IMAGINARY HEALTHCARE

NEXTGEN

EDWHAP

Enterprise Data Warehouse & Analytics Platform

APRIL 2024 – Nithin Mohan

Overview

- Introduction
- Objectives
- Functional Requirements
- Non-functional req`s
- Assumptions & Risks
- High-Level Architecture
- Solution Strategy
- Solution Benefits
- Technology Landscape
- Data Platform Recommendation
- Platform Scoring & Comparison
- Conclusion
- The Path Forward
- Appendix

Introduction

Imaginary Healthcare, a leading healthcare research organization dedicated to improving patient outcomes, is embarking on a critical initiative to leverage the power of the cloud.

This presentation outlines a strategic plan for building a modern data and analytics platform that will fuel groundbreaking research and accelerate discoveries.

The platform will address the challenges of managing and analyzing massive datasets, encompassing billions of rows of genomic data, patient records, and clinical trial results.

Executive Summary

Problem:
Our current data infrastructure struggles to handle the growing volume and complexity of genomic research data. This hinders research progress and limits our ability to gain valuable insights.

Solution:
Building a scalable and secure cloud-based data platform specifically designed for genomic research.

Value Proposition:

This platform will enable:

- Efficient processing and analysis of large datasets, both historical and real-time.
- Secure and compliant data management for patient privacy.
- Collaborative research environment for authorized researchers.
- Potential for advanced data analysis through Machine Learning integration.

Business Objectives

Building a Secure, Scalable, and Cost-Effective - Modern Data & Analytics Platform for Genomic Research

Improve Patient Care

Insights could inform personalized medicine approaches, leading to more effective treatment plans for individual patients

Accelerate Time to Market

Expedite groundbreaking discoveries in healthcare. Accelerate discovery of new treatments.

Foster Collaboraton

Break down silos between research teams, enabling collaborative analysis of large datasets across disciplines accelerating the process of discovery

Efficiency

Streamlined research workflows and data access for researchers, allowing them to focus on analysis and discovery.

Functional Requirements

- **Data Ingestion & Storage:**
  - Securely ingest and store massive datasets *(billions of rows)* encompassing structured *(patient records)* and unstructured data *(genomic data, clinical trial results)*.
- **Data Processing & Transformation**
  - Implement robust ETL pipelines for data cleansing, normalization, and handling Slowly Changing Dimensions.
- **Data Governance & Security**
  - Enforce strict access controls (RBAC) and data encryption to comply with HIPAA regulations and ensure patient privacy.
- **Data Analytics & Visualization**
  - Provide a user-friendly web-based portal with pre-built reports, dashboards, and the ability to upload custom scripts for advanced analysis.
- **Multi-tenancy & Collaboration**
  - Support multiple research teams with secure access to relevant data sets based on their roles and projects.

Non-Functional Requirements

- **Scalability:**
  - The platform must efficiently handle massive datasets and accommodate growth in data volume over time.
- **Performance:**
  - Deliver fast query response times and seamless data exploration for researchers.
- **Security:**
  - Implement robust security measures to protect sensitive patient data at rest and in transit.
- **Availability:**
  - Ensure high platform availability to minimize downtime and disruption to research activities.
- **Cost-Effectiveness:**
  - Utilize cloud resources efficiently and implement a FINOPS strategy to optimize costs.

Assumptions

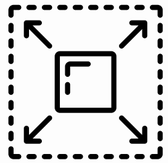
- **Cloud Provider Selection:**
  - Imaginary Healthcare has already chosen a specific cloud provider (AWS) for the platform.
  - Solution may require to suggest other best fit options and include an alternate cloud provider Azure as a secondary option to support future multi-cloud needs.
  - For deciding platform choice, vendor-lockin is also considered.
- **Connectivity:**
  - Hybrid connectivity exists between the on-premise Enterprise network and the cloud landscape to ensure the traffic happens through private network.
- **Existing Data Source:**
  - Existing data resides in a relational database management system (RDBMS) on-premises.
  - There is a need for initial load for historical data, that needs to be envisioned later phase.
- **Realtime Capabilities :**
  - There is a need for the system to handle the realtime ingestion of genomic data from various sources to provide proactive insights.
- **Technical Expertise:**
  - A team with expertise in cloud technologies, data engineering, and data security will be available for platform development and management.
- **Data Governance Framework:**
  - A foundational data governance framework may already exist within Imaginary Healthcare, which the platform can integrate with.
- **Scalability Needs:**
  - The platform's scalability requirements will be clearly defined based on projected data growth and user concurrency.
- **Artificial Intelligence and Machine Learning Support**
  - Platform should be able to support the future needs of machine learning and artificial intelligence capabilities with data.

Risks

- Data migration challenges(data quality, completeness)
- Potential for security breaches
- User adoption of the new platform
- Cost overruns due to unforeseen cloud resource utilization

Solution Objectives

Building a Secure, Scalable, and Cost-Effective - Modern Data & Analytics Platform for Genomic Research



Scalability

Scalable Data Ingestion & Storage: Securely ingest and store massive datasets (billions of rows) encompassing structured (patient records) and unstructured data (genomic data).

Efficient Data Processing & Transformation: Implement robust ETL pipelines for data cleansing, normalization, and handling Slowly Changing Dimensions (SCDs).

Privacy & Security

Data Governance & Security: Ensure the highest level of security for sensitive patient data, complying with all relevant healthcare regulations like HIPAA

Data Sharing

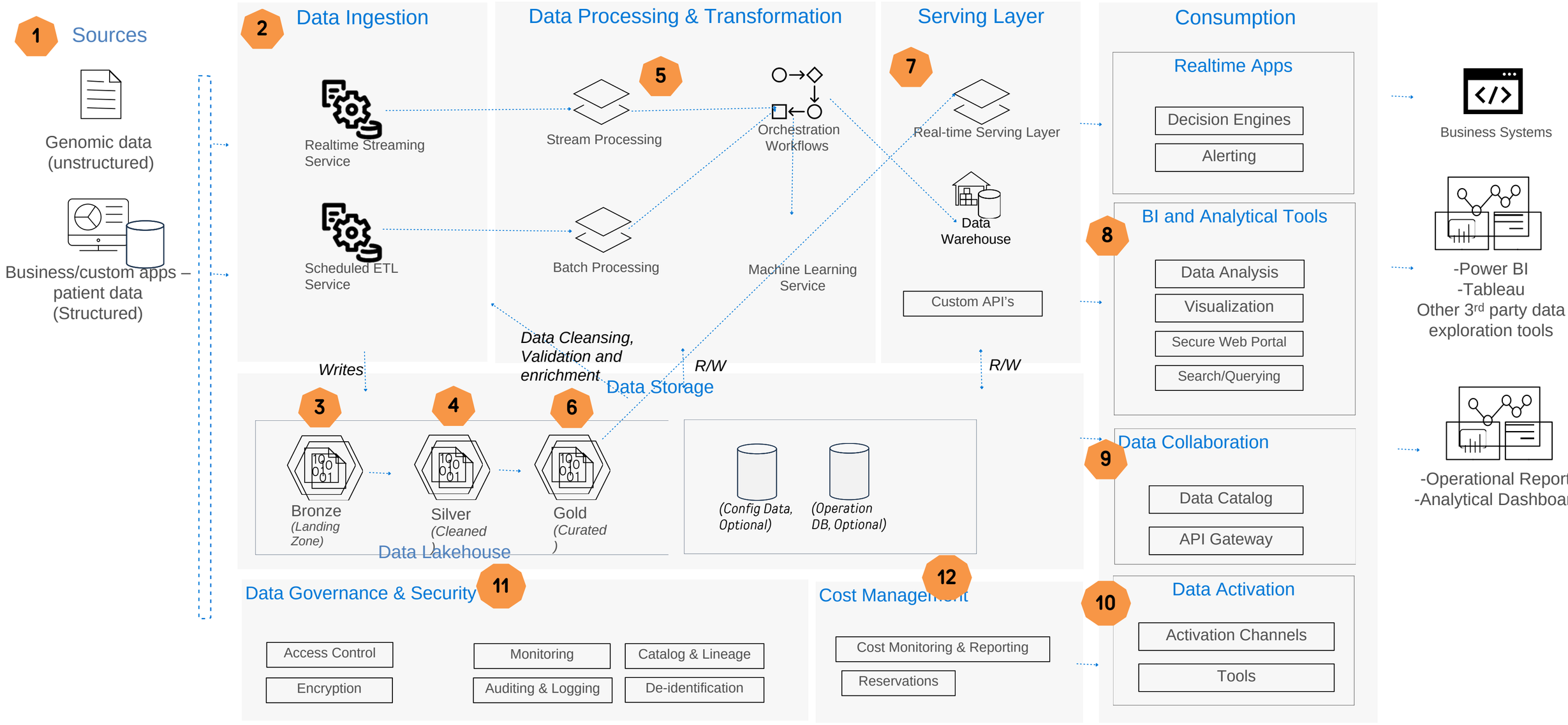
Collaboration & Sharing: Support multi-tenancy with secure access control, enabling collaboration and data sharing among research teams based on their roles and projects

Analytics

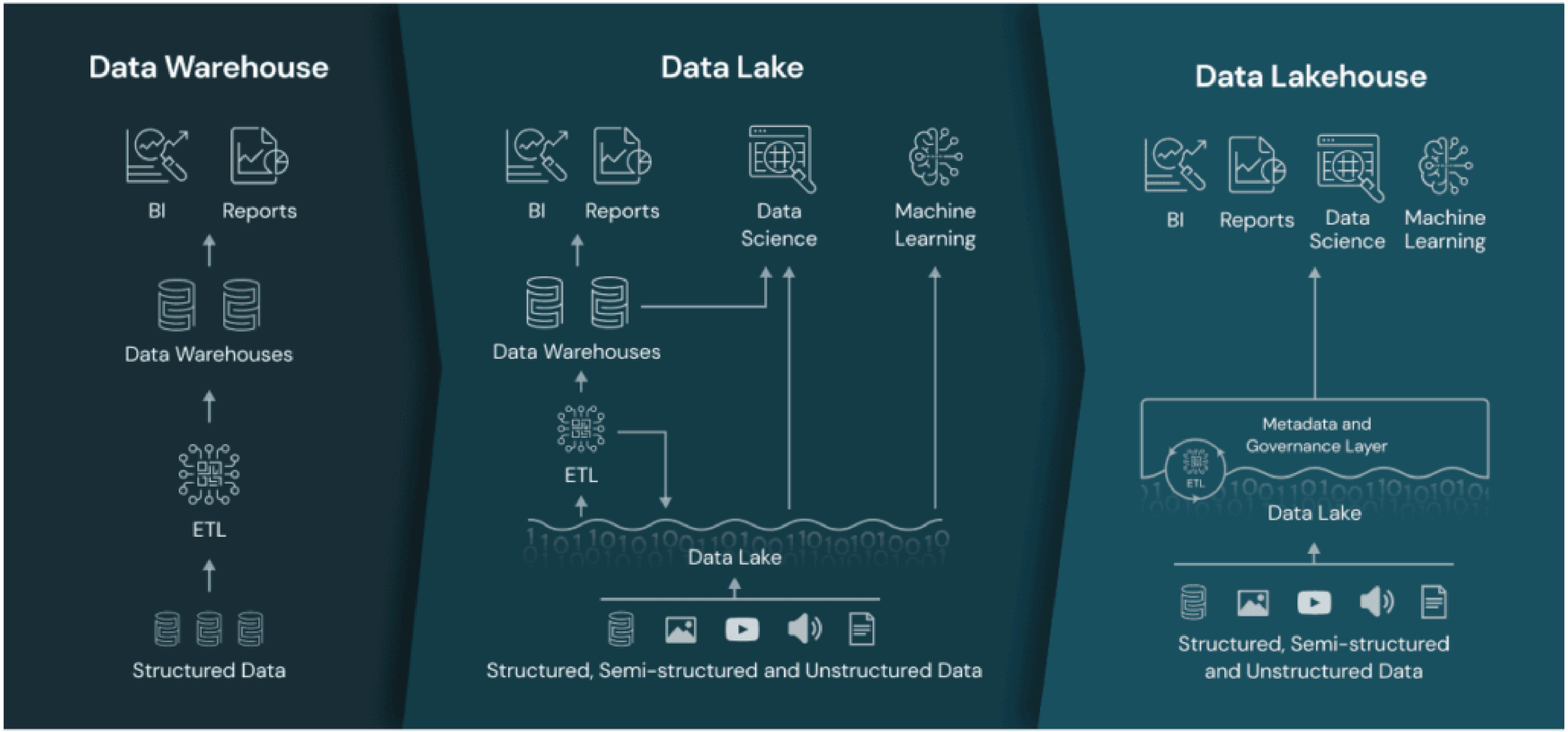
User-Friendly Analytics Portal: Develop a secure, web-based portal with an intuitive interface for researchers to access data.

Advanced Data Exploration & Analytics: Integrate BI tools (Power BI, Tableau) for pre-built reports, dashboards, and the ability to upload custom scripts for advanced analysis.

High-level Platform Architecture



Why a Lakehouse?



A visualization of the flow of data in data lakehouse architecture vs. data warehouse and data lake. Image courtesy of Databricks.

Why a Lakehouse?

“ Opt-in a Lakehouse Architecture Pattern“

- Data lakehouses reduce the complexity of managing a data lake
- ACID compliance and transactional support
- Improved updates and schema evolution
- Support for modern table formats such as iceberg, delta lake and hudi etc

“ Data lakehouses democratize access to data “

Overall, organizations that adopt a lakehouse architecture and benefit from:

- Open architecture
- Increased functionality
- Better scale
- Better economic impacts
- Fewer complexities

Solution Strategy

Sources

- Genomic Data from various systems needs to be streamed to the Data Ingestion system for realtime process.
- Utilize Change Data Capture (CDC) or a similar technology to capture changes from the RDBMS system and stream those changes to the streaming service.
- Patient data can be loaded periodically in bulk using ETL processes. This may require deploying an integration runtime in each business system or leveraging a centralized integration server located within the on-premise network.

Data Ingestion & Storage:

- Utilize a cloud object storage service (e.g., AWS S3, Azure Blob Storage, GCP Cloud Storage)) as a landing zone for raw geonomic data transfer.
- Implement a streaming service (e.g., Apache Kafka, Apache Pulsar, Google Cloud Pub/Sub, Amazon MSK, Event Hub) for real-time data ingestion.
- Utilize Delta Lake capability for reliable data storage with ACID transactions.
- Leverage a Data Lakehouse platform (e.g., Open Lakehouse, AWS Lake Formation, Azure Data Lake Storage Gen2 with Delta Lake, Google Cloud Storage with Dataproc Metastore) to store both historical and real-time data. These platforms offer the benefits of both data lakes and data warehouses. They can handle diverse data formats (structured, semi-structured, and unstructured) like a data lake, while also facilitating efficient querying through their structured data management capabilities, similar to a data warehouse.
- Data Lakehouse platforms are designed for scalability. They can automatically add storage capacity as the amount of data grows, allowing you to store billions of rows and beyond.

Solution Strategy …

Data Processing & Transformation:

- **Batch & Stream Processing:** Develop batch ETL pipelines using frameworks like Apache Spark to process historical data and load it into the data warehouse. Implement stream processing pipelines using tools like Apache Flink or Spark Streaming to continuously cleanse, transform, and store real-time data in the data lake.
- Transform data through cleansing, normalization, and SCD handling.
- Utilize orchestration tools (e.g., Airflow) for automating data pipelines.
- Utilize Apache Spark for efficient data manipulation.
- **Advanced Analytics:** Leverage Apache Spark's machine learning capabilities or integrate with specialized Machine Learning services offered by the cloud provider (e.g., AWS SageMaker, Azure Machine Learning, Google AI Platform) for advanced analytics on genomic data.

Data Warehouse & Serving Layer:

- **Data Warehouse:** Build a data warehouse (e.g., Snowflake, Databricks, Google BigQuery) for structured/semi-structured historical data, optimized for efficient querying and complex analysis of genomic research data.
- **Real-time Serving Layer:** Implement a real-time serving layer (e.g., Apache Druid, Apache HBase) for low-latency access to the most recent and frequently accessed real-time data, enabling faster insights for researchers.

Solution Strategy …

Data Security & Governance:

- Implement encryption (e.g., AES-256) for data at rest and in transit.
- Enforce Role-Based Access Control (RBAC) for user data access, ensuring only authorized researchers have access to specific datasets based on project requirements and regulations.
- Integrate with existing Data governance framework for data quality, lineage, and compliance. Or use Unity Catalog: a unified, fine-grained governance solution for data and AI.
- Utilize data lineage tracking features within the Data Lakehouse platform to track data origin and transformations, enhancing data auditability.
- Enable comprehensive logging and auditing for data activity tracking. Possibly integrate to enterprise SIEM for cybersecurity purposes.
- Implement data de-identification techniques to protect patient privacy.
- Compliance with HIPAA regulations, which is predominantly assured by the Platform and we ensure we secure all loose ends.

Data Sharing & Collaboration:

- **Multi-tenancy:** Design the platform with multi-tenancy to enable secure data sharing and collaboration among authorized researchers working on different projects.
- **Granular Access Controls:** Define granular access controls for data and functionalities based on project requirements and regulations to ensure compliance with data privacy regulations (e.g., HIPAA).
- Utilize the component like ‘Delta Sharing’ from Databricks data platform for more efficient data sharing capabilities.

Solution Strategy …

Business Intelligence & Visualization:

- Develop a web-based UI using a BI tool (e.g., Tableau, Power BI, Amazon QuickSight) for researchers to access
- Pre-built reports and dashboards for both historical and real-time data.
- Interactive data visualizations for deeper exploration.
- Self-service analytics capabilities to empower researchers to conduct ad-hoc analysis.

Cost Optimization & Management:

- Understand and implement FinOps Framework Concepts
- **Managed Services:** Leverage managed cloud services for core functionalities like data storage, compute, and analytics to reduce operational overhead and maintenance costs.
- **Autoscaling:** Implement autoscaling mechanisms for cloud resources based on usage patterns to optimize resource utilization and cost efficiency.
- **Cost Allocation & Monitoring:** Monitor cloud resource utilization and implement cost allocation models (e.g., ref: FINOPS framework) to track spending by project or research group for better cost visibility and control.
- Utilize Serverless capabilities wherever possible, there by not committing to dedicated instances.
- Estimate the Growth: in months, 1 Yr. 2 Yr, 5 year. Calculate the TCO

Solution Strategy …

Advanced Data Analysis with Machine Learning:

- The platform can be further extended to integrate with Machine Learning (ML) for advanced data analysis.
- This might involve incorporating tools and libraries like TensorFlow or PyTorch to develop ML models for tasks such as:
  - **Variant discovery:** Identifying genetic variations associated with diseases or traits.
  - **Gene expression analysis:** Understanding how genes are expressed under different conditions.
  - **Drug discovery:** Identifying potential drug targets or predicting drug response.
- A feature store (e.g., Feast, Hopsworks) can be implemented to centrally manage, version, and serve features for training and deploying ML models.
- By integrating ML, researchers can gain deeper insights from the genomic data and accelerate scientific discovery.

Data Storage . extended

Data Storage Techniques:

- **Partitioning:** Data is divided into smaller, manageable partitions based on specific criteria (e.g., date, patient ID). This allows for faster querying and retrieval of specific datasets.
- **Columnar Storage:** Data is stored by column instead of by row, which is more efficient for analytical workloads where you typically query specific columns.
- **Data Compression:** Data can be compressed using various techniques to reduce storage footprint without compromising information integrity.
- Utilize ***avro****(row)* for raw data, as it provides faster writes for stream to cloud object storage and ***parquet****(columnar)* for clean(silver) & curated(gold) data as it is efficient for querying(faster read).

  Cloud Storage Services:
- Leverage cloud object storage services offered by major cloud providers (e.g., AWS S3, Azure Blob Storage, Google Cloud Storage). These services are highly scalable and cost-effective, ideal for storing large datasets.

  Data Lifecycle Management:
- Implement data lifecycle management policies to automatically archive or delete less frequently accessed data. This optimizes storage costs and keeps the active data lakehouse performant.

  Handling Increased Demand:
- **Autoscaling:** Utilize autoscaling features offered by cloud providers. This allows the system to automatically adjust resources (compute, storage) based on real-time demand, ensuring efficient resource utilization.
- **Horizontal Scaling:** The system can be horizontally scaled by adding more nodes to the data lakehouse cluster, increasing processing power to handle larger data volumes and complex queries.

  Benefits:
- **Cost-Effectiveness:** Cloud storage and autoscaling features optimize costs by paying only for the resources you use.
- **Performance:** Data partitioning and columnar storage enable efficient querying of large datasets.
- **Flexibility:** The Data Lakehouse architecture accommodates diverse data formats and evolving data needs.

Solution Benefits

Improved Scalability

- Handles large and growing volumes of complex genomic datasets efficiently.

Real-time Insights

- Enables real-time processing and analysis of data streams for faster decision-making.

Secure and Compliant

- Ensures patient privacy compliance through robust data security practices.

Collaborative Research Environment

- Fosters collaboration and knowledge sharing among researchers.

Cost-Effective Platform

- Optimizes costs with managed services, autoscaling, and efficient resource utilization.

Technology Landscape

Data Ingestion	• Landing Zone - Cloud Storage (AWS S3, Azure Blob Storage, GCP Cloud Storage) • Streaming Service - Kafka, Apache Pulsar, Amazon MSK, Amazon Kinesis Streams, Google Pub/Sub	Cost-effective, scalable cloud storage (for raw data) with a popular streaming service for real-time data. Integrates well with other AWS services if using AWS cloud.
Data Storage	• Data Lake (e.g., AWS Glue Data Lake, Azure Data Lake Storage) • Lakehouse (e.g., Delta Lake on cloud storage, Databricks Lakehouse) for Transactional Tables	A data lake offers flexibility for storing various data formats. A lakehouse adds structure and transactional consistency for data warehousing capabilities. Choose based on data governance and future analytics needs.
Data Processing	• Apache Spark, Airflow/Luigi/Databricks Jobs (ETL Pipelines), Glue (AWS), Data Factory(Azure), Data Fusion(GC)	Spark offers powerful distributed processing for large datasets. Airflow/Luigi are popular tools for building and managing ETL pipelines.
Data Warehouse	* Snowflake, Databricks, Redshift (AWS), Synapse(Azure). BigQuery(GC)	Both offer scalable and secure cloud-based data warehousing solutions. Choice might depend on cost considerations, ease of use, and desired functionalities.
Real-time Serving Layer	• Spark Structured Streaming, Apache Druid, Hbase, Amazon Kinesis Data Streams or Kinesis Firehose, Cloud Dataflow(GC), Azure Stream Analytics	Both provide low-latency access to frequently accessed real-time data. Choice might depend on data volume and query patterns.
Real-time Machine Learning	• TensorFlow, PyTorch (Model Development), • Feature Store (e.g., Feast, Hopsworks) or Model Registry • Databricks ML, Amazon SageMaker, Google AI, Azure Machine Learning	These libraries enable development of Machine Learning models for advanced data analysis. A feature store provides centralized management of features used for training and deploying ML models.
Data Security & Governance	• Security: Encryption (e.g., AES-256), IAM roles, Cloud Logging, • Governance: Data Catalog, Data Lineage, Metadata Tagging, AWS Lake formation, Databricks UnityCatalog, Azure Purview	control. Cloud logging for activity tracking and audit purposes. Data Governance capabilities including data classification, tagging, metadata management, data lineage, traceability, data retention and archival

Data Platform Recommendation

Option 1: AWS Native Stack

- AWS offers more modular approach providing technologies to stich between various needs for the platform. However, this makes it a cloud centric approach.
- Within AWS, there is a clear distinction between the data lake components (S3 for storage, Glue for metadata, Lake Formation for security) and the data warehouse components (Redshift with Spectrum for lake integration).

Option 2: AWS + Databricks

- The Databricks platform offers a lake-centric approach to the data lakehouse. Databricks provisioned into AWS offers the best combo of modern data warehouse and analytics platform.
- Most of our transformations and other heavy lifting is happening within the data lake itself.
- Doesn't include a persistent, physical data warehouse like in AWS Redshift. Instead, Databricks platform components (Delta Lake, Unity Catalog, Databricks SQL, and more) meet the core data capabilities required to lend structure and reliability to the lake.
- The Databricks platform enables ACID and transactional support, metadata store and history, auto-scaling compute engine, governance and security model, and many more necessary features to impose order on unruly data lakes.

Note: Considering the current enterprise cloud landscape, limiting the options to AWS cloud and concluding the recommendation options with 2 choices.

Approach #1: AWS Native Stack

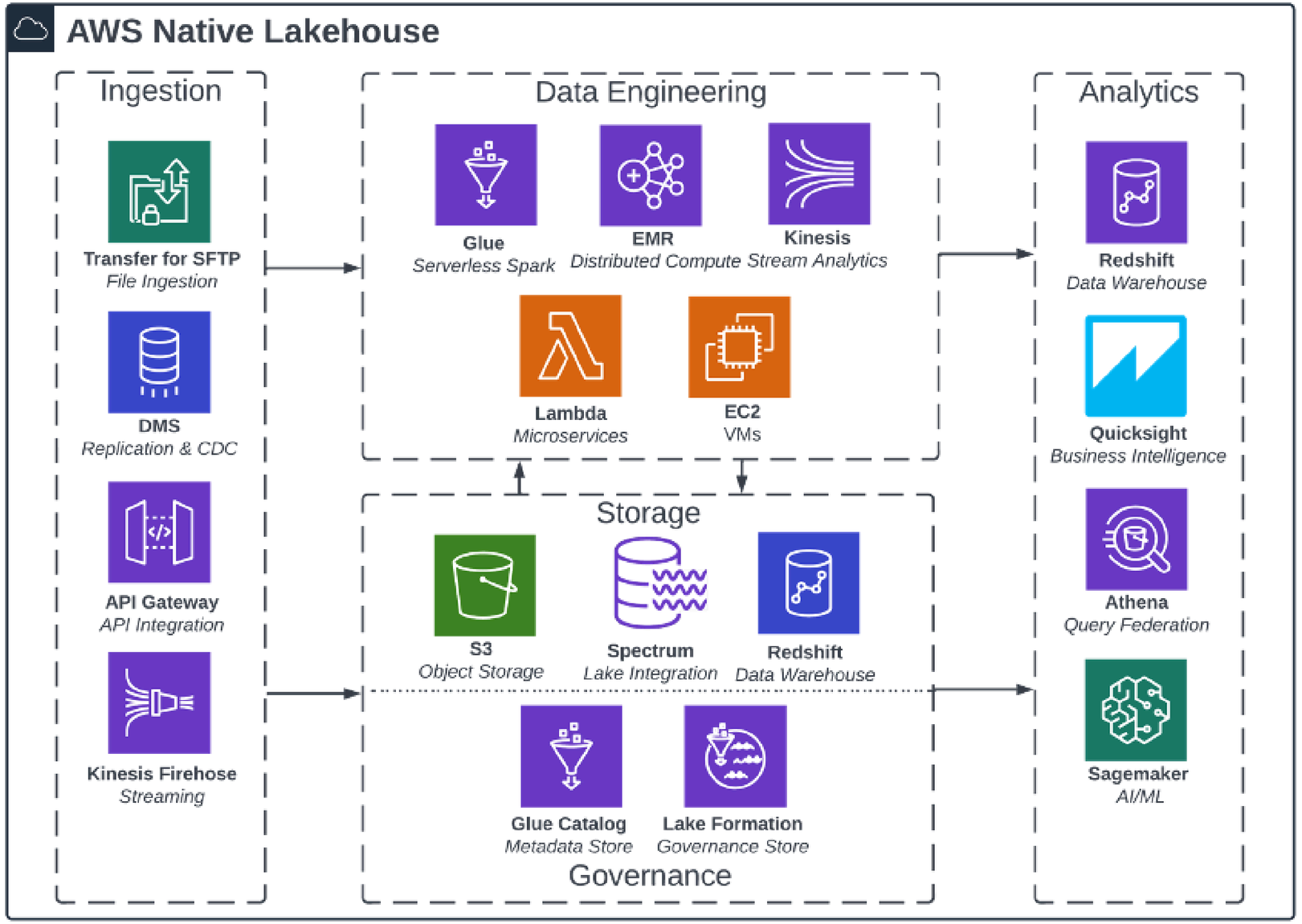


Diagram: AWS Native Lakehouse capability using various pluggable PaaS and IaaS services available in AWS.

Approach #2: AWS + Databricks

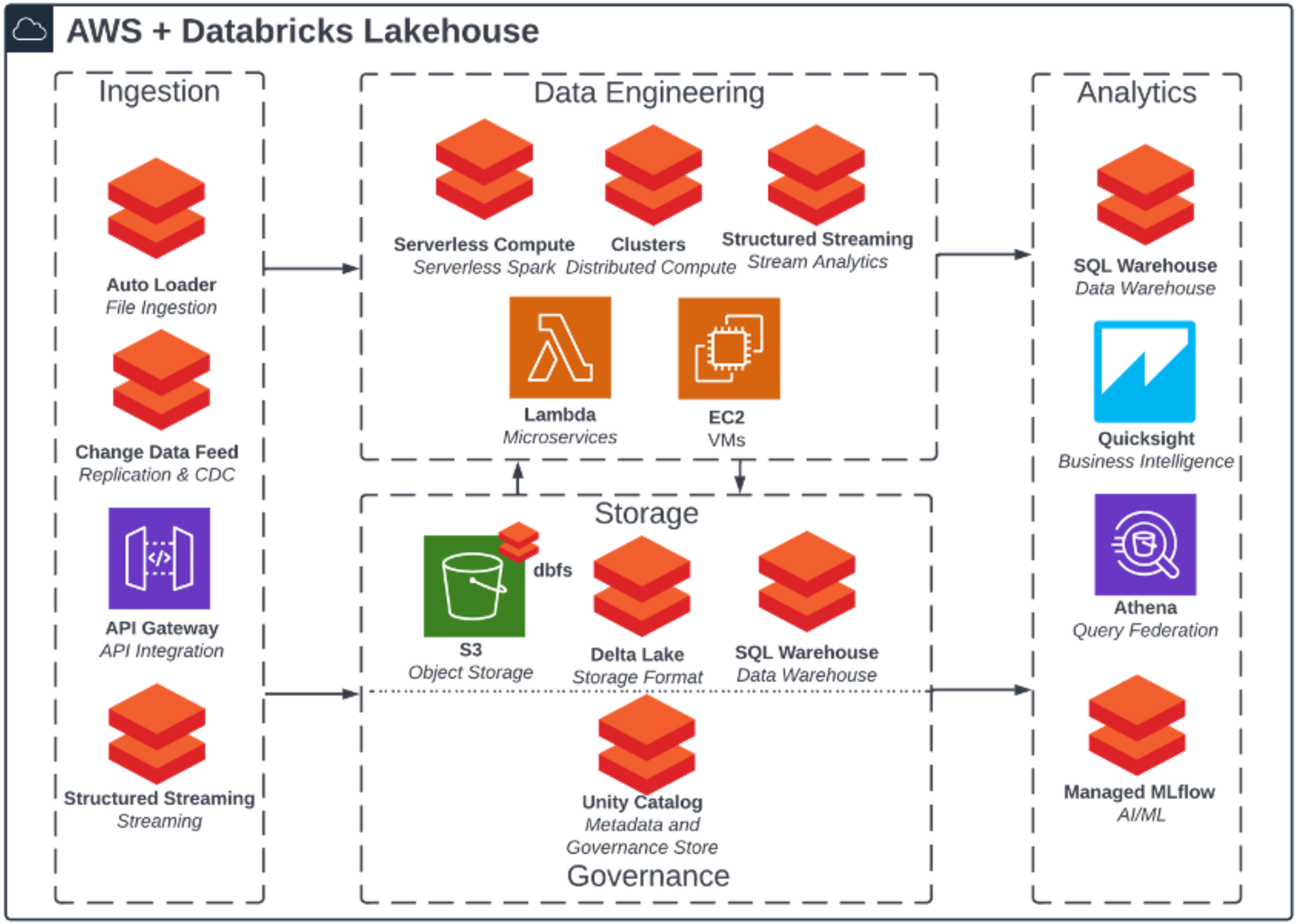


Diagram: A Mix AWS Cloud services and Modern Databricks Data Platform offerings, which gives us the best of the breeds.

Data Platform Recommendation …

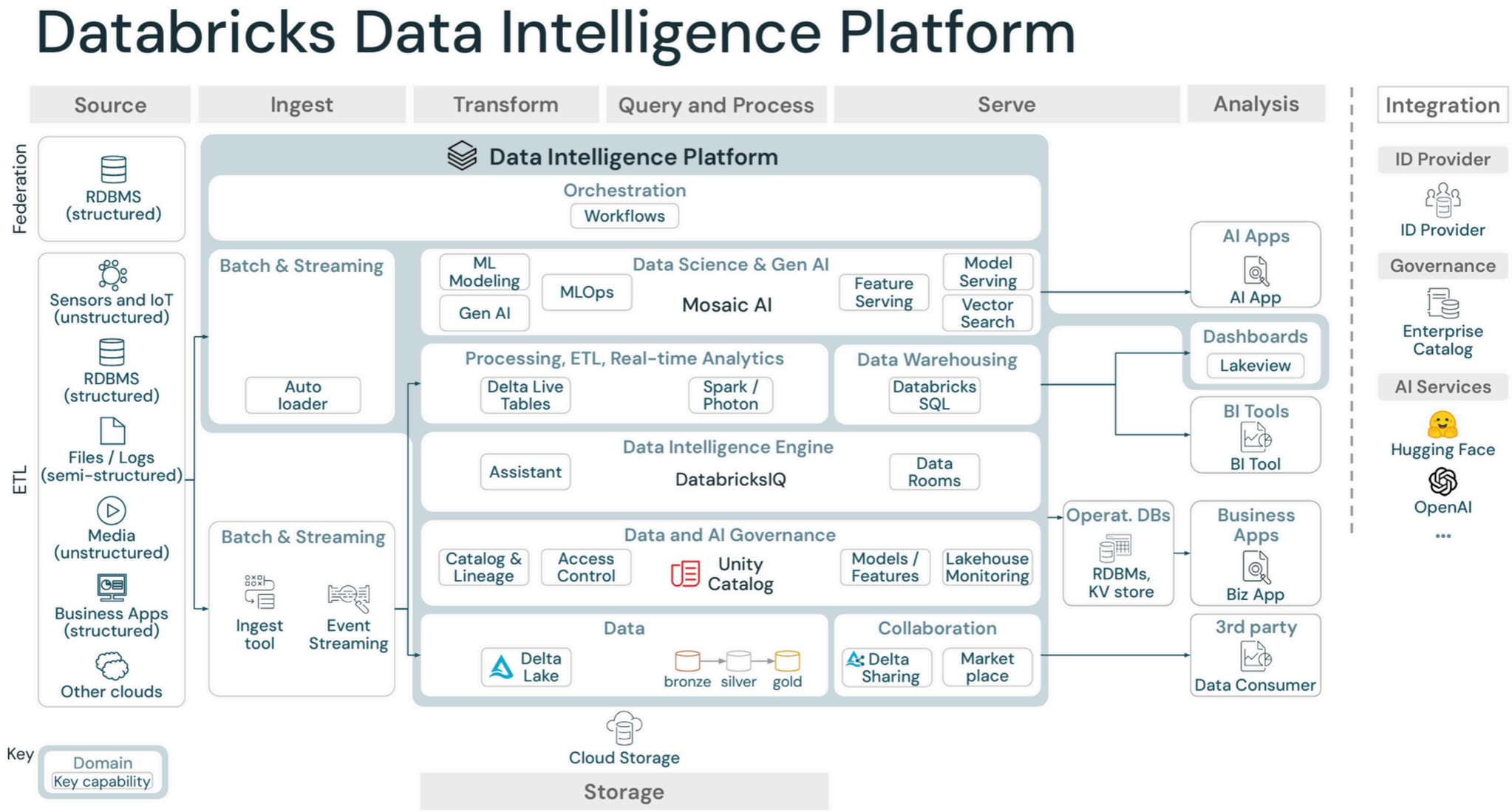
Based on our analysis, Databricks Lakehouse Platform emerges as a compelling choice as our Enterprise Data and Analytics platform for Imaginary Healthcare due to the following strengths:

- **Simplicity and Automation:** Delta Lake's built-in versioning provides automatic SCD Type 2 support, streamlining data management.
- **Advanced Analytics:** Leverages Apache Spark for powerful data processing and transformation capabilities, ideal for complex data workloads.
- **Open Lakehouse Architecture:** Supports various data formats and integrates with existing tools and ecosystems, offering flexibility for your data architecture.
- **Cloud Agnostic:** Operates across multiple cloud providers, providing freedom from vendor lock-in.

Databricks Lakehouse Platform empowers you to:

- **Simplify SCD Management:** Automate historical data tracking with Delta Lake versioning, reducing development effort.
- **Unlock Deeper Insights:** Leverage Apache Spark for complex data engineering and advanced analytics.
- **Build a Unified Data Platform:** Consolidate data warehouse and data lake into a unified platform, fostering better data collaboration.
- **Maintain Data Freedom:** Avoid vendor lock-in with a cloud-agnostic platform that integrates with your existing infrastructure.

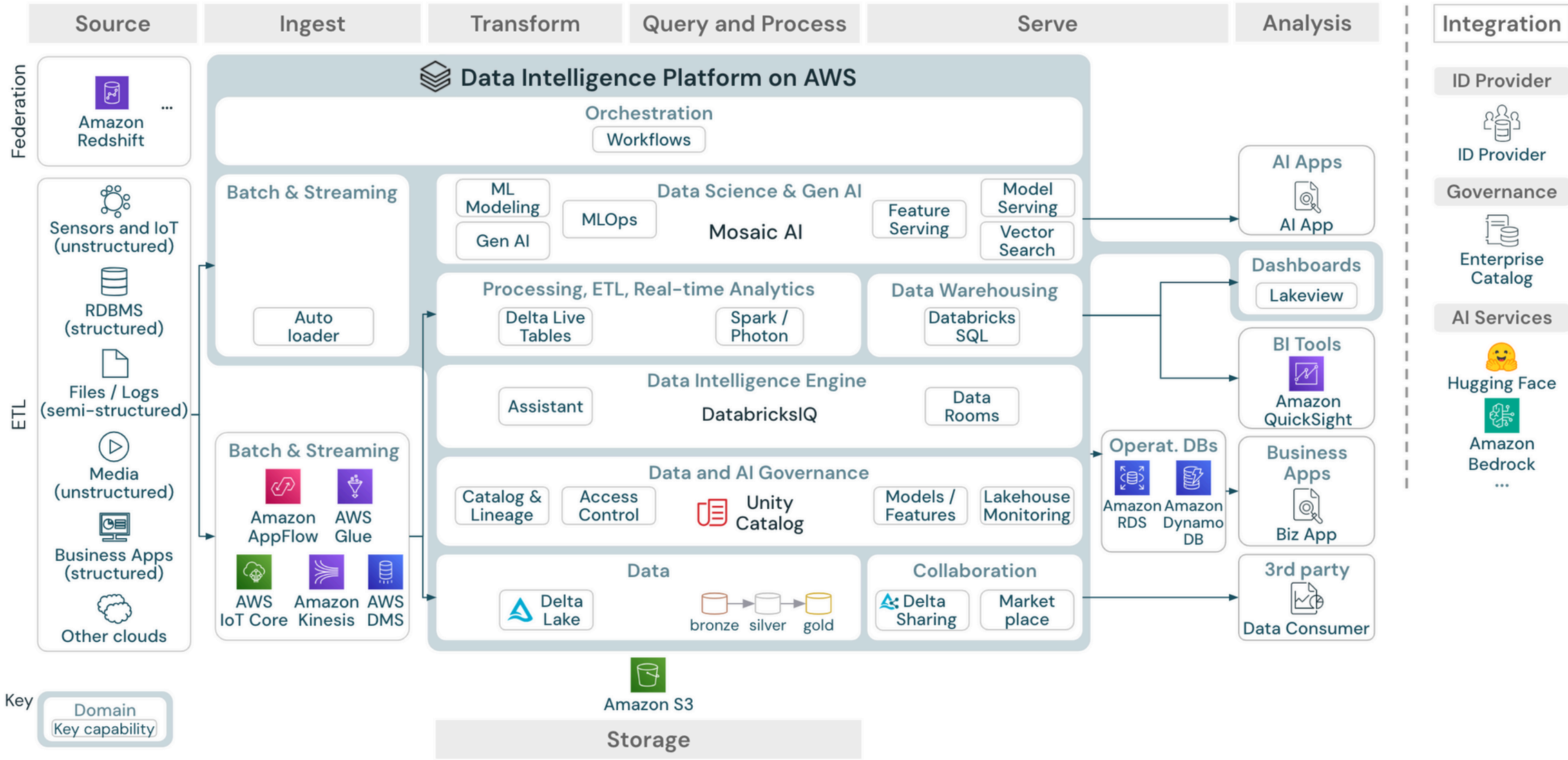
Data Platform: ONE for all needs



Note: Diagram is a reference to indicate one platform can meet all the needs of the solution.

https://docs.databricks.com/en/lakehouse-architecture/reference

Data Platform: AWS + Databricks



Note: Diagram is a reference to indicate data bricks deployed on AWS cloud.

https://docs.databricks.com/en/lakehouse-architecture/reference

Data Platform Comparison ..

Feature	Weighting	Databricks Lakhouse	Snowflake Data Cloud	AWS (Redshift & Glue)	Azure Synapse Analytics	Google BigQuery	Score (Weighted)
Lakehouse Support	High (3)	5 (Delta Lake)	5 (Delta Lake)	4 (via S3)	4 (via ADLS)	0	15 (3)
SCD Support	Medium (2)	5 (Delta Lake versioning)	5 (Advanced with Data Vault and Timetravel)	3 (custom logic)	4 (native Type 1 & 2)	3 (custom logic with Databricks)	8 (2)
Data Warehousing	High (3)	5 (integrated with Delta Lake)	5	4 (Redshift)	5	5	15 (3)
Data Processing & Transformation	High (3)	5 (advanced with Apache Spark)	5	4 (Glue)	4 (improved features)	4 (various services)	15 (3)
Cloud Agnosticism	Low (1)	4	3 (Limited Partnering with major cloud providers for data storage)	0	0	0	4 (1)
Realtime Capabilities	Medium (2)	4 (Delta Live Tables)	4 (Advanced with Data Cloud)	3 (Kinesis)	3 (Stream Analytics)	3 (Pub/Sub)	8 (2)
Scalability	High (3)	5	5	5	5	5	15 (3)
Security	High (3)	5	5	5	5	5	15 (3)
Ease of Use & Management	Medium (2)	4	4 (Complexity increases with Data Cloud features)	3 (Glue can be complex)	4	4	6 (2)
Cost-Effectiveness	Medium (2)	4 (pay-per-use)	4 (pay-per-use, can be expensive for large workloads or varies based on chosen services)	3 (Redshift cost-effective, Glue additional cost)	4 (pay-per-use)	4 (pay-per-use)	8 (2)
Total Score (Provider)		**113**	**112**	**95**	**99**	**97**	

Scoring is based on a 1-5 scale, with 5 being the best.

Note: Rough scoring based on private research and understanding via public domain documentation. This may be inaccurate.

Conclusion

This comprehensive solution strategy outlines a robust, secure, and scalable cloud-based platform specifically designed to empower Imaginary Healthcare’s genomic research endeavors.

Why This Approach?

- **Unified Data Management:** The Data Lakehouse architecture seamlessly handles all your genomic data, structured, semi-structured, and unstructured, eliminating data silos.
- **Real-time Insights:** Gain valuable insights from ongoing experiments and sensor data with real-time processing capabilities.
- **Advanced Analytics & Machine Learning:** Uncover hidden patterns and relationships within your data using built-in or integrated Machine Learning tools, driving deeper research breakthroughs.
- **Secure Collaboration:** Robust security measures ensure patient privacy compliance, while multi-tenancy fosters secure data sharing and collaboration among researchers.
- **Scalability & Cost-Effectiveness:** Managed services, autoscaling, and efficient resource utilization optimize costs, allowing the platform to adapt to your growing research needs.

By implementing this solution, Imaginary Healthcare’s can:

- Accelerate research progress through efficient data management and advanced analytics.
- Drive innovation in genomics by enabling collaborative research efforts.
- Make data-driven decisions to improve healthcare outcomes.

The Path Forward

Refine the architecture based on the chosen cloud platform and specific requirements.

Develop detailed designs for data pipelines, data governance framework, and BI functionalities.

Implement security best practices and conduct penetration testing to ensure platform security.

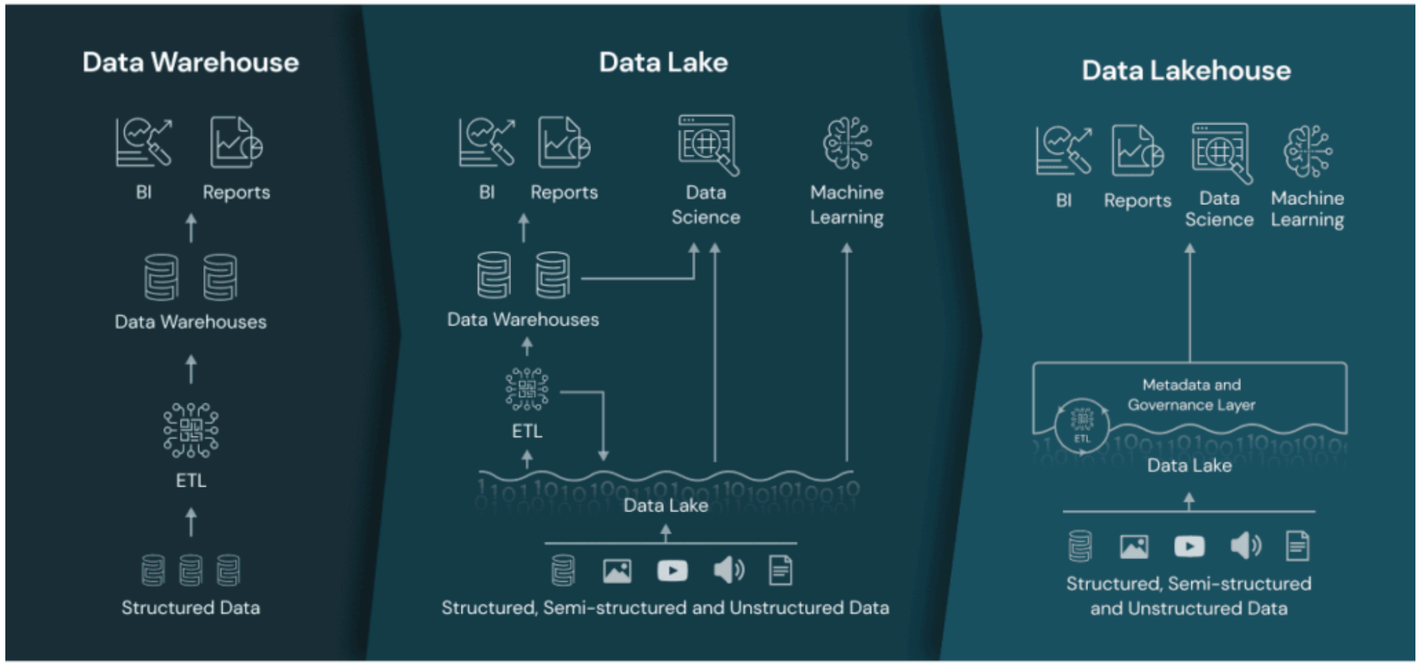
Pilot the platform for user testing and feedback

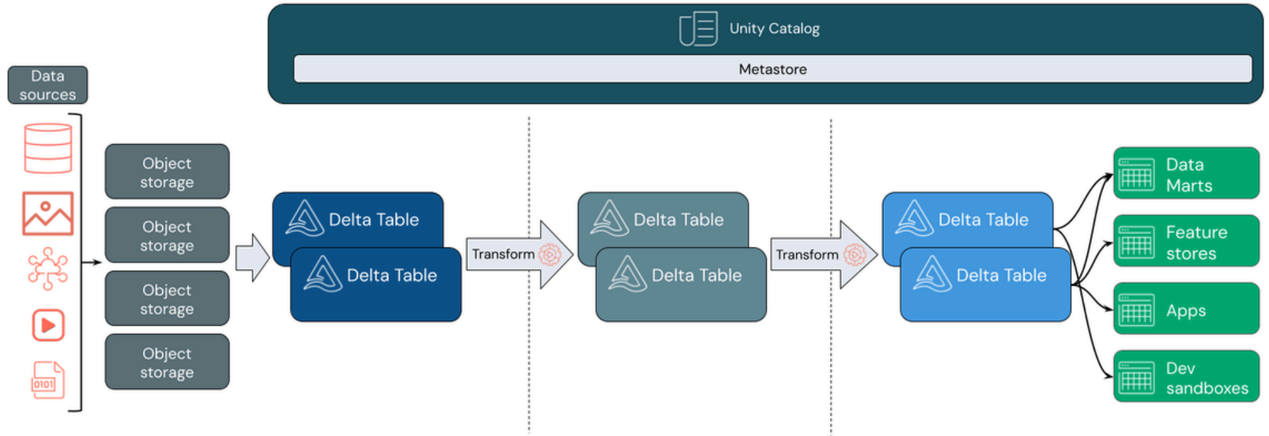
THANK YOU!

Appendix: Supporting Slides

i. Slowly Changing Dimensions(Info & Comparison)
ii. Benefits of using AWS
iii. Snowflake as a Data Warehouse (Mixed mode)
iv. Databricks as a Data Warehouse

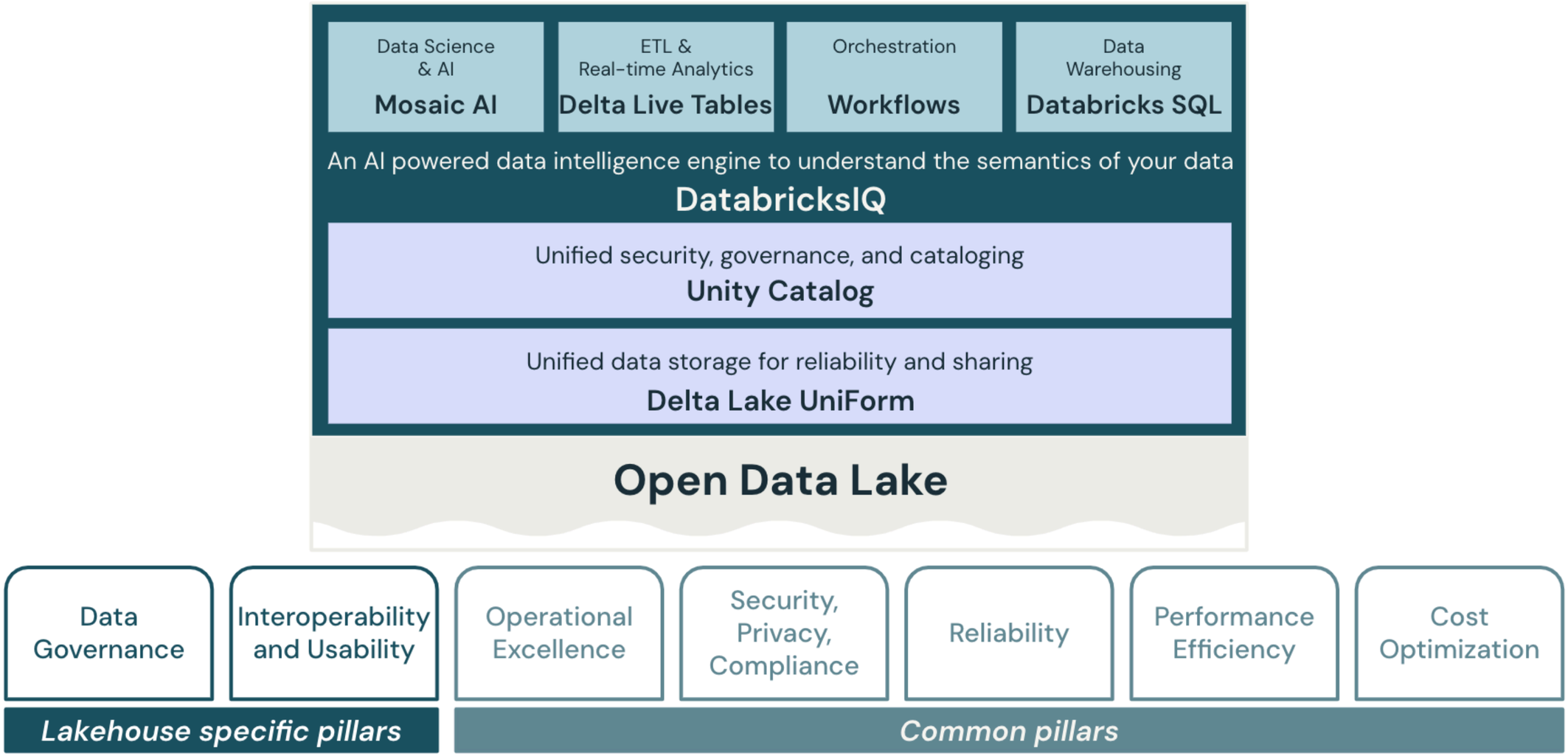
Why a Lakehouse?





A visualization of the flow of data in data lakehouse architecture vs. data warehouse and data lake. Image courtesy of Databricks

Databricks Well-architected framework



A visualization of the flow of data in data lakehouse architecture vs. data warehouse and data lake. Image courtesy of Databricks

Why a Lakehouse?

“ Data lakehouses democratize access to data “

Data lakehouses make the data lake more accessible to different people in the organization who would not otherwise have benefitted and might have been forced to use a costly data warehouse instead. Data is no longer gated and suitable only for data engineers.

At the same time, this shift comes without sacrificing the original use-cases that made data lakes popular in the first place.

Overall, organizations that adopt a lakehouse architecture and benefit from:

- Open architecture
- Increased functionality
- Better scale
- Better economic impacts
- Fewer complexities

Databricks Lakehouse - Capabilities

- **Real-time data processing:** Process streaming data in real-time for immediate analysis and action.
- **Data integration:** Unify your data in a single system to enable collaboration and establish a single source of truth for your organization.
- **Schema evolution:** Modify data schema over time to adapt to changing business needs without disrupting existing data pipelines.
- **Data transformations:** Using Apache Spark and Delta Lake brings speed, scalability, and reliability to your data.
- **Data analysis and reporting:** Run complex analytical queries with an engine optimized for data warehousing workloads.
- **Machine learning and AI:** Apply advanced analytics techniques to all of your data. Use ML to enrich your data and support other workloads.
- **Data versioning and lineage:** Maintain version history for datasets and track lineage to ensure data provenance and traceability.
- **Data governance:** Use a single, unified system to control access to your data and perform audits.
- **Data sharing:** Facilitate collaboration by allowing the sharing of curated data sets, reports, and insights across teams.
- **Operational analytics:** Monitor data quality metrics, model quality metrics, and drift by applying machine learning to lakehouse monitoring data.

Slowly Changing Dimensions (SCD)

General Considerations for SCD Implementation:

- **Choosing the Right SCD Type:** Select the appropriate SCD technique based on your specific data governance requirements and the need to maintain historical data. *Type 1 (overwrite), Type 2 (new row for changes), and Type 3 (current & historical values).*
- **Data Quality & Cleansing:** Ensure proper data quality checks and cleansing routines are in place before updating dimension tables to maintain data integrity.
- **Data Lineage & Auditing:** Implement data lineage tracking and auditing practices to understand how dimension data changes over time, especially important for regulatory compliance.

Choosing the Best Platform for SCD Support:

All three platforms offer robust capabilities for managing SCDs. The choice might depend on other factors:

- **Existing Infrastructure:** If you're already invested in the AWS ecosystem, leveraging AWS Glue and Redshift might be a natural fit.
- **Team Expertise:** Consider engineering team's experience with Apache Spark (Databricks) or SQL (AWS, Snowflake) for SCD implementation.
- **Desired Functionality:** Databricks offers advanced features like Delta Lake versioning for specific SCD needs.

Slowly Changing Dimensions (SCD)

Feature	Databricks Lakehouse Platform	Snowflake Data Warehouse	Snowflake Data Cloud (Integrated Services)	AWS (Redshift)	Azure Synapse Analytics	Google BigQuery
Native SCD Support	Yes (Delta Lake versioning)	Limited (requires manual updates)	Advanced (Data Vault & Time Travel)	No (requires custom logic)	Native Type 1 & 2	No (requires custom logic with Databricks)
SCD Techniques Supported	- Type 2 (Automatic with Delta Lake versioning)	- Type 1 & 2 (Manual)	- Type 1 & 2 (Data Vault), Type 2 (Time Travel)	- Type 1 & 2 (Custom logic)	- Type 1 & 2 (Native)	- Type 1 & 2 (Custom logic with Databricks)
Versioning	Built-in with Delta Lake	Not supported	Built-in with Time Travel	Not supported	Not supported	Not supported
Historical Data Management	Automatic with Delta Lake versioning	Requires manual updates or additional tools	Automatic with Time Travel	Requires manual ETL process	Requires custom logic or PolyBase for historical data access	Requires custom logic with Databricks
Data Lineage	Supported through Delta Lake and Apache Spark	Limited	Enhanced with Data Cloud features	Requires integration with AWS Glue or other tools	Supported	Supported through BigQuery Data Catalog
Ease of Use	Straightforward with Delta Lake	Requires additional development effort	More complex with additional features	Requires custom development	Relatively easy with native support	Requires custom development with Databricks

Benefits of using AWS

- **Scalability**
  - AWS offers on-demand resources to handle massive datasets and growing user demands.
- **Security**
  - AWS provides robust security features and compliance certifications to ensure data privacy and regulatory adherence.
- **Managed Services**
  - Leveraging managed services reduces development overhead and simplifies platform management.
- **Cost-Effectiveness**
  - AWS offers a variety of pricing models and optimization strategies to control cloud costs.

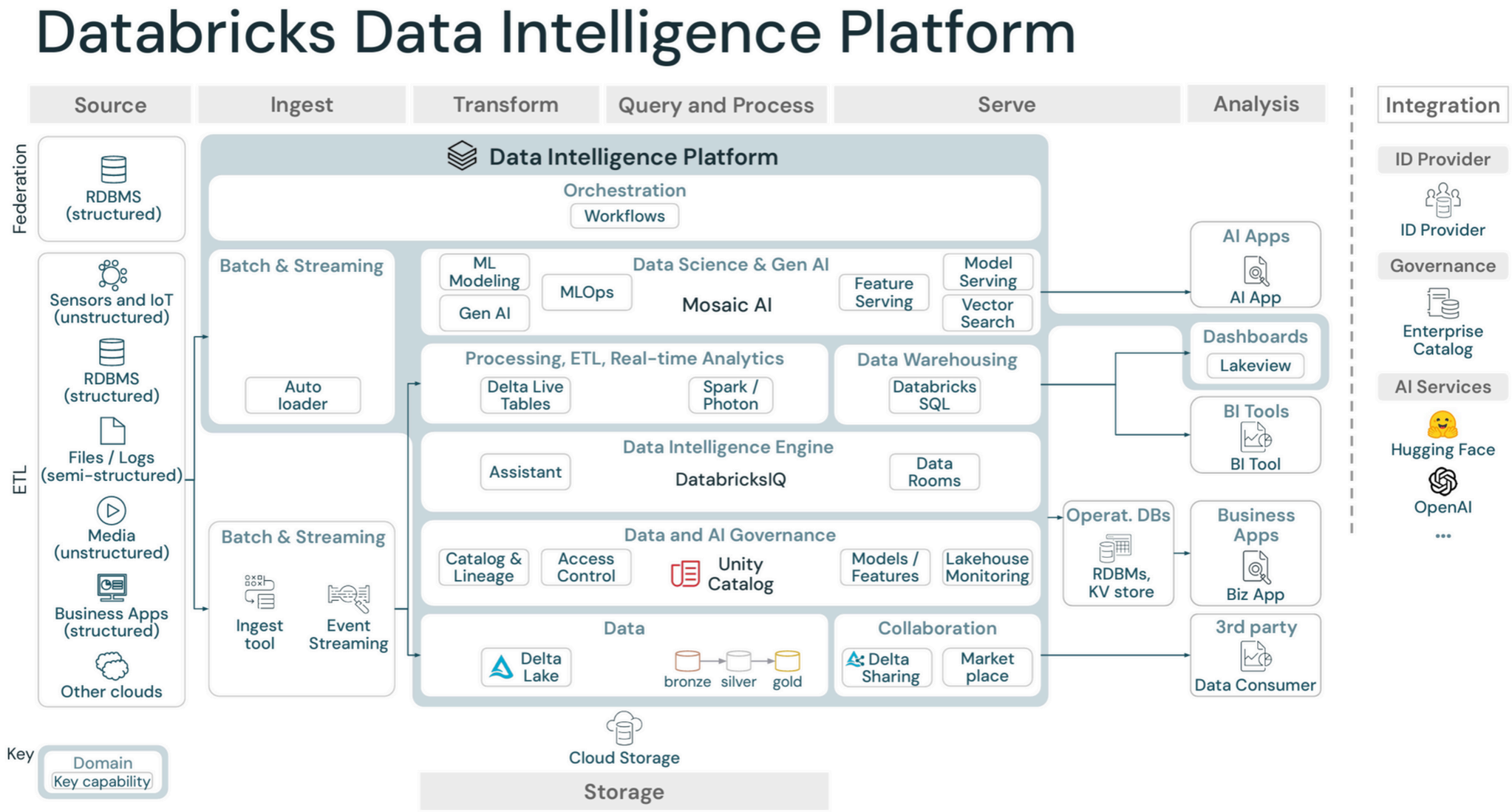
Snowflake as a Data Warehouse

- **Functionalities**
  - Snowflake is a cloud-native data warehouse offering similar functionalities to Redshift, including scalability, performance, and SQL compatibility.
  - It eliminates the need for infrastructure management and offers a pay-per-use model.
- **Integration with AWS**
  - Snowflake integrates seamlessly with AWS services for data ingestion, processing, and governance
- **Data Ingestion:**
  - Utilize AWS services like AWS S3 to stage data before loading it into Snowflake using its Snowpipe feature for efficient data movement.
- **Data Processing & Transformation**
  - AWS Glue ETL jobs can be configured to extract data from on-premises sources and transform it for loading into Snowflake.
- **Data Governance**
  - AWS services like AWS Lake Formation can be integrated with Snowflake's native security features (user roles, object ownership) for comprehensive data governance.

Databricks as a Data Warehouse

- **Functionalities**
  - Databricks is a cloud-native data warehouse offering similar functionalities to Redshift & Snowflake, including scalability, performance, and SQL compatibility.
  - It eliminates the need for infrastructure management and offers a pay-per-use model.
  - Leverages open-source technologies like Spark and Delta Lake, promoting flexibility and avoiding vendor lock-in.
  - Databricks goes beyond data warehousing by offering a comprehensive environment for data engineering, data science, and collaborative data analytics, all within a single platform.
- **Data Ingestion**
  - Read data directly from AWS S3 or Azure Data Lake Storage using Spark APIs within Databricks notebooks for data processing pipelines.
- **Data Processing**
  - Utilize the **Unit Catalog** to discover and manage metadata stored in Cloud Object Storage/Data Lake, simplifying data access within Databricks notebooks.
  - Alternatively utilize AWS Glue Data Catalog or Azure Purview
- **Data Governance**
  - Implement AWS IAM roles to control access to Databricks clusters and resources.
  - Leverage AWS KMS to manage encryption keys for data at rest within Databricks.

Databricks Data Platform



FinOps Framework

Principles

- Teams need to collaborate
- Decisions are driven by business value of cloud
- Everyone takes ownership for their cloud usage
- FinOps data should be accessible and timely
- A centralized team drives FinOps
- Take advantage of the variable cost model of the cloud

Domains & Capabilities

FinOps Foundation

Understand Cloud Usage & Cost

- Data Ingestion
- Allocation
- Reporting & Analytics
- Anomaly Management

Quantify Business Value

- Planning & Estimating
- Forecasting
- Budgeting
- Benchmarking
- Unit Economics

Optimize Cloud Usage & Cost

- Architecting for Cloud
- Rate Optimization
- Workload Optimization
- Cloud Sustainability
- Licensing & SaaS

Manage the FinOps Practice

- FinOps Practice Operations
- Cloud Policy & Governance
- FinOps Assessment
- FinOps Tools & Services
- FinOps Education & Enablement
- Invoicing & Chargeback
- Onboarding Workloads
- Intersecting Disciplines

Core Personas

- Engineering
- FinOps Practitioner
- Finance
- Leadership
- Procurement
- Product

Allied Personas

- ITAM
- ITFM
- ITSM
- Security
- Sustainability

Phases

Maturity

- Crawl
- Walk
- Run

Datawarehouse in Azure

Landing

Azure Data Lake Gen2
(Azure Storage)

Transform

Synapse (Spark)

OR

Synapse Pipelines
(Data Flows)

OR

Synapse
SQL Dedicated Pool

OR

Azure Databricks (Spark)

Serve

Synapse
SQL Dedicated Pool

Orchestrate

Synapse Pipelines

Datalake in Azure

Landing

Azure Data Lake Gen2
(Azure Storage)

Transform

Synapse (Spark)

OR

Synapse Pipelines
(Data Flows)

OR

Synapse SQL
Serverless

OR

Azure Databricks (Spark)

Serving Storage

Azure Data Lake Gen2
(Azure Storage)

Serving Compute

Synapse (Spark)

OR

Synapse SQL
Serverless

OR

Synapse – SQL Dedicated
(External Tables)

OR

Azure Databricks (Spark)

Orchestrate

Synapse Pipelines

Real-Time Big Data Analytics Architecture