Data Management & Engineering: enabling AI/ML initiatives

Data Management in the Modern Landscape

In the digital era, the exponential growth of data is transforming how organizations operate. Data management has evolved from simple database administration to an ecosystem of intelligent, automated, and scalable platforms that handle massive, heterogeneous datasets.

Cloud-based data management has become the cornerstone of modern digital transformation. Organizations now rely on distributed data platforms that can ingest, process, store, and visualize data efficiently across hybrid and multi-cloud environments. The shift to cloud has not only democratized access to data but has also enabled real-time analytics, self-service BI, and AI-driven insights.

The modern paradigm is characterized by three core transformations:

a) **Cloud-Native Architecture:** Reliance on elastic, pay-as-you-go cloud services (AWS, Azure, GCP) that decouple storage (Data Lakes like S3, ADLS) from compute (Databricks, Snowflake, BigQuery). This shift enables infinite scalability and flexible cost structures, demanding a FinOps approach to governance.

b) **The Rise of the Data Lakehouse:** The traditional separation between structured data warehouses (for BI) and unstructured data lakes (for ML) has converged into the Data Lakehouse architecture. This hybrid model leverages the scalability of data lakes while introducing the transactional consistency and governance features (ACID properties, schema enforcement) previously exclusive to data warehouses. This architecture is central to unifying AI and BI workloads.

c) **Data Decentralization and Ownership:** Architectures like **Data Mesh** have emerged, challenging the central data platform. Data Mesh advocates treating data as a product, decentralizing ownership to business domains, which are responsible for the quality, governance, and availability of their own data sets.

In 2025, the primary driver for robust data management is **AI readiness**. Organizations must ensure their data is clean, labelled, observable, and rapidly accessible to train and deploy Generative AI and Machine Learning models. Poor data quality is the single greatest bottleneck to successful AI adoption. Therefore, modern DM is inseparable from data quality, governance, and real-time processing capabilities.

Data Management Lifecycle: Steps and Challenges

The modern data engineering lifecycle is often defined by four key stages, each presenting unique technical and organizational challenges.

Ingestion: Data Acquisition and Loading

This stage involves extracting data from source systems (databases, SaaS applications, IoT devices, logs) and loading it into the target data store (Data Lake or Data Warehouse).

Step	Description	Technical Challenges
Batch ETL/ELT	Moving data at scheduled intervals (e.g., daily).	Handling large volume (petabytes); managing failed jobs; ensuring eventual consistency.
Real-Time Streaming	Capturing and processing data continuously (e.g., sensor data, clickstreams).	Ensuring low latency (sub-second); managing out-of-order events; guaranteeing delivery (exactly once semantics).
Change Data Capture (CDC)	Tracking and replicating only changes from transactional databases.	Database log parsing complexity; maintaining atomicity and consistency during replication.

Key Challenge: Data Velocity and Volume.

> The sheer speed and size of streaming data, especially from IoT and application logs, overwhelm traditional batch systems, requiring specialized tools like Apache Kafka or managed streaming services (AWS Kinesis, Azure Event Hubs).

Storage and Cataloguing: Organizing Data Assets

Once ingested, data is stored and indexed so users and systems can find, understand, and access it.

Step	Description	Technical Challenges
Storage (Data Lake)	Storing raw, unstructured data (S3, ADLS) cost-effectively.	**Data Swamps:** Uncontrolled growth leading to massive, ungoverned data lakes with poor discoverability.
Cataloguing & Metadata	Creating an index (catalogue) of all data assets, including schema, lineage, and usage.	Maintaining accuracy across rapidly evolving schemas; integrating disparate data sources into a single view.
Data Governance	Defining and enforcing policies on data quality, access, and compliance (GDPR, HIPAA).	Enforcing fine-grained access control (row-level security) across petabytes of data; tracking and proving data lineage.

Key Challenge: The "Last Mile" of Data Governance.

While data storage is cheap, ensuring metadata is accurate and access policies are consistently enforced across compute engines (SQL, Spark, ML tools) remains a complex governance hurdle.

Transformation: Cleaning, Enriching, and Modelling

Transformation converts raw data into a reliable, analytical format. The industry has shifted from ETL (Extract, Transform, Load) to **ELT (Extract, Load, Transform)**, where transformation happens *within* the data Warehouse or Lakehouse using tools optimized for cloud scale (e.g., SQL, dbt, Spark).

Step	Description	Technical Challenges
Data Cleansing	Fixing errors, handling missing values, and standardizing formats.	Identifying subtle anomalies (e.g., legitimate outliers vs. system errors); scaling complex cleansing logic across massive datasets.
Data Modelling	Structuring data into star, snowflake, or Kimball models for efficient BI.	Balancing normalization (data integrity) with denormalization (query performance); maintaining models as business logic changes.
Data Quality (DQ)	Establishing and monitoring rules to measure data fitness for use.	Defining universal quality metrics across diverse domains; implementing automated, preventative DQ checks *before* data reaches analysts.

Key Challenge: Observability and Trust.

Failures in transformation jobs directly impact business metrics. The challenge is implementing **Data Observability**—using automated tools to monitor the health, volume, schema, and latency of data pipelines in real-time, often using AI-driven anomaly detection.

Analysis and Consumption: Delivering Insights

This final stage involves querying the refined data to create reports, dashboards, operational applications, or feed ML models.

Step	Description	Technical Challenges
Business Intelligence (BI)	Running SQL queries for standardized reporting.	Maintaining low query latency on massive tables; managing resource contention between many concurrent users.
AI/ML Modelling	Using prepared data for training and inferencing.	Ensuring reproducibility of ML features; bridging the gap between data engineering pipelines and MLOps tools.
Reverse ETL	Moving cleaned, transformed data *back* into operational SaaS systems (e.g., Salesforce, Marketo).	Maintaining synchronization between the warehouse and hundreds of operational applications.

Key Challenge: Enabling Self-Service.

The goal is to democratize data access, but without proper governance, this leads to analytical silos and inconsistent metric definitions. The challenge is providing secure, governed, self-service tools for non-technical users ("Citizen Data Scientists").

Market Analysis, Players, and Opportunities

The global data management market is experiencing rapid expansion, driven by the shift to cloud-native architectures and the massive demand for AI-ready data. The Data Lakehouse architecture (estimated to grow at a CAGR of over 20% through 2030) defines the competitive landscape.

Major Market Players

The market is currently segmented into three dominant groups:

Category	Key Players & Offerings	Market Strategy
Cloud Hyperscalers	AWS (Redshift, S3, Lake Formation, Glue), Google Cloud (BigQuery, BigLake, Cloud Dataflow), Microsoft Azure (Synapse Analytics, ADLS)	Deep integration, highly optimized services, massive platform lock-in advantage. Driving AI integration directly into data services.
The Lakehouse & Cloud-Agnostic Leaders	Databricks (Delta Lake), Snowflake	**Databricks:** Pioneered the Data Lakehouse; strong focus on unified AI/ML and BI workloads using Delta Lake. **Snowflake:** Cloud-agnostic, near-zero administration, superior data sharing capabilities (Data Exchange).
Open-Source/Federation	Starburst (Trino/Presto), Dremio (Apache Arrow, Iceberg)	Focus on open data formats (Iceberg, Delta) and federation, allowing complex SQL queries across multiple data sources (Lakes, Warehouses, other databases) without physically moving the data.

Emerging Architectural Trends and Opportunities

The core opportunities in data management are no longer in basic storage but in the augmentation, governance, and monetization layers:

AI and Augmented Data Management (ADM)

AI is the single most significant trend. ADM refers to integrating AI/ML to automate repetitive tasks:

- **Opportunity:** AI-driven data quality, automatic schema detection, intelligent cataloging, and automated root-cause analysis (Observability). Vendors offering AI-powered tools that simplify the work of the data engineer will see exponential growth.

FinOps and Cloud Economics

As cloud spending soars, the focus has shifted to efficiency.

- **Opportunity:** Tools and services focused on **optimizing cloud spend**. This includes serverless architectures that precisely match compute to workload demand (pay-per-query/usage) and governance layers that automatically monitor and archive unused or stale data (data pruning).

Data Fabric vs. Data Mesh

While Data Mesh focuses on organizational change (decentralized ownership), Data Fabric focuses on technology (a unified, virtual data layer).

- **Opportunity:** The market needs technologies that enable **data virtualization** (Data Fabric) to deliver governed data products, allowing users to query data wherever it resides without physical migration.

The Data Sharing Economy

Platforms like Snowflake's Data Marketplace and Databricks' Delta Sharing enable organizations to buy, sell, and exchange data sets securely without requiring complex ETL or copies.

- **Opportunity:** Building specialized industry-specific data exchanges and platforms that connect vertical markets (e.g., healthcare data, financial market feeds).

NNP Data Management Solution

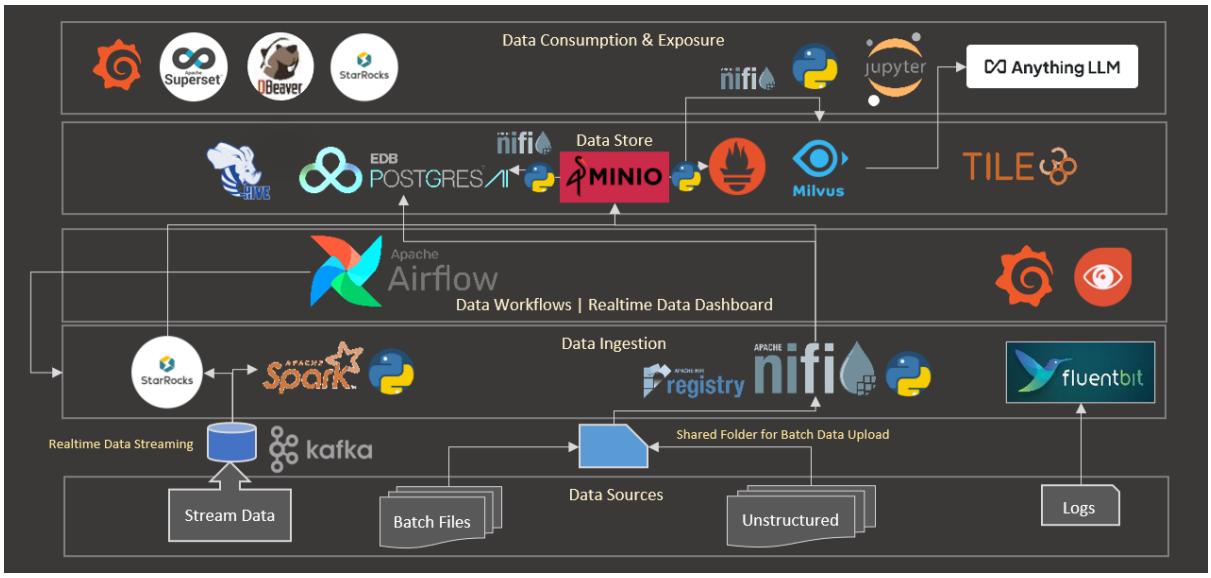
To operationalize the principles of modern data management, NUBONS has developed a Cloud Data Management Architecture designed to support both real-time and batch data pipelines on a unified, cloud-native platform.

This architecture leverages open-source technologies and containerized services to ensure scalability, cost efficiency, and interoperability across data sources and workloads.

NNP Data Management Architecture

Modern enterprises operate in a data-driven ecosystem, where information flows continuously across heterogeneous systems. To harness this data effectively, a well-structured and scalable **Data Management Architecture** is required — one that supports both real-time and batch data flows, while ensuring consistency, reliability, and analytical agility.

The following architecture represents the generic **cloud-native data management framework** implemented by NUBONS that can be adapted for multiple enterprise use cases. Each layer in this pipeline plays a vital role in ensuring data moves seamlessly from its source to consumption, providing actionable intelligence across the organization.



Data Source Layer

Data originates from diverse environments and systems, often differing in format, velocity, and quality.

- **Structured Data:** Relational databases, ERP systems, and transactional applications provide data essential for operational analytics.

- **Unstructured Data:** Text files, documents, multimedia, and sensor data collected from multiple endpoints.
- **Logs:** System, application, and infrastructure logs that provide observability and monitoring data for real-time analysis.

This heterogeneous mix establishes the foundation of the data pipeline.

Data Integration Layer

This layer is the entry point of all data into the platform, responsible for capturing, stearming and moving data from various sources into the platform in real time or batches.

- **Apache Kafka:** Serves as a distributed event streaming platform that ingests and buffers real-time data from multiple producers.
- **Apache NiFi:** Enables visual design of dataflows, allowing low-code integration, transformation, and routing between systems.
- **Apache NiFi Registry:** Provides version control, configuration management, and lifecycle governance for NiFi dataflows.
- **Apache Spark:** Provides large-scale distributed data processing for ETL, analytics, and machine learning tasks.
- **FluentD:** A unified data collector that helps in aggregating logs and event data from multiple sources.
- **Promtail:** Collects logs from servers or containers and forwards them to Grafana Loki for centralized log management.

Together, these tools form the backbone of the **data ingestion and transformation** ecosystem, ensuring flexible and scalable integration pipelines.

Workflow Orchestration Layer

To manage complex dependencies, scheduling, and automation across diverse data processes, **Apache Airflow** serves as the orchestration engine.

- It defines workflows as **Directed Acyclic Graphs (DAGs)**, enabling end-to-end control of ETL jobs, transformations, and data quality checks.
- Airflow integrates seamlessly with NiFi, Spark, and Python-based scripts, ensuring coordinated execution across both batch and real-time pipelines.

This layer ensures **automation, traceability, and fault tolerance** within the data ecosystem.

Data Storage Layer

The storage layer serves as the **data lake and warehouse foundation**, offering persistence, scalability, and query efficiency.

- **MinIO:** Acts as the object storage layer for raw and intermediate datasets, providing S3-compatible cloud-native data storage.
- **PostgreSQL:** Used for structured transactional or aggregated data, supporting analytics and downstream visualization tools.

- **Prometheus:** Stores real-time time-series metrics, useful for system monitoring and observability analytics.
- **Milvus:** A vector database designed for storing and searching unstructured embeddings used in AI/ML workloads.
- **Hive:** Supports big data warehousing and querying across large-scale datasets stored in distributed environments.
- **Grafana Loki:** A horizontally scalable log aggregation system that works seamlessly with Promtail for centralized log analysis.

This combination of databases and storage systems ensures data is stored **securely, efficiently, and in the most suitable format** for its use case.

Consumption and Exposure Layer

This layer provides the **interfaces for exploration, analytics, and data exposure** to end users and external systems.

- **Apache Superset:** Provides an open-source visualization layer enabling self-service business intelligence.
- **Grafana:** Used for observability dashboards, real-time performance monitoring, and metric visualization.
- **DBeaver:** A universal SQL client used by developers and analysts for direct data querying and exploration.
- **StarRocks:** A high-performance analytical database that allows real-time querying and federated analysis across storage systems.
- **SparkSQL:** Enables SQL-based analytics and transformations on distributed Spark datasets.
- **Apache NiFi:** Enables API-driven data exposure and seamless data delivery to downstream applications.
- **Python.ai:** Supports advanced analytics and machine learning model deployment pipelines.

This layer ensures **data accessibility, democratization, and actionability** across the enterprise.

Data Consumer Layer

The final layer represents the **end users and consuming applications** that derive value from processed data for business decisions, automations and further exposures as per needs from end users.

- **Enterprise Data Consuming Applications:** Operational systems and analytical dashboards that use processed datasets.
- **Real-time Insights and Actions:** Monitoring systems and automated workflows that act upon live data streams.
- **Data API Exposure:** APIs that make curated datasets available for partner integrations, AI applications, or customer-facing analytics.

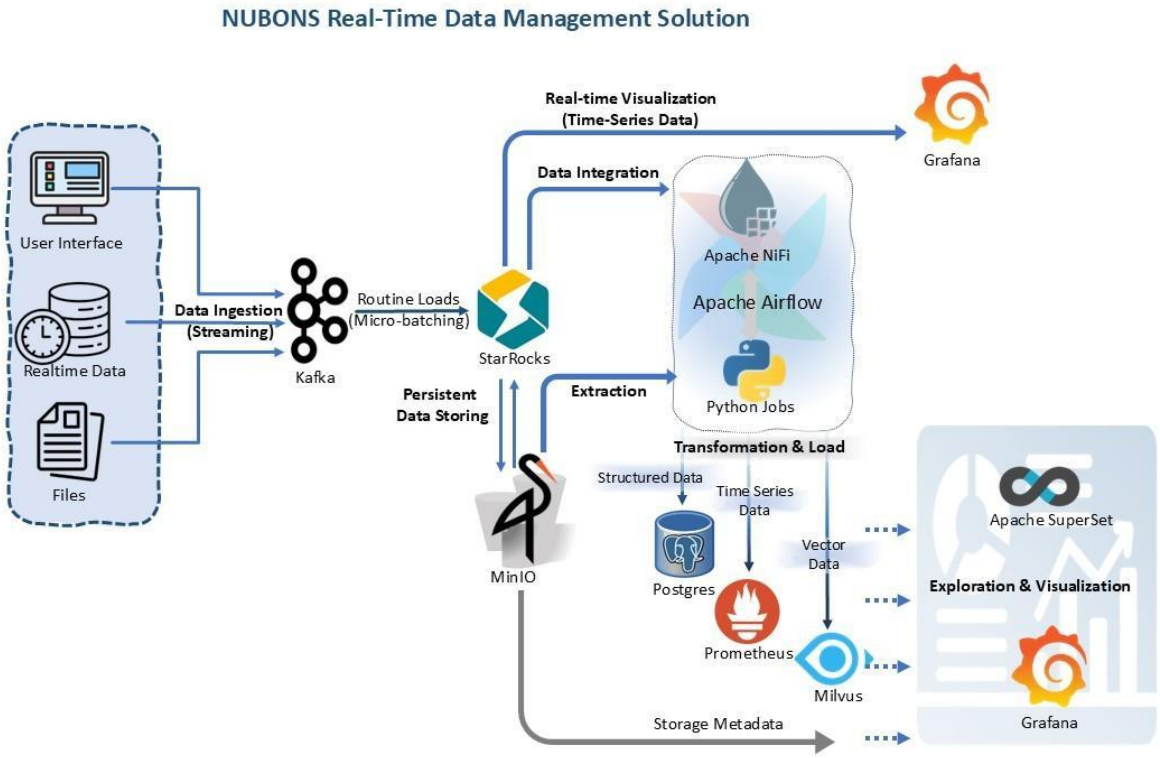
This layer closes the data loop — delivering **timely, accurate, and contextual insights** for decision-making and automation.

Let's get deeper into 2 specific implementations of the above NUBON Data Management framework based on data generation and its usage.

Real-time Data Management

NUBONS Real-time Data Management architecture has 2 complementary data pipelines that work together to enable both real-time monitoring and batch-oriented business analytics on real-time data.

1) Real-time Data Pipeline
2) Batch Data Pipeline for Real-time data Streaming



Real-time Data Pipeline

NUBONS Real-time Data Pipeline for managing the real-time data designed for low-latency ingestion and analytics, ensuring that streaming data from heterogeneous sources can be visualized within seconds for operational intelligence.

A. Heterogeneous data sources continuously stream their events into Apache Kafka
   - Applications, IoT devices, and databases push data into Kafka.
   - Kafka ensures durability, ordering, and scalability for real-time ingestion. It works as the central hub for event-driven pipelines.

B. StarRocks connects directly to Kafka to pull data
   - StarRocks continuously pulls data from Kafka topics using Routine Load jobs.
   - StarRocks acts purely as a compute and query engine in this architecture.

C. MinIO works as Unified Storage
    - All ingested data is stored durably in MinIO (S3-compatible).
    - StarRocks does not use local storage. It stores and fetches data from MinIO when queries are executed, This enables almost instantaneous analytical responses on fresh data.

D. Grafana is used for visualization
    - Grafana connects to StarRocks as a data source. StarRocks pulls data from MinIO.
    - Grafana provides vilsualization the real-time metrics, KPIs and near time-series dashboards with latency in seconds.

Batch Data Pipeline for Real-time Data Streaming

The batch data pipeline focuses on curated, transformed datasets stored over time, supporting historical analysis, compliance reporting, and advanced BI dashboards for strategic decision-making. For real-time data streaming, this batch pipeline complements the real-time flow by ensuring that curated, transformed, and business-ready datasets are available for advanced analytics and decision-making.

A. Heterogeneous data sources (applications, IoT devices, logs, etc.) continuously stream events into Apache Kafka. Kafka topics serve as the ingestion layer, ensuring durability and fault tolerance of the incoming data.

B. StarRocks subscribes to Kafka streams via routine load jobs and passes the ingested data into MinIO, which is used as the primary storage layer.

C. Apache Airflow orchestrates the batch workflows, coordinating NiFi pipelines, Python transformations, and storage updates across multiple destinations.

D. Apache NiFi runs the Extraction-Transformation-Load Routine using Python scripts.
    - NiFi fetches raw or semi-processed data directly from MinIO via S3-compatible connectors.
    - Can also query StarRocks if required, to extract curated datasets.
    - Transformations are performed using NiFi processors and custom Python scripts.
    - Data is reshaped and enriched to match downstream schema requirements.

E. Transformed data is loaded into a Polyglot Storage Layer
    - PostgreSQL for relational, transactional, and structured batch datasets.
    - Prometheus for time-series and metrics data (collected from pipeline jobs or transformed monitoring events).
    - Milvus for vector embeddings and unstructured data (e.g., images, audio features, NLP embeddings) supporting AI/ML-driven queries.

F. Apache Superset provides a BI layer to explore data and aggregated insights.
    - Superset connects to PostgreSQL for BI and historical reporting.
    - Enables interactive dashboards, SQL-based exploration, and batch analytics.

Uniqueness and Benefits NUBONS Data Management Architecture

NNP Data Management Platform stands out due to its modular, open-source, and cloud-native design philosophy. The system's flexible orchestration and data decoupling ensure superior scalability and maintainability compared to traditional monolithic ETL pipelines.

Key Strengths

- Unified Real-time and Batch Framework: A single ecosystem handles streaming and scheduled workloads, minimizing redundancy.
- Storage Agnosticism through MinIO: All data — structured or unstructured — is centrally stored, making it easily accessible for diverse analytics tools.
- High Performance via StarRocks: Enables near-instant querying across large data volumes without storing duplicates.
- End-to-End Automation: Airflow orchestrates cross-tool workflows, reducing human error and improving reliability.
- Multi-Destination Data Delivery: Postgres, Prometheus, and Milvus each address specialized analytics needs, ensuring optimized query performance.
- Open-source and Cost-efficient: The architecture eliminates vendor lock-in while leveraging enterprise-grade reliability from open technologies.
- Enhanced Observability: Grafana provides integrated monitoring and real-time performance visibility across the data landscape.

Strategic Advantage

This architecture demonstrates NUBONS commitment to open innovation, scalability, and transparency in cloud data management.

It serves as a model for organizations seeking a future-ready, modular data platform that supports analytics, machine learning, and AI-driven decision-making with minimal infrastructure overhead.

By combining the best of streaming frameworks, open storage, and workflow orchestration, XYZ has effectively created a next-generation data management ecosystem that is adaptable, efficient, and resilient.

Strategic Value of Open-Source Tools in Data Management

While proprietary cloud platforms offer convenience and managed services, open-source technologies remain the backbone of the modern data stack, providing strategic advantages in agility, cost control, and extensibility.

Advantages of Open-Source Tools

Avoidance of Vendor Lock-In: Open-source tools (like Apache Spark, Trino, Iceberg) can run on any major cloud (AWS, Azure, GCP), private cloud, or on-premises infrastructure. This provides critical migration flexibility and protects the organization from proprietary pricing or feature changes.

Cost Efficiency (Scale): While managed cloud services simplify operations, open-source solutions often offer significant cost savings at massive scale, as licensing fees are eliminated, and compute resources can be custom-optimized.

Community Innovation and Extensibility: Open-source projects evolve faster than many proprietary tools, benefiting from contributions from thousands of developers across the industry. This allows for rapid feature adoption and integration with new technologies (e.g., new connectors or ML algorithms).

Auditability and Customization: Organizations can fully audit the source code for security, compliance, or custom tuning purposes, which is essential for regulated industries.

Conclusion: Data Management as Competitive Differentiator

The data management landscape has reached a new level of maturity and complexity, moving far beyond simple storage and reporting. In the 2025 era, a successful data strategy is a direct prerequisite for competitive advantage, particularly through the application of AI and Machine Learning.

Organizations that succeed will embrace a **unified, open architecture** (the Data Lakehouse) paired with **decentralized data ownership** (Data Mesh principles). The challenges are no longer purely technical but center on governance, trust, and cost management. The implementation of **Augmented Data Management (ADM)** and **FinOps** methodologies is crucial for scaling while maintaining control.

By strategically leveraging cloud elasticity, adopting open-source flexibility to avoid vendor lock-in, and prioritizing data quality and observability, enterprises can transform their data pipelines from a necessary cost center into a resilient, real-time factory for intelligence and competitive differentiation.

Glossary of Modern Data Architecture Terms

Term	Definition
ACID Properties	Atomicity, Consistency, Isolation, Durability. Database properties ensuring reliable transaction processing.
Data Lakehouse	A hybrid architecture combining the low-cost storage and flexibility of a Data Lake with the structure and governance of a Data Warehouse.
Data Mesh	An organizational and architectural approach that decentralizes data ownership to specific business domains, treating data as a product.
Data Fabric	An architectural pattern that uses intelligence (AI/ML) and a unified set of services to access and manage data scattered across disparate systems without moving the data physically.
Data Observability	The practice of proactively monitoring the health and quality of data within a system, often focusing on metrics related to volume, freshness, schema, and lineage.
FinOps	Cloud Financial Operations—the practice of bringing financial accountability to the variable spend model of cloud computing, often involving automated cost monitoring and optimization.
Reverse ETL	The process of moving modelled, analytically-ready data from a data warehouse or Lakehouse back into operational SaaS tools (e.g., CRM, marketing platforms).

About Nubo Native Solution

Nubo Native Solution is working on a mission to democratize cloud by providing a sovereign, adaptable and comprehensive Cloud Platform referred as Nubo Native Platform (NNP) for state-of-the-art Cloud Native Development and Hosting.

Nubo Native Solution with its path-breaking Cloud Platform and associated Consulting and Professional Services enables large-scale Cloud Repatriation, complex Application Modernization, API Lifecycle Management, AI Enablement, Edge Computing and accelerated Software Development ensuring lower TCO and improved TTM, for the Enterprises worldwide.

Compiled by Nubo Native Platform team

November 2025

Website: nubons.com
Email: contact@nubons.com