

Get started with Real-Time Intelligence in Microsoft Fabric

1. Introduction

<https://learn.microsoft.com/en-us/training/modules/get-started-kusto-fabric/1-introduction>

Introduction

Completed

Imagine you're a data analyst for a delivery company, responsible for monitoring package delivery performance across your network of distribution centers, delivery vehicles, and customer routes. Your operations team needs to know immediately when delivery delays occur, which routes are experiencing issues, and how customer satisfaction is trending in real time. Currently, your delivery reports are generated overnight, meaning by the time you identify a problem—such as a vehicle breakdown or weather-related delays—hundreds of packages might already be behind schedule and customers are left waiting without updates. You need continuous monitoring of delivery trucks, customer feedback systems, and warehouse scanners to track package movements and delivery performance as events happen, not hours later.

In this scenario, Real-Time Intelligence in Microsoft Fabric can be used to help you work with streaming data from GPS trackers on delivery vehicles, package scanning systems, and customer notification platforms as events occur. Unlike traditional batch processing that shows historical snapshots of delivery status, Real-Time Intelligence could help you analyze package movements and delivery performance as it happens. This approach could help you spot route delays quickly, estimate delivery windows based on current conditions, and set up automated customer notifications while creating workflows for route optimization.

By the end of this module, you'll understand how Microsoft Fabric Real-Time Intelligence components work together to create real-time analytics solutions, how to ingest, process, store and query real-time data, and how to visualize data in motion and automate responses to changing conditions.

2. What is real-time data analytics?

What is real-time data analytics?

Completed

Real-time analytics is the practice of processing, analyzing, and acting on data as it's generated, typically within seconds to minutes of when events occur. Unlike traditional analytics that works with static snapshots of historical data stored in databases, real-time analytics operates on data that's actively flowing through your systems, enabling immediate insights and rapid responses to changing conditions. This approach is also known as *near* real-time analytics, since there's always some degree of processing and network latency involved.

Understand events and streams

Events are records of things that happen in a system. They capture moments when something occurs, changes, or is completed. Examples include website clicks, stock price changes, customer purchases, patient vital sign changes, or equipment sensor readings. Think of them as digital records or log entries that document activity across your systems.

A **stream** is essentially a sequence of events, typically ordered by the time an event occurred. Each event in the stream represents something that happened at a specific moment. Events flow through streams continuously as they occur. For example, a stream of equipment temperature sensor readings contains temperature readings over many points of time. This continuous flow of event information allows you to detect patterns over time, identify opportunities or risks, and take action immediately after something happens, or in *real-time*.

Streams are the delivery mechanism that carries events from where they happen to where they need to be processed, analyzed, or acted upon.

Components of real-time analytics solutions

To build real-time analytics solutions, you need several integrated capabilities working together:

Real-time data ingestion: Collect data from multiple sources simultaneously, as information is generated. For example: database changes from change data capture, sensors, applications, system logs, and APIs.

Stream processing: Transform and analyze data while it flows from sources to destinations. This includes filtering, aggregating, joining with other data sources, and detecting patterns with minimal latency.

Low-latency storage: Use specialized databases and storage systems designed to handle high-velocity data writes and provide fast query responses.

Interactive dashboards: Create visualizations that update automatically as new data arrives, show current state and trends in real-time.

Automated decision making: Set up event-driven rules and triggers that can initiate actions, send alerts, or start workflows based on real-time conditions.

Use real-time analytics

To use real-time data effectively, information has to be ingested, processed, stored, analyzed, and presented to be actionable. Real-time analytics enables you to:

- **Respond immediately** to opportunities or problems as they emerge
- **Optimize operations** by adjusting resources and configurations based on current conditions
- **Enhance customer experiences** through personalized, contextual interactions
- **Prevent issues** by detecting anomalies before they become critical problems

Real-Time Intelligence in Microsoft Fabric brings all these capabilities together in a single platform. Through components like Eventstreams for data ingestion and transformation, Eventhouses for analytics-optimized storage, the Real-Time hub for data discovery, Real-Time Dashboards for visualization, and Activator for automated alerts and actions, Real-Time Intelligence enables you to monitor critical events, trigger automated responses, track business processes, and analyze patterns in real-time, turning what happens in your systems into actionable insights.

3. Ingest and transform real-time data

<https://learn.microsoft.com/en-us/training/modules/get-started-kusto-fabric/3a-define-real-time-hub>

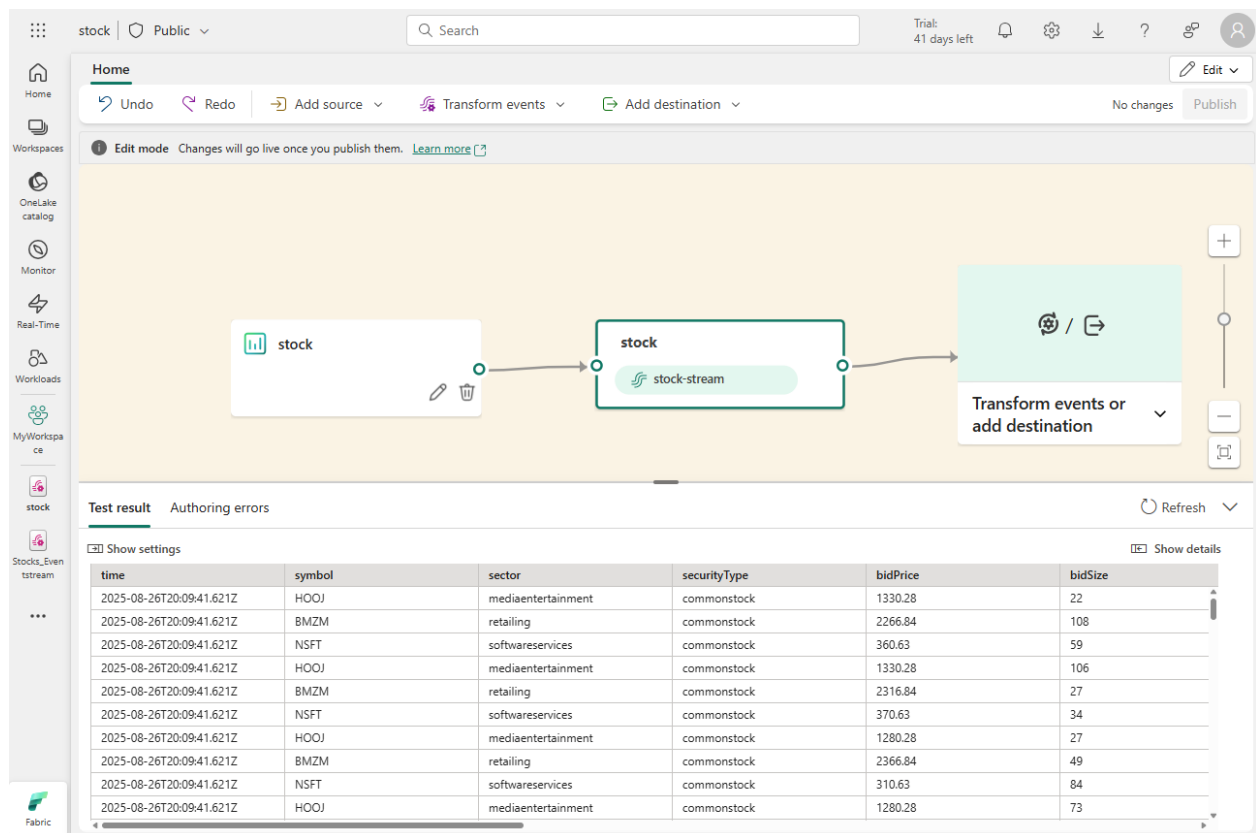
Ingest and transform real-time data

Completed

Real-Time Intelligence in Microsoft Fabric provides two primary approaches for ingesting streaming data: using eventstreams or directly ingesting data into a KQL database in an Eventhouse.

Eventstreams for data ingestion and transformation

Eventstreams are a way to bring real-time events into Fabric, to transform them, and then route data to a destination.



The image shows the three main components of an Eventstream: **sources** where data originates, **transformations** optional processing applied to the data, and **destinations** where the processed data is sent.

Think of the Eventstream components like a water pipe system. The source is your faucet, transformations are filters along the way and you need a destination like a sink or bucket to collect and use the water.

Next, let's review each component of an Eventstream.

Data sources for eventstreams

Once you create an eventstream in Fabric, you can connect it to a wide range of data 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) (in preview)

Tip

To see all supported sources, see [Supported sources](#).

Event transformations in eventstreams

Raw data from a source system is rarely in the exact format you need for analysis or storage. Transformations are what make your data useful and actionable. 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.

Tip

For more information about supported transformations, see [Process event data with event processor editor](#) and [Process events using SQL code editor](#).

Data destinations in eventstreams

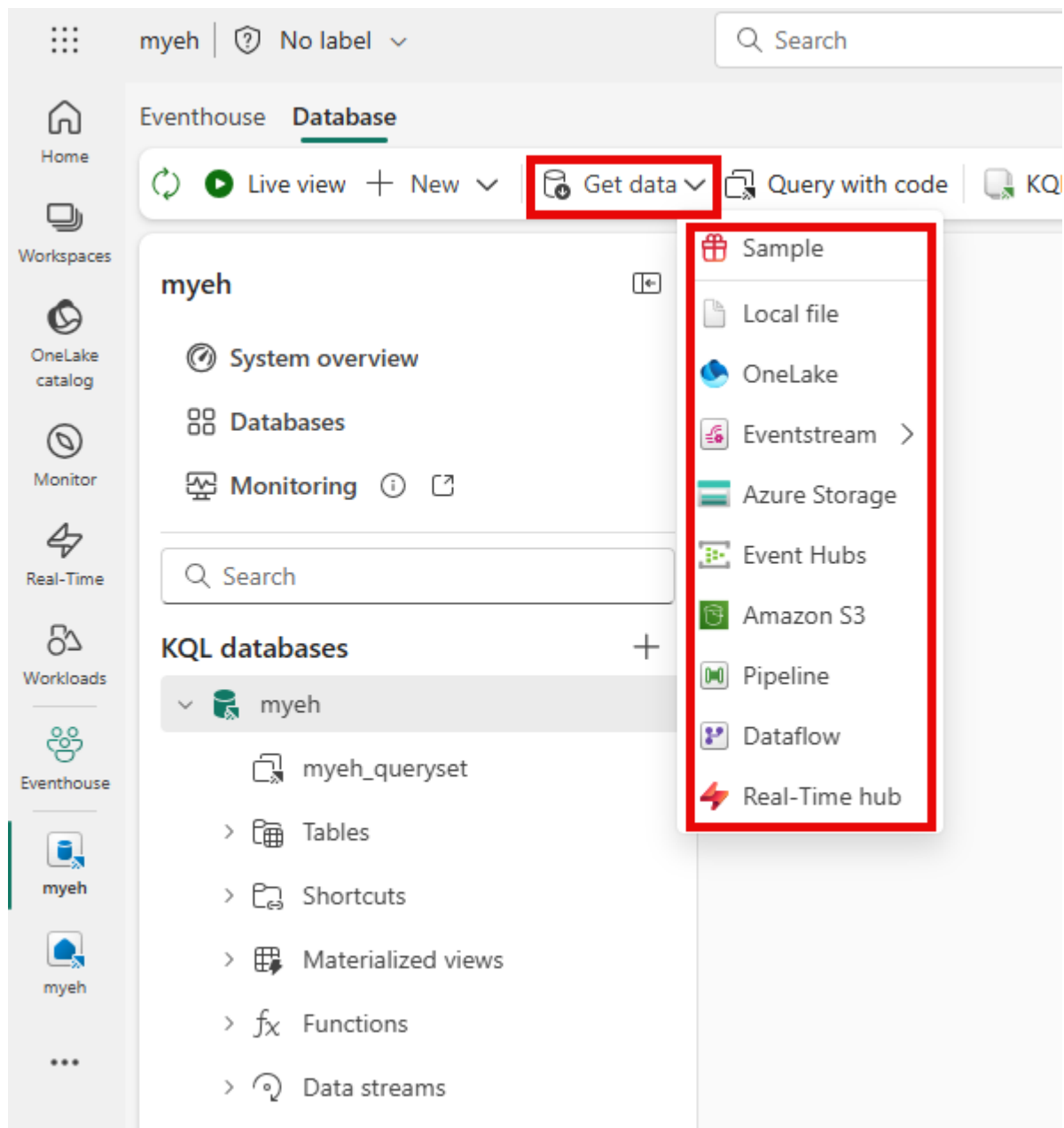
Streaming data flows continuously and is temporary by nature. It requires immediate processing and storage to retain its value. The destination in an eventstream is what makes your real-time data processing actionable. It's where your processed data becomes available for queries, reports, dashboards, alerts, actions, or integration with other systems. You can load the data from your stream into the following destinations: a KQL database in an Eventhouse, Lakehouse, a derived stream, Fabric Activator, or a custom endpoint.

Tip

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

Direct ingestion to a KQL database in an Eventhouse

Data can also be directly ingested into a KQL (Kusto Query Language) database in an Eventhouse. Some examples of data ingestion sources include: local files, Azure storage, Amazon S3, Azure Event Hubs, OneLake, and more. Data ingestion can be configured using **connectors** or through the **Get data** option in a KQL database as shown in this image.



Tip

For more information about supported ingestion sources for KQL databases in Eventhouses, see [Data sources](#) and [Data connectors overview](#).

Data transformation in a KQL database in Eventhouse with update policies

When directly ingesting data into a KQL database, data first lands in the database, then can be transformed using **update policies**. This is different from eventstream transformations that occur *during* stream processing, before routing data to a destination.

Update policies are automation mechanisms triggered when new data is written to a table. They run a query to transform ingested data and save the result to a destination table.

4. Real-Time Intelligence in Microsoft Fabric

<https://learn.microsoft.com/en-us/training/modules/get-started-kusto-fabric/3-describe-kusto-databases-tables>

Real-Time Intelligence in Microsoft Fabric

Completed

As organizations generate increasing volumes of event-driven data, the ability to process, analyze, and act on data in motion becomes essential for competitive advantage. Real-Time Intelligence provides comprehensive capabilities for working with streaming data with minimal latency.

Explore Real-Time Intelligence use cases

Unlike batch systems that process data on scheduled intervals, Real-Time Intelligence helps you respond to events as they happen, delivering near real-time insights.

Here are some common types of event data and examples of how Real-Time Intelligence can support downstream actions and business responsiveness:

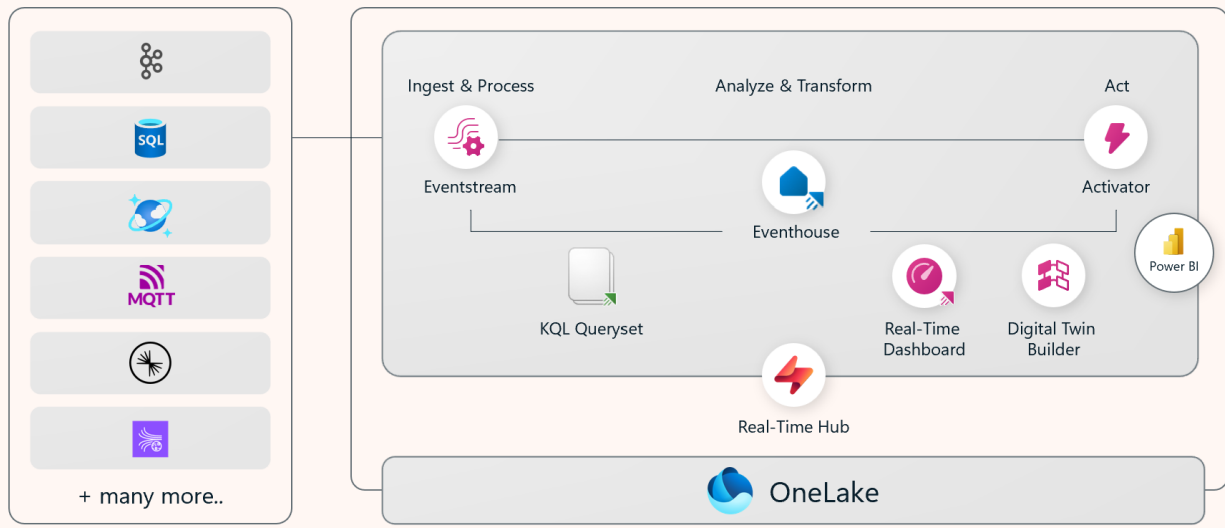
- **Delivery tracking:** Monitor vehicle locations to alert customers when packages are delayed
- **Equipment monitoring:** Track machine temperature to prevent costly breakdowns
- **Fraud detection:** Analyze purchase patterns to block suspicious transactions immediately
- **Website performance:** Monitor page load times to improve user experience
- **System health:** Track application errors to maintain service reliability

Real-Time Intelligence components

Microsoft Fabric's Real-Time Intelligence is an integrated set of components that work together to handle streaming data from capture through automated response.



Real-Time Intelligence in Microsoft Fabric



The diagram shows how Real-Time Intelligence components work together for end-to-end processing. Each component handles a specific stage of the real-time analytics process:

Ingest and process data in motion with Eventstreams

Data ingestion and processing can happen through **Eventstreams**, which capture streaming data from various sources and apply real-time transformations as data flows through the system. Eventstreams can filter, enrich, and transform your data and route it to different destinations.

Store real-time data in an Eventhouse

Real-Time Intelligence stores data in KQL (Kusto Query Language) databases in **Eventhouses**. These databases are designed for time-series data and fast ingestion of streaming data. The storage integrates with OneLake, making your data available to other Fabric tools.

Analyze data with KQL Queryset

KQL Queryset provides a workspace for running and managing queries against KQL databases. The KQL Queryset allows you to save queries for future use, organize multiple query tabs, and share queries with others for collaboration. The KQL Queryset also supports T-SQL queries, allowing you to use familiar SQL syntax alongside KQL for data analysis.

Visualize insights with Real-Time Dashboard

Real-Time Dashboards connect directly to KQL databases and refresh automatically as new data arrives. These dashboards let you explore data interactively and monitor both current conditions and

historical trends.

Act on data with Activator

Automated actions can be configured with **Activator**, which continuously monitors streaming data against user-defined rules and thresholds. When conditions are met, Activator can send notifications, trigger workflows in Power Automate, execute Fabric data pipelines or notebooks, creating event-driven automation that responds to real-time conditions.

Discover streaming data with the Real-Time hub

The Fabric **Real-Time hub** is a central location where you can discover and manage all of the data-in-motion that you have access to. It gives you a way to ingest streaming data from Azure and from external sources and it lets you subscribe to Azure and Fabric events.

Think of the Real-Time hub as your streaming data catalog where you can see what's happening in near real-time across your organization. There are connectors you can use to ingest data into Microsoft Fabric from various sources. For example, you might connect to IoT sensor streams through Azure Event Hubs, subscribe to Azure Blob Storage events, use Change Data Capture (CDC) to stream database changes, or monitor Fabric workspace events.

Once you have configured a connection to data source or event source, these items become the foundation for event driven decision making and a wide range of real-time analytics solutions, from building dashboards and setting up alerts to triggering automated workflows and analyzing trends in your data.

The screenshot displays the Microsoft Fabric Real-Time hub interface. The sidebar on the left includes navigation options: Home, Workspaces, OneLake catalog, Monitor, Real-Time (highlighted with a red box), Workloads, and RTI_WS. The main content area is titled "Discover, analyze, and act on data-in-motion" and features three interactive cards: "Subscribe to OneLake events", "Act on Job events", and "Visualize data". Below these cards, the "Recent streaming data" section is visible, which includes a filter bar and a table of data streams.

Data	Source item	Item owner	Workspace	Endorsement
fabric_events_eventstream-stream	fabric_events_eventstream	System Administrator	My workspace	—
MyTutorialEventstream-stream	MyTutorialEventstream	System Administrator	MYRTIWS	—
MyRawData	MyTutorial	System Administrator	MYRTIWS	—
TransformedData	MyTutorial	System Administrator	MYRTIWS	—
TutorialEventstream3-stream	TutorialEventstream3	System Administrator	RTI_WS	—
new_event_stream-stream	new_event_stream	System Administrator	RTI_WS	—
RTISample-stream	RTISample	System Administrator	My workspace	—
Bikestream	RTISample	System Administrator	My workspace	—

To access the real-time hub, select the **Real-Time** icon in the main Fabric menu bar.

The real-time hub organizes data-in-motion into several main categories:

- **Data sources:** Browse and connect to available streaming data sources, such as Microsoft sources, database change data capture feeds, and external sources from other cloud providers
- **Azure sources:** Discover and configure Azure streaming data sources such as Azure IoT Hub, Azure Service Bus, Azure Data Explorer DB, and more
- **Fabric events:** Subscribe to system-generated events in Fabric that you can access, like job status changes, events produced by action on files or folders in OneLake, and Fabric workspace item changes
- **Azure events:** Subscribe to system events from Azure services that can be used to trigger automated responses such as actions on files or folders in Azure blob storage

In the Real-Time hub, you can preview and explore your streaming data by navigating directly to eventstreams or KQL databases in eventhouses for deeper analysis and querying. You can also build automated responses using *Activator* rules that trigger actions like notifications, workflows, or data processing when specific patterns are detected.

5. Automate actions

<https://learn.microsoft.com/en-us/training/modules/get-started-kusto-fabric/4c-activator>

Automate actions

Completed

Activator is a technology in Microsoft Fabric that enables automated processing of events that trigger actions. For example, you can use Activator to notify you by email when a value in an Eventstream deviates from a specific range or to run a notebook to perform some Spark-based data processing logic when a Real-Time Dashboard is updated.

The screenshot displays the Activator application interface. On the left, a sidebar contains a tree view with categories such as 'Package delivery sample', 'Delivery status', and 'Delivery failed'. The 'Delivery failed' category is highlighted. The main area shows a 'Definition' panel for the 'Delivery failed' rule. This panel includes a 'Monitor' section with a dropdown for 'Delivery status', a 'Condition' section with a dropdown for 'Is equal to', and an 'Action' section with a dropdown for 'Teams message'. A 'Definition' dialog box is open on the right, showing the configuration for the 'Delivery failed' rule. The dialog box includes a 'Monitor' section with a dropdown for 'Delivery status', a 'Condition' section with a dropdown for 'Is equal to', and an 'Action' section with a dropdown for 'Teams message'. The dialog box also includes a 'Recipients' section with a dropdown for 'System Administrator' and a 'Message' section with a text input field containing 'This package had a failed delivery attempt. An attempt will be made on the next business day.'.

This image shows a rule configured to alert when package delivery failures happen, demonstrating how you can automate responses to specific business events.

Understand Activator key concepts

Activator operates based on four core concepts: *Events*, *Objects*, *Properties*, and *Rules*.

- **Events** - Each record in a stream of data represents an *event* that has occurred at a specific point in time.
- **Objects** - The data in an event record can be used to represent an *object*, such as a sales order, a sensor, or some other business entity.
- **Properties** - The fields in the event data can be mapped to *properties* of the business object, representing some aspect of its state. For example, a *total_amount* field might represent a sales order total, or a *temperature* field might represent the temperature measured by an environmental sensor.
- **Rules** - The key to using Activator to automate actions based on events is to define *rules* that set conditions under which an action is triggered based on the property values of objects referenced in events. For example, you might define a rule that sends an email to a maintenance manager if the temperature measured by a sensor exceeds a specific threshold.

Use cases for Activator

Activator can help you in various scenarios, such as dynamic inventory management, real-time customer engagement, and effective resource allocation in cloud environments. It's a powerful tool for any circumstance that requires real-time data analysis and automated actions.

Use Activator to:

- Initiate marketing actions when product sales drop.
- Send notifications when temperature changes could affect perishable goods.
- Flag real-time issues affecting the user experience on apps and websites.
- Trigger alerts when a shipment hasn't been updated within an expected time frame.
- Send alerts when a customer's account balance crosses a certain threshold.
- Respond to anomalies or failures in data processing workflows immediately.
- Run ads when same-store sales decline.
- Alert store managers to move food from failing grocery store freezers before it spoils.

Tip

For more information about Activator, see [What is Activator?](#).

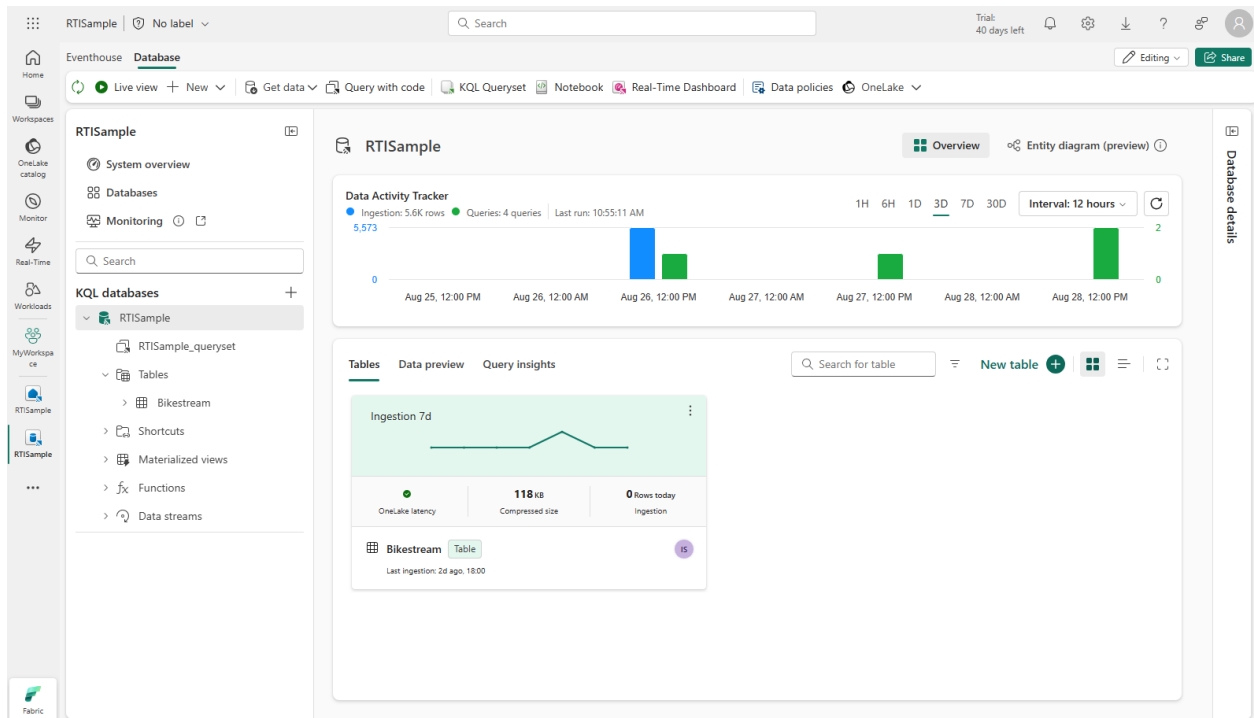
6. Store and query real-time data

<https://learn.microsoft.com/en-us/training/modules/get-started-kusto-fabric/4-write-queries-kusto-query-language>

Store and query real-time data

Completed

KQL databases in an Eventhouse are where you store and query real-time data that flows from Eventstreams and other streaming sources. Once data is loaded into tables, you can use the Kusto Query Language (KQL) or T-SQL to query your data.



Within an eventhouse, you can create:

- **KQL databases:** Real-time optimized data stores that host a collection of tables, stored functions, materialized views, shortcuts and data streams.
- **KQL querysets:** Collections of KQL queries that you can use to work with data in KQL database tables. A KQL queryset supports queries written using Kusto Query Language (KQL) or a subset of the Transact-SQL language.

Understand the power of Kusto Query Language (KQL)

To query data in a table in a KQL database, you can use the **KQL**. KQL is specifically designed for analyzing large volumes of structured, semi-structured, and unstructured data with exceptional performance. KQL databases are optimized for time-series data and index incoming data by ingestion time and partition it for optimal query performance. KQL is the same language used in Azure Data Explorer, Azure Monitor Log Analytics, Microsoft Sentinel, and in Microsoft Fabric.

Get familiar with KQL syntax

KQL queries are made of one or more query statements. A query statement consists of a table name followed by operators that `take`, `filter`, `transform`, `aggregate`, or `join` data. For example, to print any 10 rows in the `stock` table, execute:

```
stock
| take 10
```

A more complex example might aggregate data to find average stock prices over the last 5 minutes:

```
stock
| where ["time"] > ago(5m)
| summarize avgPrice = avg(todouble(bidPrice)) by symbol
| project symbol, avgPrice
```

Tip

To learn more about KQL, see [Kusto Query Language \(KQL\) overview](#).

Automate data processing with management commands

Beyond basic querying, you can automate data processing through **management commands** including:

- **Update policies:** Automatically transform incoming data and save it to different tables as it arrives.
- **Materialized views:** Precalculate and store summary results for faster queries.
- **Stored functions:** Save frequently used query logic that you can reuse across multiple queries.

Tip

For more information about working with KQL databases, including detailed examples of update policies, materialized views, and stored functions, see [Work with real-time data in a Microsoft Fabric Eventhouse](#).

Other query options

Using SQL

KQL databases in Eventhouses also support a subset of common T-SQL expressions for data professionals already familiar with T-SQL syntax. For example:

```
SELECT TOP 10 * FROM stock;
```

Use Copilot to help with queries

Microsoft Fabric includes Copilot for Real-Time Intelligence, which can help you write queries to extract insights from your Eventhouse data. Copilot uses AI to understand what you're looking for and can generate the required query code.

Tip

To learn more about Copilot for Real-Time Intelligence, see [Copilot for Real-Time Intelligence](#).

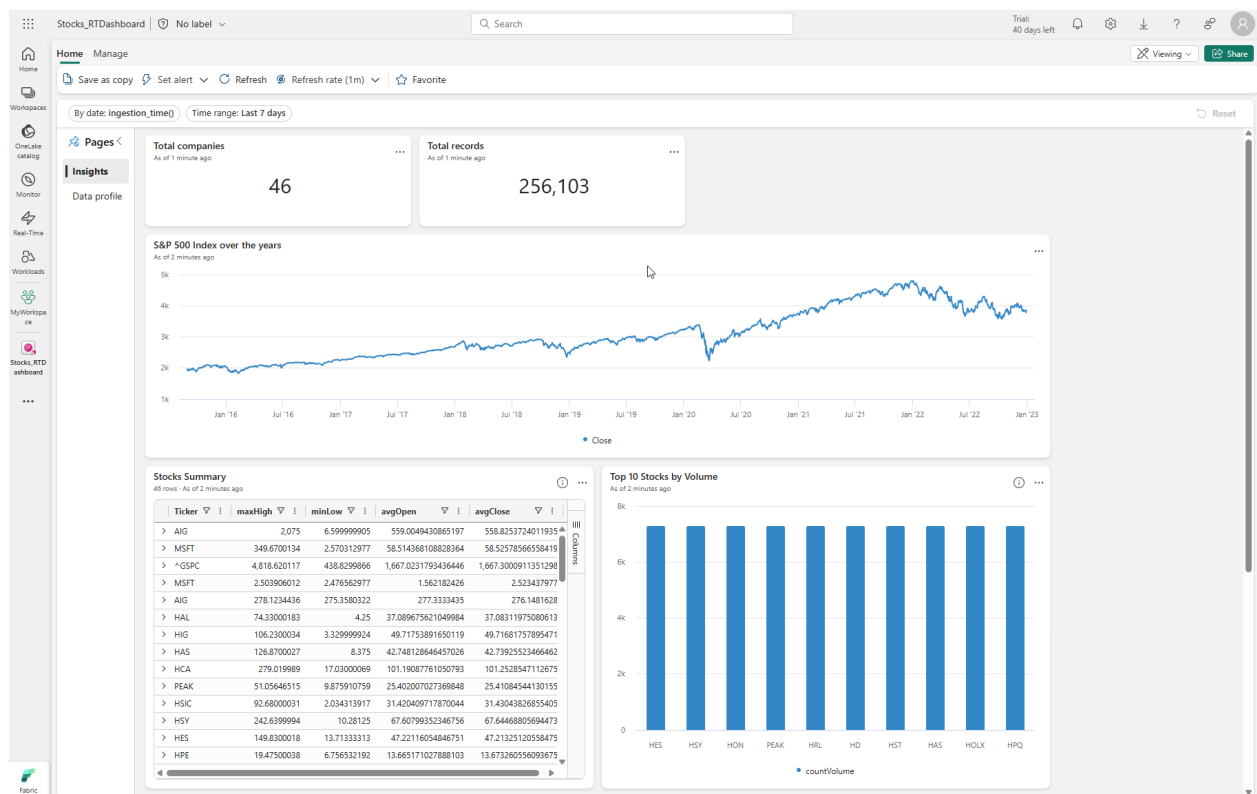
7. Visualize real-time data

<https://learn.microsoft.com/en-us/training/modules/get-started-kusto-fabric/4b-visualize-data>

Visualize real-time data

Completed

Real-Time Dashboards provide a way to pin data visualizations to a single visual interface, enabling you to surface real-time insights at a glance. Each *tile* in a dashboard shows you different information based on a KQL query that extracts real-time data from tables in an eventhouse.



Create a Real-Time Dashboard

You can create a Real-Time Dashboard in a workspace and then configure its source, or you can create one directly from a KQL queryset in an eventhouse.

Dashboards are composed of one or more *tiles*, each containing a visualization based on a KQL query expression. By default, the visualization shows the results of the query as a table; but you can edit the tile to customize how the data is displayed.

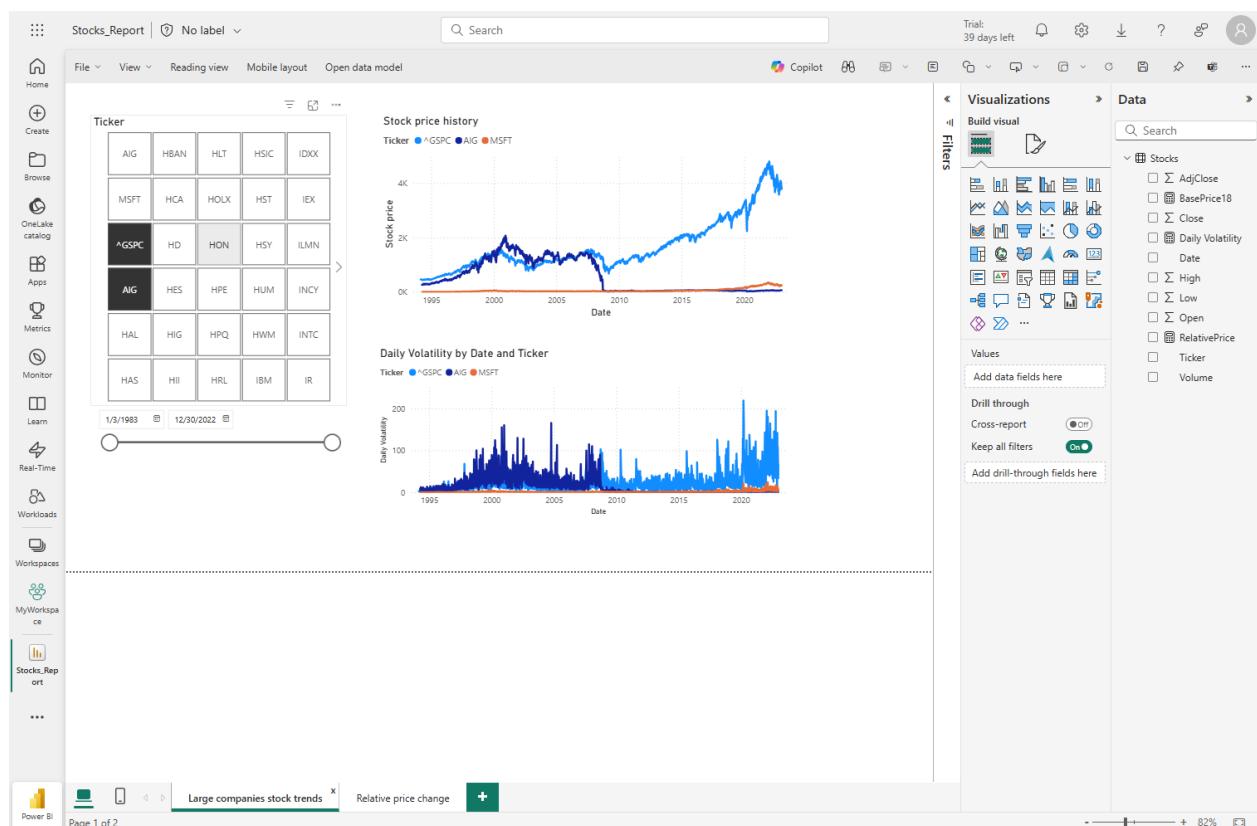
When published, tiles let you explore the data they contain interactively by drilling into the data and using a visual interface to filter and aggregate the data, and change the visualization type.

Tip

To learn more about Real-Time Dashboards, see [Create a Real-Time Dashboard](#).

Visualize real-time data with Power BI

You also can create Power BI reports from your KQL database data.



Tip

To learn more about using Power BI with real-time data in Fabric, see [Visualize data in a Power BI report](#).

8. Exercise - Get started with Real-Time Intelligence in Microsoft Fabric

<https://learn.microsoft.com/en-us/training/modules/get-started-kusto-fabric/5-exercise-use-kusto-query-data-on-lake>

Exercise - Get started with Real-Time Intelligence in Microsoft Fabric

Completed

Now it's time to try out Real-Time Intelligence yourself.

In this exercise, you'll ingest real-time data into Microsoft Fabric. You'll then query and visualize the data before defining an alert to automate an action based on a threshold value in the real-time data stream.

Note

You need a Microsoft Fabric tenant to complete this exercise. For more information about accessing Microsoft Fabric, see [Getting started with Fabric](#).

Launch the exercise and follow the instructions.

[Launch Exercise](#)

9. Module assessment

<https://learn.microsoft.com/en-us/training/modules/get-started-kusto-fabric/6-knowledge-check>

Module assessment

Completed

10. Summary

<https://learn.microsoft.com/en-us/training/modules/get-started-kusto-fabric/7-summary>

Summary

Completed

In this module, you learned how Microsoft Fabric Real-Time Intelligence enables you to ingest, transform, store, visualize, and act on data in motion. You explored the Real-Time Hub for discovering streams, Eventstreams for ingesting and processing event data, and KQL databases in Eventhouses for fast, time-based analytics using KQL. You also learned how to visualize streaming insights with Real-Time dashboards and Power BI, and how to automate responses using Activator.

Tip

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