



The Well-Architected Lakehouse

for Data and AI

Leo Mao
Sr. Specialist Solutions Architect



Leo Mao



Sr. Specialist Solutions
Architect - Architecture &
Platform Practice Lead-APJ

- 2.5 years with Databricks
- 18+ years of industry experience
- 12+ years of experience in Data and Analytics (Architecture, Engineering & DS/ML)
- I am the Architecture & Platform Practice Lead in APJ, we have a great team working closely with customers on all aspects related to Architecture and Platforms

Housekeeping

- Your connection will be muted
- Submit questions in the Q&A panel
- If we do not answer your question during the event, we will follow up with you to get you the information you need!
- We will share recording with all attendees after the session

You will learn



What is the Well Architected Framework and Why



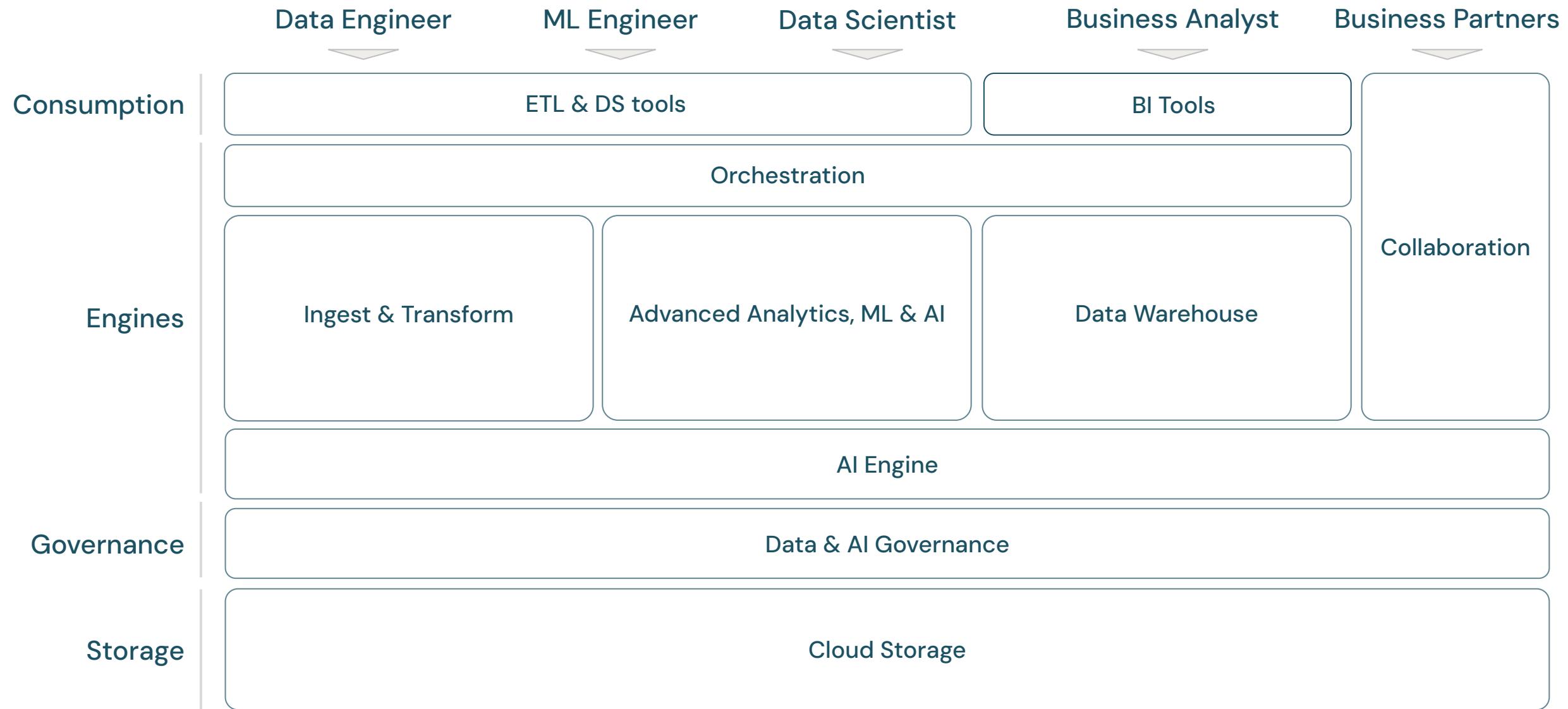
Well Architected Lakehouse and Guiding Principles



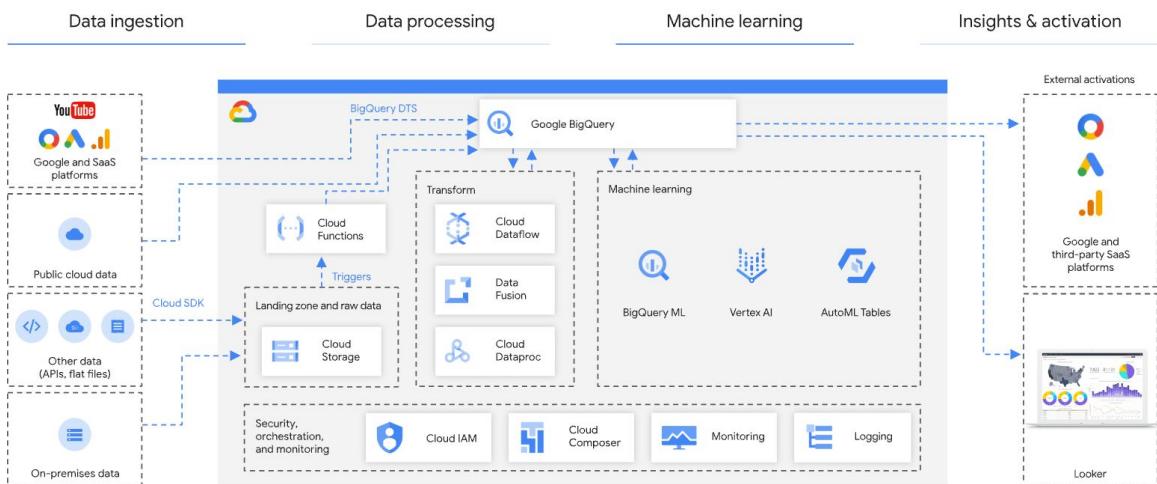
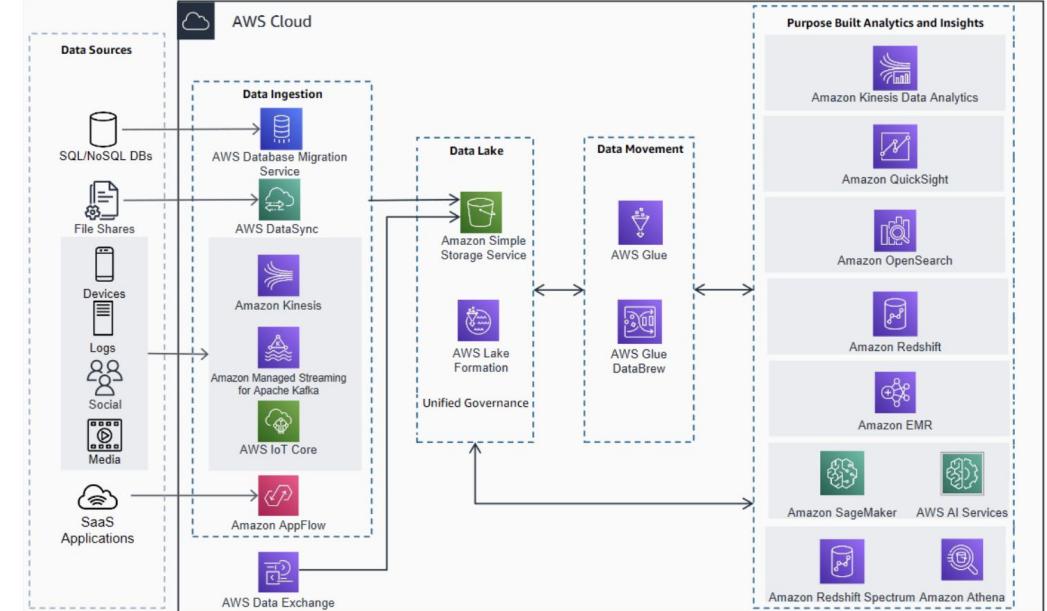
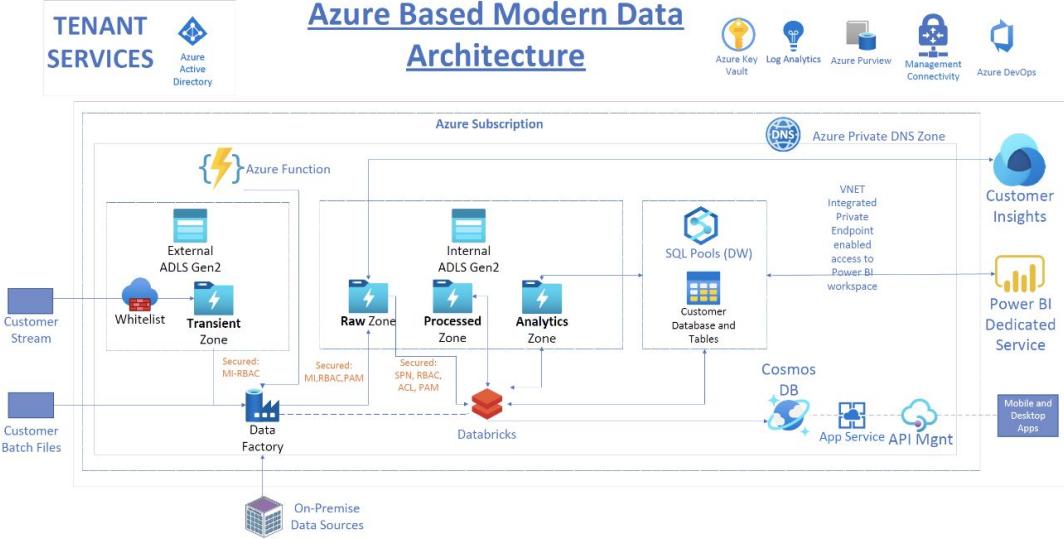
How Databricks help customers to implement a “good” Lakehouse?

What is the Well Architected Framework & Why

Modern Data Platform – Capability

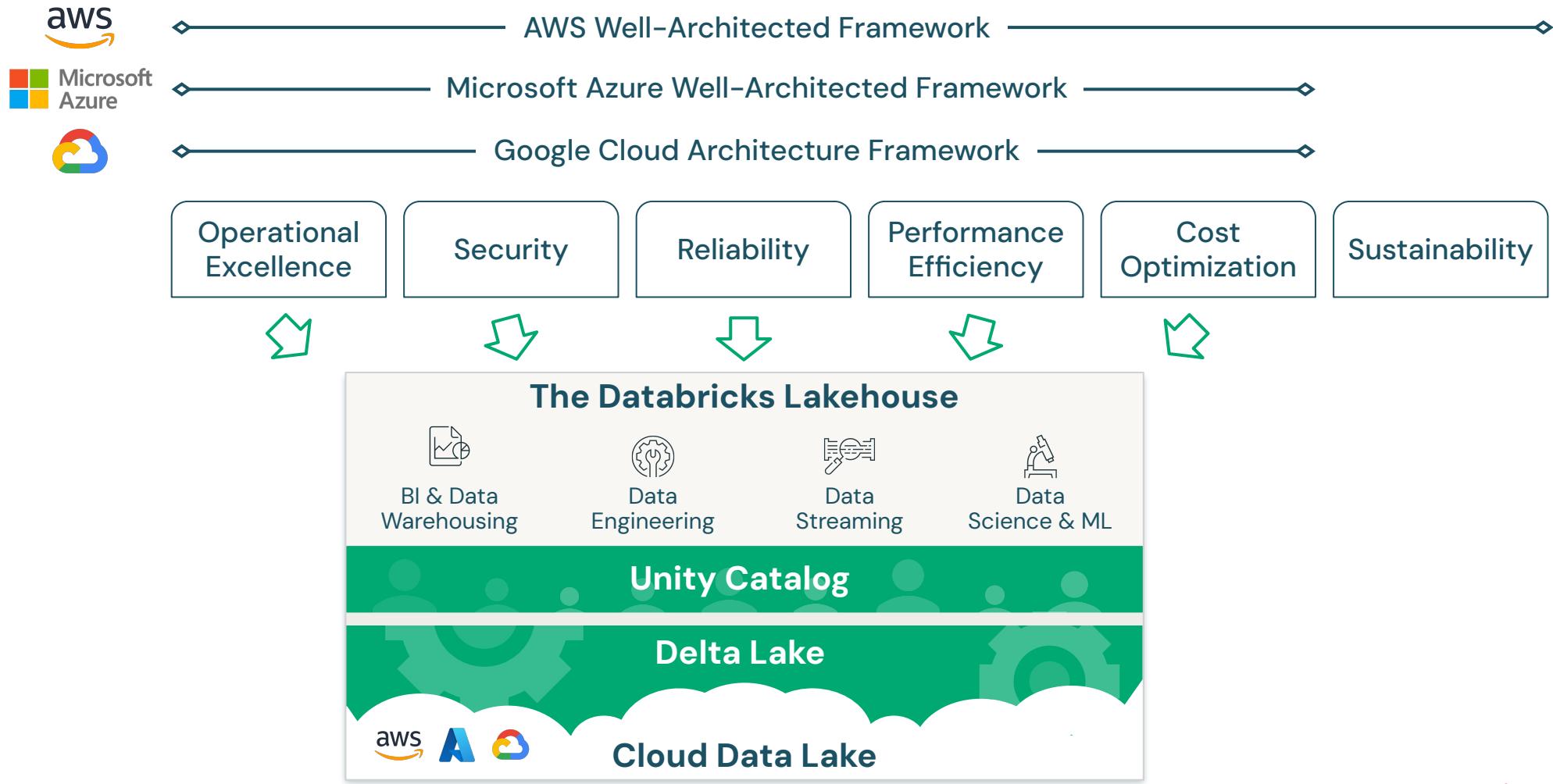


Cloud Data Architecture - Examples

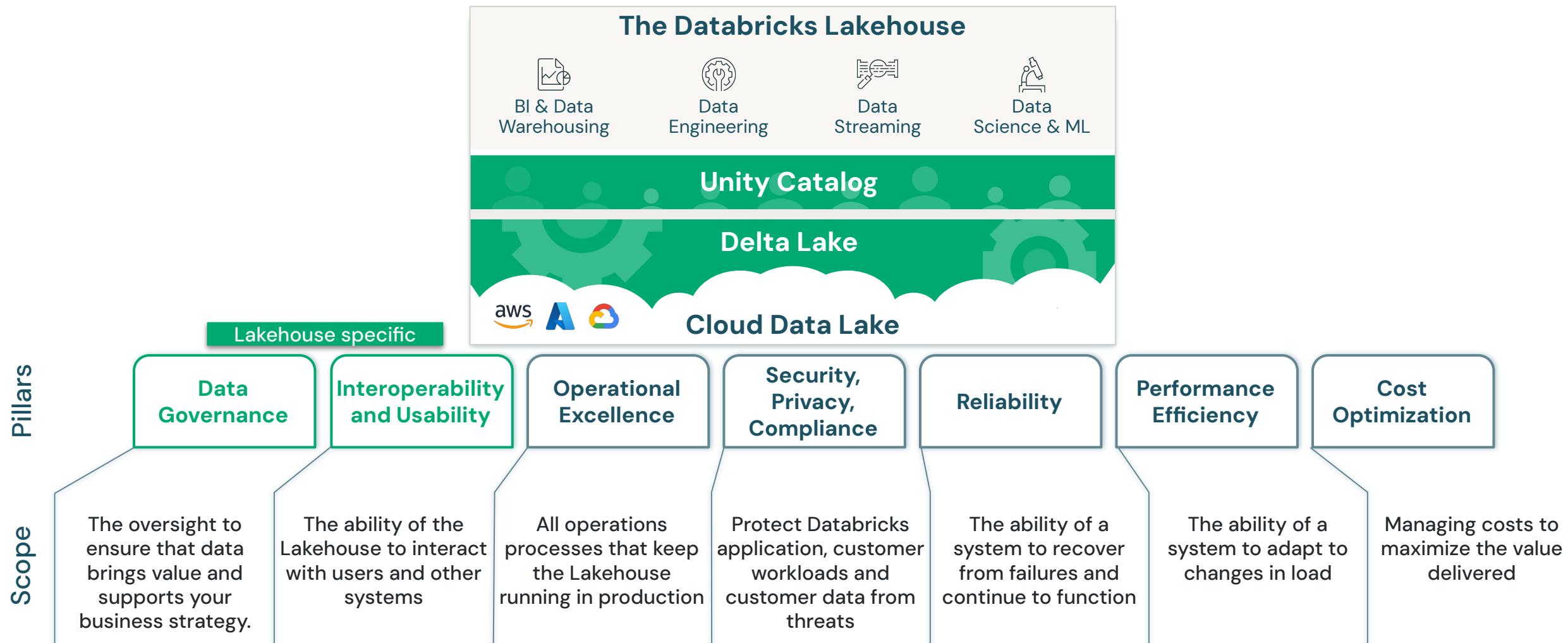


The Well-Architected Lakehouse

Extends the Cloud Well-Architected Frameworks to the Lakehouse

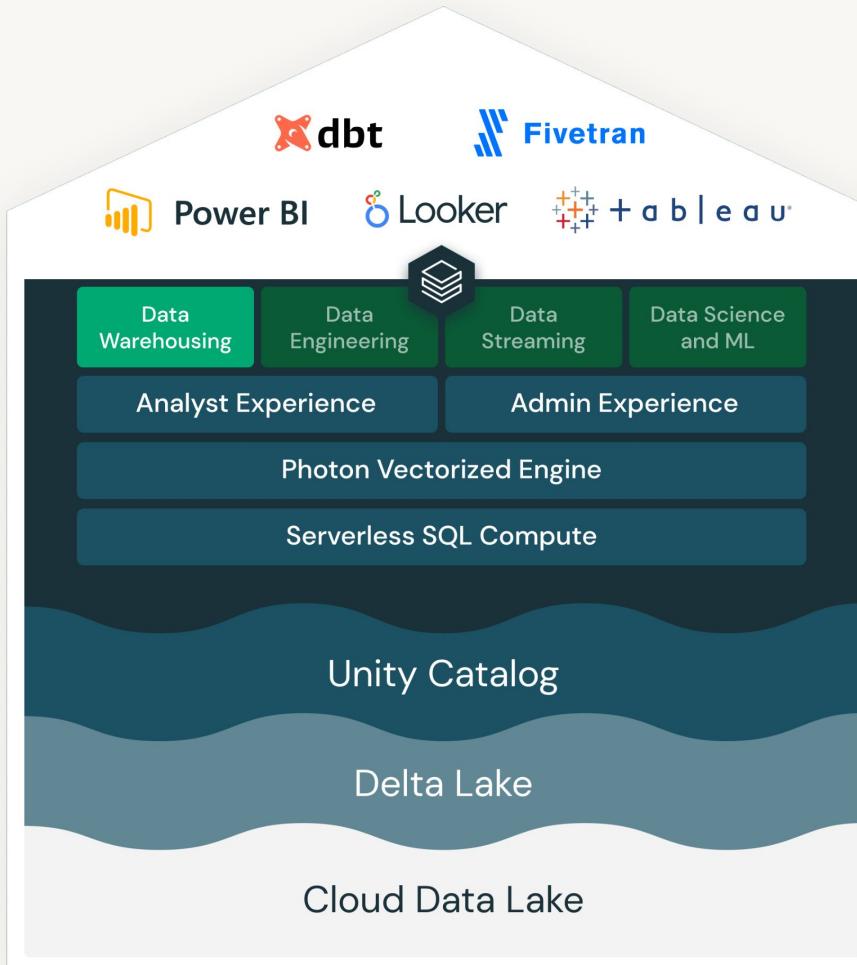


The Well-Architected Lakehouse



Lakehouse for your data platform

Powered by Databricks



Seamless Integration with the Ecosystem

Ease of Use

Real-world Performance

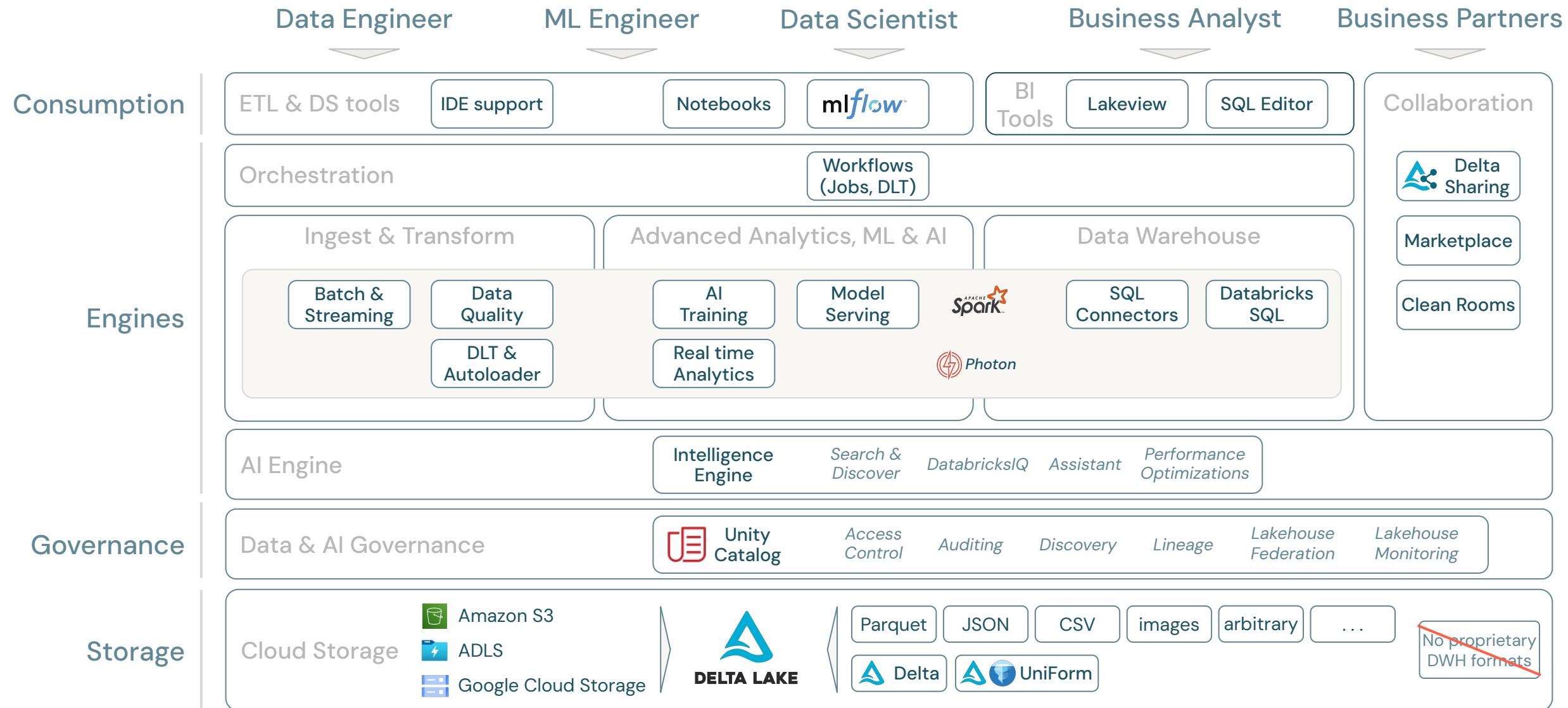
Centralized Governance

Open and Reliable Data Lake as the Foundation



Well Architected Lakehouse & Guiding Principles

Lakehouse – “Data Intelligence Platform”



Guiding Principles for the Lakehouse



Curate Data and Offer Trusted Data-as-Products



Adopt an Organization-wide Data Governance Strategy



Remove Data Silos and Minimize Data Movement



Encourage the Use of Open Interfaces and Open Formats



Democratize Value Creation through Self-Service Experience



Build to Scale and Optimize for Performance & Cost

Well documented Principles and Best Practices



https://docs.databricks.com/lakehouse-architecture/index.html

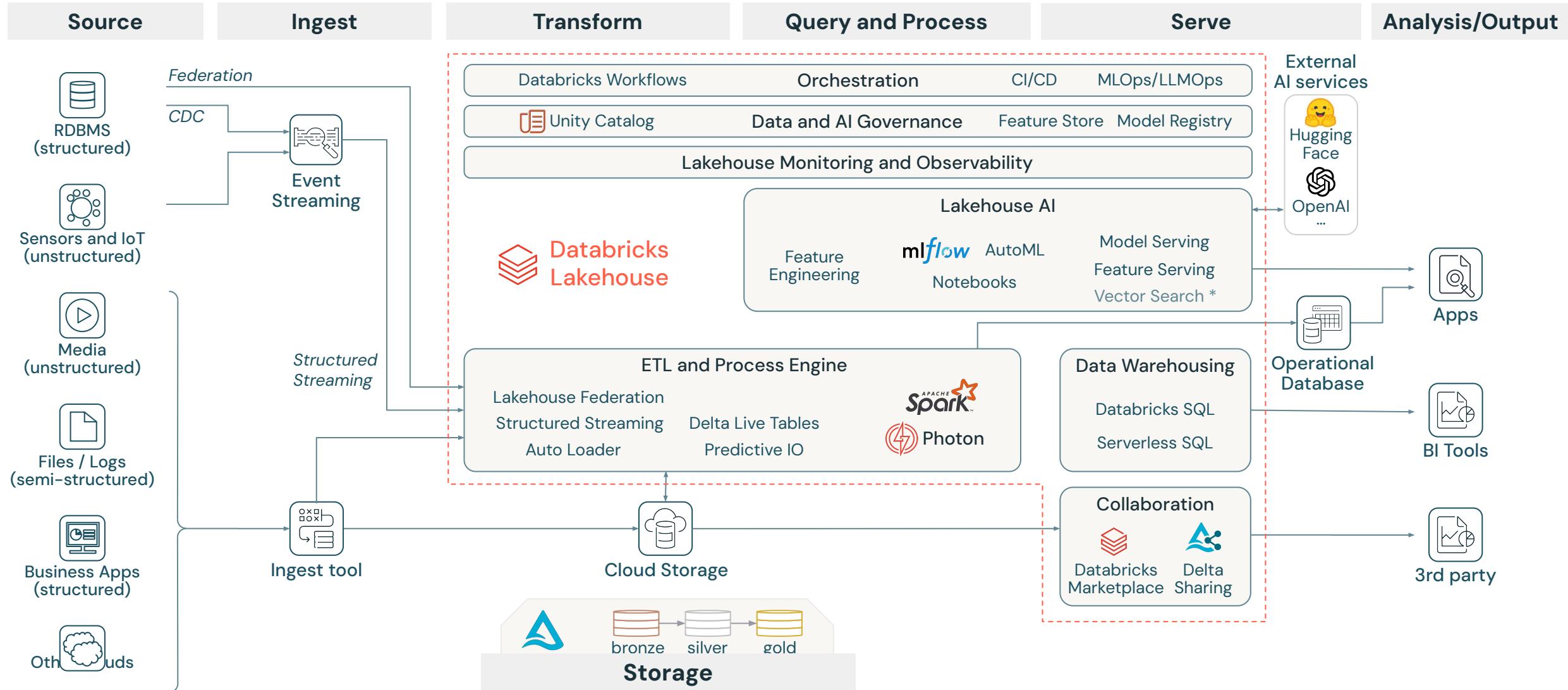
https://docs.gcp.databricks.com/lakehouse-architecture/index.html

https://learn.microsoft.com/en-gb/azure/databricks/lakehouse-architecture

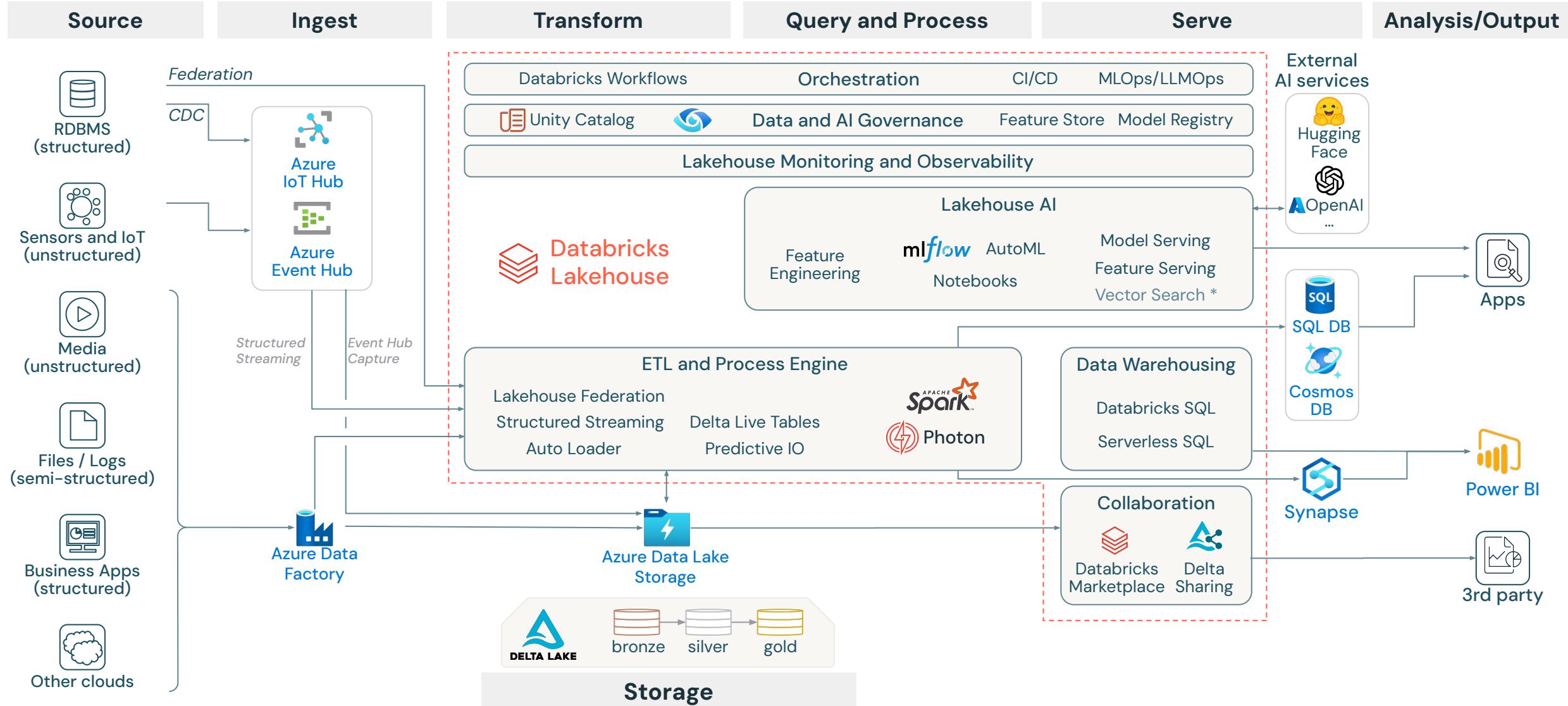
©2023 Databricks Inc. — All rights reserved

14

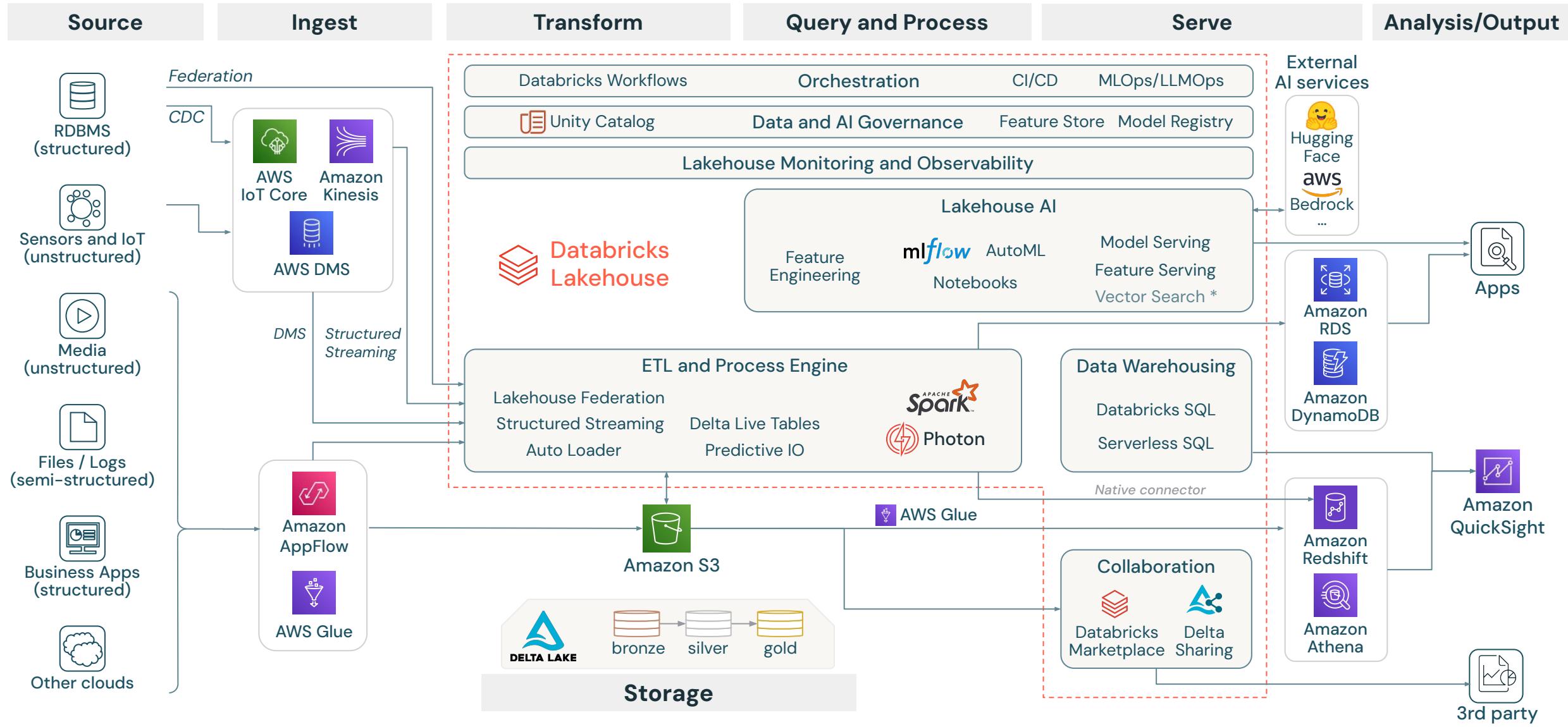
Lakehouse Reference Architecture – Cloud Agnostic



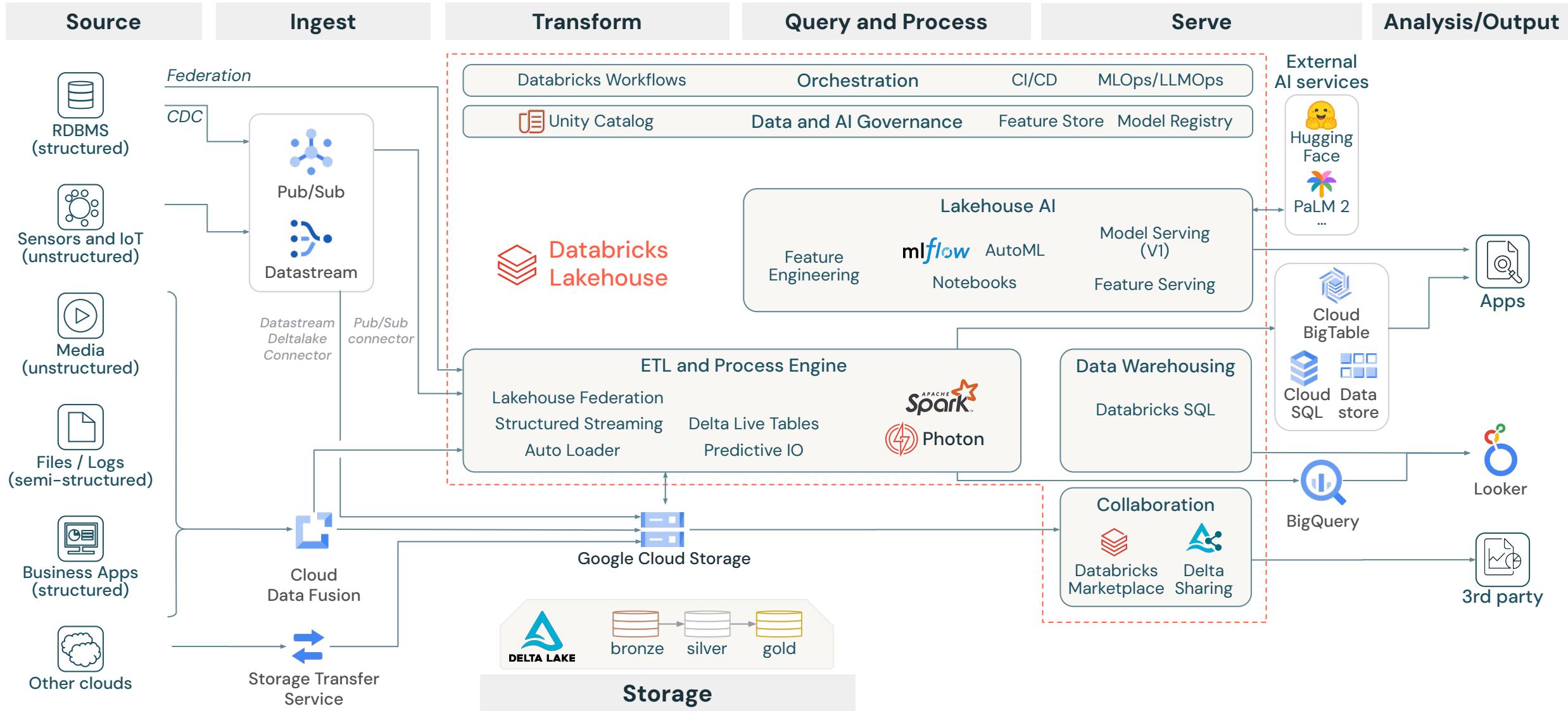
Lakehouse Reference Architecture (Azure)



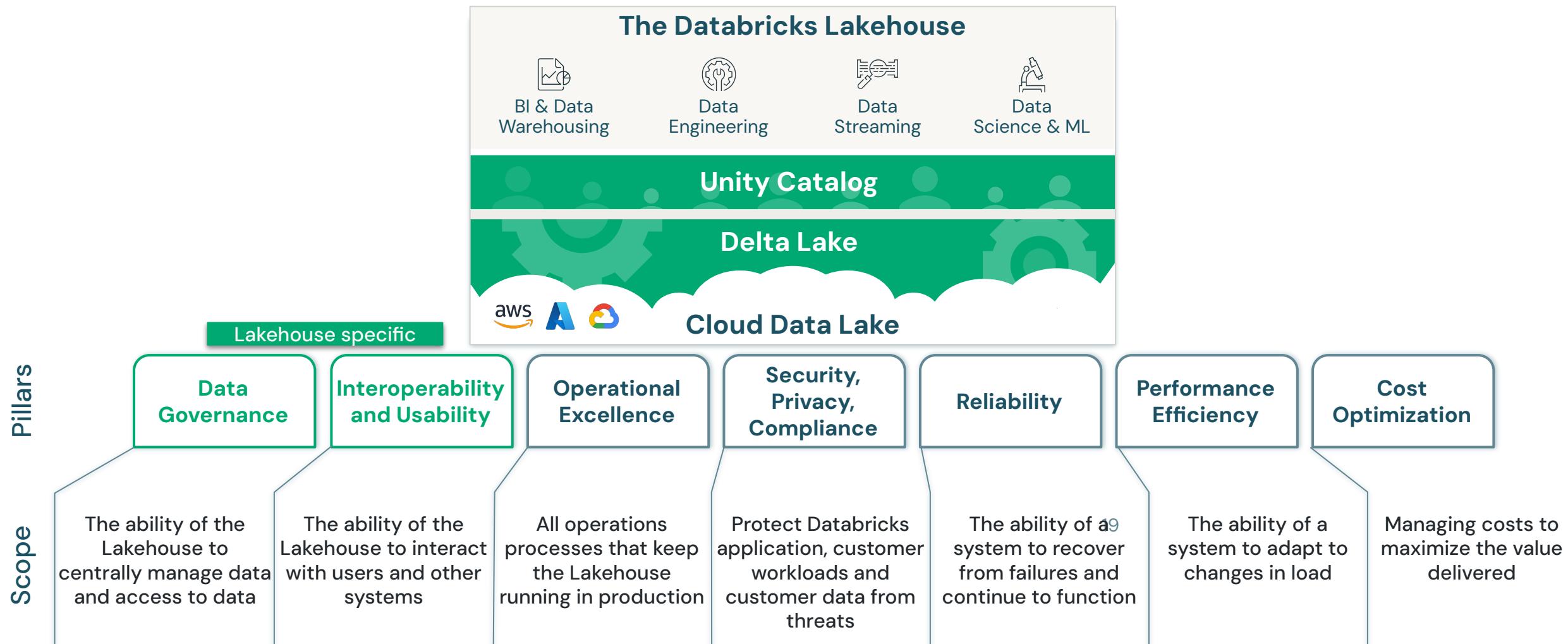
Lakehouse Reference Architecture (AWS)



Lakehouse Reference Architecture (GCP)



The Well-Architected Lakehouse



How do we operationalise it?



Well-Architected Lakehouse Assessment / Review

~ 4 weeks*

~ 2 week*

Architecture and Best Practices Assessment

Assess the current data architecture with Lakehouse Well-Architected Framework & Best Practices - based on core focus areas

~ 3 weeks*

Architectural Design Review

Review the current data architecture with Lakehouse Well-Architected Framework, design high level target architecture & high level implementation plan - based on defined focused areas

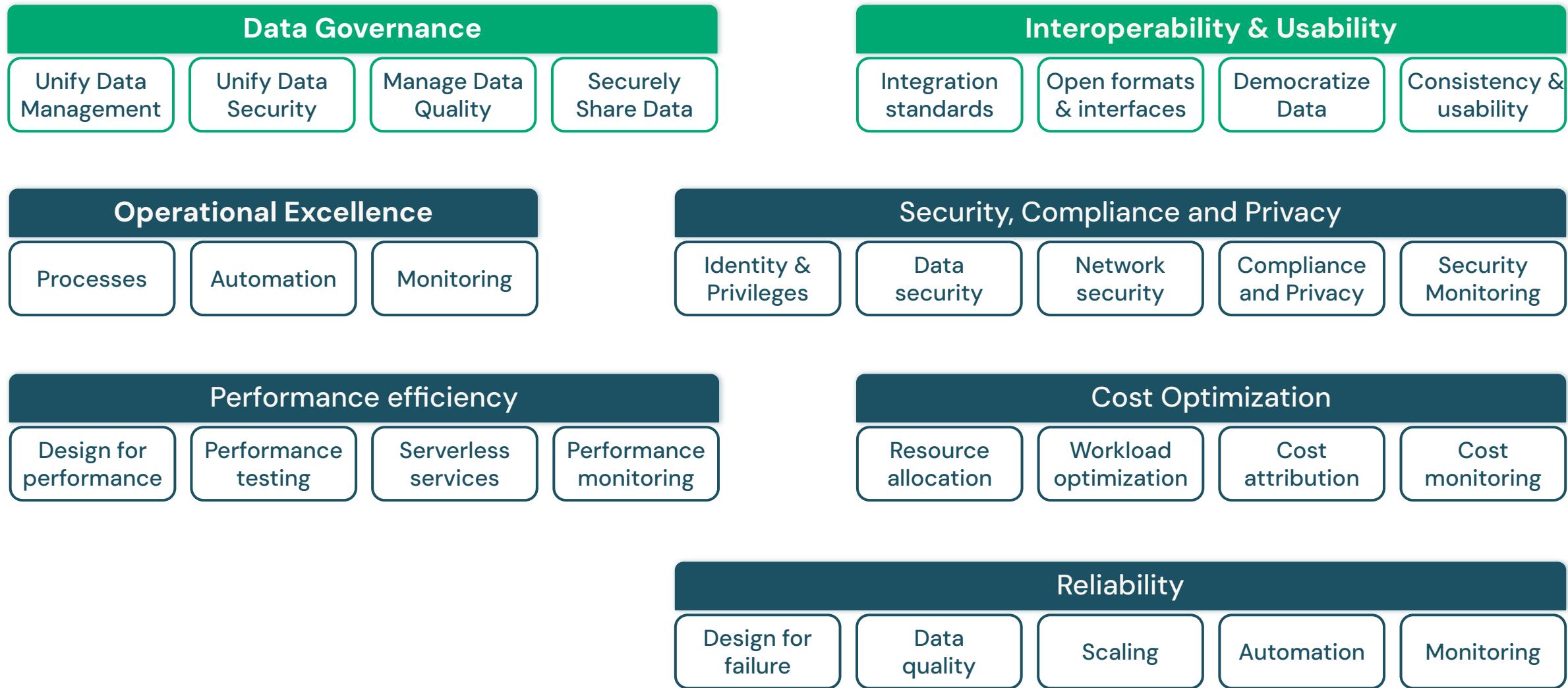
Well Architecture Lakehouse Review – Enterprise

End to End Enterprise review of the current data architecture with Lakehouse Well-Architected Framework, design high level architecture & high level implementation plan



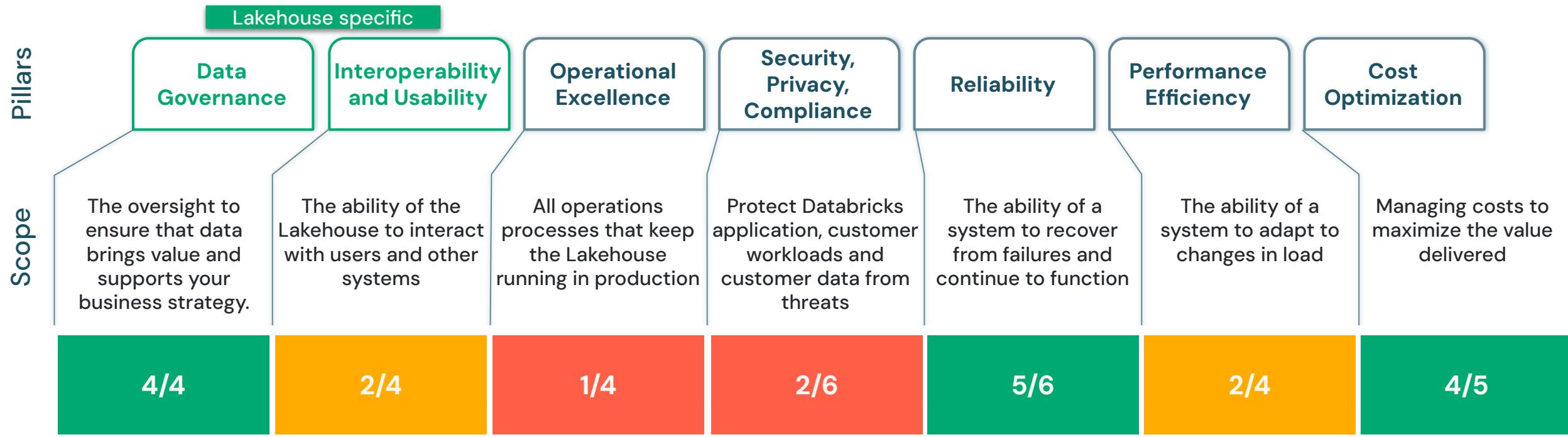
How Databricks help customers to implement a “good” Lakehouse

Grouped best practices



Assessment Summary

High level summary

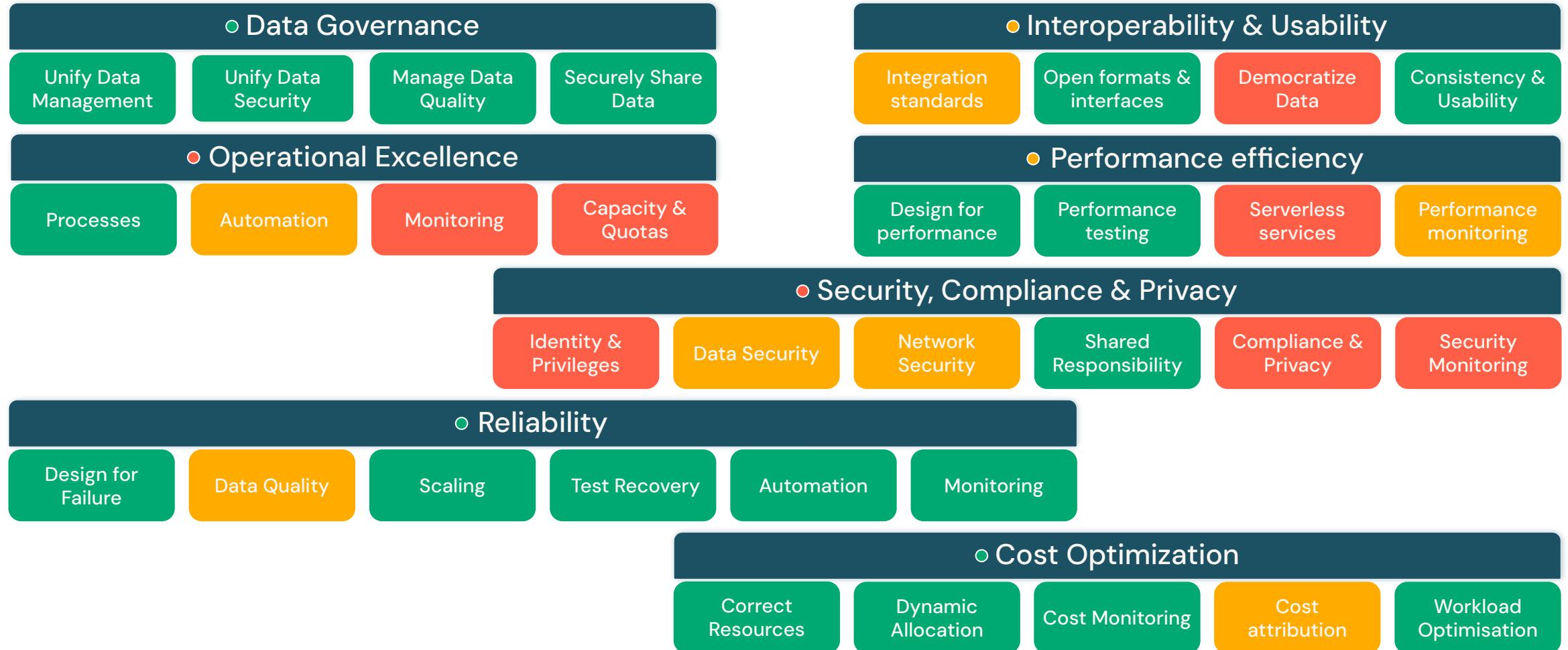


Colour Legend

- = Well Architected, following most or all Best Practices (> 80%)
- = Some Best Practices followed(> 40%)
- = Large Gaps in Best Practices (< 40%)

Assessment Summary

Best practices summary

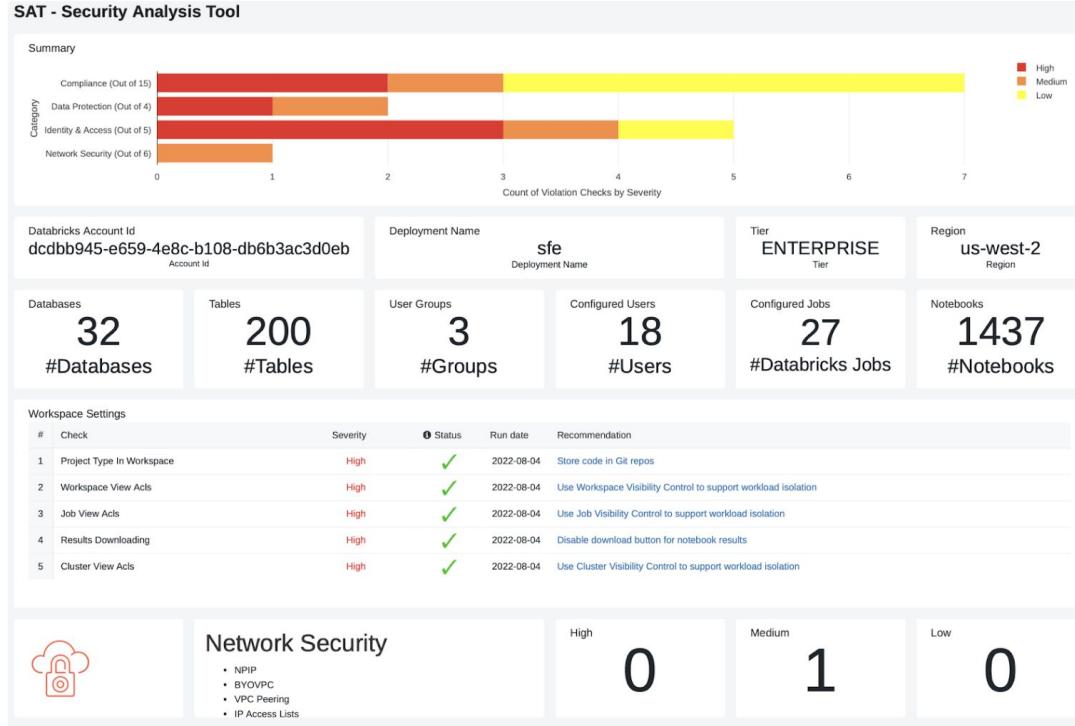


How Databricks help customers to implement a “good” Lakehouse

- PS Offerings: 3 packages
- Consistent and systematic approach
- Assessment and Review Playbooks
- Artifacts/Templates to jump start
- Assessment Tools and recommended workshop plan
- Governed by: Databricks well-architected framework



Tools and Process



Project Sunstone - Customer Recommendations Unassigned Tier (p0) Owner: Dylan L. Slack (dylansl) [Typical] Freshness: 3hr...

Note: Unless indicated otherwise, the score is out of 100.

Score 66

Workspaces Last 28d 7

Users Last 28d 308

Detailed Recommendations

Recommendation	Score	Description
Audit Logs	100	This score measures the implementation of the audit logging feature across total customer workspaces. Audit logging is a service customers can activate which sends low-latency logs in JSON format for any workspace in which it's been configured. Every 15 minutes, Databricks will pipe customer workspace-level metrics to a desired cloud storage location. They contain a rich schema of information including details on accounts, secrets, dbfs, secrets, and more.
Customer Managed Keys	100	Data encryption is an important mechanism in preventing the exposure of potentially sensitive enterprise information and general data governance. Databricks provides this functionality to three critical areas of the enterprise data lake, being workspace dbfs storage, secrets, and both the queries and query history in Databricks SQL. Customers can manage their own cloud-managed keys to encrypt those services.
IP Access Lists	100	IP addresses entering access to Databricks should be restricted to corporate VPN registered IPs, or those from internal users only. Without IP access lists, customers are exposing free Databricks workspaces to the outside world. Good security practices dictate that we only provide access to those users inside private corporate networks.

Detailed Workspace Recommendations

Recommendation	Workspaced	Workspace Name	Cloud and Tier	T-shirt Size	# Users Last 28d	Metadata
Services DBSQL Warehouses	153848019791957	global-job-1	aws ENTERPRISE	L	9	
Upgrade to DBR11 LTS	153848019791957	global-job-1	aws ENTERPRISE	M	9	
Services DBSQL Warehouses	293066574744705	global-users-1	aws ENTERPRISE	S	41	
Upgrade to DBR11 LTS	293066574744705	global-users-1	aws ENTERPRISE	L	41	
Services DBSQL Warehouses	427150201465914	all-cloud	aws ENTERPRISE	L	16	
Upgrade to DBR11 LTS	427150201465914	all-cloud	aws ENTERPRISE	L	16	

Summary: Data Governance

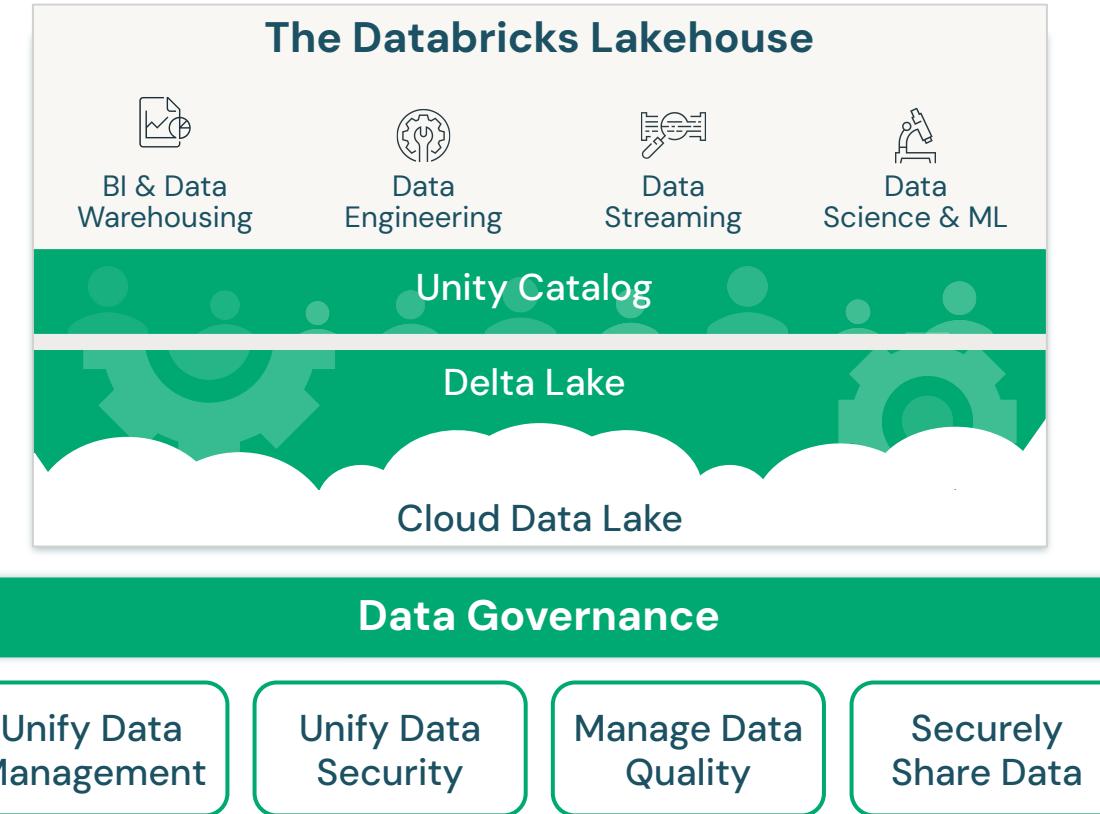
Data Governance

Dimension	Principle	Health and Key Findings	Key Recommendations
Unify Data Governance	1	Unify data management	<ul style="list-style-type: none"> Define Databricks UC Roles per data governance policies Use Catalogs to provide segregation across EQL organisation's information architecture
	2	Unify data security	None
Data Quality	3	Manage data quality	<ul style="list-style-type: none"> Define Schema / Metadata data quality rules and implementation Build data ingestion auditing frameworks
Data Sharing	4	Share data securely and in real-time	<ul style="list-style-type: none"> Consider using Delta Sharing to manage, govern, audit and track usage of shared data



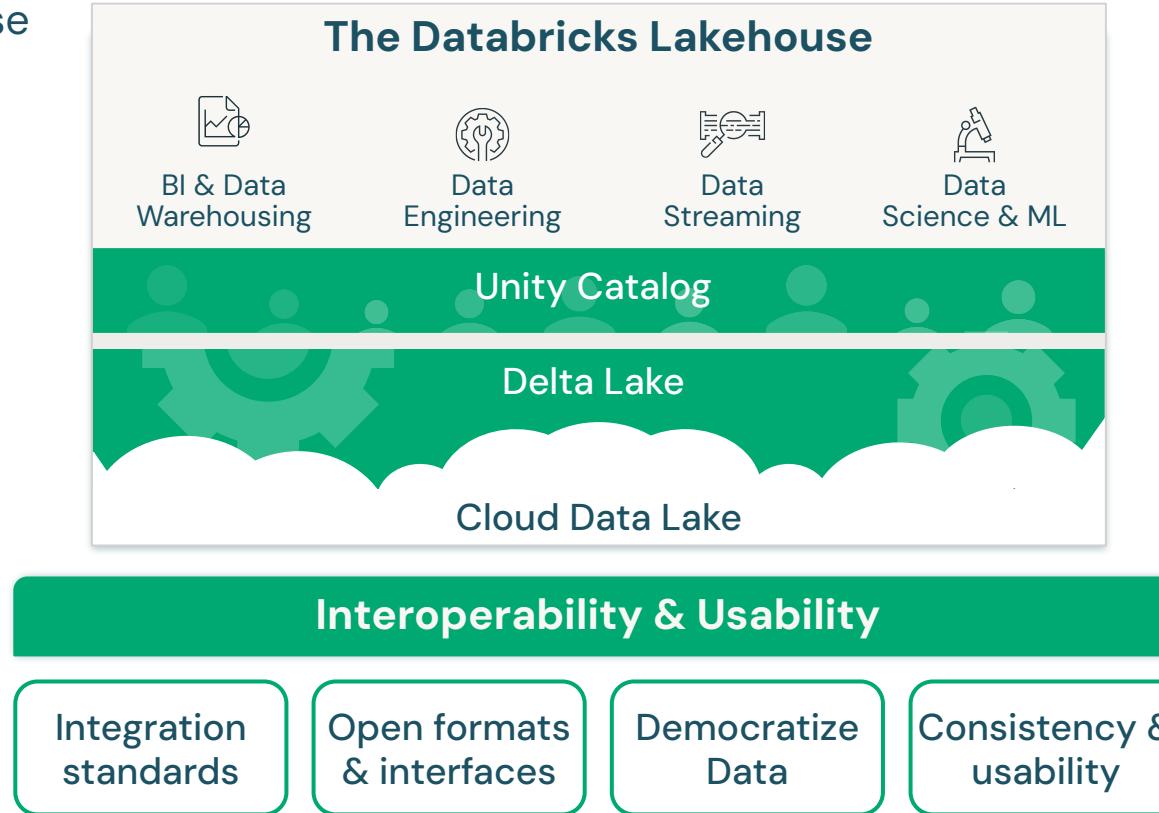
Data Governance

Centrally manage data and access to data for the Lakehouse



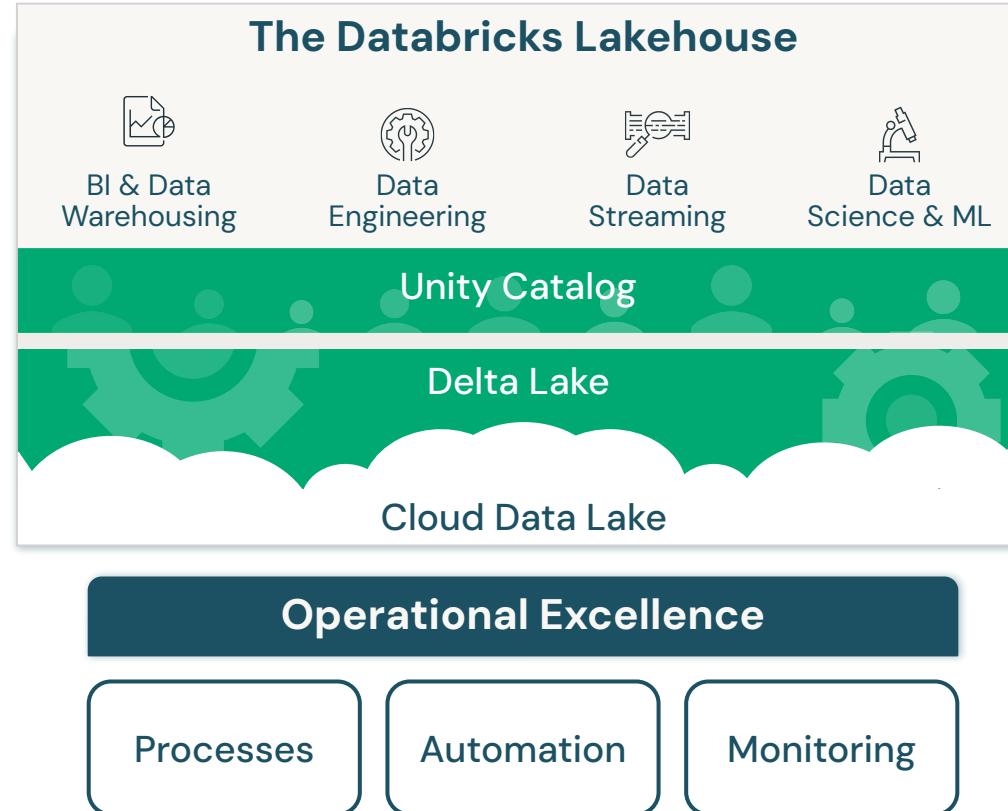
Interoperability & Usability

The ability of the Lakehouse to interact with users and other systems



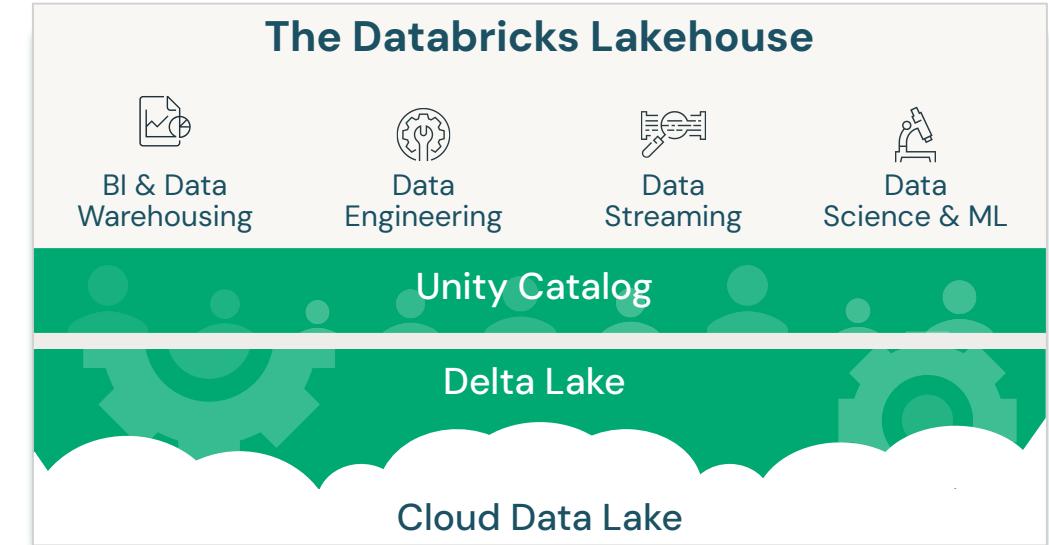
Operational Excellence

All operations processes
that keep the Lakehouse
running in production



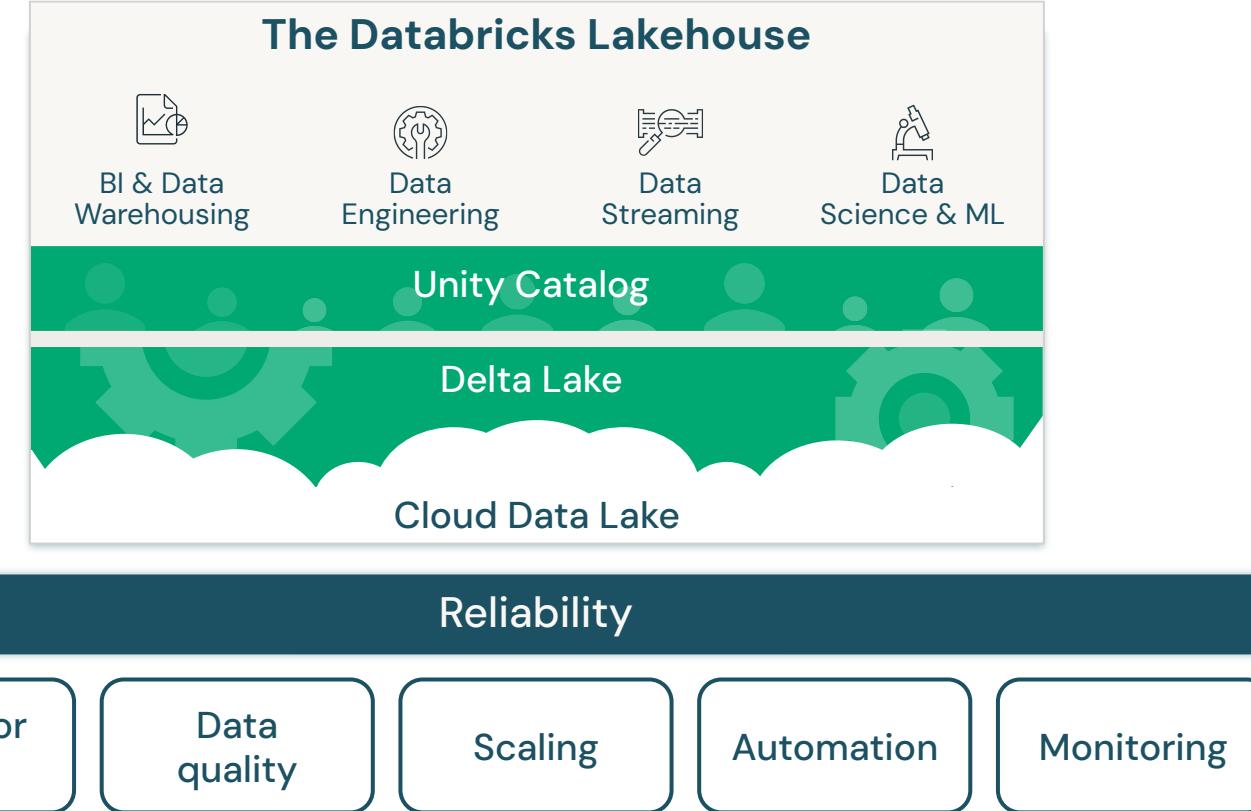
Security, Compliance and Privacy

Protect the Databricks application, customer workloads and customer data from threats



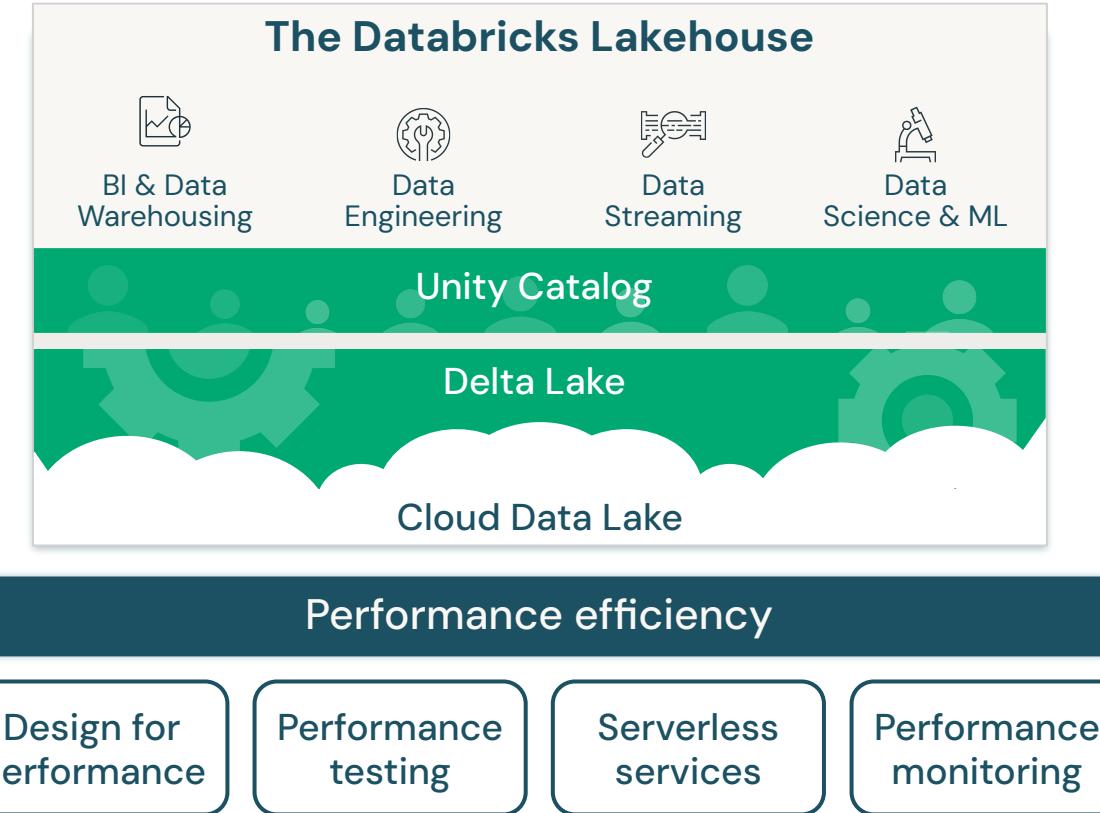
Reliability

The ability of the Lakehouse to recover from failures and continue to function.



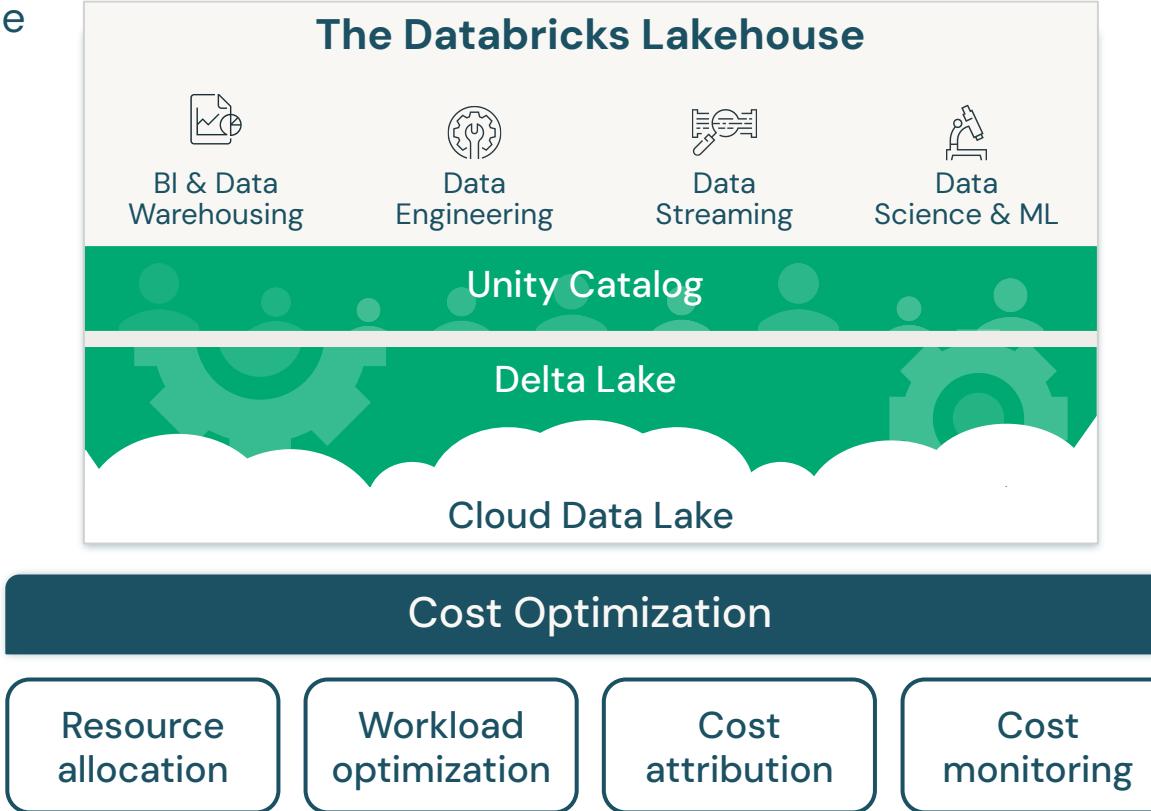
Performance efficiency

The ability of the Lakehouse
to adapt to changes in load



Cost Optimization

Managing costs to maximize the value delivered



Curious to learn more?

- Refer to the docs: [Databricks well-architected framework](#)
- Talk to your Databricks Representative or
- Contact us @ <https://www.databricks.com/company/contact>

Thank you!

Q & A

Appendix

1 Data Governance

Dimension	Principle	Features and Best Practices
Unify Data Governance	Unify data management	<ul style="list-style-type: none">• Manage metadata for all data assets in one place• Track data lineage to drive visibility of the data• Discover data and related information via the Data Explorer
	Unify data security	<ul style="list-style-type: none">• Centralize access control• Configure audit logging• Audit Unity Catalog events• Audit data sharing events
Data Quality	Manage data quality	<ul style="list-style-type: none">• Use a layered storage architecture• Improve data integrity by reducing data redundancy• Actively manage schemas• Use constraints and data expectations• Take a data-centric approach to machine learning
Data Sharing	Share data securely and in real-time	<ul style="list-style-type: none">• Use the open Delta Sharing protocol for sharing data with partners• Use Databricks-to-Databricks Delta sharing between Databricks users

2 Interoperability and Usability

Dimension	Principle	Features and Best Practices
Interoperability	Settle on standards for integration	<ul style="list-style-type: none">• Use the Databricks REST API for external integration• Use optimized connectors to access data sources from the Lakehouse• Leverage partners available in Partner Connect• Use Delta Live Tables and Auto Loader
	Prefer open interfaces and open data formats	<ul style="list-style-type: none">• Use the Delta data format• Use Delta Sharing to exchange data with partners• Use MLflow to manage Machine Learning workflows
Usability	Lower the barriers to access the Lakehouse	<ul style="list-style-type: none">• Provide a self-service experience across the platform• Use the serverless services of the platform• Offer predefined clusters and SQL Warehouses for different use cases
	Ensure data consistency and usability	<ul style="list-style-type: none">• Offer reusable data-as-products that the business can trust• Publish data products semantically consistent across the enterprise• Use Unity Catalog for data discovery and lineage exploration

3 Operational Excellence (1/2)

Dimension	Principle	Features and Best Practices
Release Processes	Optimize build and release processes	<ul style="list-style-type: none">• Use Databricks Repos to store code in Git• Standardize on DevOps processes (CI/CD)• Standardize in MLOps processes
Automation and Reproducibility	Automate deployments and workloads	<ul style="list-style-type: none">• Use Infrastructure as Code for deployments and maintenance• Use cluster policies• Use automated workflows for jobs• Use Auto Loader• Use Delta Live Tables• Follow the “deploy code” approach for ML workloads• Use a model registry to decouple code and model lifecycle• Use MLflow Autologging• Reuse the same infrastructure to manage ML pipelines

3 Operational Excellence (2/2)

Dimension	Principle	Features and Best Practices
Monitoring	Set up system / workload monitoring, alerting, and logging	<ul style="list-style-type: none">• Platform Monitoring via cloud monitoring solutions• Cluster Monitoring via Ganglia• SQL Warehouse monitoring• Auto Loader Monitoring• Delta Live Tables Monitoring• Streaming Monitoring• ML Models Monitoring• Security monitoring• Cost Monitoring
Capacity Management	Manage capacity and quota	<ul style="list-style-type: none">• Manage capacity and quota• Invest in capacity planning

4 Security, Privacy, Compliance (1/2)

Dimension	Principle	Features and Best Practices
Identity and Access	Manage identity and access using least privilege	<ul style="list-style-type: none">• Authenticate via single sign-on• Leverage multi-factor authentication• Disable local passwords• Set complex local passwords• Separate admin accounts from normal user accounts• Token management• SCIM synchronization of users and groups• Configure access control• Limiting cluster creation rights• Store and use secrets securely• Cross-account IAM role configuration• Customer-approved workspace login• Use clusters that support user isolation• Use Service Principals to run jobs and all automated tasks
Data security	Protect data in transit and at rest	<ul style="list-style-type: none">• Avoid storing production data in DBFS• Secure access to cloud storage• Leverage data exfiltration settings within the admin console• Use bucket versioning• Encrypt storage and restrict access• Add a customer-managed key for managed services• Add a customer-managed key for workspace storage

4 Security, Privacy, Compliance (2/2)

Dimension	Principle	Features and Best Practices
Network security	Secure your network and identify and protect endpoints	<ul style="list-style-type: none">• Deploy with a customer-managed VPC or VNet• Use Secure cluster connectivity• Use IP access lists• Implement network exfiltration protections• Apply VPC service controls• Use VPC endpoint policies• Configure PrivateLink• Configure domain name firewall rules
Compliance	Review shared responsibility	<ul style="list-style-type: none">• Review the Shared Responsibility Model
	Meet the compliance and data privacy requirements	<ul style="list-style-type: none">• GDPR and CCPA compliance using Delta Lake• HIPAA compliance features• PCI-DSS compliance controls• FedRAMP compliance
Monitoring	Monitor system security	<ul style="list-style-type: none">• Use Databricks audit log delivery• Configure AWS, Azure, GCP tagging to monitor usage and enable charge-back• Monitor provisioning activities• Enhanced Security Monitoring
Generic Controls	n/a	<ul style="list-style-type: none">• Service quotas• GCP org policies• Controlling libraries• Isolate sensitive workloads into different workspaces• Leverage CI/CD processes to scan code for hard coded secrets• Use AWS Nitro instances



5 Reliability

Dimension	Principle	Features and Best Practices
Resilience	Design for failure	<ul style="list-style-type: none">• Use Delta Lake• Use Apache Spark or Photon for your distributed workloads• Automatically rescue invalid or non-conforming data• Configure jobs for automatic retries and termination• Use a scalable and production grade model serving infrastructure
Data Quality	Manage data quality	<ul style="list-style-type: none">• Use a layered storage architecture• Improve data integrity by reducing data redundancy• Actively manage schemas• Use constraints and data expectations• Take a data-centric approach to machine learning
Autoscaling	Design for autoscaling	<ul style="list-style-type: none">• Enable autoscaling for batch workloads• Enable autoscaling for SQL Warehouse• Leverage DLT's enhanced autoscaling
Recovery	Test recovery procedures	<ul style="list-style-type: none">• Backup via the Databricks migration tool• Recover from Structured Streaming query failures• Recover ETL jobs based on Delta time travel• Leverage Databricks Workflows and its built in recovery• Configure a disaster recovery
Automation	Automate deployments and workloads	See Operational Excellence – Automation
Monitoring	Set up system / workload monitoring, alerting, and logging	See Operational Excellence – Monitoring



6 Performance Efficiency

Dimension	Principle	Features and Best Practices
Design	Use serverless architectures	<ul style="list-style-type: none">• Use serverless compute
	Design for performance	<ul style="list-style-type: none">• Use parallel computing where beneficial• Be aware of data access patterns• Use parallel computation where it is beneficial• Analyze the whole chain of execution• Prefer larger clusters• Use native operations• Use Photon• Use caching• Optimize file management• Join optimizations
Testing	Run performance testing in the scope of development	<ul style="list-style-type: none">• Test on data representative for production data• Take prewarming of resources into account• Control caching• Identify bottlenecks
Monitoring	Monitor performance	See Operational Excellence

7 Cost Optimization

Dimension	Principle	Features and Best Practices
Resource Selection	Choose the correct resources	<ul style="list-style-type: none">• Use Delta• Use automated Job clusters for jobs• Use SQL Warehouse for SQL workloads• Use the right runtime for your workloads• Ensure to use CPU and GPU for the right workloads• Consider using spot instances
Right Sizing	Dynamically allocate and deallocate resources	<ul style="list-style-type: none">• Use auto termination
	Optimize workloads, aim for scalable costs	<ul style="list-style-type: none">• Only use online streaming when the latency requirements demand for it• Choose the most efficient cluster size• Leverage auto-scaling compute• Use auto termination
Cost/performance profiling	Monitor and control cost	<ul style="list-style-type: none">• Monitor costs• Evaluate Photon for your workloads• Evaluate Serverless for your workloads
Cost Control and attribution	Analyze and attribute expenditure	<ul style="list-style-type: none">• Tag clusters for cost attribution

Databricks AI

End to end AI capabilities ...

Databricks AI

External Services

MLOps + LLMOps

- Move Code, Data, and Models between development and production
- Manage Models, Features, Experiments

Prepare Data

- Discover & Transform Structured Data into Features
- Chunk & Create Embeddings from Unstructured Data

Develop & Evaluate AI

- Train and Test Algorithms
- Fine-Tune & Prompt Engineer Models
- Create GenAI agents & tools
- Chain together assets
- Evaluate Experiments

Serve AI

- Low Latency Model Serving
- Log Model Requests/ Responses

Serve Data

- Low Latency Feature Serving
- Query Embeddings in Vector DB

AI Engine

- AI driven discovery and search
- AI Assistant
- AI driven performance optimization and scaling

Data & AI Governance

- Manage security & permissions
- Track model lineage
- Data monitoring
- AI monitoring (Evaluate model accuracy to maintain quality)

Cloud Storage

Databricks AI

... fully integrated into the Data Intelligence Platform

