ENTERPRISE BLUEPRINT

Data Estate Blueprint

A comprehensive architecture guide for building modern, AI-ready enterprise data platforms that scale from startup to enterprise.

VERSION 2.0

PAGES 12

LAST UPDATED February 2025

Contents

01

Executive Summary

Your data estate is the foundation of AI-driven business transformation. This blueprint provides a structured approach to designing, implementing, and scaling enterprise data architecture.

Modern enterprises generate data at unprecedented scale—from customer interactions and IoT sensors to financial transactions and operational logs. The organizations that thrive are those that can transform this raw data into actionable intelligence, predictive insights, and automated decisions.

This blueprint presents a proven approach to building a modern data estate that supports everything from basic reporting to advanced machine learning and generative AI. Whether you're modernizing legacy systems or building greenfield, these patterns will accelerate your journey.

10x	60%	95%	24/7
Faster Time to Insight	Cost Reduction	Data Quality Score	Real-time Availability

Who This Blueprint Is For

CDOs & Data Leaders

Strategic framework for data transformation initiatives and business alignment.

Data Architects

Reference architecture patterns and technology selection guidance.

Data Engineers

Implementation patterns, best practices, and technical specifications.

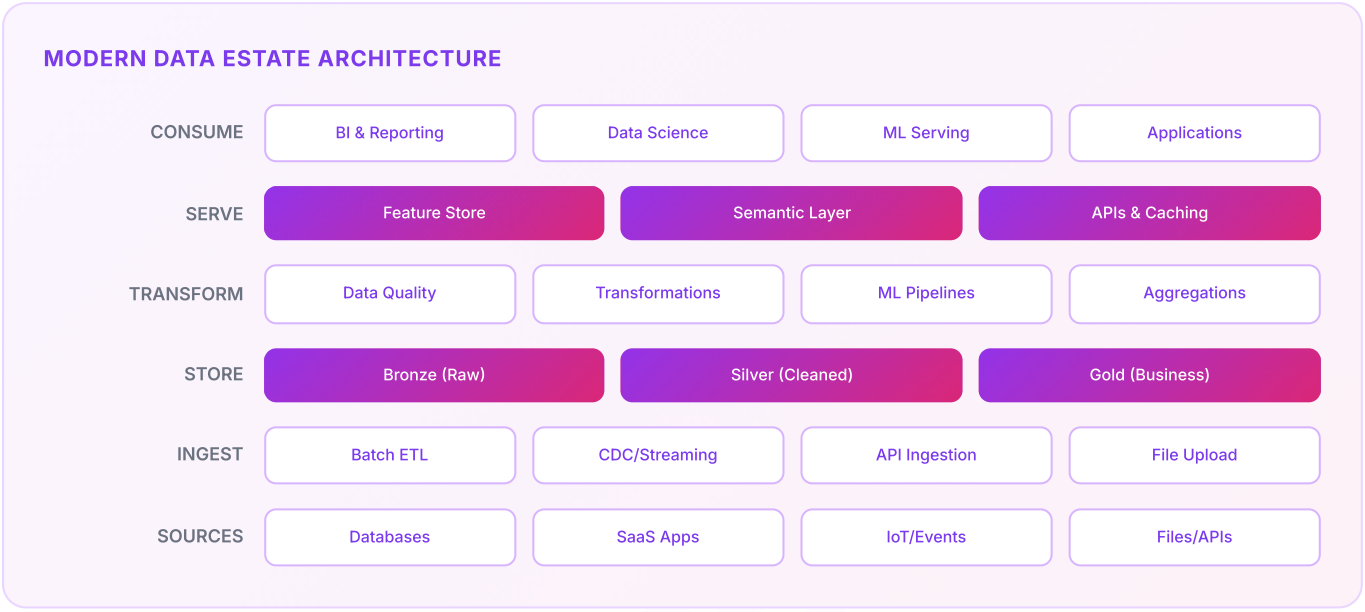
Data Scientists

ML infrastructure requirements and feature engineering patterns.

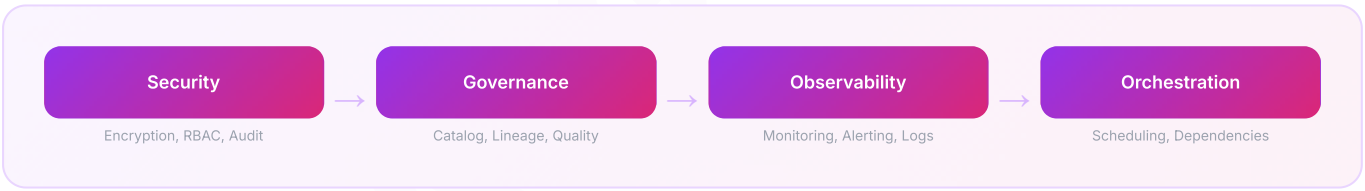
02

Reference Architecture

The modern data estate follows a layered architecture pattern that separates concerns while enabling seamless data flow from sources to consumers.



Cross-Cutting Concerns



03

Core Architecture Layers

Each layer serves a specific purpose in the data lifecycle, with clear responsibilities and interfaces.

Ingestion Layer

The ingestion layer is responsible for reliably capturing data from all sources with appropriate latency guarantees. Modern ingestion must handle both batch and real-time patterns while maintaining data lineage and quality checks at the edge.

Pattern	Use Case	Latency	Technologies
Batch ETL	Large volume historical loads, nightly syncs	Hours	Spark, Airflow, dbt
Micro-batch	Near real-time dashboards, aggregations	Minutes	Spark Streaming, Flink
CDC Streaming	Database replication, event sourcing	Seconds	Debezium, Kafka Connect
Real-time Events	User activity, IoT sensors, fraud detection	Milliseconds	Kafka, Kinesis, Pub/Sub

Storage Layer (Medallion Architecture)

The medallion architecture organizes data into progressive quality layers, enabling both raw data preservation and optimized analytics access.

Bronze Layer (Raw)

Exact copy of source data with ingestion metadata. Immutable, append-only. Enables replay and reprocessing.

Silver Layer (Cleaned)

Validated, deduplicated, and standardized data. Schema enforcement and data quality rules applied.

Gold Layer (Business)

Business-level aggregations and metrics. Optimized for consumption by analysts and applications.

Platinum Layer (ML)

Feature tables and training datasets. Versioned, reproducible, and optimized for model training.

Key Principle: Lakehouse Architecture

Modern platforms combine data lake flexibility with warehouse performance using Delta Lake, Apache Iceberg, or Apache Hudi. This enables ACID transactions, time travel, and schema evolution on cloud object storage.

04

Technology Stack

Recommended technologies for each layer, with cloud-agnostic and cloud-specific options.

Layer	AWS	Azure	GCP	Cloud-Agnostic
Lakehouse Platform	Databricks, EMR	Databricks, Synapse	Databricks, Dataproc	Databricks, Spark
Data Warehouse	Redshift	Synapse Analytics	BigQuery	Snowflake
Streaming	Kinesis, MSK	Event Hubs	Pub/Sub, Dataflow	Kafka, Confluent
Orchestration	Step Functions, MWAA	Data Factory	Cloud Composer	Airflow, Dagster
Transformation	Glue, dbt	Data Factory, dbt	Dataform, dbt	dbt, Spark SQL
Catalog & Governance	Glue Catalog	Purview	Dataplex	Unity Catalog
Feature Store	SageMaker FS	Azure ML FS	Vertex AI FS	Feast, Tecton

Recommended Stack by Maturity

Startup / Early Stage

Stack: Snowflake + Fivetran + dbt + Airflow
Why: Fast to implement, low ops overhead, scales well

Growth / Mid-Market

Stack: Databricks + Kafka + dbt + MLflow
Why: Unified analytics + ML, strong governance, flexible

Enterprise

Stack: Databricks + Confluent + Unity Catalog + Feast
Why: Full governance, real-time, ML-ready at scale

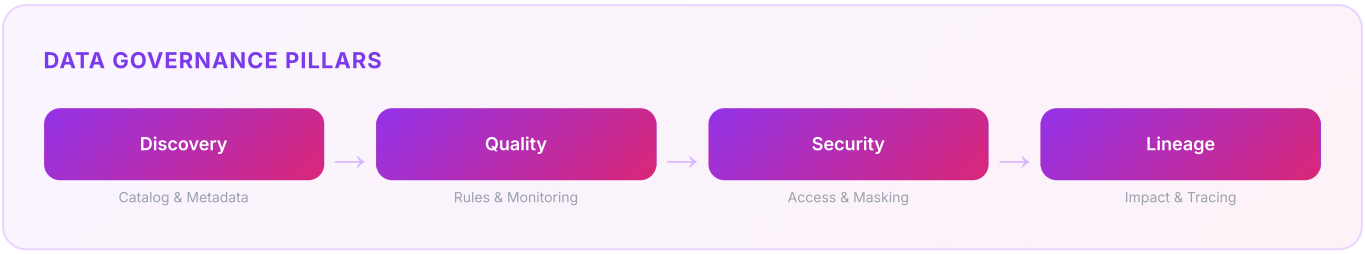
Regulated Industry

Stack: Cloud-native + Collibra + Privacera
Why: Fine-grained access control, audit trails, compliance

05

Data Governance Framework

Governance is not optional—it's the foundation that enables trust, compliance, and self-service at scale.



Data Catalog Requirements

Automated Discovery

Crawl and classify data assets automatically. Detect schema changes and new tables.

Business Glossary

Link technical metadata to business terms. Enable non-technical users to find data.

Data Ownership

Assign stewards and owners. Enable request/approval workflows for access.

Usage Analytics

Track query patterns and popular datasets. Identify unused and duplicate data.

Data Quality Dimensions

Dimension	Definition	Example Rule
Completeness	Required fields are populated	email IS NOT NULL
Accuracy	Values match expected patterns	email LIKE '%@%.%'
Consistency	Values align across systems	status IN ('active', 'inactive')
Timeliness	Data is current and fresh	updated_at > NOW() - 1 day
Uniqueness	No duplicate records	COUNT(DISTINCT id) = COUNT(*)

06

Implementation Roadmap

A phased approach that delivers value incrementally while building toward the complete data estate.

Phase 1: Discovery & Assessment

Weeks 1-4

Current state analysis, data source inventory, stakeholder interviews, and gap assessment. Define success metrics and prioritize use cases.

Phase 2: Architecture Design

Weeks 5-10

Technology selection, security design, governance framework, and scalability planning. Create detailed technical specifications and cost models.

Phase 3: Foundation Build

Weeks 11-22

Deploy core infrastructure, implement initial pipelines, establish governance framework. Deliver first production workloads.

Phase 4: Migration & Integration

Weeks 23-46

Migrate legacy workloads, integrate additional data sources, expand self-service capabilities. Enable ML and advanced analytics.

Phase 5: Optimization & Scale

Ongoing

Performance tuning, cost optimization, capability expansion. Continuous improvement based on usage patterns and feedback.

Quick Win Strategy

Identify 2-3 high-value use cases that can be delivered in Phase 3. These early wins build momentum, demonstrate value to stakeholders, and provide learning for subsequent phases.

07

Success Metrics & KPIs

Measure what matters—track these KPIs to demonstrate value and guide continuous improvement.

Operational Metrics

Metric	Target	How to Measure
Data Freshness	< 15 min for operational data	Lag between source update and availability
Pipeline Reliability	> 99.5% success rate	Successful runs / total scheduled runs
Query Performance	p95 < 5 seconds	95th percentile of query execution time
Data Quality Score	> 95% across critical tables	Passing quality checks / total checks

Business Value Metrics

Metric	Target	Impact
Time to Insight	Reduced from weeks to hours	Faster decision making
Self-Service Adoption	> 60% of analysts	Reduced data team bottleneck
Cost per TB	Reduced by 40%+	Infrastructure efficiency
ML Model Velocity	3x faster deployment	Accelerated AI initiatives

40%
Avg. Cost Reduction

10x
Faster Insights

99.5%
Pipeline Reliability

08

Common Anti-Patterns

Learn from others' mistakes—these are the pitfalls we see most frequently in data platform initiatives.

Data Swamp

Problem: Dumping raw data without schema, lineage, or governance.
Solution: Implement metadata management from day one. Enforce naming conventions and documentation requirements.

Copy-Paste Integration

Problem: Creating redundant copies across systems.
Solution: Use virtual federation and semantic layers. Establish single source of truth patterns.

Premature Optimization

Problem: Over-engineering for scale before understanding needs.
Solution: Start simple, measure, then optimize. Right-size infrastructure based on actual usage.

Governance Afterthought

Problem: Adding security and governance after the fact.
Solution: Design governance in from the beginning. It's 10x harder to retrofit.

Tool Island Syndrome

Problem: Teams selecting tools in isolation.
Solution: Establish enterprise standards while allowing flexibility. Create integration patterns.

Dashboard Overload

Problem: Too many dashboards, no single source of truth.
Solution: Curate key metrics. Implement semantic layer for consistent definitions.

09

Implementation Checklist

Use this checklist to track your progress and ensure nothing is missed.

Foundation

- [ ] Cloud landing zone provisioned with security controls
- [ ] Lakehouse platform deployed and configured
- [ ] Network connectivity to source systems established
- [ ] IAM and RBAC policies defined and implemented

Data Ingestion

- [ ] Batch ingestion pipelines for priority sources
- [ ] CDC/streaming infrastructure for real-time needs
- [ ] Data quality checks at ingestion points
- [ ] Error handling and dead letter queues configured

Storage & Processing

- [ ] Bronze/Silver/Gold layers implemented
- [ ] Transformation pipelines with dbt or equivalent
- [ ] Partitioning and optimization strategies applied
- [ ] Backup and disaster recovery tested

Governance & Operations

- [ ] Data catalog populated with metadata
- [ ] Data quality monitoring dashboards live
- [ ] Lineage tracking enabled end-to-end
- [ ] Alerting and on-call procedures documented

Ready to Build Your Data Estate?

Our data engineering experts can help you design and implement a modern data platform tailored to your specific needs. Schedule a free consultation to discuss your data strategy.

aegisit.ai/contact

(404) 490-0234 | info@aegisit.ai