#FABCONSQLCON2026
SQLCON
Microsoft SQL
COMMUNITY CONFERENCE
SQL Database in Microsoft Fabric
The Fast Path to Modern, Scalable Analytics
Chris Wagner
Director AI & Data

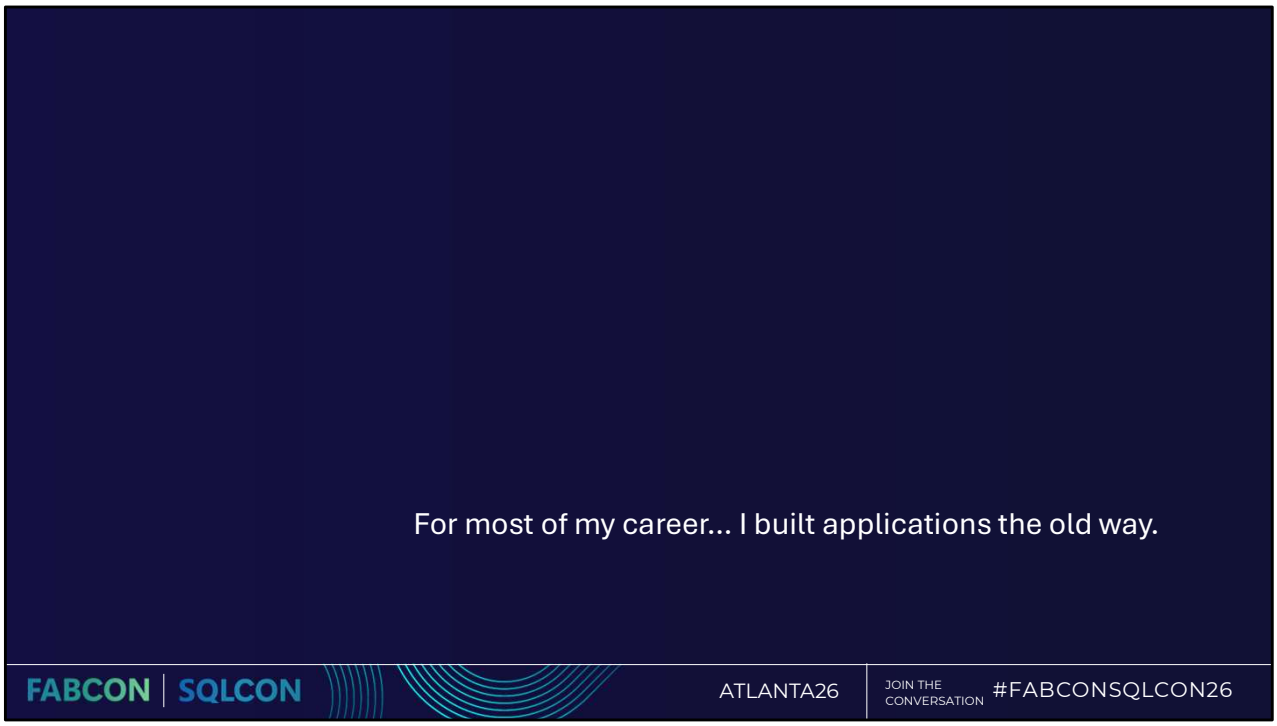
#FABCONSQLCON2026
FABCON
Microsoft Fabric
COMMUNITY CONFERENCE
SQLCON
Microsoft SQL
COMMUNITY CONFERENCE
ATLANTA MARCH 16 - 20, 2026

#FABCONSQLCON2026
FABCON
Microsoft Fabric
COMMUNITY CONFERENCE
SQLCON
Microsoft SQL
COMMUNITY CONFERENCE
ATLANTA MARCH 16 - 20, 2026

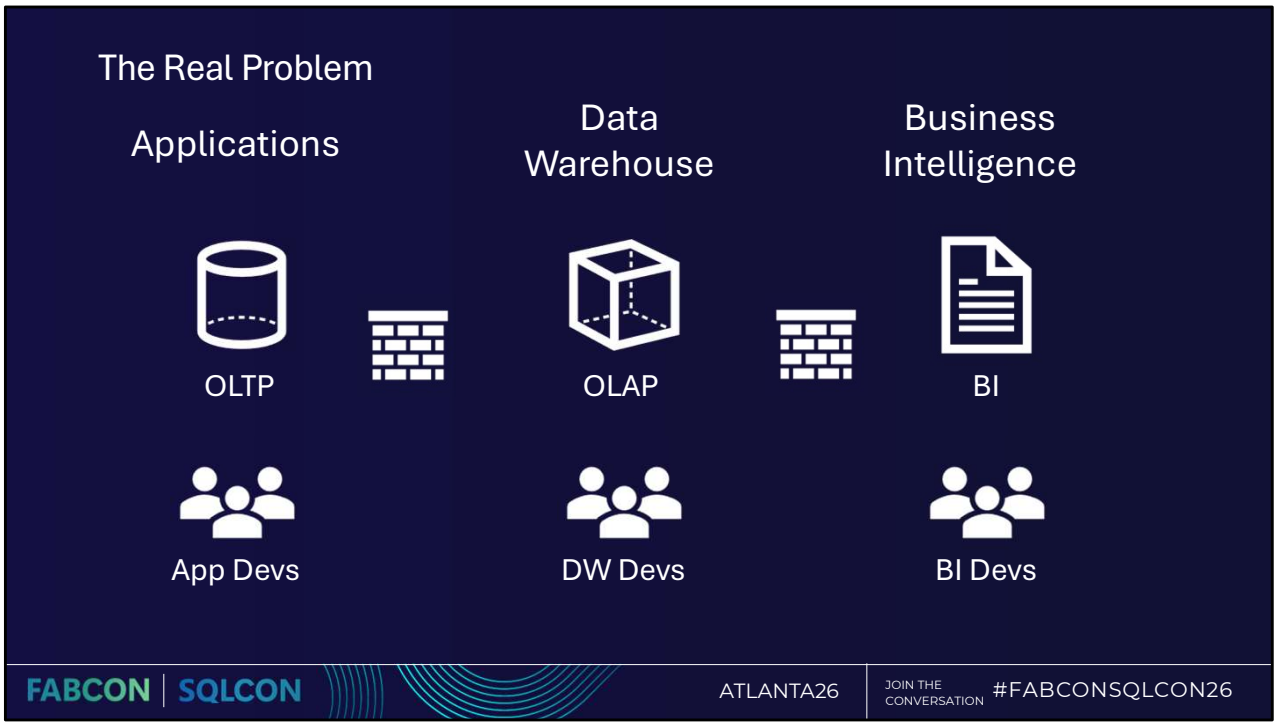
#FABCONSQLCON2026
FABCON
Microsoft Fabric
COMMUNITY CONFERENCE
SQLCON
Microsoft SQL
COMMUNITY CONFERENCE
ATLANTA MARCH 16 - 20, 2026

SQL Databases in Microsoft Fabric introduce a new way to support operational workloads within a unified analytics platform. By combining familiar SQL capabilities with the broader Fabric ecosystem, organizations can modernize applications while keeping data closely integrated with analytics.

This session explores how SQL Databases in Fabric help teams modernize relational workloads, simplify platform architecture, and accelerate access to insights. Throughout the session, the capabilities of the platform are evaluated through three key lenses: **Cost, Speed, and Control**—the factors most organizations balance when modernizing their data platforms.

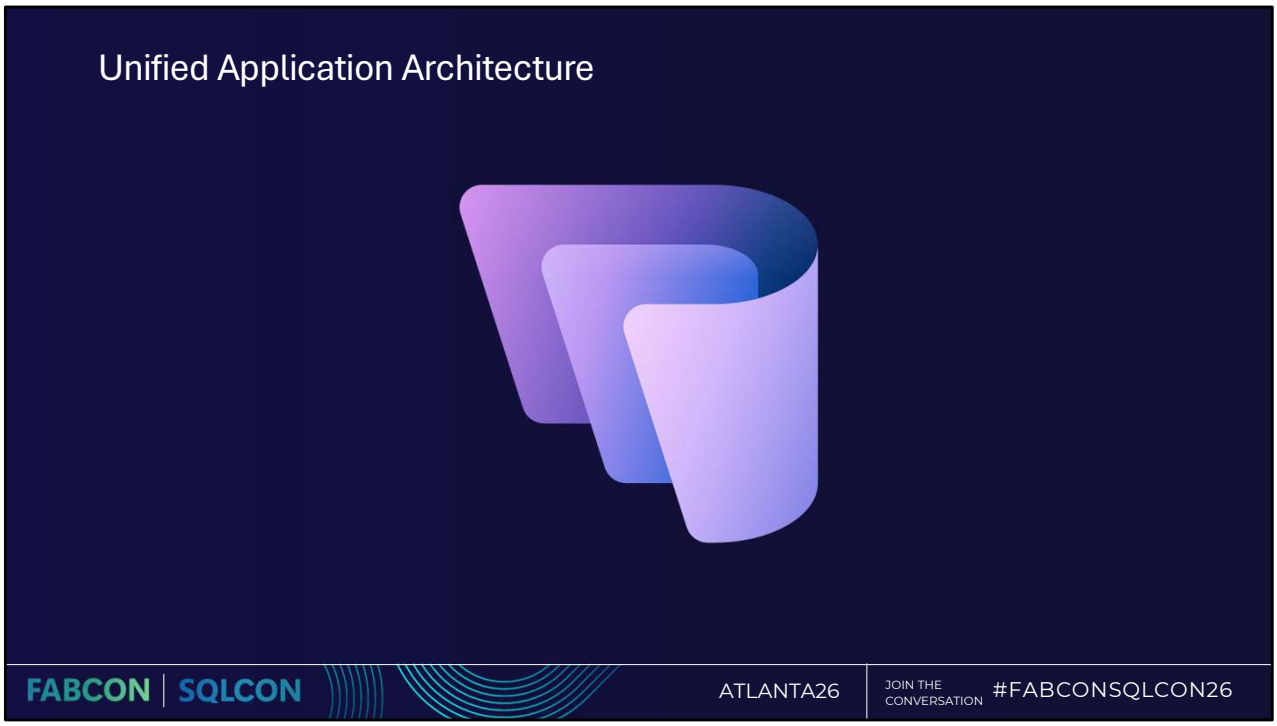


Pause.
Then continue verbally:
Separate app database
Separate reporting database
Separate analytics platform
Endless ETL pipelines
Copies of copies of copies
“And I thought that was just how it had to be.”



The Real Problem
Here's the shift.
"The problem was never SQL.
The problem was separation."
Data duplicated
Latency everywhere
Governance fractured
Developers and analysts disconnected
We accepted architectural tax as normal.
It isn't.

For many years, data architectures have separated operational systems from analytics platforms. Application databases handled transactions, while separate warehouses and reporting systems were used for analysis.
This separation often required complex ETL pipelines, duplicated data, and additional governance layers to keep systems synchronized.
These patterns became common not because they were ideal, but because earlier platforms required organizations to architect around technical limitations.
Modern data platforms are beginning to challenge that assumption by bringing operational workloads and analytics closer together within a single ecosystem.
This shift opens the door to architectures where application data, analytics, and governance are intentionally designed to work together rather than being separated by necessity.



Here's your conviction moment:
"This isn't just another database option.
This is the beginning of converged application architecture."
Application development where:
Data lands once
Operational + analytical workloads coexist intentionally
Governance is native
Scale is built-in
Analytics is not an afterthought
That's the future.
The architecture we've been trying to fake for 20 years.

transition

The Modernization Team

CFO

Cost & Consolidation

BI Director

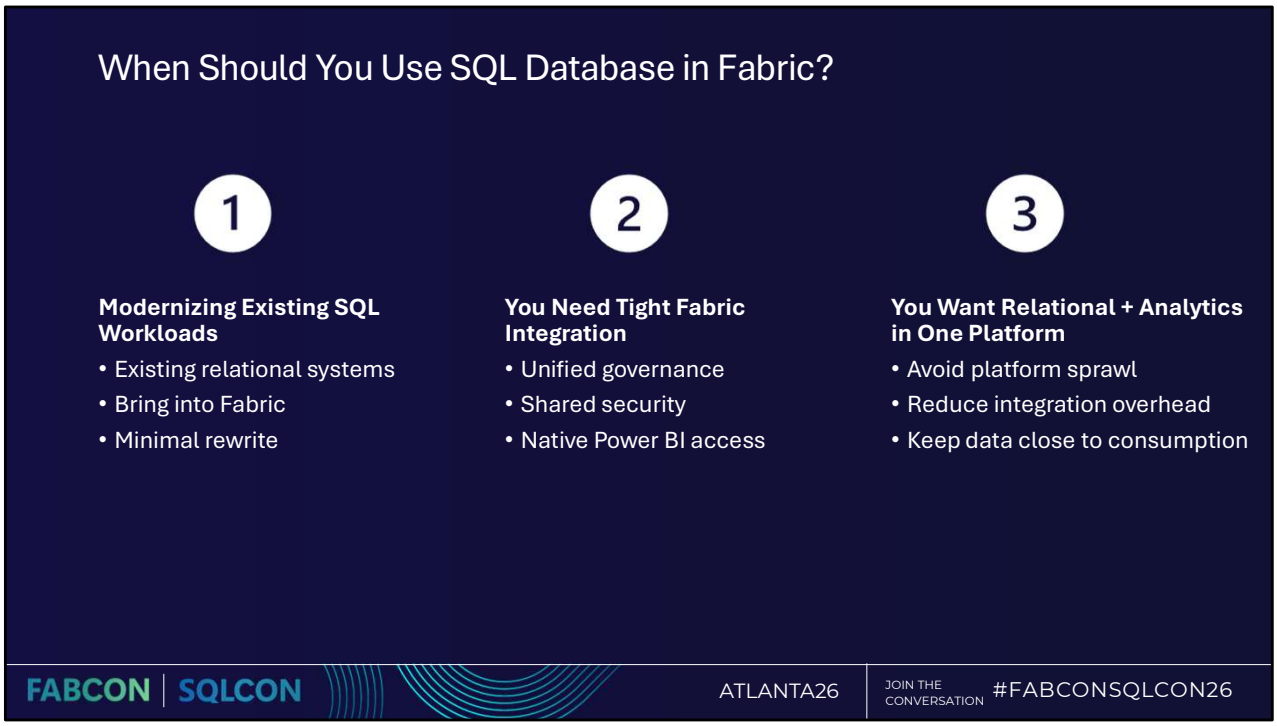
Speed & Delivery

DBA

Control & Governance

Modernizing data platforms requires alignment across multiple roles within an organization. Each stakeholder evaluates technology through a different lens.
The **CFO** focuses on **cost and consolidation**, looking for opportunities to reduce platform sprawl, simplify infrastructure, and control overall technology spending.
The **BI Director** prioritizes **speed and delivery**, ensuring teams can move data quickly, build reports faster, and provide insights to the business without unnecessary delays.
The **DBA** is responsible for **control and governance**, maintaining security, operational stability, and reliable data management as workloads scale.
Throughout the demos, these personas represent the perspectives many organizations balance when adopting new data technologies.

SQL Database in Microsoft Fabric is not intended to replace every database platform. Instead, it introduces a relational database engine that operates natively within the Fabric ecosystem.
It is designed to support structured, operational-style workloads while remaining closely integrated with the broader data platform.
Because it lives inside Fabric, the data stored in SQL databases can connect directly to analytics, reporting, and AI capabilities without requiring complex data movement or duplication.
In this architecture, the SQL database acts as a bridge between operational data and analytical workloads, enabling a more unified data platform.

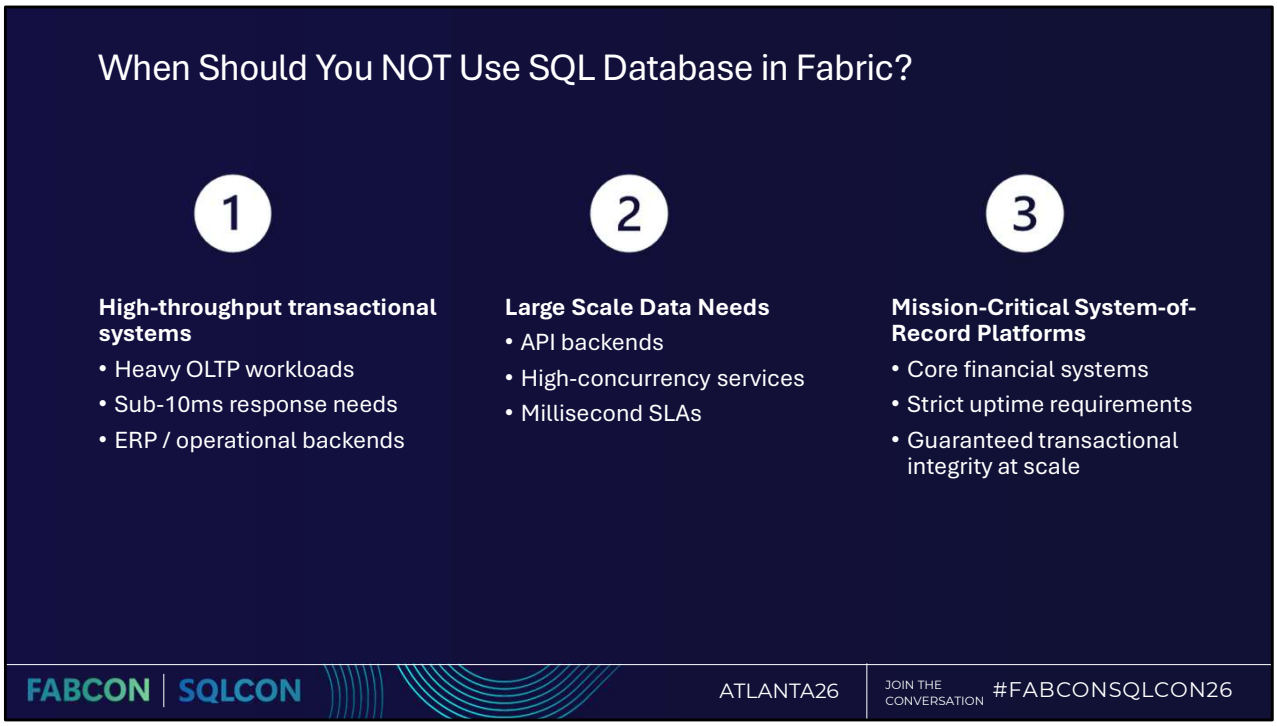


SQL Database in Fabric is a strong fit when organizations want to bring operational workloads closer to analytics while simplifying their data platform.

Modernizing existing SQL workloads is one of the most natural use cases. Existing relational applications can be brought into Fabric with minimal code changes while gaining access to the broader data platform.

Tight Fabric integration allows operational data to immediately participate in governance, security, and analytics workflows. Data stored in SQL Database in Fabric can be easily accessed by Power BI and other Fabric services.

Relational and analytical workloads in one platform reduces the need for multiple systems and complex data movement. By keeping operational data close to analytics and reporting tools, organizations can simplify architecture and reduce integration overhead.



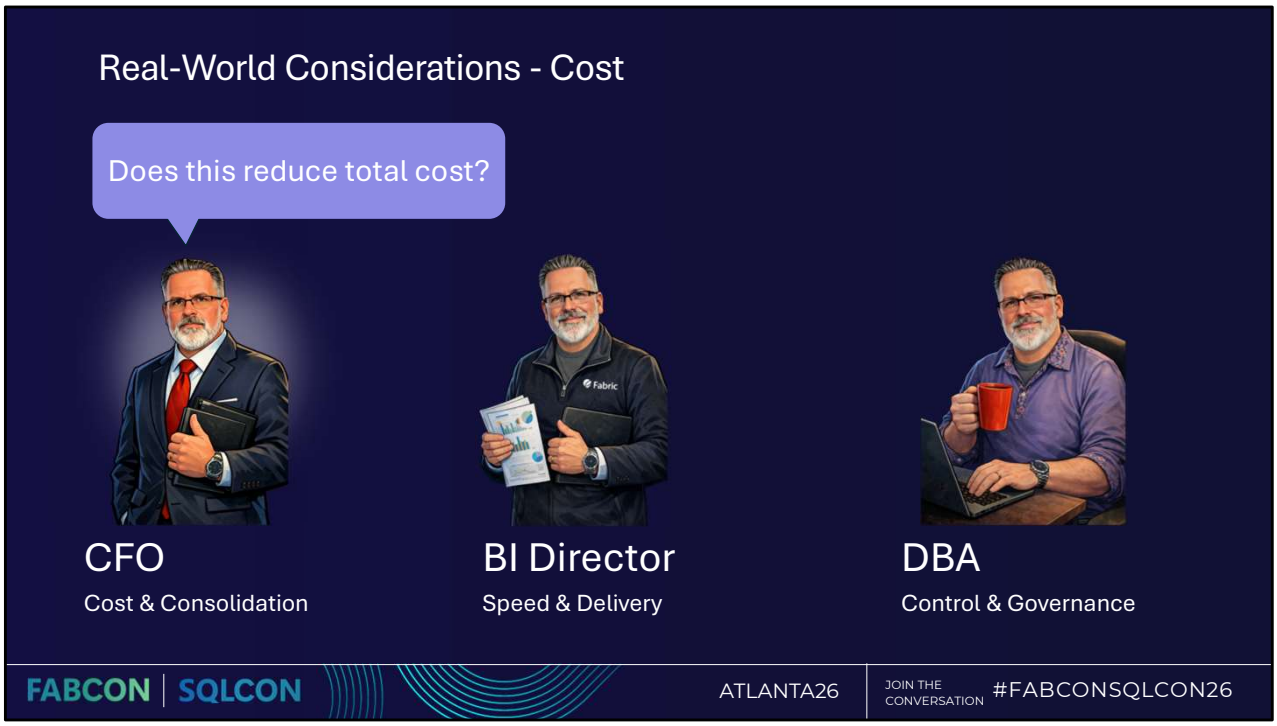
SQL Database in Fabric is designed to support operational workloads within a unified analytics platform. However, it is not intended to replace every type of database system.

High-throughput transactional systems with extremely high write volumes and sub-10 millisecond response requirements are better suited for dedicated OLTP platforms.

Large-scale API backends and high-concurrency services that require millisecond latency and massive parallel request handling may also exceed the intended operational profile.

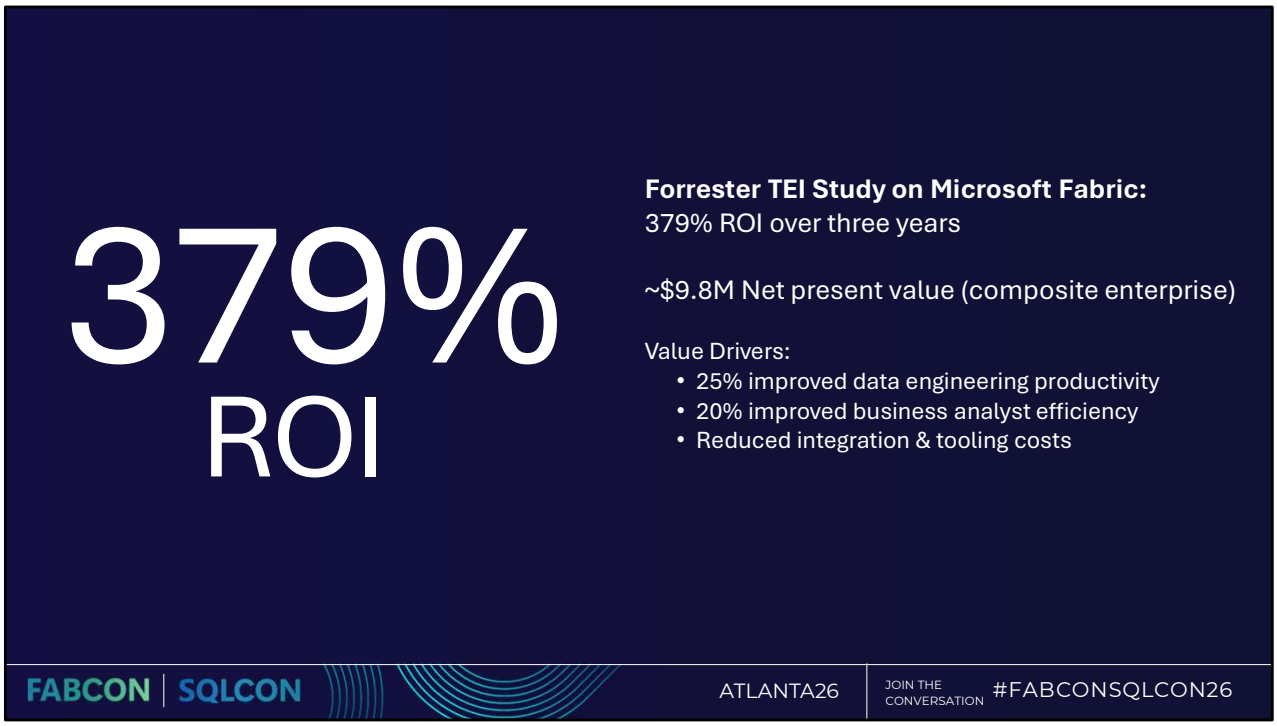
Mission-critical systems of record, such as core financial platforms with strict uptime and transactional guarantees, should continue to run on established enterprise database platforms.

Fabric SQL is best used where **operational workloads benefit from close integration with analytics, governance, and the broader Fabric ecosystem.**



One of the first questions organizations ask when evaluating new capabilities is whether it introduces **additional platform cost or complexity**.
SQL Database in Microsoft Fabric runs **within an existing Fabric capacity**, rather than requiring a separate Azure SQL subscription or independent infrastructure.
By bringing relational workloads into the Fabric ecosystem, organizations can **reduce platform sprawl** and avoid maintaining separate systems for operational databases, analytics platforms, and reporting tools.
This approach aligns relational workloads with the **same governance, capacity model, and platform strategy** already used for analytics in Fabric, helping simplify both cost management and architecture.

What you say
“When you reduce integration points,
you reduce failure points.
When you reduce platforms,
you reduce cost — not just license cost,
but operational cost.”
Pause.
“That’s the real ROI.”



Independent research highlights the potential economic impact of consolidating data workloads on Microsoft Fabric. A Total Economic Impact study conducted by Forrester reported a **379% return on investment over three years** for a composite organization adopting the platform.

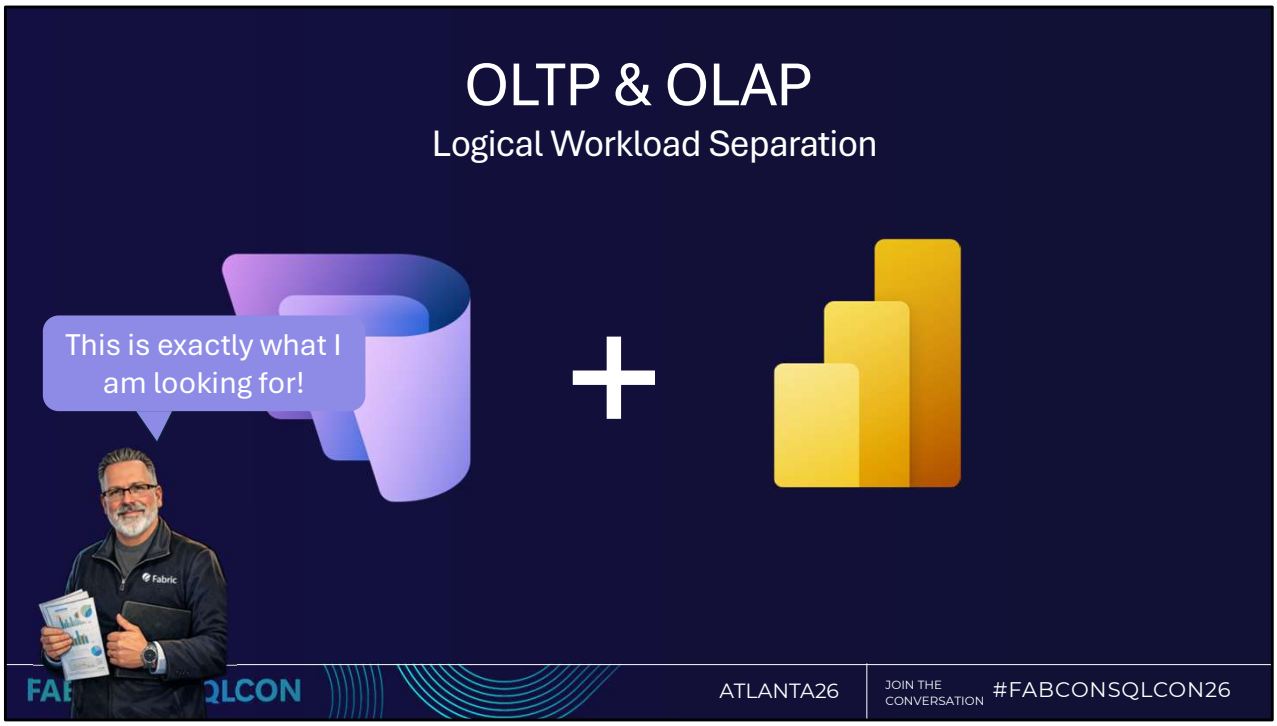
The value is not driven solely by licensing efficiencies. Much of the impact comes from **platform consolidation, reduced integration complexity, and improved engineering productivity**.

By reducing the number of tools required to manage operational data, analytics, and governance, organizations can streamline their data architecture while enabling analysts and engineers to deliver insights more quickly.

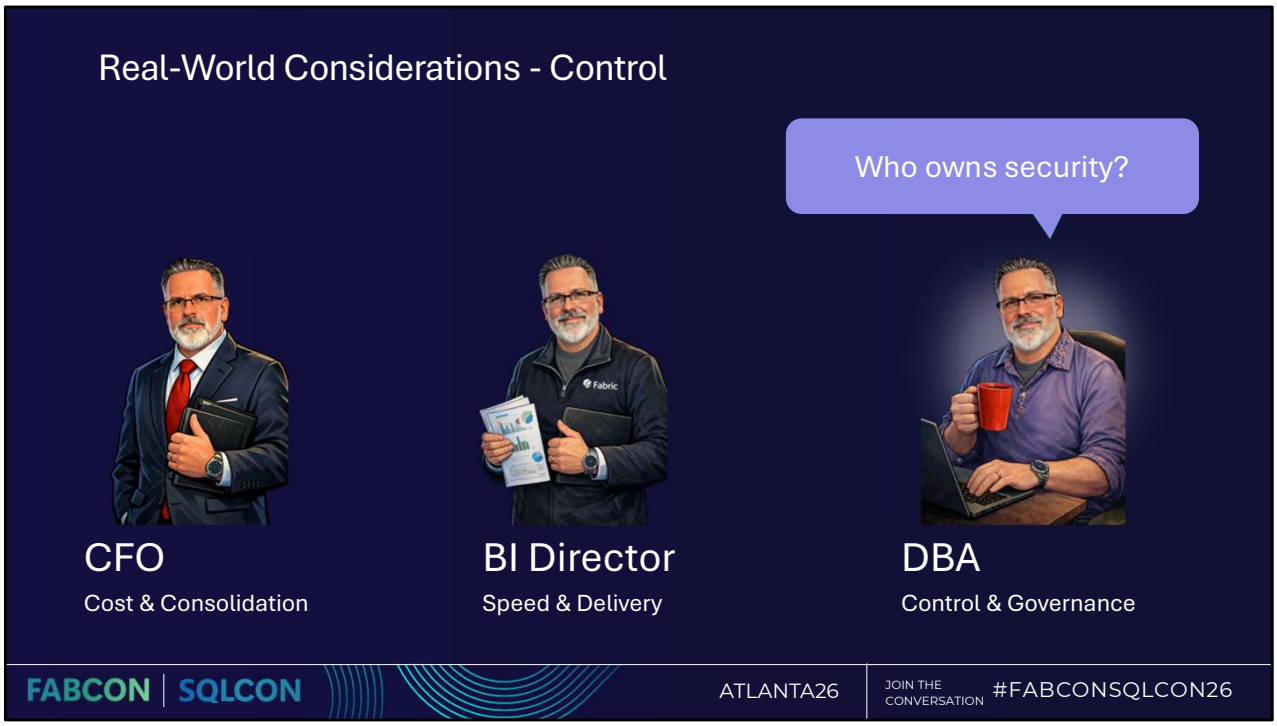
These factors combine to create measurable operational value beyond traditional cost savings.

A common question is whether SQL Database in Fabric should replace existing analytics systems used for reporting and large-scale analysis.
The answer is no. SQL databases are designed for **OLTP (Online Transaction Processing)** workloads, which focus on inserts, updates, and operational data management.
Analytics and reporting workloads are typically handled by **OLAP (Online Analytical Processing)** systems that are optimized for large-scale aggregations and analytical queries.
Within Microsoft Fabric, these roles remain distinct. **SQL Database in Fabric supports operational relational workloads**, while **warehouses and lakehouses support analytics and reporting**.
The key advantage is that these systems now operate within the **same platform, governance model, and security boundary**, reducing data movement and simplifying integration between operational and analytical workloads.

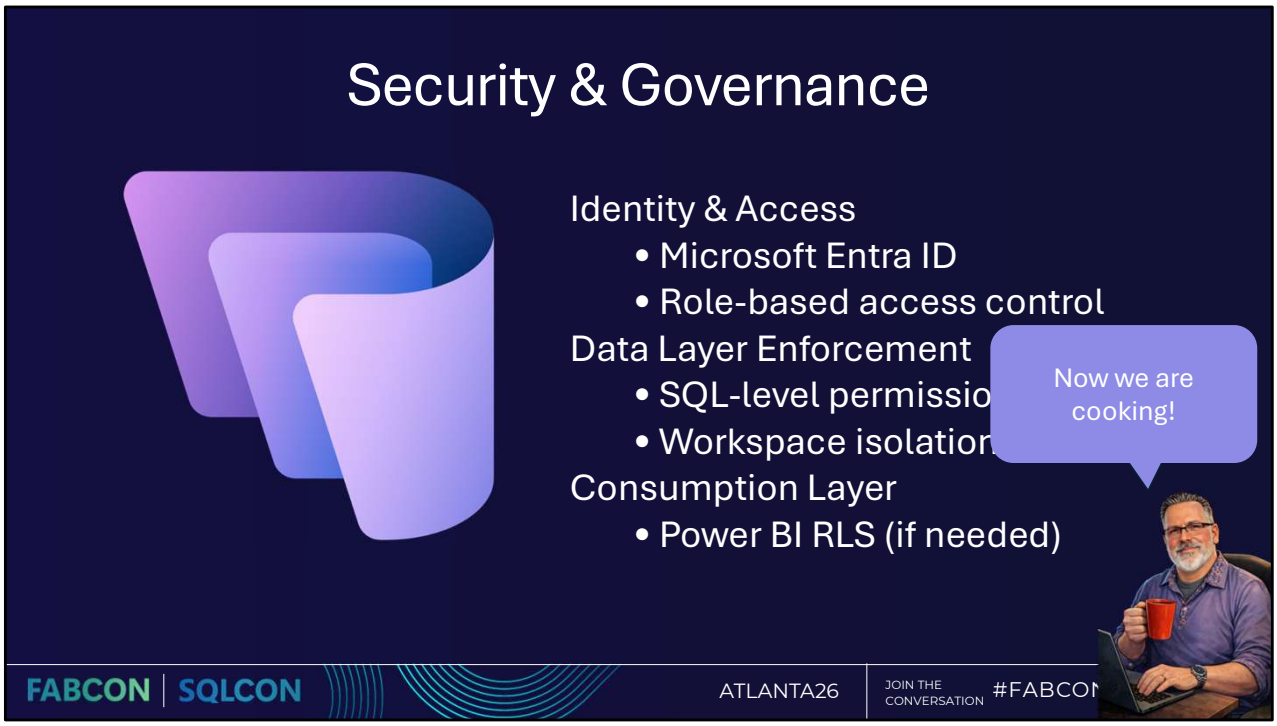
Enterprise-scale reporting and analytics typically remain best suited for dedicated analytical platforms such as warehouses or lakehouses.
However, **operational reporting** can often be performed directly against transactional systems when insights are closely tied to current application activity. Examples include order status, fulfillment progress, service ticket resolution, or other real-time operational metrics.
In these scenarios, reporting directly from the SQL Database in Fabric enables **near real-time visibility** because the data reflects the current transactional state without requiring a separate ETL pipeline.
This approach complements—not replaces—analytical systems. Operational applications can write to **SQL Database in Fabric**, while downstream analytics, dashboards, and curated models can still be developed in Fabric warehouses or lakehouses as needed.



A well-designed data platform separates **transactional workloads (OLTP)** from **analytical workloads (OLAP)** while still allowing them to work together efficiently.
In this architecture, **applications interact with the SQL Database endpoint** to handle transactional operations such as inserts, updates, and operational queries.
Analytical workloads access the data through **analytical endpoints and Fabric services**, allowing reporting, dashboards, and large-scale queries to run without impacting application performance.
This separation maintains performance discipline by ensuring that **analytical queries do not compete with transactional workloads**, while still enabling both workloads to operate within the same unified Fabric platform.



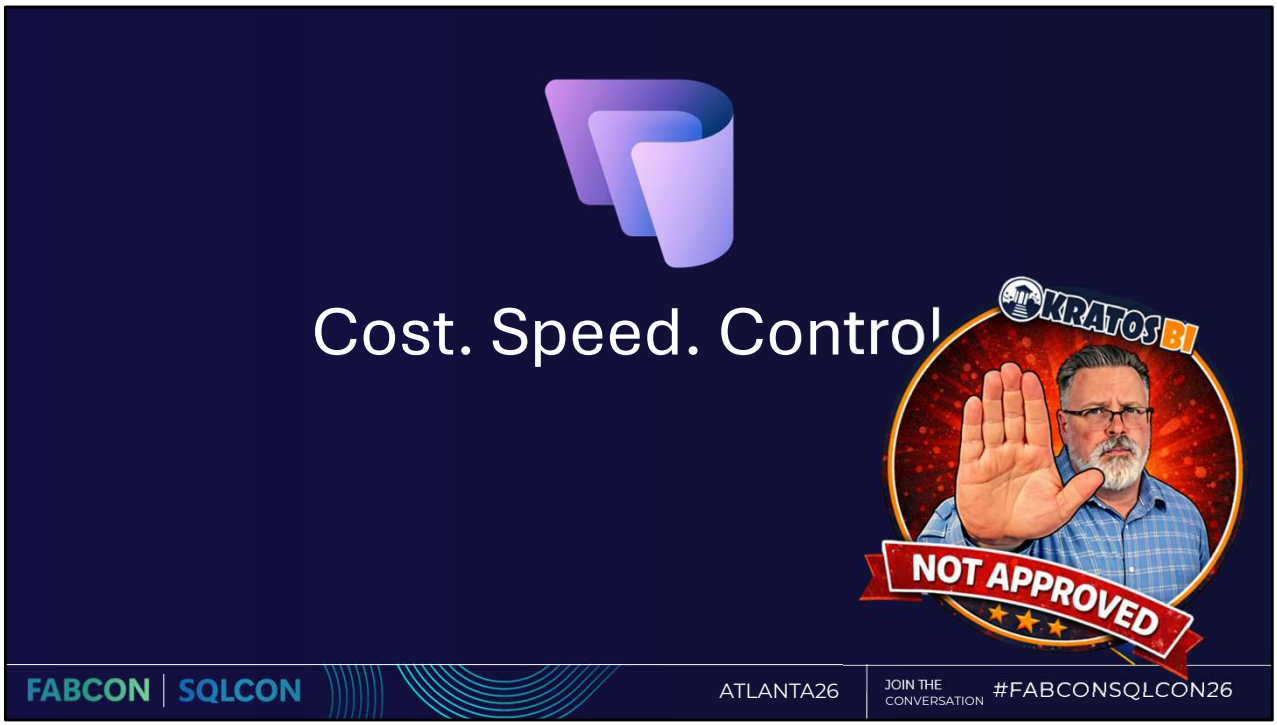
Security and governance remain central considerations when introducing new data workloads into an organization's platform.
SQL Database in Fabric operates within the **existing Microsoft Fabric governance framework**, using the same **Microsoft Entra ID identity system and role-based access control (RBAC) model** already used across the platform.
Because the database runs **inside the Fabric capacity and workspace boundary**, it does not introduce a separate security environment or independent governance model.
This approach allows organizations to maintain consistent control over identity, permissions, and data access while integrating operational relational workloads into the broader Fabric ecosystem.



Effective security in a modern data platform is implemented through **layered governance**, ensuring consistent control from identity to data access to consumption. **Identity and access** are managed through **Microsoft Entra ID**, with permissions granted using role-based access control (RBAC). This provides a consistent identity system across the Fabric platform without requiring separate authentication models for individual tools.

At the **data layer**, SQL-level permissions control access directly within the database, while **workspace boundaries provide isolation** between projects, teams, or environments.

At the **consumption layer**, additional controls such as **Power BI Row-Level Security (RLS)** can be applied to ensure users only see the data they are authorized to access. This layered approach ensures that **analytics workloads inherit governance policies rather than bypassing them**, maintaining consistent security across operational and analytical use cases.



Fabric SQL needs to prove itself in real-world scenarios.
Throughout the demos we evaluate the platform using three criteria:
Cost – Can workloads run efficiently without unnecessary infrastructure or duplicated systems?
Speed – Can teams ingest, process, and query data quickly enough to support operational and analytical needs?
Control – Can organizations maintain governance, security, and operational reliability as workloads scale?
If a solution does not deliver on **Cost, Speed, and Control**, it does not get approved.

What We're Going to Demo

- **Data Ingestion**

Bringing data into Fabric and preparing it for use across the platform.

- **SQL Development**

Working with data using familiar SQL tools and patterns for operational workloads.

- **Analytics Integration**

Using the same data instantly in Power BI and Excel for reporting and analysis.

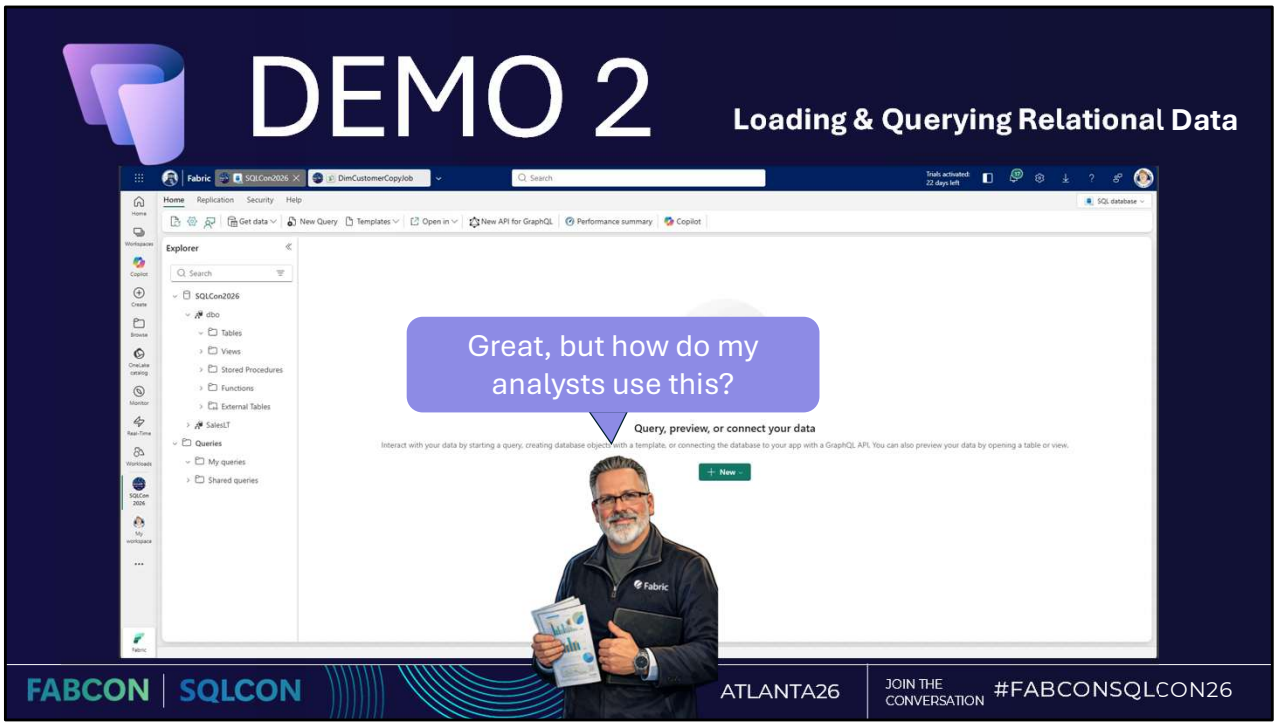
- **Governance & Management**

Monitoring, securing, and managing workloads across the Fabric environment.

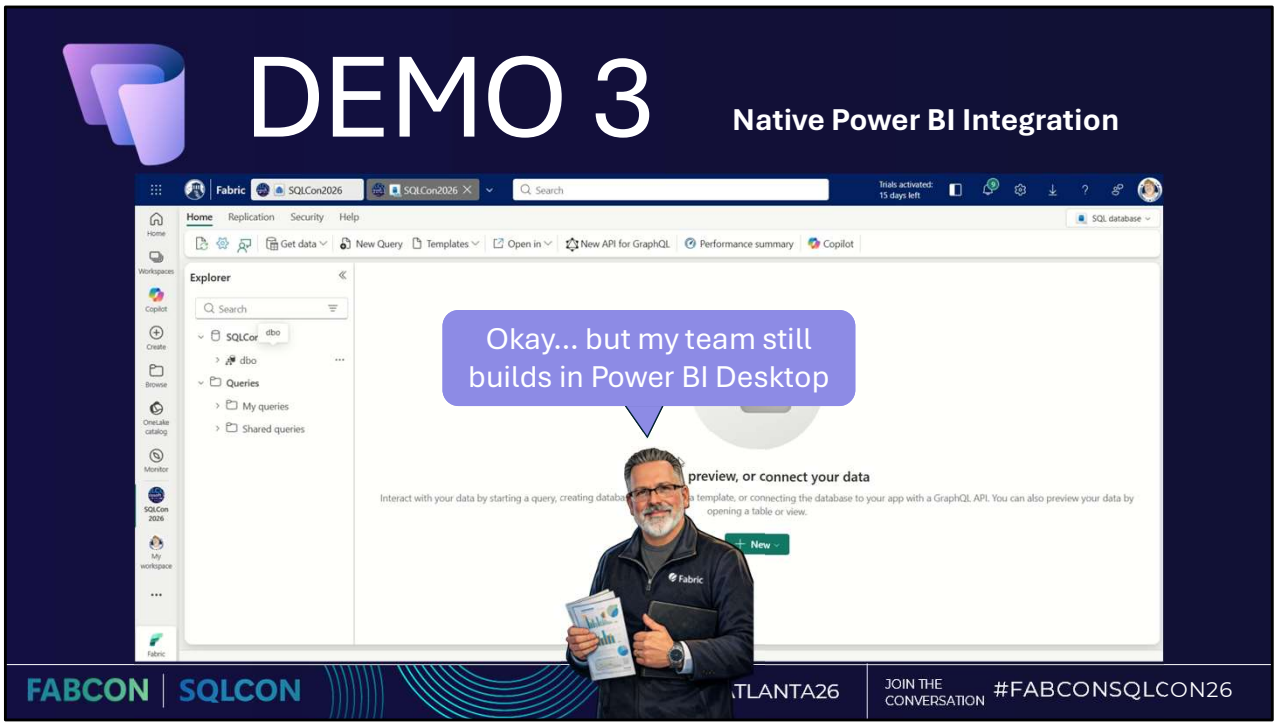
After your architecture talk:
"At this point, the only thing that matters… is whether this is easy."
Beat.
"Let's build one."

"We start by provisioning a SQL Database inside the workspace."

"Notice — this is native to the Fabric workspace. No separate portal. No separate environment."



This demo focuses on how relational data can be **loaded into and queried within SQL Database in Microsoft Fabric**.
Data engineers and developers can create tables, load relational datasets, and interact with the database using familiar SQL tools and interfaces available directly within the Fabric environment.
Once the data is available in the SQL Database, it can support **operational workloads while remaining accessible to downstream analytical tools** within the Fabric ecosystem.
This demonstration also sets up the next step in the workflow: enabling analysts and reporting tools to access the same relational data for analysis, dashboards, and insights without requiring complex data movement or additional platforms.



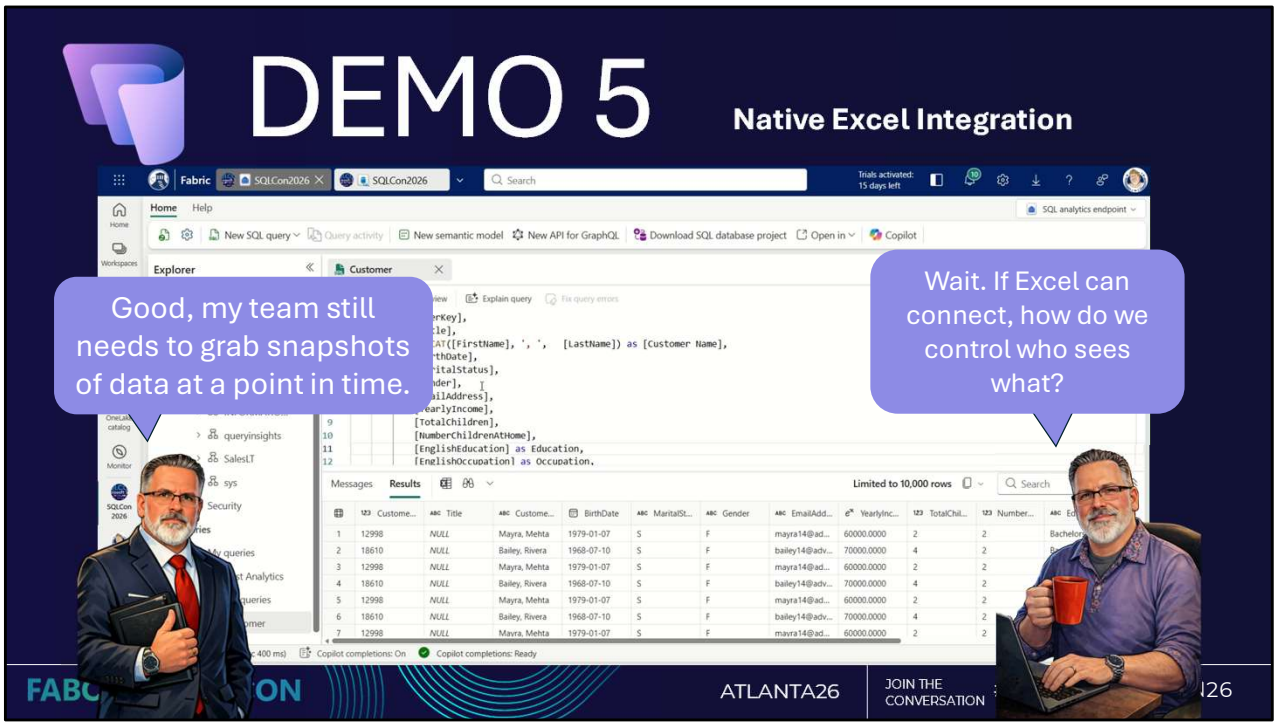
This demo highlights how **SQL Database in Microsoft Fabric integrates directly with Power BI**, enabling analysts to work with relational data without changing their existing development workflows.

Data stored in the SQL Database can be accessed from **Power BI Desktop using standard SQL connectivity**, allowing analysts to build reports, semantic models, and dashboards using the same tools they already use today.

Because the database resides within the Fabric ecosystem, **Power BI can connect to the data without complex data movement or separate infrastructure**, keeping operational data closely aligned with reporting and analytics.

This integration enables teams to continue building in Power BI while benefiting from a unified data platform that supports both operational workloads and analytical insights.

This demo shows how **Power BI Desktop can connect directly to SQL Database in Microsoft Fabric**, allowing analysts to build reports using their existing tools and workflows.
Because the database is part of the Fabric platform, reports can connect to the data without requiring separate extracts or duplicated datasets.
Teams can develop reports, models, and dashboards in **Power BI Desktop while maintaining a single relational data source**, reducing data duplication across reports and simplifying data management.
This approach supports both **analyst productivity and architectural discipline**, allowing Power BI development to continue while keeping operational data centralized within the Fabric platform.



This demo demonstrates how **Excel can connect directly to data stored in SQL Database within Microsoft Fabric**, enabling business users to continue working in familiar tools while accessing centralized data.

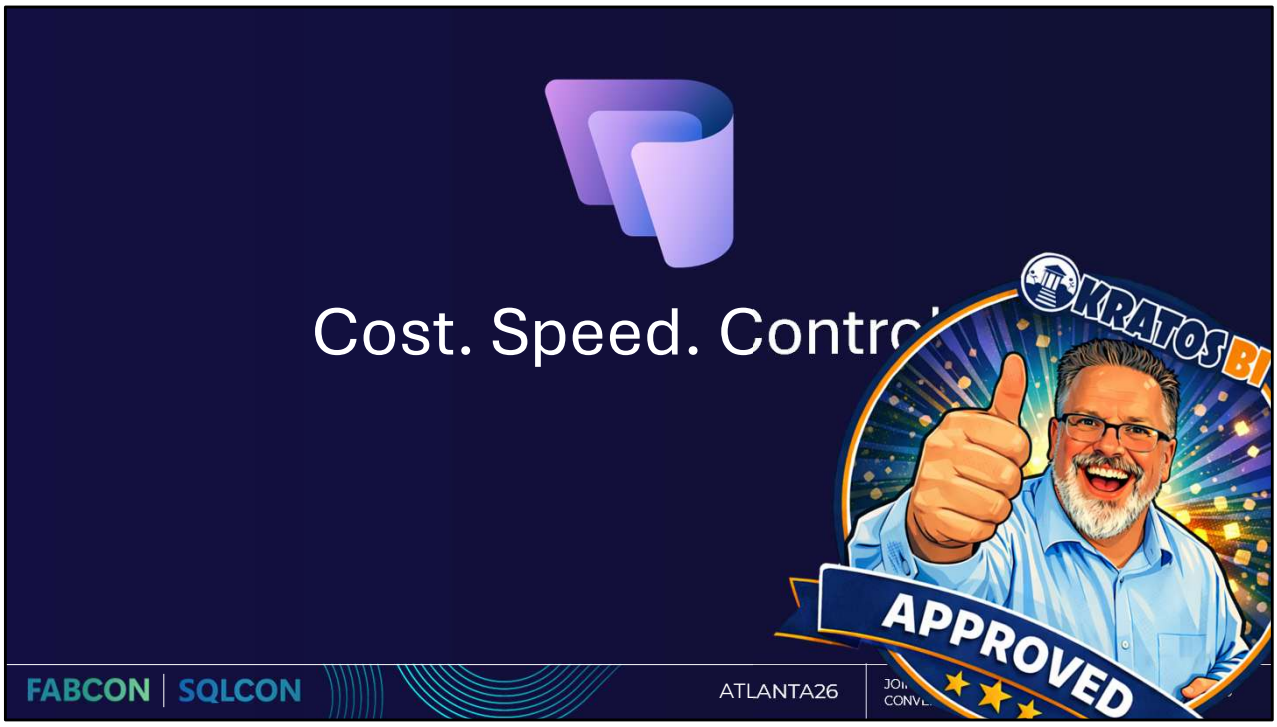
Excel users can query the database to analyze current data or capture **point-in-time snapshots** for ad hoc analysis, reporting, or operational review.

By connecting Excel directly to the Fabric SQL database, teams avoid maintaining separate data extracts while still supporting the flexible workflows many business users rely on.

Access and visibility remain controlled through the platform's **identity, permissions, and governance framework**, ensuring that users only see the data they are authorized to access.

This demo highlights how **governance and security controls extend across the analytics layer in Microsoft Fabric**.
As data from the SQL Database is exposed to reporting tools and semantic models, governance policies can continue to be enforced through **role-based access, model permissions, and row-level security when required**.
Fabric enables organizations to manage governance consistently across operational data and analytical models without creating separate security frameworks for each tool.
This ensures that as data flows from **relational workloads into reporting and analytics**, security policies remain intact and access remains controlled across the entire platform.

transition


Cost. Speed. Contro
KRATOS BI
APPROVED
FABCON | SQLCON
ATLANTA26

Sound off.
The mic is all yours.
Influence the product roadmap.

Join the Fabric User Panel

Share your feedback directly with our Fabric product group and researchers.

https://aka.ms/JoinFabricUserPanel

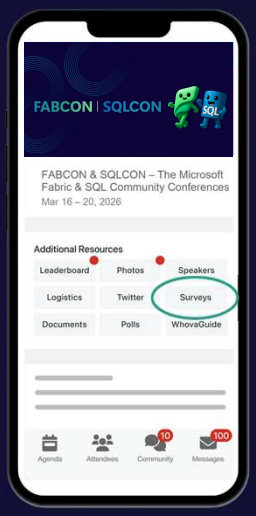
Join the SQL User Panel

Influence our SQL roadmap and ensure it meets your real-life needs

https://aka.ms/JoinSQLUserPanel

How was the session?

Complete Session Surveys in Whova for your chance to WIN PRIZES!



FABCON | SQLCON ATLANTA26 JOIN THE CONVERSATION #FABCONSQLCON26

Get Two Fabric Certifications for FREE

Attendees of FABCON can take the Fabric Analytics Engineer or Fabric Data Engineer exam for free. Be part of the 2 fastest growing role-based certifications in Microsoft history.

Request your voucher by March 23, 2026.

https://aka.ms/fabcon/cert100

FABCON

Power BI Integration

SQLCON