Especially for you (and you can refer to folks within your organization that could benefit from this information), we're pleased to share the competitive advantages of Microsoft Synapse/Azure SQL DW (per category):

Unlimited Concurrency

Unlimited concurrency requires an unlimited budget.

Microsoft's competition's approach is to continuously add more compute which is expensive. Most customers do not have unlimited budgets and on the contrary are looking for cost predictability in their analytics capability, and cloud services all up. The industry price/performance leadership of **Azure SQL Data Warehouse** combined with Workload Importance and Resultset Cache deliver the optimal blend of concurrency with a bias to high value workloads and the lowest cost.

DBA Requirement

The claims of simplicity made by various vendors are weak attempts to gloss over the missing features in their immature platform. Although the cloud platform has enabled new architectural capabilities in the platform, and new economics in how compute is deployed, the fundamentals of data structures and query optimization remain as they have been proven in industry for decades.

Security in particular is one aspect that require manual configuration. Although it is a one-off, customers should not underestimate the security configuration required. Added complexity increases the risk of errors made in implementation and may result in gaps in the implementation. False or weak attempts to gloss over the missing features in their immature platform. Although the cloud platform has enabled new architectural capabilities in the platform, and new economics in how compute is deployed, the fundamentals of data structures and query optimization remain as they have been proven in industry for decades.

Security in particular is one aspect that require manual configuration. Although it is a one-off, customers should not underestimate the security configuration required. Added complexity increases the risk of errors made in implementation and may result in gaps in the implementation.

Automatic Table Clustering

The deprecation of manual table re-clustering take all cost control away from the user. Microsoft's competition recommendations around clustering are effectively mandating it as a must-do function to maintain the performance of the warehouse, but with automatic re-clustering, customer will pay-per-index rebuild since a new warehouse is automatically created, and their algorithms will decide when to do it.

It may or may not happen multiple times within a given period. Customer should have the choice to leverage their own capacity at a performance tradeoff, or when activity is low, or alternatively subscribe to automatic clustering. This results in unpredictable costs and paying extra to optimize their performance. It's a no-win for customer who either pay more to rebuild indexes, or don't rebuild and suffer from decaying performance and need to buy larger warehouses to compensate.

SQL Data Warehouse enables customer to leverage their own capacity, schedule the work for exactly when they want to run it, and maintain a completely predictable price model.

SQL Data Warehouse supports online (and automatic) scaling. SQL Data Warehouse provides Workload Isolation (in preview) in a single cluster. Once a developer defines compute utilization thresholds for workload groups using a simple percentage, the engine will continuously scale to balance workloads without any interruption to queries being executed, and without the performance impact of queries needing to populate cache which can have an orders or magnitude impact relative to cache reuse and in-memory execution.

Auto Suspend/Resume

Auto Suspend/Resume is a useful catch-all for developers who forget to turn off warehouses at the end of the day, but it is not solution that even slightly compensate for the price/performance advantage of SQL Data Warehouse. Customers can achieve the same auto-pause and auto-resume using a programmatic approach in Azure today and for the "CEO query", Workload Importance is a vastly more appropriate feature both in terms of performance and cost.

Self-Adapting Schema

Determining optimal hash keys is not difficult for developers. Furthermore, changing these without informing developers or engineers can have significant adverse effects on performance. While some customers may have an appetite for entirely self-managing warehouses, Microsoft experience in delivering data platforms has informed us that customers appreciate control of aspects of the system that can improve or degrade performance. As such, today we remove the 'investigation burden' from our customers by surfacing recommendations via Azure Advisors regarding data model optimizations (+ others) that customers can review, test, and decide whether to implement or not. With potentially thousands of users running hundreds of thousands of queries per day, and small, unannounced "adaptive optimization" may have unintended consequences, and other DWs often do not provide the knobs to control or rollback these changes.

Multi-Cloud

Running a single workload across multiple clouds adds significant complexity and cost overhead that is not obvious.

In reality this is not so easy. Data synchronization between clouds can result in excessive egress costs, and the commonality of the surrounding ecosystem such as security and data loading tools is questionable. At best, a customer would need to take a dependency on non-native Cloud services from any of the hyper scale providers since, for example, Azure Data Factory only runs on Azure. Once customers bet on a cross-cloud best of vendor approach from providers such as Informatica and Tableau, then the TCO of the solutions grows quickly.

The tradeoffs should be fully understood. Azure provides the most comprehensive integrated stack for the entire modern data warehouse ranging from ingestion, through data lakes and warehouses,
and up to cubes and visualization. Azure's competition and the multi-cloud DWs are only one piece of this solution and bridging across clouds creates fragmentation in the all up analytics platform.

Fast JSON Query Performance

SQL Data Warehouse also supports loading, storing, and querying JSON is its native nested format using the SQL language and this capability if optimized for data exploration scenarios.

Faster Data Sharing

Microsoft's competition claims they are more nimble and can share data 'faster'.

Microsoft recently announced Azure Data Share that enables sharing of any data from a variety of source securely between two or more parties within Azure. Our approach does not make any assumptions or place any restrictions on the type
of data a customer wishes to share, not does it lock the recipient into a particular technology vendor. In contrast to other Data Warehouses and platforms, Azure Data Share is open and flexible.

Using Azure Data Share today, data from Azure SQL Data Warehouse can be shared via snapshot in a format that can be consumed by non-SQL Data Warehouse such as Open Source, Databricks, or ETL/ELT systems for further processing.