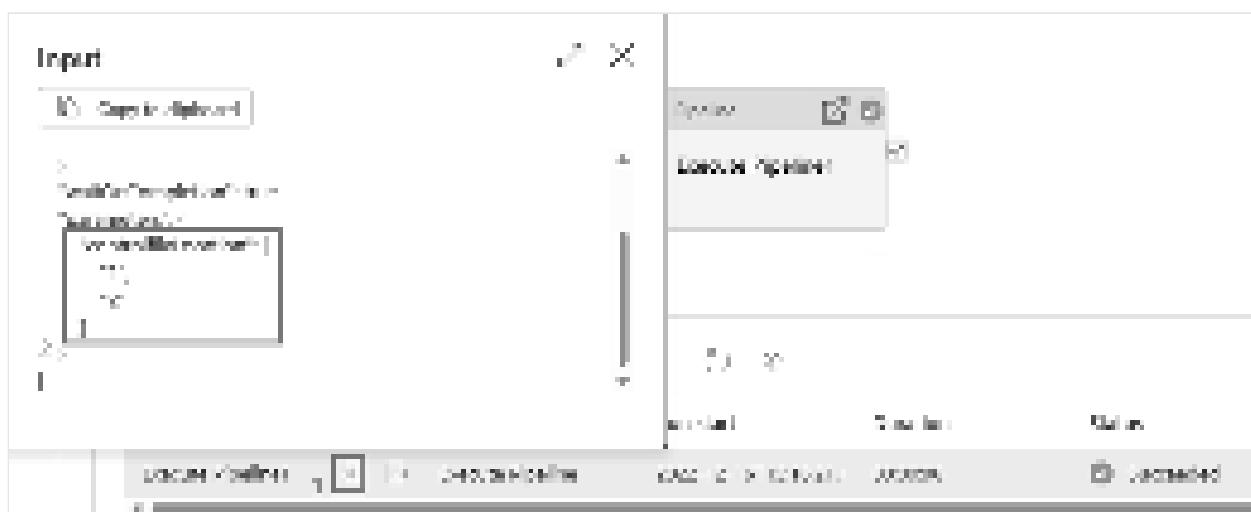
This is due to the fact that the payload is passed from the parent pipeline to the child as string. We can see it when we check the input passed to the child pipeline.



Recommendation: To solve the issue we can leverage the create array function as shown in the below image.



Then our pipeline will succeed. And we can see in the input box that the parameter passed is an array.



Next steps

For more troubleshooting help, try these resources:

- Data Factory blog ↗
 - Data Factory feature requests
 - Stack Overflow forum for Data Factory ↗
 - Twitter information about Data Factory ↗
 - Azure videos ↗
 - Microsoft Q&A question page
-

Feedback

Was this page helpful?



Yes



No

Provide product feedback ↗ | Ask the community ↗

Troubleshoot the Azure Data Explorer connector in Data Factory in Microsoft Fabric

Article • 11/15/2023

This article provides suggestions to troubleshoot common problems with the Azure Data Explorer connector in Data Factory in Microsoft Fabric.

Error code:

KustoMappingReferenceHasWrongKind

- **Message:** Mapping reference should be of kind 'Csv'. Mapping reference: '%reference;'. Kind '%kind;' .
- **Cause:** The ingestion mapping reference is not CSV type.
- **Recommendation:** Create a CSV ingestion mapping reference.

Error code: KustoWriteFailed

- **Message:** Write to Kusto failed with following error: '%message;' .
- **Cause:** Wrong configuration or transient errors when the destination reads data from the source.
- **Recommendation:** For transient failures, set retries for the activity. For permanent failures, check your configuration and contact support.

Next steps

For more troubleshooting help, try these resources:

- Data Factory blog ↗
- Data Factory community ↗
- Data Factory feature requests ideas ↗

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Troubleshoot the Azure Database for PostgreSQL connector in Data Factory in Microsoft Fabric

Article • 11/15/2023

This article provides suggestions to troubleshoot common problems with the Azure Database for PostgreSQL connector in Data Factory in Microsoft Fabric.

Error code:

AzurePostgreSqlNpgsqlDataTypeNotSupported

- **Message:** The data type of the chosen Partition Column, '%partitionColumn;', is '%dataType;' and this data type is not supported for partitioning.
- **Recommendation:** Pick a partition column with int, bigint, smallint, serial, bigserial, smallserial, timestamp with or without time zone, time without time zone or date data type.

Error code:

AzurePostgreSqlNpgsqlPartitionColumnNameNotProvided

- **Message:** Partition column name must be specified.
- **Cause:** No partition column name is provided, and it couldn't be decided automatically.

Next steps

For more troubleshooting help, try these resources:

- Data Factory blog ↗
- Data Factory community ↗
- Data Factory feature requests ideas ↗

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback ↗ | Ask the community ↗

Troubleshoot the Azure Table Storage connector in Data Factory in Microsoft Fabric

Article • 11/15/2023

This article provides suggestions to troubleshoot common problems with the Azure Table Storage connector in Data Factory in Microsoft Fabric.

Error code:

AzureTableDuplicateColumnsFromSource

- **Message:** Duplicate columns with same name '%name;' are detected from source.
This is NOT supported by Azure Table Storage destination.
- **Cause:** Duplicated source columns might occur for one of the following reasons:
 - You're using the database as a source and applied table joins.
 - You have unstructured CSV files with duplicated column names in the header row.
- **Recommendation:** Double-check and fix the source columns, as necessary.

Next steps

For more troubleshooting help, try these resources:

- Data Factory blog ↗
- Data Factory community ↗
- Data Factory feature requests ideas ↗

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback ↗ | Ask the community ↗

Troubleshoot the delimited text format connector in Data Factory in Microsoft Fabric

Article • 12/22/2023

This article provides suggestions to troubleshoot common problems with the delimited text format connector in Data Factory in Microsoft Fabric.

Error code: DelimitedTextBadDataDetected

- **Message:** `Bad data is found at line %rowIndex; in source %fileName;.`
- **Cause:** Invalid delimited text data.
- **Recommendation:** Ignore the bad data by setting **Fault tolerance** to **Skip incompatible rows** in the copy activity.

Error code:

DelimitedTextColumnNameNotAllowed

- **Message:** `The name of column index %index; is empty. Make sure column name is properly specified in the header row.`
- **Cause:** When 'firstRowAsHeader' is set in the activity, the first row is used as the column name. This error means that the first row contains an empty value (for example, 'ColumnA, ColumnB').
- **Recommendation:** Check the first row, and fix the value if it is empty.

Error code:

DelimitedTextMoreColumnsThanDefined

- **Message:** `Error found when processing '%function;' source '%name;' with row number %rowCount;: found more columns than expected column count: %expectedColumnCount;.`
- **Causes and recommendations:** Different causes may lead to this error. Check below list for possible cause analysis and related recommendation.

Cause analysis	Recommendation
The problematic row's column count is larger than the first row's column count. It might be caused by a data issue or incorrect column delimiter or quote char settings.	Get the row count from the error message, check the row's column, and fix the data.
If the expected column count is "1" in an error message, you might have specified wrong compression or format settings, which caused the files to be parsed incorrectly.	Check the format settings to make sure they match your source files.
If your source is a folder, the files under the specified folder might have a different schema.	Make sure that the files in the specified folder have an identical schema.

Next steps

For more troubleshooting help, try these resources:

- Data Factory blog [↗](#)
- Data Factory community [↗](#)
- Data Factory feature requests ideas [↗](#)

Feedback

Was this page helpful?



Yes



No

Provide product feedback [↗](#) | Ask the community [↗](#)

Troubleshoot the XML format connector in Data Factory in Microsoft Fabric

Article • 11/15/2023

This article provides suggestions to troubleshoot common problems with the XML format connector in Data Factory in Microsoft Fabric.

Error code: XmlSinkNotSupportedException

- **Message:** Write data in XML format is not supported yet, choose a different format!
- **Cause:** XML data was used as destination data in your copy activity.
- **Recommendation:** Use data in a different format from that of the destination data.

Error code: XmlAttributeNameConflict

- **Message:** Column names %attrNames; for attributes of element '%element;' conflict with that for corresponding child elements, and the attribute prefix used is '%prefix;' .
- **Cause:** An attribute prefix was used, which caused the conflict.
- **Recommendation:** Set a different value for the "attributePrefix" property.

Error code: XmlValueColumnNameConflict

- **Message:** Column name for the value of element '%element;' is '%columnName;' and it conflicts with the child element having the same name.
- **Cause:** One of the child element names was used as the column name for the element value.
- **Recommendation:** Set a different value for the "valueColumn" property.

Error code: XmlInvalid

- **Message:** Input XML file '%file;' is invalid with parsing error '%error;' .

- **Cause:** The input XML file is not well formed.
- **Recommendation:** Correct the XML file to make it well formed.

Next steps

For more troubleshooting help, try these resources:

- Data Factory blog ↗
- Data Factory community ↗
- Data Factory feature requests ideas ↗

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback ↗ | Ask the community ↗

Troubleshoot the REST connector in Data Factory in Microsoft Fabric

Article • 11/15/2023

This article provides suggestions to troubleshoot common problems with the REST connector in Data Factory in Microsoft Fabric.

Error code: RestSinkCallFailed

- **Message:** Rest Endpoint responded with Failure from server. Check the error from server:%message;
- **Cause:** This error occurs when a data factory or Synapse pipeline talks to the REST endpoint over HTTP protocol, and the request operation fails.
- **Recommendation:** Check the HTTP status code or the message in the error message and fix the remote server issue.

Error code: RestSourceCallFailed

- **Message:** The HttpStatusCode %statusCode; indicates failure.
Request URL: %requestUri;
Response payload:%payload;
- **Cause:** This error occurs when Data Factory talks to the REST endpoint over HTTP protocol, and the request operation fails.
- **Recommendation:** Check the HTTP status code or the request URL or the response payload in the error message and fix the remote server issue.

Error code: RestSinkUNSupportedCompressionType

- **Message:** User Configured CompressionType is Not Supported By Data Factory: %message;
- **Recommendation:** Check the supported compression types for the REST destination.

Unexpected network response from the REST connector

- **Symptoms:** The endpoint sometimes receives an unexpected response (400, 401, 403, 500) from the REST connector.
- **Cause:** The REST source connector uses the URL and HTTP method/header/body from the connection/data/copy source as parameters when it constructs an HTTP request. The issue is most likely caused by some mistakes in one or more specified parameters.
- **Resolution:**
 - Use 'curl' in a Command Prompt window to see whether the parameter is the cause (**Accept** and **User-Agent** headers should always be included):

```
curl -i -X <HTTP method> -H <HTTP header1> -H <HTTP header2> -H "Accept:  
application/json" -H "User-Agent: azure-data-factory/2.0" -d '<HTTP body>'  
<URL>
```

If the command returns the same unexpected response, fix the preceding parameters with 'curl' until it returns the expected response.

You can also use 'curl--help' for more advanced usage of the command.

- If only the REST connector returns an unexpected response, contact Microsoft support for further troubleshooting.
- Note that 'curl' might not be suitable to reproduce an SSL certificate validation issue. In some scenarios, the 'curl' command was executed successfully without encountering any SSL certificate validation issues. But when the same URL is executed in a browser, no SSL certificate is actually returned for the client to establish trust with server.

Tools like **Postman** and **Fiddler** are recommended for the preceding case.

Next steps

For more troubleshooting help, try these resources:

- Data Factory blog ↗
- Data Factory community ↗
- Data Factory feature requests ideas ↗

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Troubleshoot the SharePoint Online list connector in Data Factory in Microsoft Fabric

Article • 11/15/2023

This article provides suggestions to troubleshoot common problems with the SharePoint Online list connector in Data Factory in Microsoft Fabric.

Error code: SharePointOnlineAuthFailed

- **Message:** The access token generated failed, status code: %code;, error message: %message;.
- **Cause:** The service principal ID and key might not be set correctly.
- **Recommendation:** Check your registered application (service principal ID) and key to see whether they're set correctly.

Connection failed after granting permission in SharePoint Online List

Symptoms

You granted permission to your data factory in SharePoint Online List, but you still fail with the following error message:

```
Failed to get metadata of odata service, please check if service url and credential is correct and your application has permission to the resource. Expected status code: 200, actual status code: Unauthorized, response is : {"error":"invalid_request","error_description":"Token type is not allowed."}.
```

Cause

The SharePoint Online List uses ACS to acquire the access token to grant access to other applications. But for the tenant built after November 7, 2018, ACS is disabled by default.

Recommendation

You need to enable ACS to acquire the access token. Take the following steps:

1. Download SharePoint Online Management Shell [↗](#), and ensure that you have a tenant admin account.
2. Run the following command in the SharePoint Online Management Shell. Replace <tenant name> with your tenant name and add -admin after it.

PowerShell

```
Connect-SPOSERVICE -Url https://<tenant name>-admin.sharepoint.com/
```

3. Enter your tenant admin information in the pop-up authentication window.

4. Run the following command:

PowerShell

```
Set-SPOTenant -DisableCustomAppAuthentication $false
```



5. Use ACS to get the access token.

Next steps

For more troubleshooting help, try these resources:

- Data Factory blog [↗](#)
- Data Factory community [↗](#)
- Data Factory feature requests ideas [↗](#)

Feedback

Was this page helpful?

Yes

No

Provide product feedback [↗](#) | Ask the community [↗](#)

Data Factory limitations overview

Article • 11/15/2023

There are certain limitations to the current Data Factory in Microsoft Fabric features. Before submitting a support request, review the lists in this section to determine if you're experiencing a known limitation.

For service level outages or degradation notifications, check Microsoft Fabric support [↗](#).

Currently active limitations

The following links go to articles that discuss known limitations for Data Factory pipelines and Dataflow Gen2. Select the appropriate link to determine if the limitation is known for a specific feature.

- Pipeline limitations
- Pipeline resource limits
- Dataflow Gen2 limitations

Next steps

- Service level outages [↗](#)
- Get your questions answered by the Data Factory community [↗](#)

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback [↗](#) | Ask the community [↗](#)

Data Factory data pipeline limitations in Microsoft Fabric

Article • 11/15/2023

The following list describes the current limitations of pipelines in Data Factory in Microsoft Fabric.

- Most of the Azure Data Factory copy and orchestration patterns are applicable to Fabric pipelines, but tumbling window and event triggers aren't yet available.
- Pipelines don't support Continuous Integration and Continuous Delivery (CI/CD).
- Connectors don't support OAuth, Azure key vault (AKV), and Managed System Identity (MSI).
- Connectors can't use parameters.
- GetMetaData activity can't have a source from Fabric KQL databases.
- Script activity can't have a source from Fabric KQL databases.
- Copy activity uses a Web connector, whereas Web/Webhook activities use a Web v2 connector that supports richer functionality, like audience and resource URI.
- Custom activities aren't available in Fabric pipelines. Use Azure Batch activity instead.
- Data pipelines are scoped to their workspace, and can't interact with items in other workspaces.
- The on-premises data gateway can be used with Dataflow Gen2 to ingest on-premises data now. You can orchestrate on-premises data ingestion with a data flow activity in the pipeline.
- Pipelines can't use a managed VNet.
- Web activity does not support service principal based authentication.

Next steps

For information about the resource limitations for data pipelines, see [Data pipeline resource limits](#).

Feedback

Was this page helpful?



Yes



No

Data pipeline resource limits

Article • 11/15/2023

The following table describes the resource limitations for pipelines in Data Factory in Microsoft Fabric.

Pipeline Resource	Default limit	Maximum limit
Total number of entities, such as pipelines, activities, triggers within a workspace	5,000	5,000
Concurrent pipeline runs per data factory that's shared among all pipelines in workspace	10,000	10,000
External activities like stored procedure, Web, Web Hook, and others	3,000	3,000
Pipeline activities execute on integration runtime, including Lookup, GetMetadata, and Delete	1,000	1,000
Concurrent authoring operations, including test connection, browse folder list and table list, preview data, and so on	200	200
Maximum activities per pipeline, which includes inner activities for containers	40	40
Maximum parameters per pipeline	50	50
ForEach items	100,000	100,000
ForEach parallelism	20	50
Lookup Activity item count	5000	5000
Maximum queued runs per pipeline	100	100
Characters per expression	8,192	8,192
Maximum timeout for pipeline activity runs	24 hours	24 hours
Bytes per object for pipeline objects	200 KB	200 KB
Bytes per payload for each activity run	896 KB	896 KB
Data Integration Units per copy activity run	Auto	256
Write API calls	1,200/h	1,200/h
Read API calls	12,500/h	12,500/h

Pipeline Resource	Default limit	Maximum limit
Maximum time of data flow debug session	8 hrs	8 hrs
Meta Data Entity Size limit in a factory	2 GB	2 GB

For information about limitations for data pipelines, go to [Pipeline limitations](#).

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Data Factory Dataflow Gen2 limitations

Article • 11/15/2023

The following list describes the limitations for Dataflow Gen2 in Data Factory in Microsoft Fabric.

- Data Factory Fast Copy isn't yet available.
- Data destination to Lakehouse:
 - Spaces or special characters aren't supported in column or table names.
 - Duration and binary columns aren't supported while authoring Dataflow Gen2 dataflows.
- You need to have the latest version of the gateway installed to use with Dataflow Gen2.
- The incremental refresh feature isn't available yet in Dataflow Gen2.
- The Delta Lake specification doesn't support case sensitive column names, so `MyColumn` and `mycolumn`, while supported in Mashup, results in a "duplicate columns" error.
- Dataflows that use a Gateway and the data destination feature are limited to an evaluation or refresh time of one hour. Read more about the gateway considerations when using data destinations.

The following table indicates the supported data types in specific storage locations.

Supported data types per storage location:	Dataflow	Staging	Lakehouse	Azure DB (SQL) Output	Azure Data Explorer Output	Fabric Lakehouse (LH) Output	Fabric Warehouse (WH) Output
Action	No			No	No	No	No
Any	No			No	No	No	No
Binary	No			No	No	No	No
DateTimeZone	Yes			Yes	Yes	No	No
Duration	No			No	Yes	No	No
Function	No			No	No	No	No
None	No			No	No	No	No
Null	No			No	No	No	No
Time	Yes			Yes	No	No	No

Supported data types per storage location:	Dataflow	Staging	Lakehouse	Azure DB (SQL) Output	Azure Data Explorer Output	Fabric Lakehouse (LH) Output	Fabric Warehouse (WH) Output
Type	No			No	No	No	No
Structured (List, Record, Table)	No			No	No	No	No

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Data Factory known issues

Article • 12/11/2023

ⓘ Important

For important considerations when using the output destination feature in Dataflow Gen2 with an on-premises data gateway, check the article on [gateway considerations for output destination](#).

This page lists known issues for Data Factory features. Before submitting a support request, review this list to see if the issue you're experiencing is already known and being addressed.

For service level outages or degradation notifications, check Microsoft Fabric support [↗](#).

Currently active known issues

Select the **Title** to view more information about that specific known issue.

[+] Expand table

Area	Title	Issue publish date
Refreshes through a gateway	Gateway refreshes longer than an hour fail	August 3, 2023
Refresh history reports in progress for some tables that failed	Refresh history reports in progress for some tables that failed	August 24, 2023
Dataflow Gen1 admin switch affects Dataflow Gen2	Dataflow Gen1 admin switch affects Dataflow Gen2	November 6, 2023
Failed to publish Dataflow Gen2	Failed to publish Dataflow Gen2	November 11, 2023

Recently closed known issues

Select the **Title** to view more information about that specific known issue. Fixed issues are removed after 46 days.

 Expand table

Area	Title	Issues publish date	Status
Edit a Dataflow Gen2	Lineage view edit dataflow button takes user to Dataflow Gen1 experience	July 5, 2023	Fixed: September 25, 2023
Output destination in Dataflow Gen2	Staging items aren't available or are misconfigured	June 6, 2023	Fixed: October 31, 2023

Related content

- Service level outages 
- Get your questions answered by the Data Factory community 

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Known issue - Lineage view edit dataflow button

Article • 11/15/2023

A known issue is posted where a customer gets taken to the Dataflow Gen1 experience when selecting the **Edit** button from within the Lineage view of a workspace.

APPLIES TO:  Dataflow Gen2 in Microsoft Fabric

Status: Resolved

Problem area: Data Factory

Symptoms

When opening a Dataflow Gen2 from within the Lineage view of the workspace, selecting the **Edit** button for a Dataflow Gen2 takes you to the Dataflow Gen1 experience where the Output destination entry point and other functionality isn't available.

Solutions and workarounds

Open the Dataflow Gen2 from within the Workspace list view.

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Known issue - Staging items aren't hidden and can be modified by the end-user

Article • 11/15/2023

In a new workspace, when an end-user first launches a Dataflow Gen2, a set of staging items, such as a new Lakehouse and a new Data Warehouse, are automatically created for the whole workspace. These items can be used for a set of functionalities available for Dataflow Gen2.

A user is able to see these staging items in the workspace list and lineage and being able to interact with these items.

If a user deletes or modifies any of these staging items, the workspace can be set in a state that won't be able to use these items because of misconfiguration.

APPLIES TO: ✓ Dataflow Gen2 in Microsoft Fabric

Status: Closed

Problem area: Data Factory

Symptoms

When trying to refresh a Dataflow gen2 and the error refers to:

- *null Error: Couldn't refresh the entity because of an issue with the mashup document
MashupException.Error: PowerBIEntityNotFound: PowerBIEntityNotFound*

After deleting the staging items, you're unable to get them back.

Solutions and workarounds

Recreate your Dataflow Gen2 solution in a new workspace.

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#) | [Ask the community](#)

Known issue - Gateway refreshes longer than an hour fail

Article • 11/15/2023

A known issue is posted where a refresh taking longer than an hour fails with "InvalidConnectionCredentials" or "AccessUnauthorized" error reasons.

APPLIES TO: ✓ Dataflow Gen2 in Microsoft Fabric ✓ On-premises data gateway in Microsoft Fabric ✓ VNet data gateway in Microsoft Fabric

Status: Open

Problem area: Data Factory

Symptoms

When you're using OAuth2 credentials, the gateway currently doesn't support refreshing tokens automatically when access tokens expire (one hour after the refresh started). If you get the errors "InvalidConnectionCredentials" or "AccessUnauthorized" when accessing cloud data sources using OAuth2 credentials even though the credentials have been updated recently, you might be hitting this error. This limitation for long running refreshes exists for both VNet gateways and on-premises data gateways.

Solutions and workarounds

Shorten queries to run in less than an hour.

Feedback

Was this page helpful?



Yes



No

Provide product feedback ↗ | Ask the community ↗

Known issue - In progress refresh history

Article • 11/15/2023

Users are experiencing an intermittent issue. Refresh history reports in progress for some tables while the dataflow has failed.

APPLIES TO:  Dataflow Gen2 in Microsoft Fabric.

Status: Active

Problem area: Data Factory

Symptoms

Failed dataflows show in progress refresh history for some tables, while it's not refreshing anymore.

Solutions and workarounds

Retry to refresh the dataflow again and check the refresh history.

Feedback

Was this page helpful?

 Yes	 No
---	--

Provide product feedback  | Ask the community 

Known issue - Dataflow Gen1 admin switch affects Dataflow Gen2

Article • 11/15/2023

Users aren't able to use Dataflow Gen2 when Dataflow Gen1 admin switch is disabled.

APPLIES TO: ✓ Dataflow Gen2 in Microsoft Fabric.

Status: Active

Problem area: Data Factory

Symptoms

When the Dataflow Gen1 admin switch is disabled, users aren't able to use Dataflow Gen2. When the user tries to create a Dataflow Gen2, the user gets the following error message: Request failed with 401(Unauthorized): Unauthorized.

Solutions and workarounds

Enable the Dataflow Gen1 admin switch to use Dataflow Gen2. For more information, see Dataflow Gen1 admin switch

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback  | Ask the community 

Known issue - Dataflow Gen2 failed to publish

Article • 11/14/2023

Users get the following error message when they try to publish a Dataflow Gen2:
"Something went wrong"

APPLIES TO: ✅ Dataflow Gen2 in Microsoft Fabric.

Status: Active

Problem area: Data Factory

Symptoms

When a user tries to publish a Dataflow Gen2, they get the following error message:
"Something went wrong"

Solutions and workarounds

If you encounter this issue, try the following: Refresh the Dataflow Gen2 and try to publish again.

Feedback

Was this page helpful?



Provide product feedback ↗ | Ask the community ↗