

Monitor activities in Microsoft Fabric

1. Introduction

<https://learn.microsoft.com/en-us/training/modules/monitor-fabric-items/1-introduction>

Introduction

Completed

In a data analytics solution, you create, manage, and optimize the frameworks for collecting, storing, processing, and analyzing data. Data in its raw form isn't meaningful for making business decisions. Data from different sources has to be combined and loaded or ingested into data stores, and transformed into formats that are useful for analysis. The ingestion and transformation process must be monitored so errors can be identified and remediated before they affect users.

Microsoft Fabric provides capabilities for monitoring Fabric activities. In this module, we explore how to use Monitor Hub to monitor ingestion and transformation activities in your Fabric environment and how to use Activator to automatically take action when patterns or conditions are detected in changing data.

By the end of this module, you'll understand the monitoring capabilities in Microsoft Fabric. You'll also get to complete a practical exercise using Monitor Hub.

2. Understand monitoring

<https://learn.microsoft.com/en-us/training/modules/monitor-fabric-items/2-understand-monitoring>

Understand monitoring

Completed

Monitoring is the process of collecting system data and metrics that determine if a system is healthy and operating as expected. Monitoring exposes errors that occurred and when they happened. To

investigate issues and remediate errors, historical data is analyzed to get a picture of the health of a system or process.

Monitoring Fabric activities

In Fabric, you schedule activities and jobs that perform tasks like data movement, and transformation. Activities have dependencies on one another. You need to make sure that data arrives in its expected location on time and that system errors or delays don't affect users or downstream activities. End-to-end processes need to be managed to ensure they're reliable, performant, and resilient. One aspect of this monitoring is identifying and handling long-running operations and errors effectively. By doing this, you can minimize downtime and quickly address any underlying issues.

The following activities in Fabric allow you to perform tasks that deliver data to users. These activities should be monitored:

- **Data pipeline activity** - A data pipeline is a group of activities that together perform a data ingestion task. Pipelines allow you to manage, extract, transform, and load (ETL) activities together instead of individually. Monitor the success or failure of jobs and pipeline activities. Look for errors if the pipeline failed. View job history to compare current activity performance to past job execution performance to gain insight into when errors were first introduced into a process.
- **Dataflows** - A dataflow is a tool for ingesting, loading, and transforming data using a low-code interface. Dataflows can be run manually or scheduled or run as part of pipeline orchestration. Monitor start and end times, status, duration, and table load activities. To investigate issues, drill down into activities and view information about errors.
- **Semantic model refreshes** - A semantic model is a visual representation of a data model that's ready for reporting and visualization. It contains transformations, calculations, and data relationships. Changes to the data model require the semantic model to be refreshed. Semantic models can be refreshed from data pipelines using the *semantic model refresh* activity. Monitor for refresh retries to help identify transient issues, before classifying an issue as a failure.
- **Spark jobs, notebooks and lakehouses** - Notebooks are an interface for developing Apache Spark jobs. Data can be loaded, or transformed for lakehouses using Spark and notebooks. Monitor Spark job progress, task execution, resource usage, and review Spark logs.
- **Microsoft Fabric Eventstreams** - Events are observations about the state of an object, like a timestamp for weather sensors. Eventstreams in Fabric are set up to run perpetually to ingest real-time or streaming events into Fabric and transform them for analytics needs, and then route them to various destinations. Monitor streaming event data, ingestion status, and ingestion performance.

Monitoring best practices

Continuously monitor the data ingestion, transformation, and load processes to ensure they're running smoothly. Monitoring best practices include:

- **Identifying what to monitor** and tracking metrics.
- **Collecting and analyzing data on a regular basis** to identify normal behavior so you can spot anomalies when they occur.
- **Reviewing logs and metrics regularly** to identify and establish parameters for normal system behavior.
- **Taking action** to resolve problems when metrics and logs show deviations from normal behavior.
- **Optimizing performance** by using monitoring data to identify bottlenecks or performance issues.

3. Use Microsoft Fabric Monitor Hub

<https://learn.microsoft.com/en-us/training/modules/monitor-fabric-items/3-use-monitor-hub>

Use Microsoft Fabric Monitor Hub

Completed

Visualization tools make monitoring easier. They help you identify trends or anomalies. Monitor hub is the monitoring visualization tool in Microsoft Fabric. Monitor hub collects and aggregates data from selected Fabric items and processes. It stores Fabric activity data in a common interface so you can view the status of multiple different data integration, transformation, movement, and analysis activities in Fabric in one place, rather than monitor each separately.

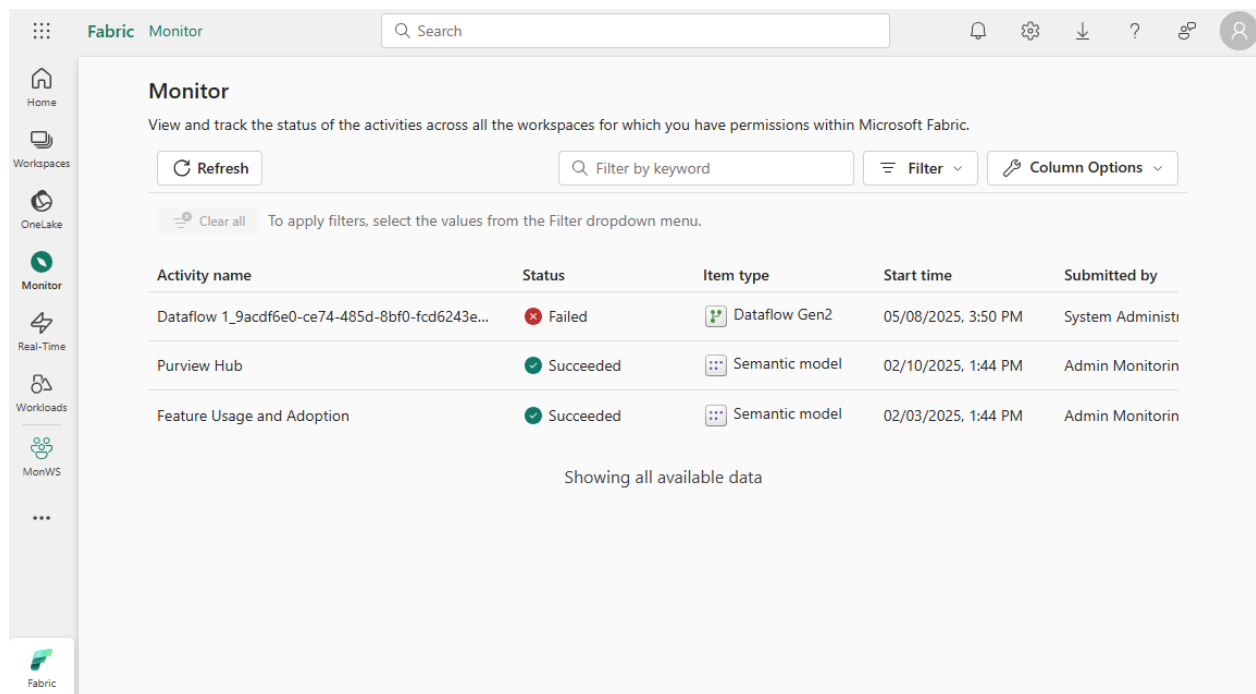
Activities displayed in the Monitor hub

Some of the activities you can see monitoring metadata for in the Microsoft Fabric Monitor hub include:

- Data pipeline execution history
- Dataflow executions
- Datamart and semantic model refreshes
- Spark job and notebook execution history and job details

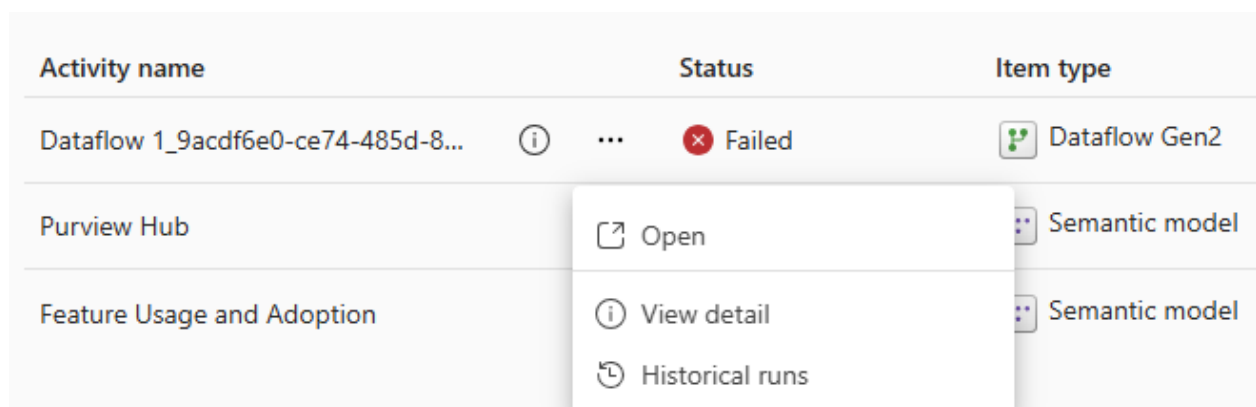
View the Monitor Hub

The Monitor hub can be opened by selecting **Monitor** from the Fabric navigation pane.



View Fabric activity detail

Each activity in Monitor hub can be selected and several actions can be performed for the selected activity. Actions vary by activity and include options such as: opening the activity, retrying it, viewing activity details or historical runs. To view this information, select the ellipsis that appears when you hover over an activity.



When you select **View detail**, the screen that appears is customized for the activity you select and provides clarity about what happened during the activity. You can view metadata such as:

- Activity status
- Start and end time
- Duration
- Activity statistics

Investigate what happened during a Fabric activity

To investigate what happened during an activity some activities in Monitor hub include hyperlinks that allow you to drill down into execution details. Information is provided about any errors and the success or failure of an execution. Spark activities across items can be viewed. You can view Spark applications triggered from Notebooks, Spark Job Definitions, and Pipelines. For more information, see: [Apache Spark monitoring overview](#)

4. Take action with Microsoft Fabric Activator

<https://learn.microsoft.com/en-us/training/modules/monitor-fabric-items/4-take-action-with-activator>

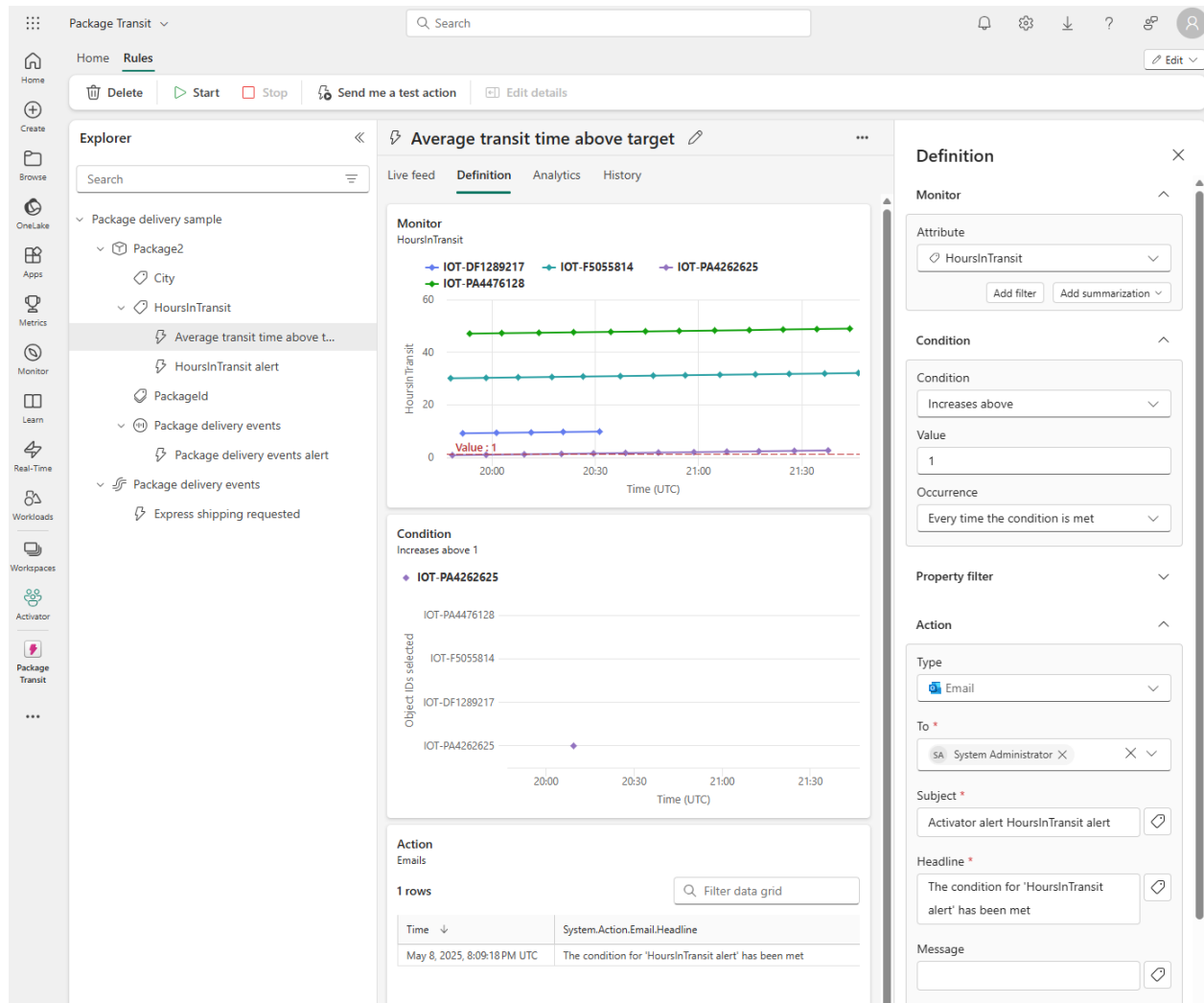
Take action with Microsoft Fabric Activator

Completed

When monitoring surfaces changing data, anomalies, or critical events, alerts are generated or actions are triggered. Real-time data analytics is commonly based on the ingestion and processing of a data stream that consists of a perpetual series of data, typically related to specific point-in-time events. For example, a stream of data from an environmental IoT weather sensor. Real-Time Intelligence in Fabric contains a tool called Activator that can be used to trigger actions on streaming data. For example, a stream of data from an environmental IoT weather sensor might be used to trigger emails to sailors when wind thresholds are met. When certain conditions or logic is met, an action is taken, like alerting users, executing Fabric job items like a pipeline, or kicking off Power Automate workflows. The logic can be either a defined threshold, a pattern like events happening repeatedly over a time period, or the results of logic defined by a Kusto Query Language (KQL) query.

What is Activator

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.



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 potent tool for any circumstance that requires real-time data analysis and 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 working with Activator, see [Tutorial: Create and activate an Activator rule](#).

5. Exercise - Monitor Fabric activity in the Monitor hub

<https://learn.microsoft.com/en-us/training/modules/monitor-fabric-items/5-exercise>

Exercise - Monitor Fabric activity in the Monitor hub

Completed

Now it's your turn to monitor Fabric activity in the Monitor hub.

Note

To complete this exercise, you need a Microsoft Fabric tenant. See [Getting started with Fabric](#) to find out how to enable a Fabric trial license.

Launch the exercise and follow the instructions.

[Launch Exercise](#)

6. Module assessment

<https://learn.microsoft.com/en-us/training/modules/monitor-fabric-items/6-knowledge-check>

Module assessment

Completed

7. Summary

<https://learn.microsoft.com/en-us/training/modules/monitor-fabric-items/7-summary>

Summary

Completed

In this module, you learned how monitoring is implemented in Microsoft Fabric. You learned how to use Monitor Hub to monitor Microsoft Fabric activities from a central location and how to use Activator to automatically take action when patterns or conditions are detected in changing data.

Tip

For more information about monitoring Fabric activities and responding to detected changes in data, see:

- [Use the Monitor hub](#)
- [What is Activator?](#)