Microsoft Fabric

Microsoft

Lakehouse Analytics

with Microsoft Fabric and Azure Databricks

Lakehouse Analytics

With Microsoft Fabric and Azure Databricks

Unlock the value of your data

Today, data is often likened to gold, with enterprises globally racing to extract, refine, and capitalize on their vast data assets. This transformative wave is powered by public cloud infrastructure that provides enterprises with massive compute power and storage capacity to process their growing data estates. Microsoft Azure, today's leading cloud platform, provides services that empower organizations to meet these modern data challenges. At the forefront is Microsoft Fabric, a new analytics powerhouse that is reshaping the realm of data management. Coupled with Azure Databricks, enterprises have a wide array of tools to harness the power of the cloud and tackle their most ambitious data initiatives.

From data engineers to decision makers, Fabric provides all users with the necessary tools and insights, propelling businesses into the future. By presenting a cohesive platform that combines diverse analytics functionalities, Fabric ensures that businesses unlock the potential of their data without the intricacies of managing multiple tools. With OneLake, a pioneering data platform, Fabric enables end-to-end analytics, providing a secure, managed storage layer for building and maintaining a modern data lakehouse.

What is a lakehouse?

The lakehouse architecture is the modern approach to building a scalable analytics platform for an enterprise's growing data estate. It combines the precision of a traditional data warehouse with the massive scale and flexibility of a data lake. By using modern tools such as Microsoft Fabric, Microsoft Power BI, and Azure Databricks, enterprises can build a lakehouse that meets the needs of data engineers, business analysts, and data scientists—all sharing a single copy of the data, stored in an open format, and governed by a unified catalog.

The lakehouse architecture on Azure fully embraces the power of Apache Spark, a scalable analytics engine for enterprises. With AI assistants in both Fabric and Azure Databricks, teams can work more efficiently to share data, find new insights, and build advanced AI models.

The traditional data lake and warehouse architecture

Data lakes, as the name suggests, are vast reservoirs that store a myriad of data types in their raw, unprocessed form. As centralized repositories, data lakes are designed to ingest and store vast volumes of data in its most authentic form. This architecture is characterized by its flexibility, allowing businesses to store structured, semi-structured, and unstructured data without the constraints of a predefined schema.

Data warehouses are structured repositories, designed for efficient querying and analysis. A data warehouse stores data that has been cleaned, organized, processed, and transformed, making it ready for business intelligence (BI) applications.

Area	Data lake	Data warehouse
Data store	It can capture and retain unstructured, semi-structured, and structured data in its raw format. A data lake stores all types of data, irrespective of the source and structure.	It can capture and retain only structured data. A data warehouse stores data in quantitative metrics with its attributes. Data is transformed and cleansed.
Schema definition	Typically, the schema is defined after data is stored. This offers high agility and data capture quite easily, but it requires work at the end of the process (schema-on-read).	Typically, a schema is defined prior to when data is stored. It requires work at the start of the process, but it offers performance, security, and integration (schema-on-write).
Data quality	Any data that may or may not be curated (such a raw data).	Highly curated data that serves as the central version of the truth.
Users	A data lake is ideal for users who indulge in deep analysis, such as data scientists, data engineers, and data analysts.	A data warehouse is ideal for operational users such as business analysts due to it being well structured and easy to use and understand.
Price and performance	The storage cost is relatively low, compared to a data warehouse, and querying results is quicker.	The storage cost is high, and querying results is time consuming.
Accessibility	A data lake has few constraints and is easily accessible. Data can be changed and updated quickly.	A data warehouse is structured by design, which makes it difficult to access and manipulate.

Table 1: Data lake and data warehouse comparison

Data management and analytics with Microsoft Fabric and Azure Databricks

Microsoft Fabric is a unified analytics platform that brings together all the data and analytics tools that organizations need. Azure Databricks is a fully managed Azure first-party service that enables analytics and AI in a lakehouse in Azure.

Fabric and Azure Databricks both provide end-to-end solutions enabling data engineers, data scientists, data administrators, data analysts, and data consumers to work together to find valuable insights. Because both platforms are built on Delta Lake storage, the two services can operate together, sharing the same copies of the data. When combined, Fabric and Azure Databricks offer a powerful synergy that enhances data analytics, processing, and AI solutions. Here are five concrete examples of how Fabric works better with Azure Databricks:

- **Efficient data pipelines**: Azure Databricks allows data engineers to use the power of Apache Spark with Photon acceleration to construct efficient, scalable data pipelines. These pipelines can feed data into OneLake, ensuring that data is readily available for analytics and other operations. This integration ensures that data engineers and scientists have a seamless experience when working with data across both platforms.

- **Unified data storage with OneLake**: Azure Databricks can directly interact with the data stored in OneLake. Whether the data originates from on-premises systems or is ingested from sources such as Azure Databricks, OneLake provides the tools to consolidate this data. This architecture minimizes data copies, offers consolidated governance, and enables users to use preferred applications such as Azure Databricks for querying and data science.

- **AI and machine learning synergy**: Azure Databricks helps build complex AI and machine learning models. When combined with AI-based Copilot features in Fabric, businesses can derive insights from their data more efficiently. This synergy ensures that data scientists can build and deploy models in Azure Databricks and then use Fabric for BI analytics, using the AI capabilities of both platforms.

- **Flexibility with data storage**: There are two potential approaches to loading data into OneLake using Azure Databricks:

  - **ADLS-based lakehouse**: Data can be stored in Delta Lake tables in an Azure Data Lake Storage (ADLS) account. A shortcut to this storage can then be created in the Fabric Lakehouse database, ensuring that data is easily yet securely accessible within Fabric.
  - **OneLake-based lakehouse**: Data can be stored directly in the ADLS location for OneLake. This approach requires identifying the default storage location for the Fabric Lakehouse. Once data is stored in this location, it can be accessed and used within Fabric, ensuring seamless integration between the two platforms.

- **Enhanced data visualizations**: Once data is stored and processed, creating visualizations becomes a crucial step. With the integration of Azure Databricks and Fabric, users can easily create Power BI datasets on the processed data. This integration ensures that data scientists and analysts can visualize their data, derive insights, and make informed decisions based on the analytics provided by both platforms.

The integration between Fabric and Azure Databricks offers a comprehensive solution for an organization's data analytics needs. With the release of the OneLake public preview, there's potential for these two powerhouses to simplify analytical tasks.

OneLake: The core of Microsoft Fabric

OneLake, also referred to as Microsoft Fabric Lake, is the foundational element of all Fabric services. It provides a unified storage hub for organizational data, built on the robust ADLS Gen2. OneLake caters to a diverse user base, from seasoned professionals to budding developers, with its primary goal being to dissolve data silos. It promotes easy data discovery and sharing, and ensures centralized security compliance.

Unify data with Microsoft Fabric

Microsoft Fabric goes beyond traditional analytics tools. It provides a unified platform, streamlining the analytics process from data integration to real-time insights. It offers a comprehensive analytics solution for enterprises, eliminating the need for multiple vendor-specific services. Fabric brings together new and existing components from Power BI, Azure Synapse, and Azure Data Factory into a single software as a service (SaaS) platform, ensuring a cohesive user experience.

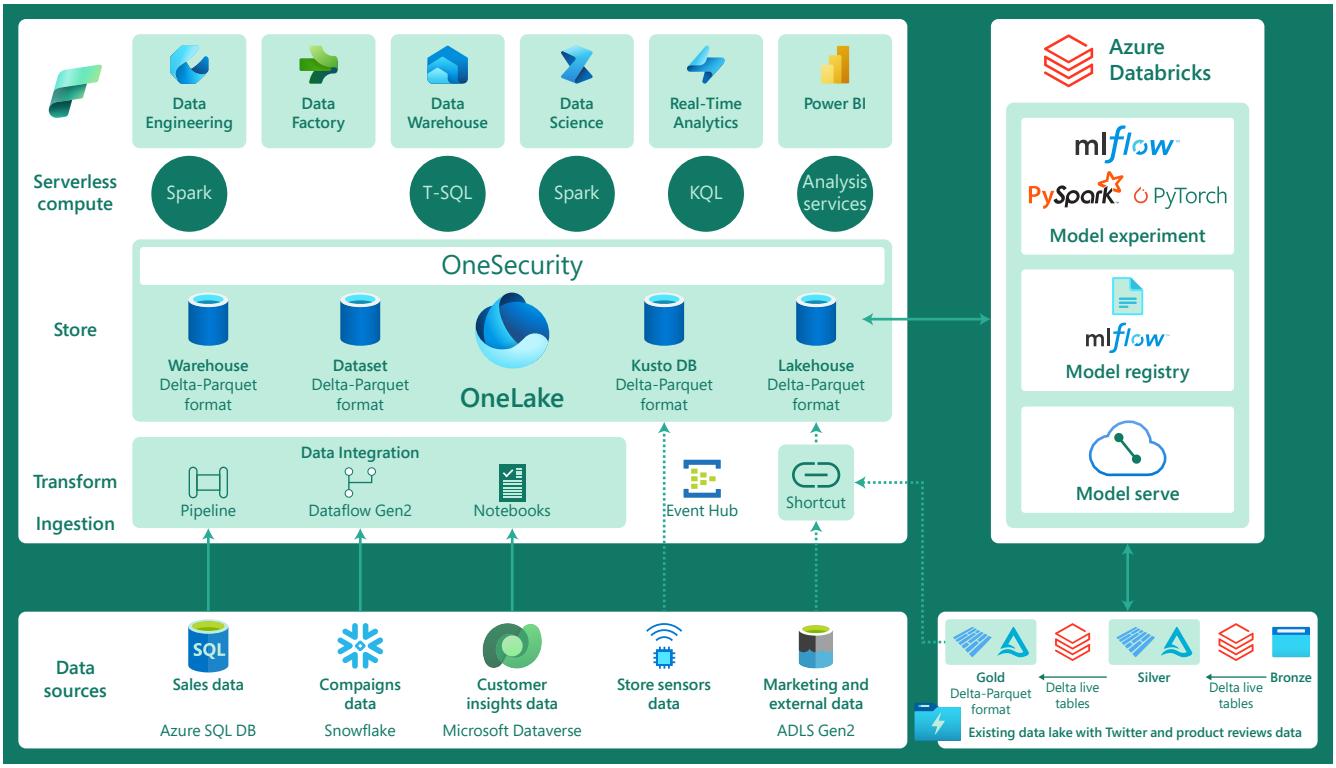

Figure 1: Microsoft Fabric and Azure Databricks architecture

Components of Microsoft Fabric

Microsoft Fabric offers end-to-end analytical solutions. It connects every data source and analytics service together through industry-leading experiences across a wide range of categories.

- **Synapse Data Engineering**: A core experience of Microsoft Fabric, Synapse Data Engineering empowers data engineers with a premier Spark platform, facilitating large-scale data transformations within the lakehouse paradigm. It streamlines the process of ingesting, transforming, and sharing organizational data in an open format.

- **Data Factory**: Data Factory is designed for complex hybrid extract-transform-load (ETL), and data integration projects. It helps in transforming raw, unorganized data into actionable insights. Data Factory allows the creation of data-driven workflows, known as pipelines, to ingest data from various data stores. These pipelines can use dataflows or compute services such as Azure HDInsight Hadoop, Azure Databricks, and Azure SQL Database to transform data.

- **Synapse Data Science**: Synapse Data Science unlocks the value of data in an organization's analytics workflow. It provides tools and capabilities that enable data scientists to build, deploy, and manage machine learning models seamlessly using Azure Machine Learning integrations.

- **Synapse Data Warehouse**: Integrated with Power BI, Fabric offers next-generation data warehousing capabilities. It supports an open data format, ensuring that data engineers and IT teams can work seamlessly by storing data in the open Delta Lake format, ensuring top-tier SQL performance.

- **Synapse Real-Time Analytics**: Synapse Real-Time Analytics is designed to handle observational data from various sources, processing semi-structured data in high volumes. It provides real-time insights, ensuring that businesses can make informed decisions based on the latest data.

- **Power BI**: Power BI is a comprehensive collection of software services, apps, and connectors designed to transform unrelated data sources into coherent, visually immersive, and interactive insights. Whether data resides in an Excel spreadsheet or a combination of cloud-based and on-premises data warehouses, Power BI facilitates easy connection, visualization, and sharing of insights. Its compatibility with both on-premises and cloud solutions ensures that businesses can choose where to store their reports based on their specific needs.

- **Data Activator**: Data Activator offers a no-code experience, empowering business analysts to drive actions automatically from data. This simplifies the process of data activation, making it easy for users to derive actionable insights from their data.

Benefits of Microsoft Fabric

Enterprises are in constant pursuit of tools that can seamlessly integrate vast amounts of data, offering insights that drive decision-making and innovation. While many solutions promise comprehensive analytics capabilities, Microsoft Fabric delivers a truly integrated experience tailored to the diverse needs of modern enterprises. It is designed to elevate simplicity and integration, offering a comprehensive suite of analytics services. When compared to traditional data lakes and warehouses, Fabric offers several advantages:

- **Comprehensive analytics suite**: Fabric serves as an all-encompassing analytics solution, covering data movement, data science, real-time analytics, and BI, streamlining the analytics process.
- **Integrated ecosystem**: Fabric provides an integrated ecosystem of services, including data lake, data engineering, and data integration. This ensures smooth dataflow throughout the analytics pipeline, reducing data silos and promoting efficiency.
- **Simplified deployment and management**: Being a SaaS platform, Fabric manages infrastructure setup and maintenance, allowing organizations to focus on analytics rather than infrastructure management.
- **Scalability and flexibility**: Designed to scale with ease, Fabric can handle varying workloads, ensuring efficient data processing without performance issues.
- **Real-time analytics**: Fabric supports real-time analytics, enabling organizations to derive insights from live data streams and make informed decisions promptly.
- **Advanced data science capabilities**: The platform offers tools for data scientists to conduct intricate data analyses, develop machine learning models, and gain predictive insights.
- **End-to-end solution**: Fabric provides a holistic analytics solution, streamlining the entire workflow from data ingestion to visualization.
- **Security and compliance**: Being part of the Microsoft ecosystem, Fabric ensures robust security measures and compliance certifications, safeguarding sensitive data.
- **User-friendly interface**: Designed to be accessible, the platform's interface caters to both technical and non-technical users, democratizing data analytics across organizations.
- **Cost-effectiveness**: By integrating a range of services, Fabric can lead to cost savings compared to managing multiple separate analytics solutions.

By offering these services and features, Fabric ensures that enterprises have a comprehensive set of tools to handle all their analytics needs, from data integration to deriving real-time insights.

AI integration

Azure OpenAI Service is built on Azure's purpose-built, AI-optimized infrastructure. This infrastructure powers various Microsoft products. AI integration is a cornerstone of Microsoft Fabric, setting it apart from other analytics platforms:

- **Unified AI-powered platform**: Fabric is not just an analytics platform; it's an AI-powered analytics platform. This means that every layer of Fabric is integrated with AI capabilities, making data analytics more intuitive, efficient, and powerful. This allows organizations to harness the full potential of their data.

- **Generative AI and language models**: Fabric uses generative AI and language model services, such as Azure OpenAI Service. These services enable the creation of everyday AI experiences that redefine how employees utilize their time. For instance, powering organization-specific AI experiences requires a steady stream of clean data from a well-integrated analytics system. Fabric ensures that this data is readily available and processed efficiently.

- **Copilot in Microsoft Fabric**: One of the standout features of Fabric is Copilot. With Copilot integrated into every data experience in Fabric, users can employ conversational language to perform a range of tasks. This includes creating dataflows and pipelines, generating code, building machine learning models, and visualizing results. This conversational AI approach simplifies complex tasks and makes the platform more user-friendly. Moreover, Copilot adheres to commitments made by Microsoft in terms of data security and privacy, ensuring that organizational data remains protected.

- **AI-driven insights**: Fabric and Power BI integration enables AI-driven analytics, enabling business analysts and users to derive valuable insights from data. The Power BI experience within Fabric is also deeply integrated into Microsoft 365, ensuring that insights are readily available where business users operate.

- **Data Activator**: Although this feature is coming soon, Data Activator in Fabric offers real-time detection and monitoring of data. It triggers notifications and actions based on specified patterns in the data, all within a no-code environment. This AI-driven feature further enhances the platform's capabilities to provide timely and relevant insights.

Responsible AI approach

Microsoft recognizes the importance of responsible AI, especially with powerful technologies such as generative models. They have taken an iterative approach to large models, working closely with OpenAI and their customers to assess use cases and address potential risks.

Azure OpenAI Service has implemented guardrails that align with the Microsoft Responsible AI principles. Developers are required to describe their intended use case or application before accessing the service. Content filters are in place to monitor both the input provided to the service and the generated content to ensure it adheres to policy guidelines.

Microsoft has identified the following six principles to guide AI development and use:

- **Fairness**: Microsoft emphasizes the importance of fairness in AI systems. The company seeks to ensure that AI systems allocate opportunities, resources, or information in ways that are equitable to all users.
- **Reliability and safety**: Microsoft prioritizes the reliability and safety of AI systems, ensuring that they function consistently and prevent any potential harm.
- **Privacy and security**: Protecting user data is paramount. Microsoft ensures that AI systems uphold the highest standards of privacy and security, safeguarding user information from unauthorized access and breaches.
- **Inclusiveness**: Microsoft believes in creating AI systems that cater to everyone, ensuring that all users, regardless of their background or abilities, can benefit from the technology.
- **Transparency**: Microsoft is committed to making AI systems transparent, ensuring that users understand how these systems operate and make decisions.
- **Accountability**: Microsoft holds itself accountable for the AI systems it develops, ensuring that there are mechanisms in place to address any issues or concerns related to the technology.

Essentially, these principles for guiding responsible AI development prioritize fairness, safety, privacy, inclusivity, transparency, and accountability, ensuring AI technologies are trustworthy and beneficial for all users.

Simplified pricing model

Microsoft Fabric has developed a streamlined pricing model, designed to optimize efficiency and align with enterprise financial goals. Central to this strategy is the acquisition of a single set of compute credits, dedicated to powering all operations within the Fabric framework.

The pricing model is built around a unified compute credits system. Be it data engineering, integration, or analytics, Fabric employs a consolidated credit pool. This approach offers both operational flexibility and potential cost savings. While there are specialized tools that cater to specific needs, Fabric integrates various functions into one cost-efficient platform, facilitating easier financial planning. However, challenges might arise when different departments, such as data engineers and business analysts, have separate budgets. Collaborative decision-making becomes essential for efficient credit allocation.

The Microsoft Fabric comprehensive pricing model offers:

- **Cost-effectiveness**: Merging diverse analytics tasks into one pricing structure can lead to substantial savings. It eliminates the hassle of tracking individual service costs, ensuring more predictable expenses.
- **Adaptability**: The shared credit system is versatile. Credits can be channeled to where they're most needed, aligning with ever-changing business requirements.
- **Uncomplicated choices**: A unified pricing model cuts through the maze of multiple service costs, streamlining decisions and trimming administrative tasks.
- **Boost in platform usage**: Fabric offers straightforward pricing and comprehensive solutions, which can be a magnet for businesses.
- **Future readiness**: To address the expansion of companies and their shifting data needs, this pricing model helps organizations forecast growth and understand variable analytics demand.

Code faster with GitHub Copilot, Visual Studio Code, and Azure Databricks

Tools such as GitHub Copilot, powered by OpenAI Codex, are changing how organizations design and use systems. When combined with application features in Visual Studio Code, Copilot offers a top-notch experience for developers. But to create AI systems such as GitHub Copilot, a simple code editor isn't enough. This is where the Databricks extension for Visual Studio Code comes in handy, which lets developers work in Visual Studio Code but use the power of Azure Databricks for deploying their projects. It's a blend of easy development and powerful execution, with the coding tips from Copilot.

Create an AI coding helper for Azure Databricks

This tutorial will show how you can code faster using Visual Studio Code, GitHub Copilot, and the Databricks extension for Visual Studio Code. With the Databricks extension for Visual Studio Code, any code made in Visual Studio Code can run smoothly on Azure Databricks. It also helps in logging models, making deployment easy.

To follow this example, you need to:

- Have Visual Studio Code 1.69.1 or higher installed and configured for Python coding.
- Ensure Visual Studio Code is running and has a local project opened.
- Generate a Azure Databricks personal access token for your target Azure Databricks workspace.
- Add your Azure Databricks personal access token as a token field along with your workspace instance URL (e.g., `https://dbc-a7h3345c-d6e7.cloud.databricks.com`) to the `DEFAULT` configuration profile in your local `.databrickscfg` file.
- Make sure Visual Studio Code is set up for Python coding. This typically involves installing the Python extension from the Visual Studio Code Marketplace and configuring the Python interpreter.
- Generate an Azure Databricks personal access token for your target Azure Databricks workspace. This token will be used to authenticate your Visual Studio Code with Azure Databricks.

- Have access to GitHub Copilot. Please review the following for more information on a free trial: https://github.com/features/copilot.

Install the Databricks extension for Visual Studio Code

To install the Databricks extension for Visual Studio Code:

1. Open Visual Studio Code.
2. Go to the `Extensions` view by pressing `Ctrl+Shift+X` or clicking on the square icon on the sidebar.
3. In the search bar, type `Databricks extension for Visual Studio Code`.
4. From the search results, find the extension and click the green `Install` button.
5. Wait for the installation to complete.

Configure the Databricks extension for Visual Studio Code

To configure the Databricks extension for Visual Studio Code, follow these steps:

1. Click the Azure Databricks logo on the sidebar of Visual Studio Code.
2. In the `Configuration` pane that appears, click `Configure Databricks`.
3. A command palette will pop up asking for the `Databricks Host`. Enter your Azure Databricks workspace URL and press `Enter`.
4. Next, you'll be prompted to authenticate. Click on `DEFAULT: Authenticate using the DEFAULT profile`.
5. To set up your Azure Databricks cluster, go back to the `Configuration` pane, click `Cluster`, and then click the gear icon that appears. This will let you select and configure the cluster you want to use.
6. If your chosen cluster isn't running, click the `Start Cluster` icon next to `Cluster`.
7. Now, set up the sync destination. In the `Configuration` pane, click `Sync Destination`, and then click the gear icon. A command palette will pop up with a randomly generated sync destination name. Click on it to select.

Install the GitHub Copilot extension

To install the GitHub Copilot extension, follow these steps:

1. Open Visual Studio Code:
2. Press `Ctrl+Shift+X` or click on the square icon on the sidebar to open the Extensions view.
3. Type `GitHub Copilot` in the search bar and locate the GitHub Copilot extension in the search results.
4. Click the green `Install` button next to the GitHub Copilot extension and wait for the installation to finish.

Once installed, the GitHub Copilot extension will prompt you to authenticate using your GitHub credentials. Follow the on-screen instructions to log in and grant the necessary permissions. With both extensions now installed and configured, you're all set to develop code in Visual Studio Code, using the capabilities of Azure Databricks and the AI-powered suggestions from GitHub Copilot.

Writing a Python script with AI-powered suggestions

Follow these steps to generate a Python script with AI-powered suggestions:

1. Create a new Python file.
2. In Visual Studio Code, click `File > New File`.
3. Save the file with a `.py` extension, for example, `script.py` or `codehelping.py`.
4. Start typing your Python code. As you begin typing, GitHub Copilot will automatically provide real-time code suggestions.
5. For instance, type `How to create a Azure databricks dataframe in python`.

6. Once prompted, you will see a result similar to *Figure 2*.

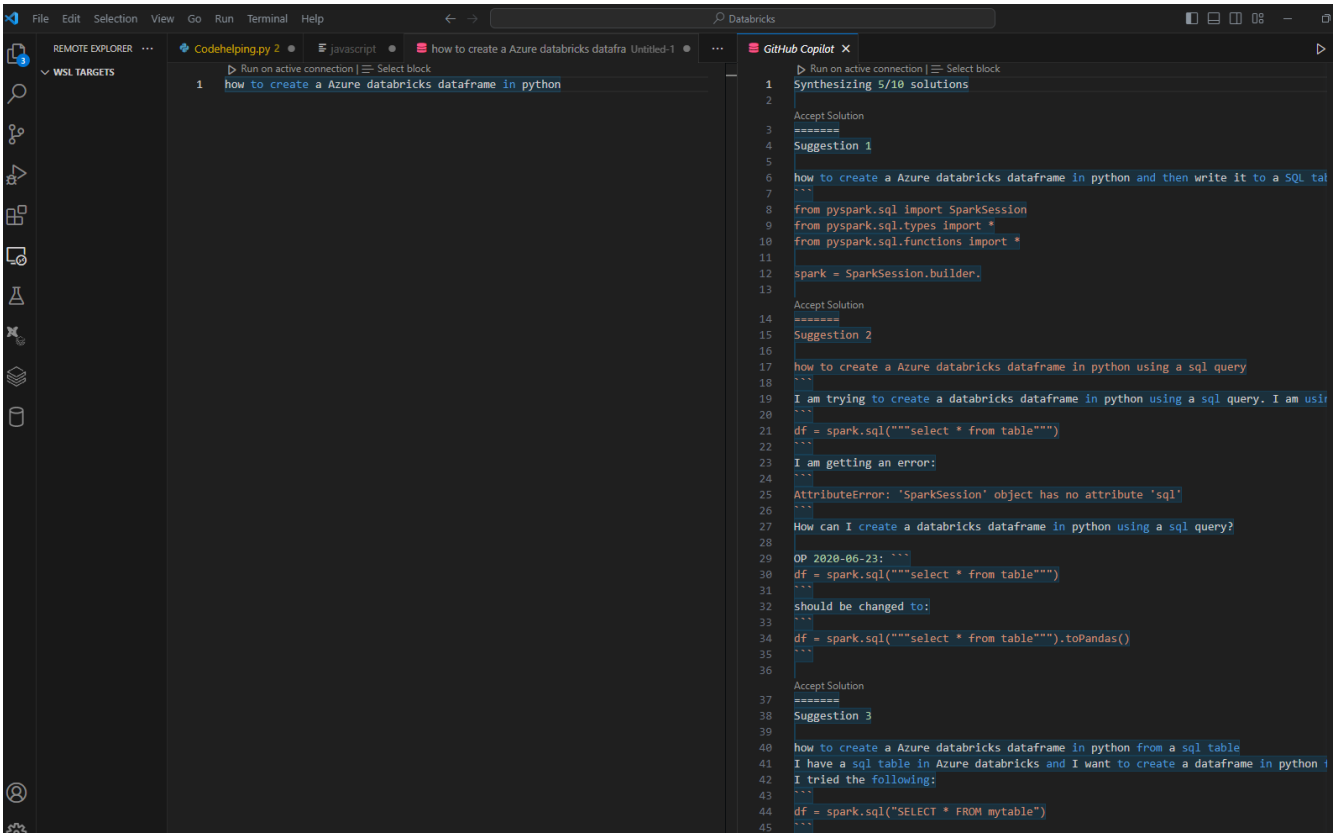

Figure 2: Python script generated by AI

This demonstrates the power of Azure Databricks as a leading platform for data and AI when combined with developer tools such as Visual Studio Code and GitHub Copilot.

Integrating Azure Databricks with OneLake

OneLake, within Microsoft Fabric, serves as a unified location for storing an organization's data. The power of the lakehouse architecture comes from allowing multiple analytics engines to process the data stored in OneLake. These could be engines within Fabric, such as Synapse Data Warehouse. However, other Azure services, such as Azure Databricks, can also process data stored in OneLake. This enables organizations to utilize the full power of Azure within the Fabric data estate.

The following tutorial shows how to connect to OneLake via Azure Databricks. Connecting to OneLake enables reading and writing data to Microsoft Fabric Lakehouse from an Azure Databricks workspace.

To follow this example, you will need:

- An established Fabric workspace and Fabric Lakehouse
- A premium Azure Databricks workspace

> **Note:**
> Only premium Azure Databricks workspaces support Microsoft Azure Active Directory credential passthrough, which is essential for this integration.

Step-by-step guide

Follow these steps to integrate Azure Databricks with OneLake:

1. To get started, you will need to configure your Azure Databricks clusters to use `credential passthrough`. This allows Azure Databricks to use your user identity when connecting to an Azure Storage account, in this case your OneLake account.

2. Open your Azure Databricks workspace. In the pane on the left, select `Compute`. Then click the Create `compute button`.

3. Follow your usual procedure for creating a new Azure Databricks cluster. However, be sure to expand the `Advanced options` section and activate the `Enable credential passthrough for user-level data access` option. This will enable you to use your Microsoft user identity to connect to OneLake.

4. When you have set all your desired configuration options, click `Create compute`.

5. Once your new cluster is ready, click on `Workspace` in the panel on the left side of the screen. Click the `Add` button in the upper-right corner of the screen and select `Notebook` from the drop-down menu. This will create a new Azure Databricks notebook.

6. When the notebook opens, use the drop-down menu at the top of the notebook to make sure you are connected to the new cluster you just created.

7. To continue the tutorial, you will need to get the details of your OneLake storage account. To find this information, head to your Fabric Lakehouse and locate the Azure Blob Filesystem (ABFS) path. This can be found in the `Properties` pane of the workspace page.

> **Note:**
> Azure Databricks exclusively supports the ABFS driver for interactions with ADLS Gen2 and OneLake. An example path might look like this: `abfss://myWorkspace@onelake.dfs.fabric.microsoft.com/`.

8. You are now ready to use Azure Databricks to load data into your OneLake storage account. Paste the following code into a cell in your notebook. This code will create a new dataset in Microsoft Fabric based on sample data in Azure Databricks:

```
oneLakePath = 'abfss://myWorkspace@onelake.dfs.fabric.microsoft.
com/myLakehouse.lakehouse/Files/'

# Load data from a Databricks public dataset into a dataframe.
#Alternatively, you can source data from elsewhere in Fabric or
#from another ADLS Gen2 account you own. For example:

yellowTaxiDF = spark.read.format("csv").option("header", "true").
option("inferSchema", "true").load("/databricks-datasets/nyctaxi/
tripdata/yellow/yellow_tripdata_2019-12.csv.gz")

#Process your data as needed. This could involve filtering,
#joining with other datasets, or other transformations. For
#instance:

filteredTaxiDF = yellowTaxiDF.where(yellowTaxiDF.fare_amount<4).
where(yellowTaxiDF.passenger_count==4)
display(filteredTaxiDF)

#Write the processed dataframe to your Fabric Lakehouse using the
#OneLake path you saved earlier:

filteredTaxiDF.write.format("csv").option("header", "true").
mode("overwrite").csv(oneLakePath)

#To ensure your data was written successfully, read the newly
#loaded file and display a sample:

lakehouseRead = spark.read.format('csv').option("header", "true").
load(oneLakePath) display(lakehouseRead.limit(10))
```

With this tutorial, you've successfully integrated Azure Databricks with OneLake in Fabric. You can now seamlessly read and write data in Fabric using Azure Databricks.

Conclusion

Effective data management and analytics are paramount for businesses to gain a competitive edge. Microsoft Fabric is a pivotal solution, offering a unified, scalable, and secure platform designed with the needs of modern enterprises in mind. This platform not only ensures seamless data integration and management but also fosters collaboration, bolsters security, and drives innovation.

Next steps

- Learn more about Microsoft Fabric, and get started with a free trial.
- Enable data, analytics, and AI use cases with Azure Databricks.
- Learn how to create a lakehouse with OneLake in Fabric.