Below is a **complete, enterprise-grade Data Architecture** for a **multi-product healthcare platform**, including **Analytics, ML, and GenAI capabilities**, written at **Chief / Principal Data Architect + Solution Architect depth**.

Data Architecture – Multi Product Healthcare Platform

Scope: Clinical, Member, Provider, Claims, Operations, Finance, Digital, Devices
Design goals: Trust, safety, interoperability, analytics at scale, explainable AI

1️⃣ Data Architecture Principles

1. **Data is a product** (owned, versioned, governed)
2. **Patient safety first** (clinical data ≠ marketing data)
3. **Federated ownership, centralized governance**
4. **Schema contracts over pipelines**
5. **Event-first for operational data**
6. **Analytics isolated from operational workloads**
7. **Explainability over accuracy (for clinical AI)**
8. **Privacy by design (HIPAA, state laws)**
9. **AI-ready, not AI-hyped**
10. **Lineage everywhere**

2️⃣ Logical Data Architecture (End-to-End)

```
┌─────────────────────────────────────────────────────────┐
│           Data Producers (Sources)           │
│ EHR | Claims | Pharmacy | Labs | Devices | Apps | Partners │
└─────────────────────────────────────────────────────────┘
           │
           ▼
```

```
┌──────────────────────────────────────────────────────────┐
│           Data Ingestion Layer                │
│ Events | APIs | CDC | Batch | Streaming           │
└──────────────────────────────────────────────────────────┘
            │
            ▼
┌──────────────────────────────────────────────────────────┐
│         Operational Data Layer (ODS)             │
│ Service DBs | Event Stores | Cache | Search          │
└──────────────────────────────────────────────────────────┘
            │
            ▼
┌──────────────────────────────────────────────────────────┐
│          Data Platform (Lakehouse)              │
│ Raw → Refined → Curated                          │
│ Quality | Lineage | Catalog | Governance             │
└──────────────────────────────────────────────────────────┘
            │
            ▼
┌──────────────────────────────────────────────────────────┐
│          Semantic & Serving Layer               │
│ FHIR | Metrics | Features | BI Models | APIs          │
└──────────────────────────────────────────────────────────┘
            │
            ▼
┌──────────────────────────────────────────────────────────┐
```

```
│      Analytics | ML | GenAI | Decision Support        │
└────────────────────────────────────────────────────────┘
```

3️⃣ Data Source Layer (Producers)

Internal

- EHR (Epic)
- Claims & billing
- Pharmacy & lab
- Scheduling
- Devices (IoT, wearables)
- Digital apps
- Operations

External

- HIEs
- Payers
- CMS
- Vendors
- Research partners

4️⃣ Data Ingestion Layer

Patterns (choose by use case)

Pattern	Used For
Event streaming (Kafka/Kinesis)	Clinical events, updates
CDC (Debezium)	EHR extracts
API ingestion	Partner data

Pattern	Used For
Batch	Claims, finance
File-based	Legacy

Rule:
👉 Events preferred over batch wherever possible

5️⃣ Operational Data Layer (ODS)

Purpose

Support real-time operational use cases without hitting EHR or analytics platforms.

Includes:

- Per-service databases
- Event stores
- Caches
- Search indexes
- Materialized views

Key rule:
👉 ODS is *not* the data warehouse

6️⃣ Data Platform (Lakehouse Architecture)

Zones

Zone	Purpose
Raw	Immutable ingest
Refined	Cleaned, standardized
Curated	Analytics-ready
Feature	ML features

Zone	Purpose
Secure	PHI-restricted

Technology

- Delta / Iceberg / Hudi
- Spark / Flink
- Object storage
- Data catalog
- DQ engine
- Lineage engine

7️⃣ Data Governance (Mandatory)

Ownership

- Data domain owners (per product line)
- Stewards
- Custodians

Controls

- Schema registry
- Data contracts
- DQ rules
- Lineage
- Access policies
- Retention policies

Tools

- Collibra / Alation
- Great Expectations

- OpenLineage
- IAM-integrated access

8️⃣ Semantic Layer (CRITICAL)

Purpose

Make data usable without breaking safety or meaning.

Includes:

- FHIR-aligned models
- Business metrics
- Clinical concepts
- Time-aware measures
- Versioned definitions

This is where trust is built

9️⃣ Analytics Architecture

Types

Type	**Examples**
Descriptive	Dashboards, reports
Diagnostic	Root cause
Predictive	Risk models
Prescriptive	Care suggestions
Operational	Near real-time KPIs

Tools

- BI (Power BI, Tableau)
- SQL endpoints

- Metrics layer
- Data APIs

🔟 Machine Learning Architecture

Lifecycle

Ingest → Feature Engineering → Training → Validation → Registry → Serving → Monitoring

Components

- Feature store
- Model registry
- Training pipelines
- Model serving (real-time/batch)
- Drift detection
- Bias detection
- Human-in-the-loop review

Healthcare rule:
👉 Models must be explainable and auditable

1️⃣ 1️⃣ GenAI Architecture (SAFE & CONTROLLED)

Use Cases

- Clinical summarization
- Documentation assist
- Coding & billing assist
- Knowledge search
- Member communication
- Agent assist (call centers)

GenAI Reference Flow

User

↓

App

↓

AI Gateway

↓

Prompt Management

↓

Retrieval (RAG)

↓

LLM

↓

Validation

↓

Audit

Guardrails (Non-negotiable)

- No PHI sent to public LLMs
- RAG only from governed sources
- Prompt versioning
- Output validation
- Full audit trail
- Human approval for clinical output

1️⃣2️⃣ Data Products by Product Line

Each product line publishes:

- Domain datasets
- Metrics
- Events
- Features
- APIs

This enables **data mesh with governance**

1️⃣3️⃣ Security & Privacy Architecture

Area	**Implementation**
Encryption	At rest + transit
Access	IAM + ABAC
Masking	Dynamic
Tokenization	PHI
Audit	Immutable
Consent	Platform-managed
Residency	Regional rules

1️⃣4️⃣ Observability for Data

- Pipeline health
- Freshness
- Completeness
- Volume anomalies
- Cost observability
- Model drift

- GenAI hallucination detection

1️⃣5️⃣ Common Failure Modes & Mitigations

Failure	**Mitigation**
Data inconsistency	Contracts + semantic layer
Broken pipelines	Observability + retries
PHI leakage	DLP + masking
AI hallucination	RAG + validation
Untrusted dashboards	Certified datasets
Model bias	Bias monitoring
Vendor lock-in	Open formats

1️⃣6️⃣ Why This Data Architecture Works

- Supports **multi-product scale**
- Separates operational vs analytical workloads
- Enables **safe AI & GenAI**
- Enforces governance without killing speed
- Supports regulatory audits
- Enables real-time + batch analytics
- Reduces EHR load
- Builds clinician trust

🎯 Interview-ready summary

"My data architecture treats data as a product, enforces governance through contracts and semantics, and enables analytics, ML, and GenAI safely. In healthcare, trust and traceability matter more than raw model accuracy — so everything is auditable and explainable.”
