Microsoft Fabric Mega-FAQ

Fabric CAT

Summary

This document is a collection of frequently asked questions from customers, the field, partners, and MVPs. This document is currently internal only, but a lot of the information will likely be released publicly in a FAQ eventually. The purpose of this document is to equip Microsoft FTEs with the information they need to have conversations and build solutions with Microsoft Fabric.

Contents

General

What is Microsoft Fabric

Fabric is an all-in-one analytics solution that covers everything from data movement to data science, real-time analytics, data engineering, data warehousing, business intelligence, and more. Learn details in the Fabric Documentation.

What is the impact of turning Fabric on/off?

Fabric can be disabled at the capacity level. All fabric workloads store their data in OneLake. Even after turning off Fabric, Fabric's artifacts and associated data don't get deleted automatically. Users could still view those artifacts in the workspaces but can't be edited. So, there shouldn't be any impact on the capacity just because Fabric is turned off.

In Fabric, what happens to the things IT would typically manage, like DW's, Spark clusters, etc?

Microsoft Fabric is a SaaS service where you no longer have to worry about managing PaaS resources like clusters.

How do professional developers collaborate with low-code ones, business users, analysts, and other personas?

There's always way more business users and low code developers than there are high code developers or people available in the central IT teams. The more we can empower these low code and high code folks to work together, the more we can unblock the business to move at the pace they need to while the high code developers focus on the really hard technical stuff.

We don't expect business users to write python code, nor do we expect high code developers to become low code developers - but the more we can empower these teams to collaborate, hand off work, and improve time to value, the better off everyone is!

We view Fabric as the perfect environment where Business and IT can meet and collaborate.

We provide a set of workloads that provide a spectrum from Pro-Dev experiences to Citizen dev experiences.

We believe that pro-devs would love the productivity of the SaaS platform and the easy UX. We also believe that some citizen developers will choose to skill-up and use some of the pro-dev capabilities.

At the end of the day, creators will choose which of the workloads and technologies they feel comfortable with and will work with others who have the skills to complement them.

Collaboration across different personas is a key design point of Fabric.

Power BI users are often on a spectrum from an end user who opens a report to get a number, someone who slices / explores the data, someone who does light creation, more advanced authors, all the way to professional BI folks who build complex semantic models in Power BI.

The goal with Microsoft Fabric and Power BI is to help empower everyone to move slightly up the spectrum to be empowered to do more with their data. Maybe that means a consumer of BI reports gets fresher data because the report was built with DirectLake mode, or a report developer can easily build off of a DW built in Fabric with fewer hurdles, or a Pro BI developer not having to deal with ETL and incremental refreshing large datasets since they can use data directly in OneLake.

Analytics is a team sport, the more we can empower teams to work together and not in their own individual silos, the faster time to value.

Azure Synapse Analytics

What is happening to Azure Synapse Analytics now that Fabric is announced?

Azure Synapse Analytics is here to stay with maintenance continuing for many years. Everything you build will continue to work as these PaaS services are not going anywhere.

The DW in Fabric represents the next gen of DW architecture and most of the future innovation will happen there, but you may stay on dedicated pools for many years with no pressure to move.

If you need GA products for your solutions today, then PaaS is the way to go. If you are looking to start a new project, you should evaluate Fabric's capabilities for improved functionality.

Is there a plan for Fabric DW features to be integrated into Azure Synapse Analytics?

No, we made significant changes in Fabric that cannot be ported to the PaaS engine because it is not compatible with the architecture.

What is the process to migrate from Azure Synapse Analytics to Microsoft Fabric?

We have plans to enable easy migrations into Fabric. These plans are not final at the time of this writing.

Synapse and ADF pipelines will be able to be imported into fabric.

Existing Azure resources will be able to be mounted into Microsoft Fabric so that you can access your data through OneLake and experience the benefits of all the new features while keeping your existing applications as-is without migration.

You can start using OneLake as your storage tier for pipelines since OneLake is ADLS compatible.

We are building migration tools that will allow you to transfer your current investments in T-SQL code for ETL in Synapse dedicated pools into Microsoft Fabric. If your current solution is already designed in a lake-first architecture, then the transition will be simpler. If you currently do your transformations in DW then this code will need to be transferred over to work in Fabric DW.

How does a Fabric Workspace compare to a Synapse workspace and Power BI workspace?

Conceptually, they are all of group of resources in a project. Fabric workspaces are more similar to Power BI workspaces and can include Power BI artifacts as well as Fabric Artifacts.

Are there plans for Synapse Link in Fabric?

Making significant investments in what's known today as "Synapse Link", which is the ability to reflect an OLTP DB thru CDC streams (Change Data Capture) into OneLake. Coming in the next 6 months.

Licensing/Billing

What happens to my existing Synapse reserved instances once Fabric is released?

We are working on a plan to convert reserved instances into Fabric capacity, but this is not yet finalized.

What happens to my existing PowerBI P-SKU licenses and capacities?

PowerBI P-SKUs will become Fabric capacities. All PowerBI items will continue to work as usual and v-cores will translate into Fabric CUs – eg. P1 = F64. OneLake requires a capacity and DirectLake requires OneLake, so you must have a capacity to use DirectLake.

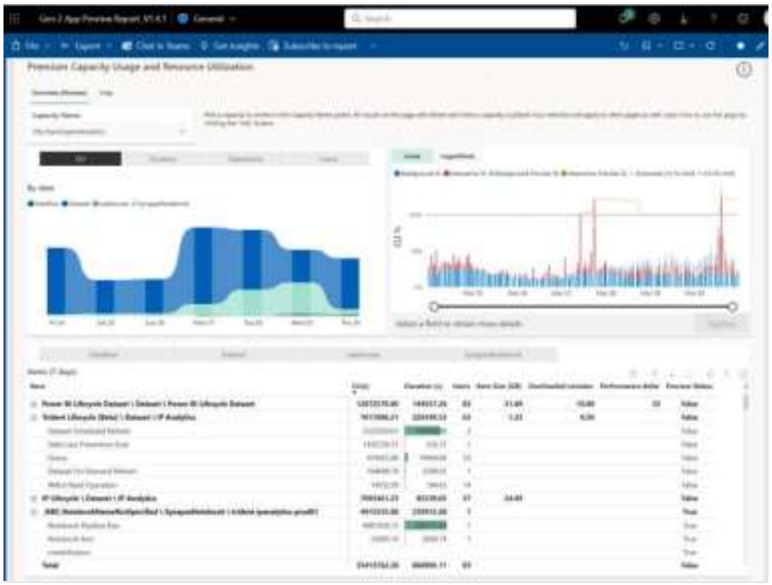
In Fabric we are lowering the entry price for capacities, they will start around $160 a month with an annual commitment – essentially 8 PPUs.

You can use the capacity for everything in Fabric including as many lakehouses and warehouses as you want across as many workspaces as you like; consumed compute will be your limit. It is not a per-user fee meaning you can have many users share a single capacity and it is not limited to a single workspace.

Fabric capacities smaller than F64 (equivalent to Premium P2) require a Pro License for report consumption. You most likley don't need a PPU if you have a Fabric Capacity. For capacities F64+ you only need PRO for report authoring.

What happens when my existing Power BI premium capacity when Fabric is enabled in my tenant?

If your administrator has turned on Microsoft Fabric at the tenant level, all Premium capacities will automatically be upgraded to support Microsoft Fabric. In addition, capacity administrators can turn on Microsoft Fabric at the capacity level (as described int the previous section), allowing anyone with access to this capacity to use the new Fabric experiences. To help you prepare for how this will impact your usage of Power BI Premium, new Fabric experiences will not draw down usage from your capacity before August 1st, 2023. You can, however, monitor how Microsoft Fabric impacts your capacity usage through the Capacity Metrics app:



Universal Fabric capacity is free to try for 60 days. If you do not yet have Power BI Premium, you can get access to a free Fabric trial (Learn more about the Fabric trial here). If trials are enabled by your tenant administrator, you will automatically be granted a Fabric trial capacity when you

(1) create an item that requires a capacity

or

(2) lick on "Start trial" from your "Account Manager" in the upper right-hand side of the portal.

This capacity can be used with one or more workspaces, allowing you to create data warehouses, lakehouses, notebooks, and more. After 60 days, you can purchase Fabric capacities in the Azure Portal (available after June 1 st). And as a tenant admin you will have visibility into all active trial capacities provisioned for users within the tenant in the Power BI admin portal. Purchase Fabric Capacities in Azure Starting on June 1st, you will be able to purchase Fabric Capacities in Azure. Login to the Azure

Portal or navigate to this link, and search for Microsoft Fabric in the search bar. Instructions on how to create a new Fabric Capacity are here

How is storage billed in Fabric?

Storage charges are part of capacity charges. Each capacity has two meters – a compute meter and a storage meter. This allows you to ensure storage is paid by those who consumed it.

This will be familiar for Synapse users but different than historical PowerBI storage billing, which was included in the SKU, because PowerBI datasets tended to be small. In Fabric we are looking at petabyte-sized lakehouses and we cannot give a fixed price for unlimited storage.

Unofficial/not final: There will likely be some included storage in Fabric that will allow you to get started. You will also not need to pay for the storage of the Power BI datasets.

OneLake

What is OneLake?

OneLake is an abstraction layer over Azure Storage and third-party storage services (AWS S3, GCP) attached to a Fabric tenant. It comes pre-provisioned with each Fabric tenant, and it enables building a single logical lake over 1P and 3P storage. It is pre-wired to Fabric data artifacts, and all the data maps to a unified Fabric API (application programming interfaces) and can be accessed by ADLS APIs as if they were a single physical storage account.

Can users create OneLake?

Users cannot create OneLake storage. OneLake storage (ADLS Gen2) managed by OneLake API is attached to Fabric tenant. When a workspace is created, a folder is created in OneLake storage (ADLS Gen2 behind the scenes) on a customer tenant. Data persisted as files, tables in a Lake House, Data Warehouse is stored in this workspace folder.

Is there any file format not supported in OneLake?

There are no restrictions on the type of files which can be stored in OneLake. However, to surface the data via Lakehouse or Warehouse artifacts, data must be stored in Delta\Parquet format. Furthermore, any Delta\Parquet format files can also be automatically scanned and registered as tables in the Lakehouse artifact.

We already use ADLS Gen2 for storing all our data, do we need to copy that data across to OneLake so that we can analyze it using one of Fabric analytics engines?

Shortcut for ADLS Gen2 can be used to read data from existing ADLS Gen2 storage without having to copy data physically to OneLake.

What is the security model for OneLake?

OneLake brings flexible security to the data lake with multi-layered access controls. Grant access to data directly in OneLake or limit access to specific query engines to keep sensitive data secure. OneLake adds a layer of security to Fabric by allowing data lake access to be configured independently of Fabric item access.

The following security modes are supported in OneLake:

- Access to a specific subsection of the lake, such as a workspace or lake house.
- Access to a specific query engine, but no direct OneLake access.

Data Integration

What is Data Integration in Fabric?

Data Integration in Fabric brings the best of ADF and Power Query together in one experience. The goal is to make sure Data Integration in Fabric is for both Citizen and Pro Data Developers.This brings low-code, AI-enabled data preparation and transformation experiences, petabyte-scale transformation, 100s of connectors with hybrid, multi-cloud connectivity, governed via Purview with enterprise promises of Data/Op, CI/CD, Application Lifecycle management, and monitoring.

What is Fabric Data Factory?

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

What is Dataflows Gen2?

This is the evolved generation of Dataflows Gen1. Dataflows provide a low-code interface for ingesting data from 100s of data sources, transforming your data using 300+ data transformations and loading the resulting data into multiple destinations such as Azure SQL Databases, Lakehouse, and more.

Dataflows Gen2 is a substantial evolution of the Power BI / Power Platform Dataflows Gen1, with a number of improvements that provides a complete ETL/ELT data integration experience.

- Dataflows Gen2 introduces the concept of Output Destinations (targets), enabling writing out the results of the transformation into a number of destinations (Fabric/Synapse Lakehouse, Warehouse, Real Time Analytics, and SQL; more to come).
- Dataflows Gen2 by default uses Fabric Lakehouse for staging the query results, making it perform a lot better when you are connecting to / consuming your dataflows as a data source.
- Dataflows Gen2 is built on top of Fabric compute engines, bringing a level of scale to your transformations that was previously not possible.
- Dataflows Gen2 will soon integrate the petabyte scale copy (first available in Azure Data Factory, now also available in Pipelines in Fabric) as part of the data flow "Get Data" experience. This will enable even faster data import/copy as part of the dataflow.
- Dataflows Gen2 fully integrates with Monitoring as part of the Fabric Monitoring hub.
- Also improved the overall authoring/save model as part of the overall experience enhancements.

Dataflows Gen2 is a Data Factory (Microsoft Fabric) feature – and works with the capacity-based licensing constructs in Fabric (Fabric Capacities) and Power BI Premium Capacities.

What is Fabric Pipelines?

Fabric 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 a number of different tasks at scale. Control flow capabilities are built into pipelines that will allow you to build workflow logic which provides loops and conditional

What is a connector?

Data factory offers a rich set of connectors that allow you to connect to different types of data stores. You can leverage those connectors to transform data in Dataflow Gen2 or move PB-level of dataset with high-scale in Data pipeline

Are connectors shared at the workspace or tenant level?

Connectors are shared at the workspace level

Will I be able to use spark compute for scale-out processing?

Yes, this is planned to be available by GA.

Power Query uses Gateway and not SHIR for on-prem data ingestion. Is this still the case for Fabric?

We plan to have a unified on-prem data gateway on Fabric, so SHIR will be integrated with on prem data gateway, which can also get the benefit of the load balance of on prem data gateway.

If the Fabric Datawarehouse is in a VNET, how does Power Query connect to the database?

You can install on-premise data gateway to connect the database.

1. Follow these instructions to install the on-premises data gateway: Install an on-premises data gateway | Microsoft Learn
2. Follow these steps to add the Lakehouse connector to the gateway folder: Use custom data connectors with the on-premises data gateway - Power BI | Microsoft Lear

Does Fabric data integration convert the source sink data into Delta/Parquet?

You can use Dataflows or Copy activity to convert the data into Delta/Parquet.

Data Engineering

What are performance improvements on delta support? Will it support features like optimize write, Z-Order, and delta caching for better performance?

Optimize, Z-order and but there are no plans for delta caching for delta logs. However Intelligent cache will cache delta files.

Commented [AA1]: V-Ordering or VertiParquet is differentiator here, which is unique to Fabric.

What will be the options to run 100s of jobs concurrently as part of ETL packages? Will it support sharing the same cluster across different spark applications launched by users & MSI?

Instant attach will be enabled for users and pipeline for sharing Spark cluster across applications.

Commented [AA2]: With High Concurrency feature, you should be able to execute multiple notebooks on the same underlying Spark cluster by a user. This will save cost for customers.

What are cost control features to minimize the cost of a non-production environment? For example, support for a single node cluster instead of 3 nodes?

The number of executors and its compute to be dynamically allocated by Fabric Spark scale service.

Commented [AA3]: Yes, you can do it with custom pool. You can disable auto-scaling and set 1 as node/executor.

With starter pool auto-scaling is enabled by default, which starts from 1 node and can go up to 10 nodes (or 9 executors).

What will be the average startup time for a new spark application especially for a Data scientist persona? Do they need to wait for 2 to 3 minutes?

Live pool will be able to reduce the startup time to less than a minute.

Commented [AA4]: With starter pool, the session should be up and running in 10-15 seconds. If you are using custom pool, it might take 2-3 minutes. However, we are coming up with live pool of custom pool, which will pre-provision resources and you should be up in few seconds.

How can you add security to artifacts like notebooks, spark pools etc. across different teams (like Data engineers, Data scientists) working on an enterprise. For example, how to hide a notebook with key business logic from other developers?

Each team will have a folder with security controls.

Will Spark support multiple IDE like VS Code/PyCharm/ IntelliJ, including a complete interactive notebook experience?

VS Code extension for notebooks/Spark currently in development that features a local-only development experience.

Commented [AA5]: This is currently available. Launch the Fabric VS Code extension, navigate through your workspace and work with Notebooks, Spark Jobs and Lakehouses directly in the IDE.

Fully remote way of working with vscode.dev is coming soon.

What will be the mechanism for continuous Spark streaming with mission-critical workloads?

Long running Spark streaming cluster will be supported in Fabric.

Commented [AA6]: This is currently supported. You can run your streaming job as SJD and setup for automatic restart of this job.

Go to Settings - Optimization and define retry policy.

Can the developer integrate .whl and python packages directly from the notebook without pool startup?

There will be an ability to specify package deployment at workspace level and that means every Notebook will have that package pre-deployed as well as an ability to deploy a package at notebook level with "%pip install" command.

Commented [AA7]: along with pip install you can also use conda install

Will Fabric support R for data scientist persona?

Yes, R will be supported

What is the authentication supported for users to interactive with Spark?

It will be based on AAD authentication

How will Fabric integrate with the AML service?

We have native integration with AML within Synapse already, however it lacks features like model tracking, execution tracking etc. In Fabric, these gaps are expected to be bridged so that customers don't need to switch back and forth between AML and Fabric

How customers will be able to set # of executors and machine type?

Customer will not have to set, it will be set by engine (planned). It will be truly dynamic scaling. However, customers will have choice to choose either use CPU vs GPU based clusters

Commented [AA8]: Yes, you are. In addition to Starter pool, you can also use custom pool if you need more control - size, number, scaling of nodes etc.

If you are using custom pool, it might take 2-3 minutes for session to start. However, we are coming up with live pool of custom pool, which will pre-provision resources and you should be up in few seconds.

Lakehouse

What are shortcuts and how are they used?

Shortcuts provide a way to connect to existing data without having to directly copy it to OneLake. This eliminates the need to set up and monitor data movement jobs and keep data in sync across sources. Shortcuts are embedded references within OneLake that point to other files' store locations. They can only be created in Lakehouses. The embedded reference makes it appear as though the files and folders are stored locally but in reality; they exist in another storage location.

From within the Lakehouse artifact, shortcuts can be created to point to anything else in OneLake, ADLS Gen2, or S3 (and eventually GFS), given the appropriate permissions.

Read more about OneLake shortcuts in Fabric Documentation

What are the performance and cost considerations for using shortcuts vs ingesting the data into the lakehouse?

Commented [NS9]: additional question - under the best case scenario for shortcuts with a close region, etc, will importing into lakehouse be similar performance or should the imported version still be faster?

A Shortcut is similar to a symbolic link, and as a result, performance characteristics will be heavily influenced by factors such as physical location of a file; distance of external storage from a Fabric tenant; throughput and concurrency supported by external storage where the files physically reside.

In terms of egress, costs will vary depending on whether a Shortcut points to ADLS storage; AWS S3; or GCP Storage. Data movement should be minimal because Shortcuts do not physically move entire datasets and uses advanced caching for higher performance and improved reusability of data.

For data sources that are not supported by shortcut (e.g. Blog, SQL DW, Snowflake), you can use Pipelines or Dataflows depending on your needs to ingest your data into the Lakehouse. (There's a feature being developed that will allow "mounting" DB engine-based systems, to act similar to a shortcut. This may sound familiar, as it derives from Azure Synapse Link functionality.)

Is there support to create shortcuts to Snowflake?

No, because Snowflake does not use an open format. They currently have beta support for Iceberg and we may support Iceberg in the future enabling a Snowflake/Fabric scenario, but as of now we cannot read their proprietary storage format.

Is there support to create shortcuts to Dataverse?

Dataverse will support shortcuts in OneLake with no data movement required. It is currently in private preview and you can read about it here: [New Dataverse enhancements and AI-powered productivity with Microsoft 365 Copilot - Microsoft Dynamics 365 Blog]

What is the difference between the 'tables' folder and 'files' folder in the lakehouse?

When a Lakehouse is created, it is created with two physical locations.

The "Tables" folder represents the managed portion of the Lakehouse to host tables of all file types (CSV, Parquet, Delta, managed, unmanaged tables). All tables, whether automatically or explicitly created, will show up as a table under this managed area of the Lakehouse.

The "Files" folder represents the unmanaged area for your Lakehouse to store data with virtually any file formats. Any delta lake format files (parquet + transaction log) stored in this area are not automatically recognized as tables. If a user wants to create a table over a delta folder in the unmanaged area, they will have to explicitly create an external table with the location pointer to the unmanaged folder containing the delta lake file

Commented [AA10]: Creating an external or unmanaged table is not currently recognized by SQL endpoint or DirectLake/PBI and coming soon though.

Can I use Databricks instead of Notebook for data engineering?

Yes, you can use Databricks to read the data from Lakehouse, transform it, write it back to Lakehouse.

How can you integrate Databricks Lakehouses in Fabric for a medallion architecture?

We chose to standardize on Delta Parquet partly because of its wide use by Azure customers and our partnership with Databricks. Customers can keep their data in their Databricks ADLS gen2 and use shortcuts to consume the data from their GOLD medallion layer in a Fabric lakehouse. All engines will work on top of this, including Power BI.

In the future we believe customers will find our spark engine to be valuable in these scenarios.

Can you use DirectLake datasets directly on Databricks tables through shortcuts?

You can use DirectLake datasets directly on Databricks tables through shortcuts, but unfortunately Unity Catalog in Databricks is not an open format at the moment. This means you will still be able to connect to the delta tables, but it would be a direct link to the table in the storage layer. Any read operation within Fabric wouldn't make any changes to the tables.

With that said, non-vOrdered tables will not have the same performance as tables that have vOrder applied. If you want to apply vOrder to the delta tables, you can do that in a Spark notebook which would do an in-place optimization with the vOrder parameter.

Are there any special best practices for medallion architecture for lakehouses in Fabric?

The preparation of a Medallion lakehouse is exactly the same, nothing changes with Fabric. However, Fabric may provide more options and features that you may opt to use. For example, typically the GOLD layer feeds a DW for serving users. In Fabric, your GOLD tables could directly serve queries through the SQL Endpoint without the need to load it into a separate DW and also serve Power BI with DirectLake. Also, you may allow your data scientists to access the SILVER layer to produce more data with the Data Science engine. This results in more flexibility in your solution.

Warehouse

What is the difference between a Lakehouse and a Warehouse in Fabric?

A lakehouse is a DB whose tables are managed by Spark.

Lakehouse facts:

- A lakehouse at its base is just a collection of folders and files in the lake.
- A special folder named ‘Tables’ holds structured data.
- The “files” folder can hold anything.
- It is simple, no measures, no relationships.
- A SQL Endpoint is automatically created that allows you to run SQL queries on top of the structured data files in the ‘Tables’ folder. This essentially makes it a read-only DW.
- The SQL Endpoint is a separate artifact that is not in the lakehouse.

A Warehouse is a database whose tables are managed by TSQL.

Warehouse Facts:

- Lake-centric with open data format.
- Rich capabilities of the SQL engine.
- Transactionally consistent.
- Includes autonomous workload management.
- Data can be ingested into the warehouse through pipelines, dataflows, cross database querying, or the COPY INTO command.

Both lakehouse and warehouse storage tables in Delta Parquet format.

Which engine you choose only depends on whether you need to use Spark or not.

Read more about data warehousing in Fabric

Regarding workload management, the Fabric documentation references a spark concept in that queries are presented to the distributed query processing scheduler (DQP) as a directed acyclic graph (DAG) of tasks. In this comparison, does the DQP play the role that cluster manager would play in spark?

DQP is managing nodes used by the system. You can think of DQP as being similar to cluster manager if you would like. It manages the topology of the backend compute to fit the operator graph that is currently being acted upon for the server.

Commented [PS11]: From workload load management standpoint, DQP also govern resources by applying different dynamic resource policies based on system usage and predefined memory/disk-based footprint policies to manage overall system workload and before allowing new queries to run.

How does tempdb fit into the Fabric Warehouse architecture?

TempDB is Fabric is used for:

- data movements
- storage for intermediate query results (CTAS, data shuffling between nodes, partially aggregated results, etc.)
- sorting data (GROUP BY, ORDER BY, etc.)
- Intermediate transaction information for in-flight transactions

Unlike Synapse Dedicated Pool and Serverless Pool, user-created temp-tables are no longer stored in TempDB. TempDB cannot be accessed or configured by users.

The Documentation mentions that ingestion jobs are run on dedicated nodes that are optimized for ETL and do not compete with other queries or applications for resources. Can you clarify how nodes are "optimized for ETL"?

The separation of storage and compute allows the SQL system to provision multiple backed nodes as needed. Specifically, an insert operation can run on 1 or a few nodes, while a few completely independent nodes are used as backends (potentially with caches) for query execution. There are multiple optimizations happening, but the separation of compute is the major improvement that allows separation of ETL and ingestion from other queries.

We also internally classify ingestion activities differently than query activities and place them on separate backend compute. With this distinction, we can optimize nodes that run ingestion activities differently than query activities, such as not requiring disk space for caching.

Can one large file be split up between compute nodes for faster data load?

Today, parquet files are not split and each parquet file is processed by a single compute node. Parquet file splitting is coming soon. During ingestions, as of now, each parquet file is generated by a single backend node, but we are looking at further optimizations beyond this.

Is data movement still required to move data between compute nodes in Fabric DW?

Yes, some operations require moving data between compute nodes, or from compute to front ends. New optimizations for this are currently being worked on.

How does Fabric Data Warehouse save data to storage? Is this happening on a compute node level or at the SQL Front End?

In General, ingestion does not happen on the front end, but on compute nodes. No data is stored on the compute nodes themselves, but rather it is all written to OneLake where it can be directly read from during selects.

Commented [PS12]: Are we sure about this? My understanding is that DQP determines if a query can run on FE or not before adding topology. Is this applicable only to selects or inserts/updates/deletes as well?

Do we plan to introduce Workload Groups like Azure Synapse Dedicated Pools?

Workload groups still exist in Fabric Data Warehouse, but they are automatically generated based on the source artifact (pipeline, report, etc), source workspace, and query type. Workload group classification is an internal job of the system and is NOT a user take. From an end-user perspective, workload groups and classifiers are not a concept they need to manage.

What execution behavior should users expect when submitting queries to DW/LH?

Commented [PS13]: Please review

Commented [NS14R13]: Added to the question.

Fabric DW is built in such a way that it would run as many queries as possible in parallel and re-run parts of the query in case of any failure instead of re-running the entire query. In most cases, Fabric provides consistent query execution time, but it can vary due to high system usage, failure retries and to avoid duplicate work.

As a user, you will also see query cancellations if the system is busy and cannot accept new queries when submitted interactively. We proactively check for system availability throughout various stages of execution and appropriately respond to system behavior.

What is the difference between SQL End Point for Lakehouse and a Data Warehouse? If there is no difference from the end-users' perspective, why do we need to show both Warehouse and SQL Endpoint in a workspace?

Commented [PS15]: My understanding is that there is no real meaning to an endpoint in Fabric. It is being loosely used. Maybe we need to clarify this without providing lot of information. This is what happens today. When a lake house or warehouse is provisioned by user, we just create a lh/dw db alias entries for a workspace. When user creates a connection for the first time, then DMS service will create a database and assign the database alias entry (weird database name that we see today - x6eps4xrq2xudenlfv6naeo3i4-bmmuvve2hnru3anhfqidprgjai) to a FE node. If FE node is not assigned, DMS service will assign a FE node from a pool and add DW/LH aliases (which we are terming as endpoints)

The SQL endpoint allows you to run SQL queries over a data artifact which is not a full warehouse – e.g. Lakehouse and soon a Kusto DB as well as many others. A SQL endpoint does not allow data write operations however. A user can still create views, functions, etc just no INSERT/DELETE, etc. Data is written by other engines.

A warehouse is a space fully managed by the SQL engine – nobody else can write in a Warehouse. It offers full transactional behavior, while data is still stored in open format.

Given this distinction, you need both in a workspace, particularly if you are in the majority of users whom use Spark for data preparation. In this case, there is no need to copy data into a warehouse if it is already in a Lakehouse or other source served by the SQL Endpoint.

Further, a Workspace contains a single SQL Front-end, where each Lakehouse, Warehouse, and other SQL endpoints are registered as databases. Thus, you can cross-join any endpoint with any Warehouse in queries.

Do we have plans to allow customer to collect diagnostic data into, for example, a Log Analytics Workspace?

Fabric Data Warehouse provides a new built-in Query Store which automatically collects and stores up to 30 days of query and execution history for analysis. Fabric query store also auto-generates insights into customer workloads to help users monitor and conduct performance tuning.

More monitoring options are being actively developed and will come later.

Do we have plans to allow customer to obtain query execution plans to allow data engineers flexibility to optimize queries?

We have mechanisms to collect all query execution telemetries for internal teams to support customer issues. We are actively collecting customer feedback and gauging the demand for exposing subset of execution plan information that is actionable to users to meet any manual tuning needs. This plan is not yet finalized.

Does the Fabric SQL Engine leverage adaptive cache for columnstore data?

Fabric includes several layers of cache; some are implemented and some still in development. Adaptive cache is simply keeping recently used column store segments in memory and eventually on SSD. This storage is temporary however and currently only persists for as long as the compute nodes are active before downscaling due to inactivity. Currently, segments are held for a minimum of 10 minutes after retrieved. As nodes start to relinquish due to inactivity, it is possible to have a "partial cache hit" if some data is still cached while other data needs to be retrieved from disk again.

Caching is an area of investment we are actively working on.

Do we still use NVME SSD for TempDB in Fabric Compute nodes?

No, user temp tables will be stored in OneLake just like regular user tables, backed by parquet. However, they will not have a delta lake log published, nor will they be accessible by the customer direclty in OneLake – meaning OneLake explorer will not be able to navigate to them.

What is the T-SQL syntax to create temp tables?

Temp tables have not yet been released, but are expected by GA. We are starting with session scoped temp tables (#tables) and the syntax is consistent with other SQL platforms.

What data movement operations will exist in Fabric SQL and how does it happen in Fabric DW/LH?

There are two main types of data movement in Fabric Warehouse. One is data being moved from the backend nodes to the front end before being sent back to the client. This applies to queries that require backend nodes for execution. Another kind of data movement is data moved between backend nodes during query execution to resolve aggregations like joins, group bys, etc. This applies to queries that require one or more backend nodes for execution. The warehouse engine will choose a shuffle operation for large volumes of data and a broadcast operation for small volumes of data.

Commented [PS16]: Maybe its worth mentioning about DMS service is integrated within SQL server process (sqlsrvr.exe) to move data in compute nodes/containers?

We should also talk about partition/compute/container - cell mapping between operators to explain how the data movement is done in Fabric.

Commented [NS17R16]: I think we should steer away from even mentioning the DMS service at all since we want to focus on the SaaS aspect. This question is basically asking - will we still have shuffles, broadcasts, etc. The answer to which is yes

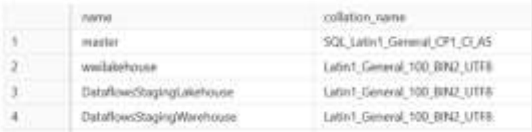
How does the new native shuffle mechanism work?

.

Can you please clarify how to obtain the data from the new in-build query store?

.

Is there a plan to allow end users to use specific collations in the future? Currently we are using different collations for master database and the rest of the SQL Endpoints/warehouses.



	name	collation_name
1	master	SQL_Latin1_General_CP1_CI_AS
2	wwilakehouse	Latin1_General_100_BIN2_UTF8
3	DataflowsStagingLakehouse	Latin1_General_100_BIN2_UTF8
4	DataflowsStagingWarehouse	Latin1_General_100_BIN2_UTF8

We plan on expanding collation support in the future, post-GA.

What are "streaming statistics" and how do they work on a static table?

.

What’s the difference between an Azure Synapse Dedicated Pool and a Fabric Data Warehouse?

In Azure Synapse we have dedicated and serverless options that come with their own advantages and disadvantages. In Fabric we re-wrote the DW engine as one product that includes the advantages of both:

- Elasticity of serverless
- Indexing, caching, and performance from Dedicated Pool.

Along with these benefits, the new engine enables:

- Separation of compute and storage
- The new engine is optimized for Delta/Parquet
- no need to create heap tables because it’s fast

The new architecture allows for countless new features that were not available in Dedicated Pool:

- Separation of compute – no more noisy neighbors

How will the data in a Fabric Data Warehouse be partitioned?

At GA we’ll recommend using Z-ordering for partitioning because it is well understood by other engines. Shortly after GA we plan to add Hash Distributions and other partitioning mechanisms. Hash distributions are easily leveraged by the SQL Query Optimizer, but we plan to upgrade our other query optimizer’s engines (ie. Spark) to use that meta-information because it results in a clean separation of parquet files.

What is the difference between Azure SQL and a Fabric Data Warehouse?

Azure SQL and Fabric Data Warehouse are two very different engines for very different purposes. Azure SQL is an OLTP system optimized for transactions and records which Fabric DW is an MPP analytics engine optimized for querying and summarizing vast amounts of data. While both engines may be able to accomplish the same tasks, your experience and performance will be much better by choosing the right engine for the right job.

What authentication type is supported in Fabric Data Warehouse?

Fabric supports Azure Active Directory authentication for all artifacts and data access within Fabric warehouse.

*For automation purposes, we recommend you use service principal authentication. Add the display name of application registration to a fabric workspace as an admin/contributor/member/viewer.

How are workspace roles mapped to data warehouse permissions in Public Preview?

Workspace Role	Trident Permissions	Login	Data plane permission	Equivalent SQL Role
Admin	Read, Write, ReadData	y	All permissions	db_owner--
Member	Read, Write, ReadData	y	All permissions	db_owner--
Contributor	Read, Write, ReadData	y	All permissions	db_owner--
Viewer	Read, ReadData	y	Read access only	db_datareader
N/A	Read + ReadData permission (artifact)	y	Read access only	db_datareader
NA	Read	Y	Connect only	public

How can I control 'bad actor' queries?

Fabric compute is designed to automatically classify queries to allocate resources and ensure high priority queries (ETL, data prep, reporting) are not impacted by potentially poorly written ad hoc queries. Add some more here

How is classification of incoming queries determined?

Queries are intelligently classified by a combination of the source, like Pipeline or Power BI, and the query type, like INSERT or SELECT. This is handled automatically by the system and does not require a user to create classifications as in a Synapse dedicated pool.

Can more than one capacity be connected to a Datawarehouse, for instance, one to handle data writes and one to handle data reads?

Currently, a capacity is assigned to the workspace level and a Data Warehouse is associated to a single workspace. This means all artifacts in the workspace will share the same capacity and all read/write operations will use the same capacity.

If you want to enable auto scaling, you can do it at a capacity level or set feature level scale to enable scaling on a data warehouse. Remember, the scaling applies to all Data warehouses hosted in a workspace as the underlying capacity is the same.

Does Fabric Data Warehouse support fine grained access control like row-level security, column-level security, dynamic data masking?

Fabric Data Warehouse will support fine grained access control using T-SQL constructs by GA.

Commented [PS18]: @Kevin Conan (HE/HIM) , @Charles Webb , Could you please confirm?

Commented [K(19R18]: @Dharini Sundaram how would you like this answer worded?

These security constructs are not available today and will integrate with Fabrics universal security model.

How are connections to the data warehouse encrypted?

Fabric data warehouse uses TLS 1.2 encryption for all connections.

Real-Time Analytics

What is Fabric Real-time Analytics?

Real-time Analytics is a portfolio of capabilities that provides end-to-end analytics streaming solution across Fabric experiences. It supplies high velocity, low latency data analysis, and is optimized for time-series data, including automatic partitioning and indexing of any data format and structure, such as structured data, semi-structured (JSON), and free text.

Real-time Analytics delivers high performance when it comes to your increasing volume of data. It accommodates datasets as small as a few gigabytes or as large as several petabytes and allows you to explore data from different sources and a variety of data formats.

See Fabric Documentation on Real-Time Analytics for more information.

What are the relevant use cases for Real-Time Analytics?

Real-time Analytics delivers value when the organizations need to analyze time series, telemetry, or log type of datasets at any scale. Such data is often present across many verticals such as:

- Manufacturing: IoT sensor data from buildings, factories, plants, machines

- Telco: Radio Access Network (RAN), Electronic Data Records (EDRs), Call Data Records (CDRs) data
- Energy: Telemetry data from renewable energy sources such as Wind Turbines, Solar Panels, as well as non-renewable sources such as offshore oil rigs, refineries, etc.
- Automotive: Connected vehicles telemetry data
- Transportation: Connected trains, bus, locomotives data

Across all verticals, logs are generated by a variety of embedded, on-prem or cloud hosted applications. Such logs are also highly suitable dataset for Real-time Analytics.

Organizations use such data to predict failures, detect anomalies, generate forecasts, optimize the efficiency of their assets, improve resilience of their software and hardware (cloud or hosted).

Any use case that requires low latency, sub-second response times on gigabytes to petabytes scale data is suitable for Real-Time Analytics

Do I have to know KQL to use Real-Time Analytics?

To query data stored in KQL Database in Real-time Analytics end-to-end scenario, it is advisable to use Kusto Query Language (KQL). Kusto Query Language is a powerful language to explore your data and discover patterns, identify anomalies and outliers, create statistical modeling, and more. The query uses schema entities that are organized in a hierarchy similar to SQL: databases, tables, and columns. You can read more about KQL.

You can also use T-SQL to query data from KQL Database. You can read more about this here. You can also use SQL to KQL Cheatsheet to construct KQL using the familiar SQL syntax.

Using Power BI, you can also construct visuals from the data stored in KQL Database without the need to write KQL. Power BI connector for KQL Database translates DAX and M query to KQL for you.

What is the performance expectation from Real-Time Analytics?

Real-time Analytics is designed and optimized for low-latency (from milli-seconds to few seconds) streaming type ingestion and sub-second query response times. Real-time analytics can query or process up to billion records in a second. You can load gigabytes to terabytes of data, in just seconds. You can query up to petabytes of data, with results returned within milliseconds to seconds. It provides high velocity (millions of events per second), low latency (seconds), and linear scale ingestion of raw data.

How will Real-Time Analytics integrate with OneLake?

Real-time Analytics integrates with OneLake in the following ways:

1. Ingest data from OneLake: Using ‘Get data’ menu, you can ingest files from OneLake into KQL Database.
2. Create shortcut to the data in OneLake: From OneLake experience, you can create a shortcut to KQL Database.

3. Mirror data from KQL Database to OneLake: With this feature, all the ingested data in KQL Database will be made automatically available in OneLake on a continuous basis.

Are there differences between Azure Data Explorer (ADX) and Fabric Real-Time Analytics?

In the documentation on Fabric Real-Time Analytics, we have provided a feature comparison table between Azure Data Explorer and Real-Time Analytics. Please refer to this document for the most accurate comparison.

Power BI

What is DirectLake mode for Power BI?

DirectLake is a groundbreaking new dataset capability, currently in preview, for analyzing very large volumes of data in Power BI. The Analysis Services engine was re-written to support Delta Parquet format directly from Onelake with performance similar to Import mode. You can even create shortcuts to external data to OneLake to use in Directlake mode, however parquet files produced by Fabric workloads (Lakehouse/DW/etc) will typically be faster and more compressed thanks to vertiparquet compression. You will still have the option to use Import or DirectQuery Mode when desired.

Read more about DirectLake mode in Fabric Documentation.

How does DirectLake (parquet) performance compare to IDF?

There will be a greater perf hit for cold cache for Parquet vs IDF files, but obviously we'll keep working on optimizing it now that we're at Public Preview. By the way, even Import datasets in Large Model mode will page in columns on-demand in PBI-P. So, there is a cold cache cost for columns being touched for the first time in both cases – that cost can be a bit higher for Parquet-based columns. Note that the cost can be different depending on column type/cardinality/Parquet rowgroups/etc. With Import datasets, you were in direct control of your partitioning schema – with Parquet, you need to control the physical data format on the ETL/ELT side.

Warm cache perf will typically be similar – sometimes one will be faster, other times the other. Again, there are optimizations planned (some in flight) to tackle some of the regressions.

In general, once the necessary columns have been paged in, you should see really good storage engine performance. It should be very similar to import mode as the actual query engine is the same. It is just the I/O system that is changing (reading parquet files instead of proprietary Vertipaq files). So, while we are not done and perf will improve even further, I think you will be very pleased with the experience, and you will find very similar efficiencies in DirectLake mode as in Import mode.

What compute is required to use Power BI DirectLake mode?

Compute resources are required to prep the data and create the delta/parquet files in the OneLake, which can be done using Spark or Dataflow/pipelines.

The compute required for queries is essentially the same as is it today with Power BI. The significant difference is compute resources are no longer needed for data refresh since this is no longer a necessary step.

How does partition elimination work in Power BI DirectLake mode?

DirectLake benefits from most of the same performance optimizations that Import tables receive from the Vertipaq engine, and automatic partition elimination isn't something that is needed because the Vertipaq engine has other optimizations that make it less necessary.

Why Datamarts in Fabric?

Datamarts are for business users just using Power BI for small scale SaaS service analytics, low to no code, visual querying, etc. It also has an important role in a "Hub & Spoke" BI architecture. Will become GA soon, and would switch the engine to Synapse Engine, so migrations and scaling up into full blown DW will be easy.