SPYGLASSMTG

Women Owned/Women Led

Microsoft Fabric 1:1 Workshop

Spyglass MTG Overview

Expertise, Experience & Excellence

Spyglass is a Specialized Expert Consulting firm focused on helping clients select, architect, implement, migrate to and manage their Microsoft Technology.

- A Women Owned/Women Led Company
- Headquartered in Lincoln, RI
- 25+ years providing Microsoft Solutions to US clients
- 25+ years average experience on our team
- 18+ years as a Gold Partner
- 5 Solutions Badges
- 5 Advanced Specializations
- FastTrack Ready Partner
- 90% of our business is repeat business
- 80% of our new business comes from Microsoft

Competencies and Specializations

Microsoft
Advanced Specializations

Teamwork Deployment
Adoption & Change Management
Analytics on Azure
Azure Data Warehouse Migration
Infra & Database Migration to Microsoft Azure

Microsoft
PowerBI Partner

Microsoft
Solutions Partner

Infrastructure Azure
Data & AI Azure
Digital & App Innovation Azure
Modern Work
Security

Microsoft
FAST TRACK
READY PARTNER

PARTNER SUCCESS SERIES
OEA Advanced
Partner
Microsoft

Microsoft
Cloud Solution Provider

SPYGLASSMTG

Demo: Deploy a Landing Zone in Fabric

Working With Fabric

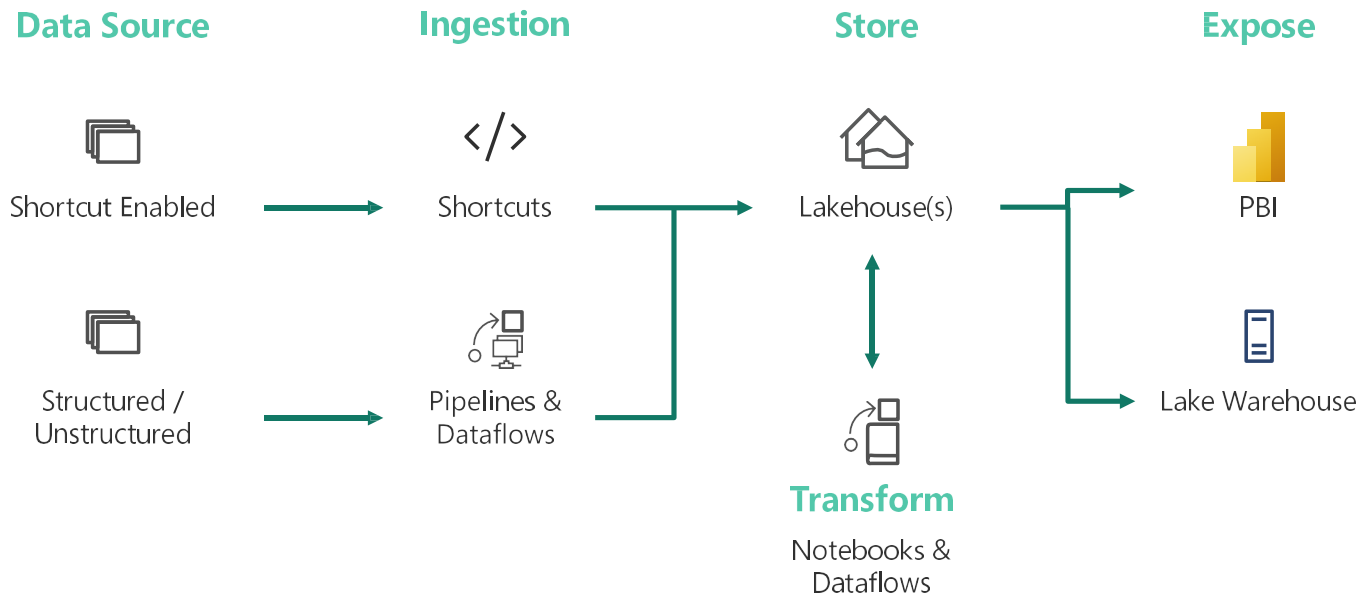
Lakehouse

The Microsoft Fabric Lakehouse analytics scenario makes it so that data can be ingested into OneLake with shortcuts to other clouds repositories, pipelines, and dataflows in order to allow end-users to leverage other data.

Once that data has been pulled into Microsoft Fabric, users can leverage notebooks to transform that data in OneLake and then store them in lakehouses with medallion structure.

From there, users can begin to analyze and visualize that data with Power BI using the see-through mode or SQL endpoints.

The Data Lakehouse scenario



Build and implement an end-to-end lakehouse for your organization:

1. Create a Microsoft Fabric workspace
2. Quickly create a lakehouse – an optional module to implement medallion architecture (Bronze, Silver, and Gold)
3. Ingest, transform and load data into the lakehouse – bronze, silver and gold zones as delta lake tables for medallion architecture
4. Explore OneLake, OneCopy of your data across lake mode and warehouse mode
5. Connect to your lakehouse using TDS/SQL endpoint
6. Create Power BI reports using DirectLake – to analyze sales data across different dimensions
7. Orchestrate and schedule data ingestion and transformation flow with Pipeline
8. Cleanup resources by deleting the workspace and other items

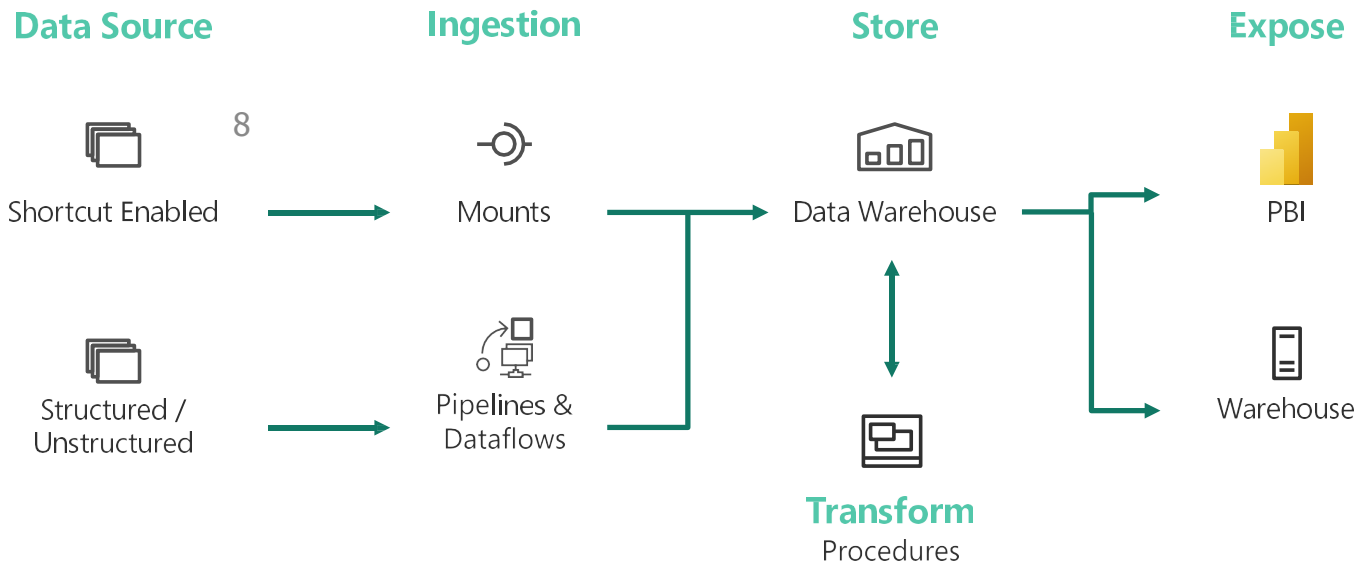
Data Warehouse

The Data Warehouse analytics scenario takes existing sources that are mounted, while pipelines and dataflows can bring in all other data that is needed.

IT teams can then define and store procedures to transform the data, which is stored as Parquet/Delta Lake files in OneLake.

From there, business users can analyze and visualize data with Power BI, again using the see-through mode or SQL endpoints.

The Data Warehouse scenario



Build and implement an end-to-end data warehouse for your organization:

1. Enable Microsoft Fabric in your tenant
2. Create a Fabric workspace
3. Quickly create a data warehouse
4. Ingest data from source to the data warehouse dimensional model
5. Transform the data to create aggregated datasets using T-SQL
6. Perform orchestration, data ingestion, and data transformation with pipelines
7. Query the data warehouse using T-SQL and a visual query editor
8. Create Power BI report using DirectLake mode to analyze the data in place
9. Cleanup resources by deleting the workspace and other items

Choosing Lakehouse or Data Warehouse

Fabric Lakehouse

- Key props
  - Easy to get started with and explore the lake
  - Better TCO for ad-hoc and intermittent workloads
  - Optimized for Spark workloads(write)
  - SQL workloads are read/write
  - Unstructured, semi-structured, structured data workloads
- Data organized by Folders and files,
- databases and tables
- Low to moderate database security features
  - Row level
  - Table level (when using T-SQL)
  - None for Spark
- Limit: Spark only write operations
  - Scala, PySpark, Spark SQL, R

Fabric Data Warehouse

- Key props
  - Full management capabilities of a data warehouse
  - Predictable and high perf for continuous workloads
  - Optimized for T-SQL workloads (read/write)
  - Spark workloads are read only
  - Structured data workloads
- Data organized by Databases, schemas, and tables
- Moderate/high database security features
  - Object level (table, view, function, stored procedure, etc.)
  - Column level
  - Row level
  - DDL/DML
- Limit: T-SQL only write operations
  - T-SQL

SPYGLASSMTG

Choosing Fabric or Synapse Dedicated Pools

Fabric Lakehouse & Data Warehouse

- Current Value Props
  - Easy to get started with no infrastructure management
  - Better TCO for ad-hoc and intermittent workloads
  - System managed Compute (auto scale)
  - Synapse Data Warehouse in Microsoft Fabric is an evolution of the dedicated pool as a SaaS solution
- Future Value Props
  - Auto Stats
  - DW Deployment Pipelines
  - GIT Integration
  - Monitoring
  - Relationship Editor
  - Cross-joining warehouse and lake house
- Ad-hoc / intermittent workloads
  - No need for loading: easier to start & run
  - Charging per usage will be cost-beneficial
- Moderate/high latency
  - this release is not focused on performance, concurrency, and scale

Synapse Dedicated Pool

- Current Value Props
  - Full network integration and isolation
  - Full management capabilities of a data warehouse
  - Predictable and high perf for continuous workloads
  - Optimized for managed (loaded data)
  - User managed Compute (no auto scale)
  - Azure Synapse Dedicated SQL Pools will continue to provide a robust, enterprise-grade PaaS solution
- Continuous workloads
  - Loading gives perf, with additional complexity
  - Charging per DWU (sized well) will be cost-beneficial
- Low latency
  - Richer Caching and Indexing + materialized views

SPYGLASSMTG

Choosing Notebooks or Data Flows

Spark Notebooks

- Data Engineering/Data Preparation
  - Data cleaning
  - Transforming semi-structured data
  - Preparing data for machine learning
- Code Centric
  - Spark Python/Scala/SQL-based
- Machine Learning on big data
- Customers with existing Spark/Hadoop data engineering
- Spark supports hundreds of destinations
- Persona
  - Data engineer
  - Data scientist
  - Data developer

Data Flows

- Data Engineering/Data Preparation
  - Data cleaning
  - Transforming semi-structured data
  - Preparing data for analytics
- Low-Code/No-Code
  - Power Query editor, Visual Drag and Drop editor
- Customers with existing Power Query data engineering
  - ADF, Power BI, Power Apps
- Supports the core Azure destinations
  - Lakehouse
  - Azure SQL database
  - Azure Data explorer
  - Azure Synapse analytics
- Persona
  - Data engineer
  - Data integrator
  - Business analyst

SPYGLASSMTG

Choosing Pipelines or Data Flows

Pipelines

- Orchestration/Data Integration
  - Orchestration of other Data Transformation Activities
    - Notebooks, Scripts, Copy, Dataflow
  - Sourcing Data with Copy Activities
- Low-Code/No-Code
  - Wizard/Canvas
- Customers with existing ADF or Synapse Pipelines
- Supports 30+ cloud and on-prem destinations
  - Connector overview - Microsoft Fabric | Microsoft Learn
- Persona
  - Data engineer
  - Data integrator

Data Flows

- Data Engineering/Data Preparation
  - Data cleaning
  - Transforming semi-structured data
  - Preparing data for analytics
- Low-Code/No-Code
  - Power Query editor, Visual Drag and Drop editor
- Customers with existing Power Query data engineering
  - ADF, Power BI, Power Apps
- Supports the core Azure destinations
  - Lakehouse
  - Azure SQL database
  - Azure Data explorer
  - Azure Synapse analytics
- Persona
  - Data engineer
  - Data integrator
  - Business analyst

SPYGLASSMTG

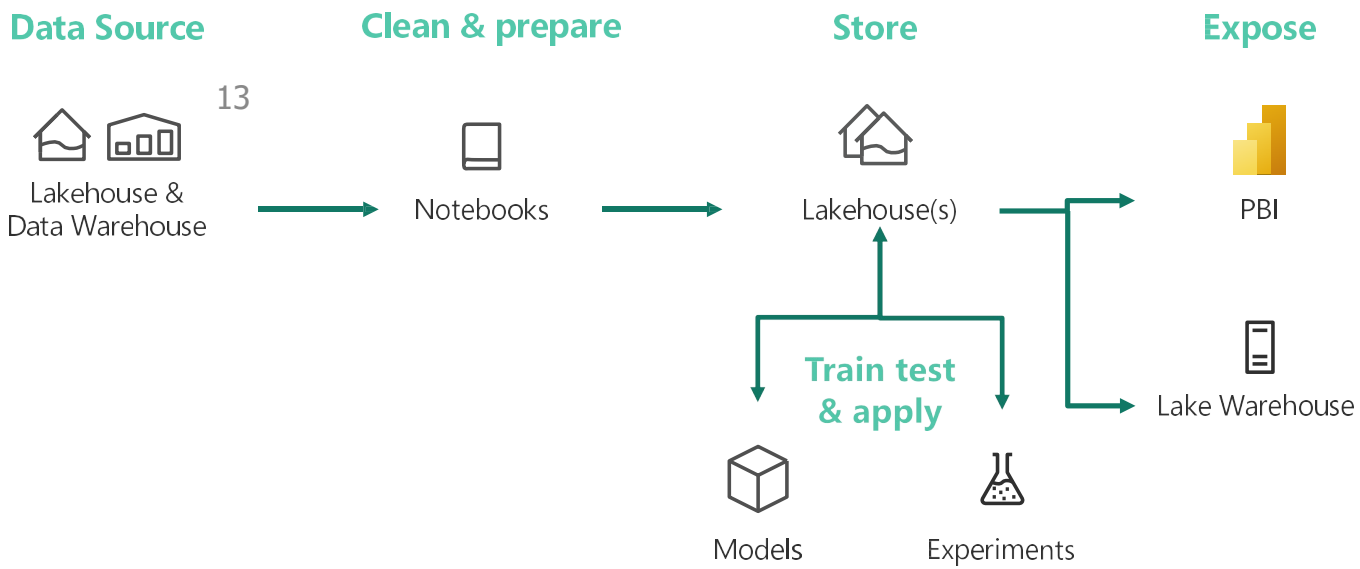
Data Science

The Data Science analytics scenario can be ingested similarly to the Lakehouse and Data Warehouse paths.

Once the data is ingested, it is cleaned and prepared using notebooks and then stored in the lakehouse with medallion structure.

After the data is cleaned and stored, machine learning models can be trained and tested directly on the lakehouse.

The Data Science scenario



In this scenario, consider performing the following activities:

1. Use the Microsoft Fabric notebooks for data science scenarios
2. Ingest data into Microsoft Fabric lakehouse using Apache Spark
3. Load existing data from the lakehouse delta tables
4. Clean and transform data using Apache Spark
5. Create experiments and runs to train a machine learning model
6. Register and track trained models using MLflow and the Microsoft Fabric UI
7. Run scoring at scale and save predictions and inference results to the lakehouse
8. Visualize predictions in PowerBI using DirectLake

Real Time Analytics

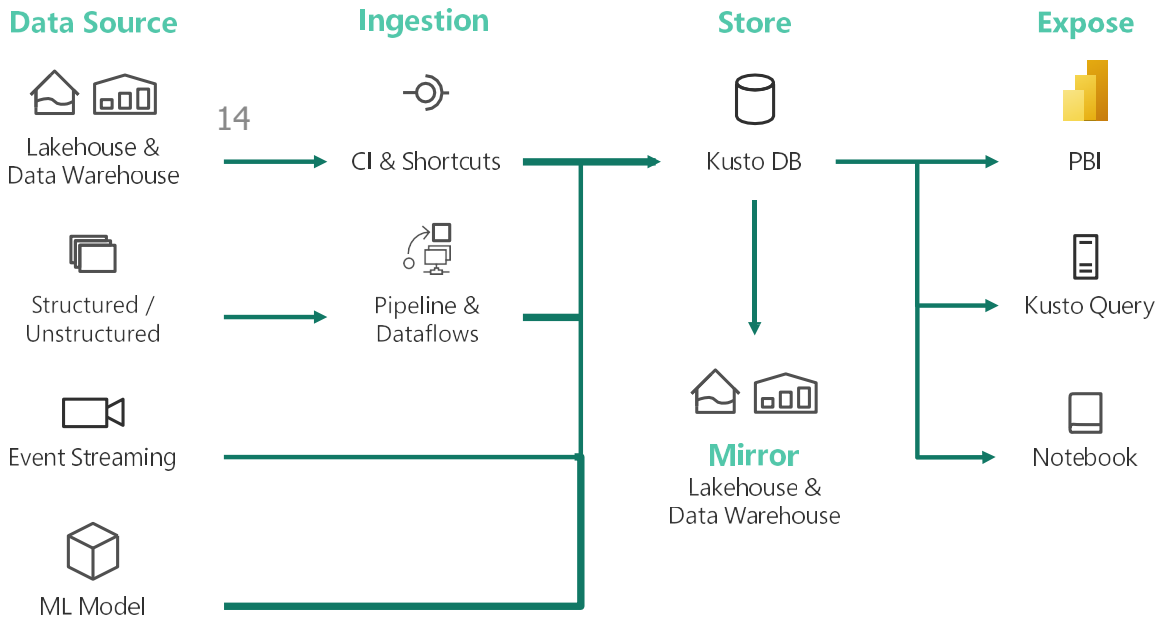
Unlike the Data Science, Lakehouse, and Data Warehouse analytics scenarios, streaming data can be ingested into the Microsoft Fabric in several ways to achieve real time analytics.

Users can leverage Event Hub, IoT Hub, pipelines, dataflows, notebooks, or open-source products like Kafka, Logstash, and more.

Once ingested into Microsoft Fabric, streaming data can be stored in Kusto DB and mirrored into the lakehouse. After the data has been stored, machine learning models can be trained and tested directly on the lakehouse with experiments.

Like the other scenarios, business users can analyze and visualize the data with Power BI using the see-through mode or SQL endpoints. Data can also be exposed through KQL or notebooks using Spark.

The end-to-end Real-Time Analytics scenario



In this scenario, consider performing the following activities:

1. Create a KQL Database
2. Create Eventstream
3. Stream data from Eventstream to KQL Database
4. Check your data with sample queries
5. Save queries as a KQL Queryset
6. Create a Power BI report
7. Create a OneLake shortcut

One Lake in Microsoft Fabric

OneLake is Fabric's lake-centric architecture that provides a single, integrated environment for data professionals and the business to collaborate on data projects.

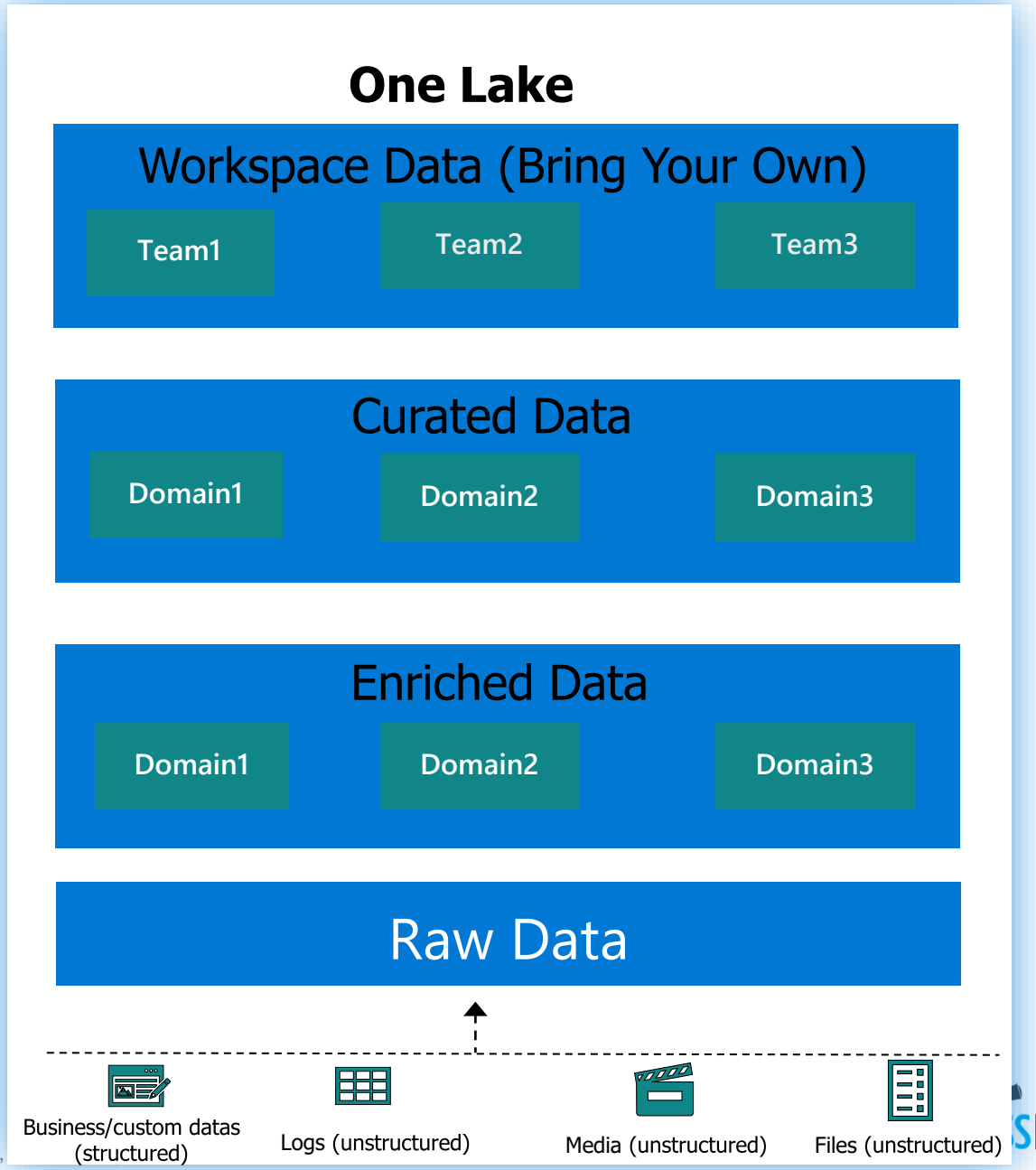
One Lake is built on top of Azure Data Lake Storage (ADLS).

Support in Microsoft Fabric covers regional resiliency with availability zones.

One Lake Organization
Medallion Architecture

A common pattern is to segment your data into different zones due to the lifecycle of the data:

- **Raw Data:** data directly from the source systems in its original format
- **Enriched Data:** data that has been cleaned of all erroneous data, modified, and enriched
- **Curated Data:** data has been aggregated and is ready to be served to users via reporting services or other down-stream systems
- **Workspace Data:** This is a zone that contains data ingested by each individual team/data consumers to provide greater value to their specific teams



Fabric Storage Optimization

- **Choose Appropriate Storage Format**
  - OneLake and supports data in any file format, including Delta Parquet, CSV, JSON, and more.
  - One Lake default format is Parquet/Delta Lake
  - Only the lakehouse tables in Delta format are available in the SQL Endpoint
- **Co-locate Storage with Capacities Locations**
  - Reduces network latency
  - Provides data residency
- **Establish Storage tier strategy**
  - Ex: Bronze/Silver/Gold layers
  - Fabric provides a medallion module

Delta Lake

Fabric Lakehouse pro-actively enables the important parameters to minimize common problems associated with big data tables compaction and small file sizes, and to maximize query performance

- **Parquet file rules apply**
- Perform File Compaction regularly
  - Currently with repartition
  - Bin compaction with OPTIMIZE (preview)
- Create Indexes on Delta Lake tables
  - Via Hyperspace for Delta Lake
  - Z-order indexing support coming soon
- Do not specify wildcards
  - Query will automatically identify Delta Lake partitions
- Use Spark pools for
  - Updating Delta Lake files
    - Serverless does not support updates
  - Time Travel queries
    - Serverless can only query the latest version

SPYGLASSMTG