

Use Eventstream in Microsoft Fabric

1. Introduction

<https://learn.microsoft.com/en-us/training/modules/explore-event-streams-microsoft-fabric/1-introduction>

Introduction

Completed

Eventstream in Real-Time Intelligence enables you to capture, transform, and route real-time events without needing to write any code. You can set up event sources, destinations, and transformations in the eventstream, enabling you to collect, then optionally process data before routing it into a destination.

Some of the main benefits of Eventstream in Fabric are:

- You can capture, transform, and route data to various destinations. You can use the drag and drop feature to create your event data processing logic without writing any code.
- There are multiple source connectors to ingest event data from different sources, such as Azure Event Hubs, Azure IoT Hub, Azure storage, Apache Kafka, and others.
- You can send data to a wide range of destinations for storage, further analysis or processing.

In this module, you'll learn how to create eventstreams and use them to ingest and transform data in motion, and route it to destinations for further processing and analysis.

2. Components of Eventstream

<https://learn.microsoft.com/en-us/training/modules/explore-event-streams-microsoft-fabric/2-eventstream-components>

Components of Eventstream

Completed

The Eventstream feature in Fabric works by creating a pipeline that ingests events from streaming data sources, processes them through optional transformations, and delivers them to various destinations. Eventstream is the delivery mechanism that carries events from where they happen to where they need to be processed, analyzed, or acted upon.

You can use the eventstream canvas, which is a visual editor, to design your pipeline by dragging and dropping different nodes, such as sources, transformations, and destinations. You can also see the event data flowing through the pipeline in real-time. You don't need to write any code or manage any infrastructure to use Eventstream.

The screenshot displays the Microsoft Fabric Eventstream canvas. On the left, a sidebar lists 'Data' sources and destinations. The 'Bicycles' source is inactive, while the 'Bikes-by-street-table' destination is active. The main canvas shows a pipeline flow: 'Bicycles' (Source) → 'Bicycle-data' (Eventstream) → 'GroupByStreet' (Transformation) → 'Bikes-by-street-table' (Destination). The 'GroupByStreet' transformation is configured with 'Tumbling' and 'Sum of No_Bikes'. Below the canvas, the 'Data preview' tab is active, showing a table of data.

Street	SUM_No_Bikes	Window_End_Time
St. Luke's Church	19	2025-08-31T21:53:25Z
Chelsea Green	10	2025-08-31T21:53:25Z
Kensington Gore	11	2025-08-31T21:53:25Z
Montpelier Street	16	2025-08-31T21:53:25Z
Cadogan Place	5	2025-08-31T21:53:20Z
Seville Street	12	2025-08-31T21:53:20Z
Sloane Avenue	3	2025-08-31T21:53:20Z
Denyer Street	18	2025-08-31T21:53:20Z
Pont Street	11	2025-08-31T21:53:15Z
Holy Trinity Brompton	4	2025-08-31T21:53:15Z

This image shows the eventstream canvas. There's a real-time data source called **Bicycles**, which includes: city bike rental data including bike locations, bike station street names and more. **Bicycle-data** is an eventstream that ingests data from the **Bicycles** data source. The data is transformed by an operation named **GroupByStreet** that sums the number of bikes by bike station street name. This data is stored in a table in an Eventhouse called **Bikes-by-street-table**.

The main components of an eventstream are:

- **Sources:** Sources are where your event data comes from. You can stream data from Microsoft sources and also ingest data from non-Microsoft platforms.
- **Transformations:** You can transform the data as it flows in an eventstream, enabling you to filter, summarize, and reshape it before storing it. Examples of available transformations include: SQL code, filter, manage fields, aggregate, group by, expand and join.
- **Destinations:** Destinations are where your transformed event data goes for storage, further processing, alerts, or integration with other systems. You can route the data from your stream to various destinations such as tables in an Eventhouse or lakehouse, custom endpoints, derived streams for more processing, or Fabric Activator to trigger actions.

3. Eventstream sources and destinations

<https://learn.microsoft.com/en-us/training/modules/explore-event-streams-microsoft-fabric/3-setup-eventstream>

Eventstream sources and destinations

Completed

Once you create an eventstream in Fabric, you can connect it to a wide range of data sources, optionally transform it, and route the transformed, or processed data to multiple destinations. In this unit, we'll review eventstream sources and destinations.

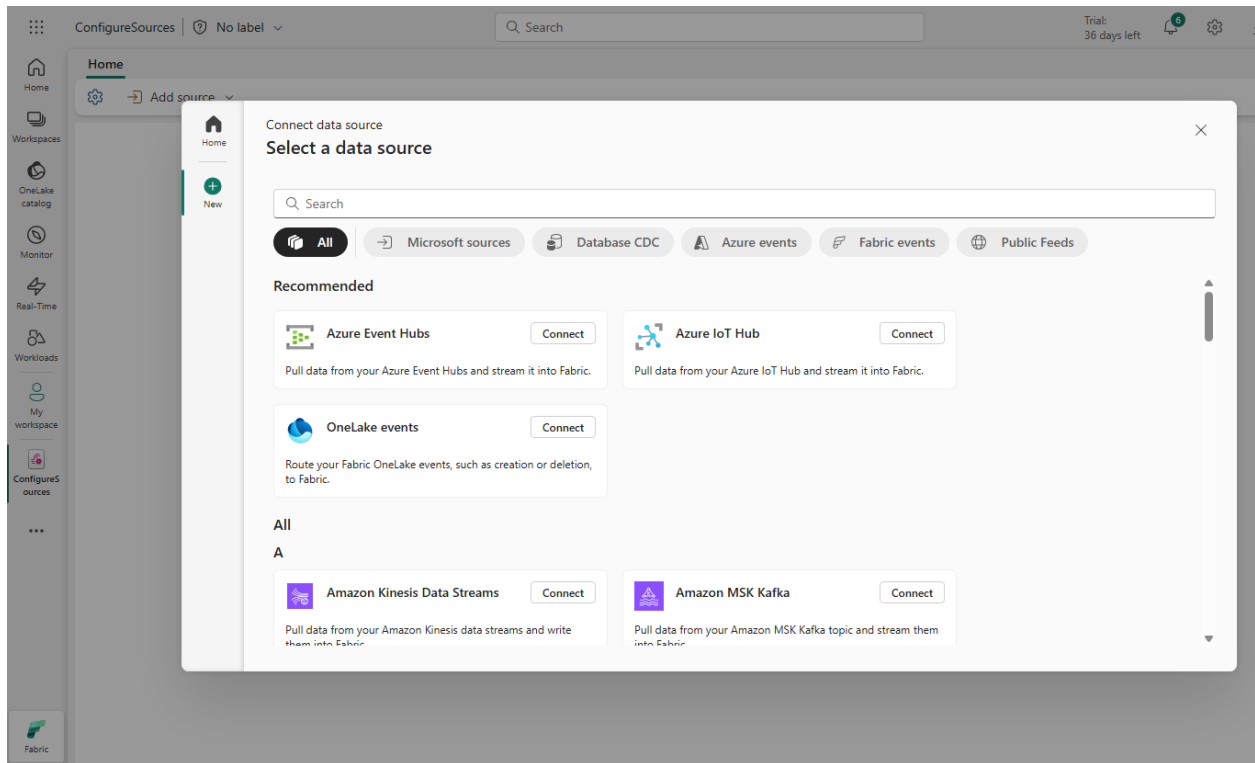
Eventstream sources

You can stream data from Microsoft sources and also ingest data from non-Microsoft platforms including:

- **Microsoft sources**, like Azure Event Hubs, Azure IoT Hubs, Azure Service Bus, Change Data Capture (CDC) feeds in database services, and others.
- **Azure events**, like Azure Blob Storage events.
- **Fabric events**, such as changes to items in a Fabric workspace, data changes in OneLake data stores, and events associated with Fabric jobs.
- **External sources**, such as Apache Kafka, Google Cloud Pub/Sub, and MQTT (Message Queuing Telemetry Transport)

Configure eventstream sources

After you create an eventstream, you can add data sources using the eventstream canvas. You can either create a new source or connect to an existing source from the Real-Time Hub:



Tip

To see all supported sources, see [Add and manage eventstream sources](#).

Eventstream destinations

Streaming data requires immediate processing and storage to retain its value. Destinations in an eventstream serve as endpoints where your processed data becomes available for queries, reports, dashboards, alerts, actions, or integration with other systems. You can load the data from your stream into the following destinations:

- **Eventhouse:** This destination lets you ingest your real-time event data into an Eventhouse, where you can use Kusto Query Language (KQL) to query and analyze the data.
- **Lakehouse:** This destination gives you the ability to transform your real-time events before ingesting them into your lakehouse. Real-time events are converted into Delta Lake format and then stored in designated lakehouse tables.
- **Derived stream:** You can think of derived streams as transformed versions of your original data stream that enable **content-based routing**. Derived streams let you route subsets of data from your *default* or original stream to different destinations based on the *content* of data. For example, you could filter IoT sensor data to send high-temperature alerts to Fabric Activator while routing hourly averages to a KQL database.
- **Fabric Activator:** Directly connect your real-time event data to an event detection engine that automatically triggers actions when specific patterns or conditions are detected in your streaming data. When data reaches certain thresholds or matches patterns, Activator can send notifications, launch Power Automate workflows, or trigger other automated responses.

- **Custom endpoint:** With this destination, you can route your real-time events to a custom endpoint. This destination is useful when you want to direct real-time data to an external system or custom application outside Microsoft Fabric.

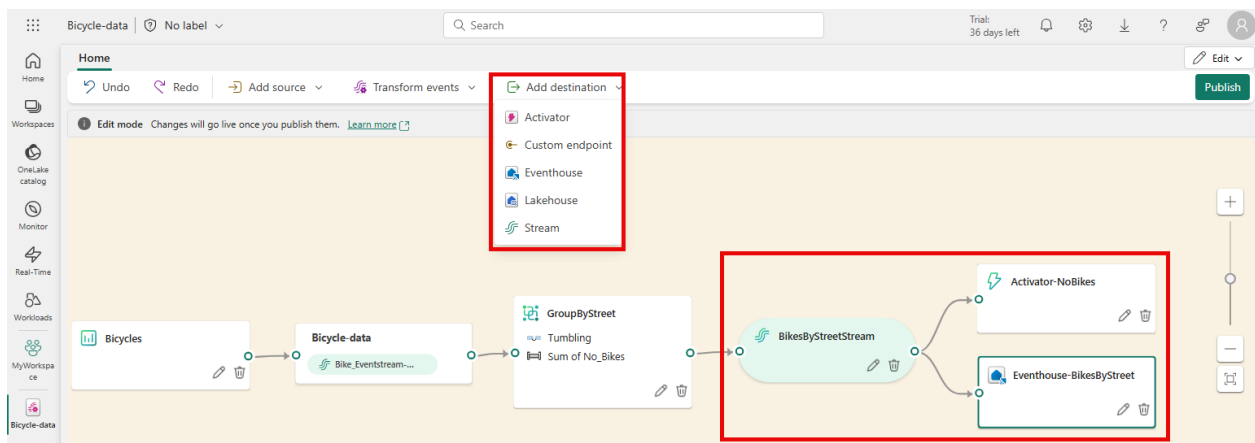
You can attach to multiple destinations within an event stream at the same time without impacting or colliding with each other.

Tip

For more information about supported destinations, see [Add and manage a destination in an eventstream](#).

Configure eventstream destinations

Eventstream destinations can be configured in the eventstream canvas. A destination can be specified after a datasource is connected or after optional transformations are applied.



The eventstream canvas in the image shows:

- **Add destination dropdown:** for configuring new destinations
- **Three configured destinations:** a derived stream, a Fabric Activator, and an Eventhouse
- **Content-based routing** where the output of the GroupByStreet transformation is routed to a derived stream that's then routed to both an Activator to check if there are bikes at every station and to an Eventhouse to insert bike counts by street into a KQL database

4. Eventstream transformations

Eventstream transformations

Completed

Raw streaming data rarely arrives in the exact format needed for analysis or action. Transformations allow you to clean, enrich, and reshape data before routing it to destinations, ensuring each endpoint receives data optimized for its specific purpose.

Common transformation scenarios include:

- **Data quality:** Filter out invalid or incomplete data before processing
- **Content-based routing:** Route different data subsets to appropriate destinations based on the actual data values or content
- **Data enrichment:** Add calculated fields, rename columns for clarity, or convert data types for downstream compatibility
- **Aggregation and summarization:** Calculate running totals, averages, or counts over time windows for dashboard displays
- **Format standardization:** Ensure consistent data structure across multiple data sources before combining streams

Transform event data

The eventstream canvas gives you a way to create event data processing workflows. Eventstream provide several no-code transformations that you can drag onto the canvas:

- **Filter:** Filter events based on the value of a field in the input. Keep only events that meet specific conditions. For example: temperature > 80°, status = "error", customer type = "premium".
- **Manage fields:** This transformation allows you to add, remove, change data type, or rename fields coming in from an input or another transformation. Add calculated fields, remove unnecessary columns, rename fields, or change data types to match destination requirements.
- **Aggregate:** Use the aggregate transformation to calculate an aggregation (Sum, Minimum, Maximum, or Average) every time a new event occurs over a period of time. This operation also lets you rename calculated columns, and filter the aggregation based on other dimensions in your data. You can have one or more aggregations in the same transformation.
- **Group by:** Calculate aggregations across events within time windows, for example, hourly sales totals, or daily temperature averages. This transformation supports various time windows including *tumbling windows* (fixed intervals) and *sliding windows* (overlapping intervals).
- **Union:** Use the union transformation to connect two or more nodes in the event canvas and add events with shared fields (with the same name and data type) into one table. Fields that

don't match are dropped and not included in the output.

- **Join:** Combine data from two streams based on a matching condition between them.
- **Expand:** Use this array transformation to create a new row for each value within an array.

Create transformation workflows

Transformations can be used together to create data processing pipelines. For example, if you had a stream of equipment temperature readings, you could start by using **filter** to remove sensor errors from incoming IoT data. Next, you might use **manage fields** to add a calculated "priority" column based on temperature thresholds. Then **group by** could calculate hourly averages by location. Finally, you'd route the processed data to appropriate destinations: temperature data to Fabric Activator for rule evaluation and hourly summaries to a Lakehouse for historical analysis.

5. Exercise - Ingest real-time data with Eventstream in Microsoft Fabric

<https://learn.microsoft.com/en-us/training/modules/explore-event-streams-microsoft-fabric/5-exercise>

Exercise - Ingest real-time data with Eventstream in Microsoft Fabric

Completed

In this exercise, you'll explore how to use an eventstream to capture a source of streaming data, transform it, and load the transformed event data into a table for further analysis.

This lab takes approximately **30** minutes to complete.

Important

You'll need a **Microsoft Fabric license** to complete this exercise. See [Getting started with Fabric](#) for details of how to enable a free Fabric trial license.

Launch the exercise and follow the instructions.

[Launch Exercise](#)

6. Module assessment

<https://learn.microsoft.com/en-us/training/modules/explore-event-streams-microsoft-fabric/6-knowledge-check>

Module assessment

Completed

7. Summary

<https://learn.microsoft.com/en-us/training/modules/explore-event-streams-microsoft-fabric/7-summary>

Summary

Completed

In this module, you learned how Microsoft Fabric Eventstream enables real-time data processing through a no-code interface. You explored using eventstream to capture data from various sources, transform it as needed, and route it to various destinations.

Tip

For more information about Real-Time Intelligence in Microsoft Fabric, see [Microsoft Fabric Eventstream - overview](#).