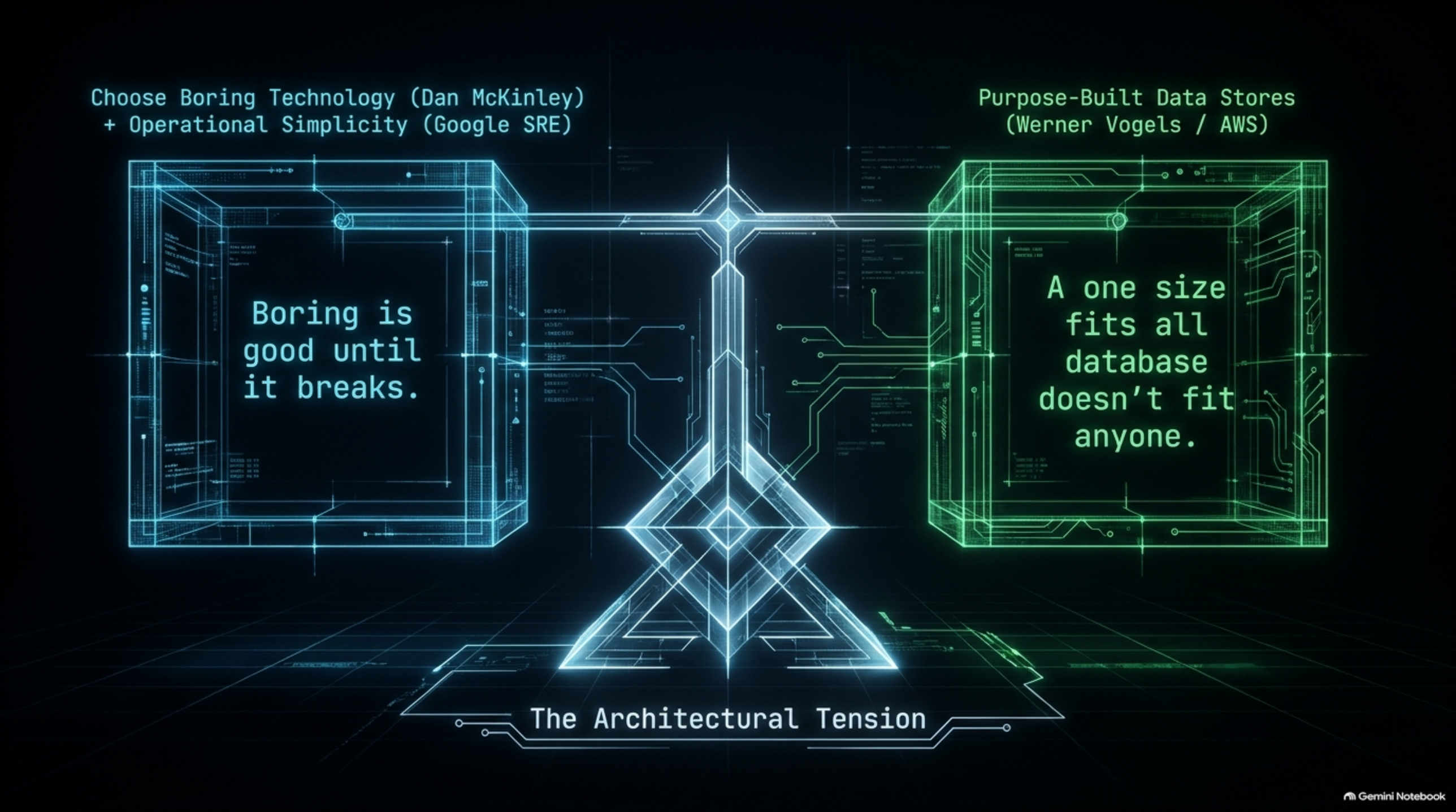
Modern Data Stores — S05E07

The Architecture Decision Framework, Polyglot Tax & Production Case Studies

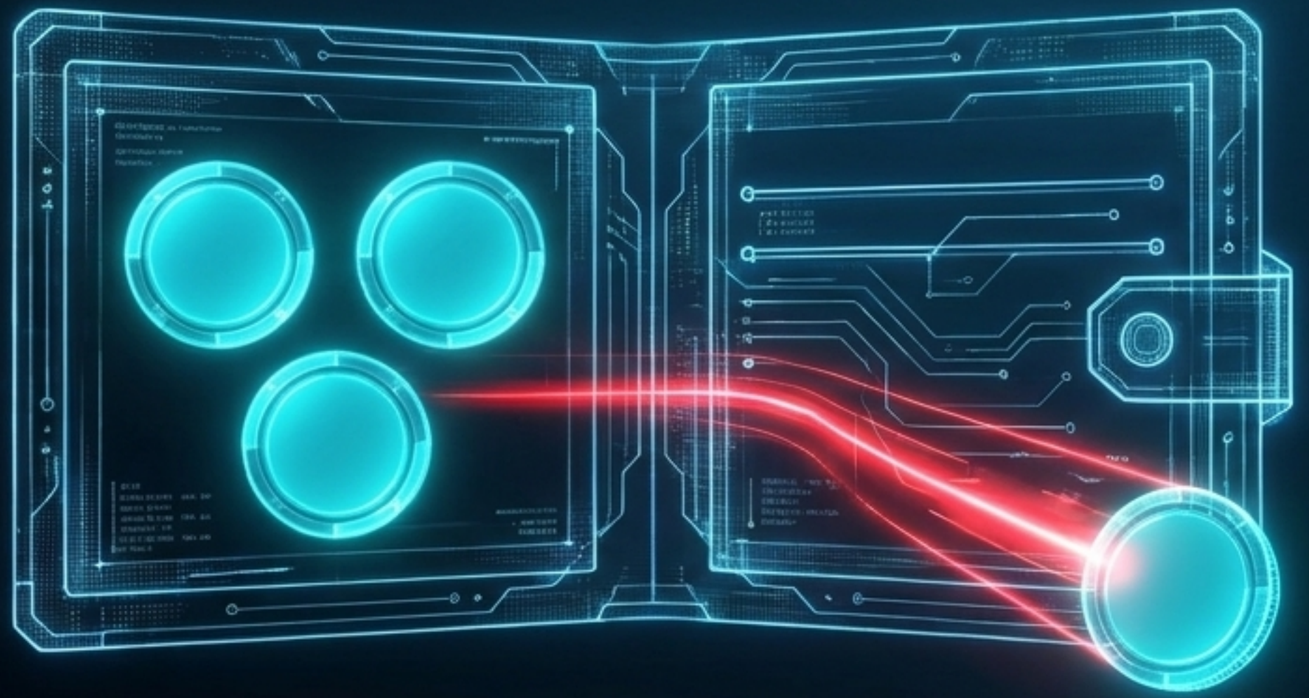
A Capstone Playbook for Software Engineers & Systems Architects

Gemini Notebook



Gemini Notebook

The Innovation Token Wallet



Every company gets ~3 tokens.

Spending one on a shiny new database drains resources that could be spent on product iteration.

WARNING: If you choose to write your own database, you have spent all your tokens and are in trouble.

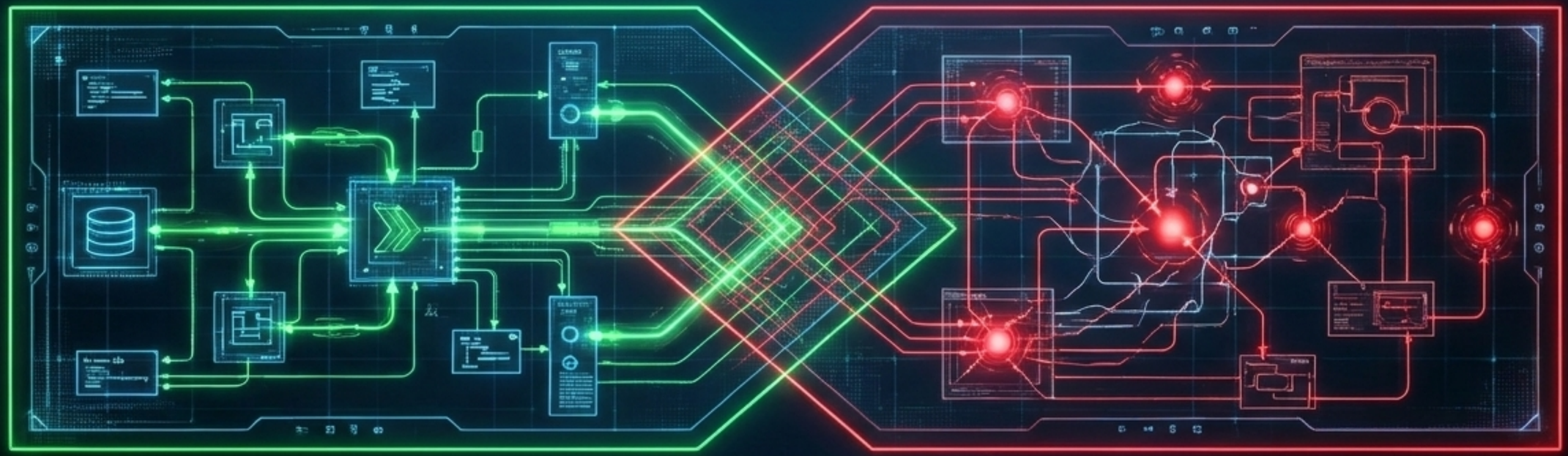
Gemini Notebook

Essential Complexity

The unavoidable difficulty of solving the actual business problem (e.g., serving web pages quickly).

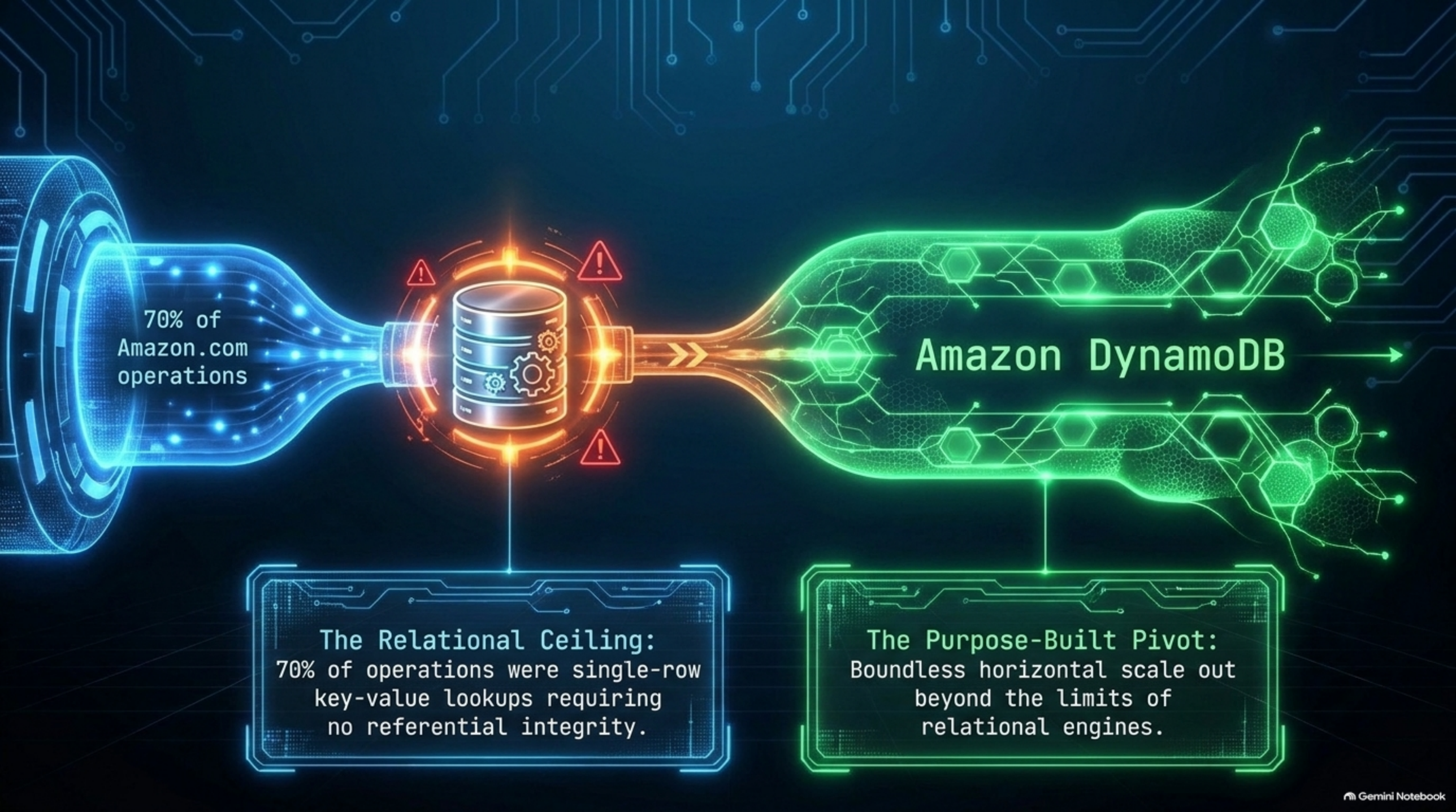
Accidental Complexity

The self-inflicted pain of maintaining the tools (e.g., managing JVM garbage collection pauses in legacy datastores).



> “At the end of the day, our job is to keep agility and stability in balance. Surprises in production are the nemeses of SRE.”

Gemini Notebook



Gemini Notebook

The 7 Quantitative Workload Questions

Architectural decisions cannot be made on vibes.
Gather these exact metrics before choosing a datastore.

ROW 1. [__ ___ __]
ROW 2. [__ ___ __]

ROW 3. [__ ___ __]
ROW 4. [__ ___ __]
ROW 5. [__ ___ __]

ROW 6. [__ ___ __]
ROW 7. [__ ___ __]

Gemini Notebook

Q1: Read/Write QPS & Ratio (Are you read-heavy or write-heavy?)
Q2: Volume (Working-set RAM vs Total TB on disk).



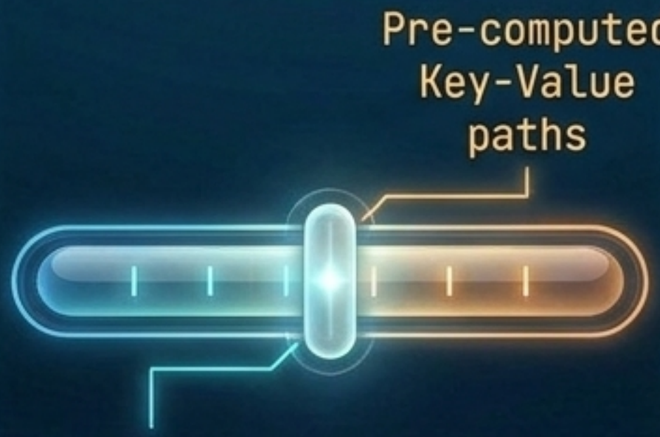
Gemini Notebook

Q3: p99 Latency. Q4: Query Flexibility. Q5: Consistency Needs.

SPEED



SHAPE



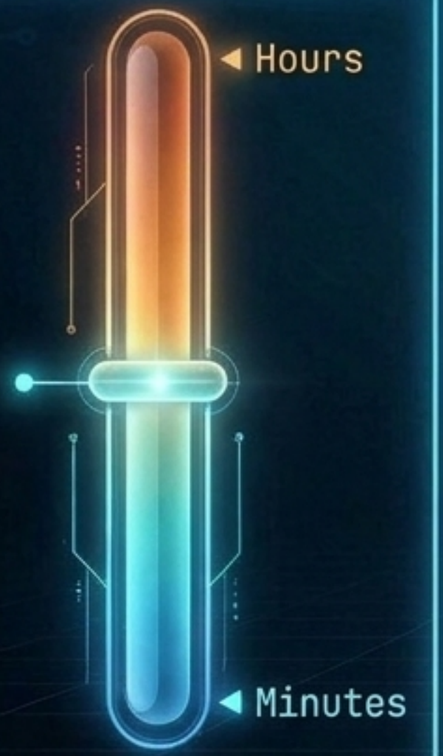
STATE



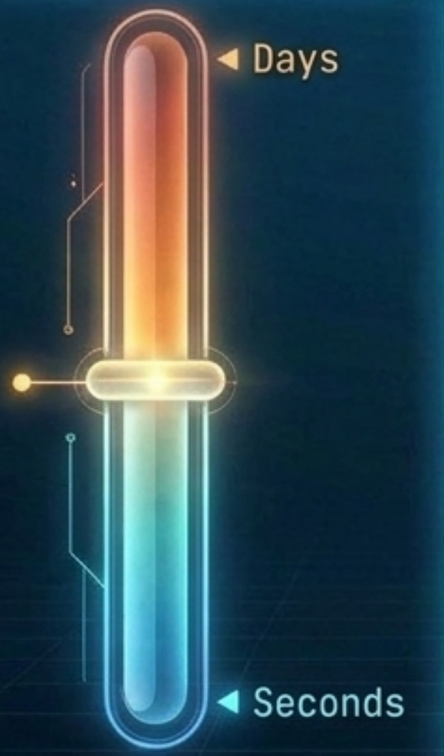
Gemini Notebook

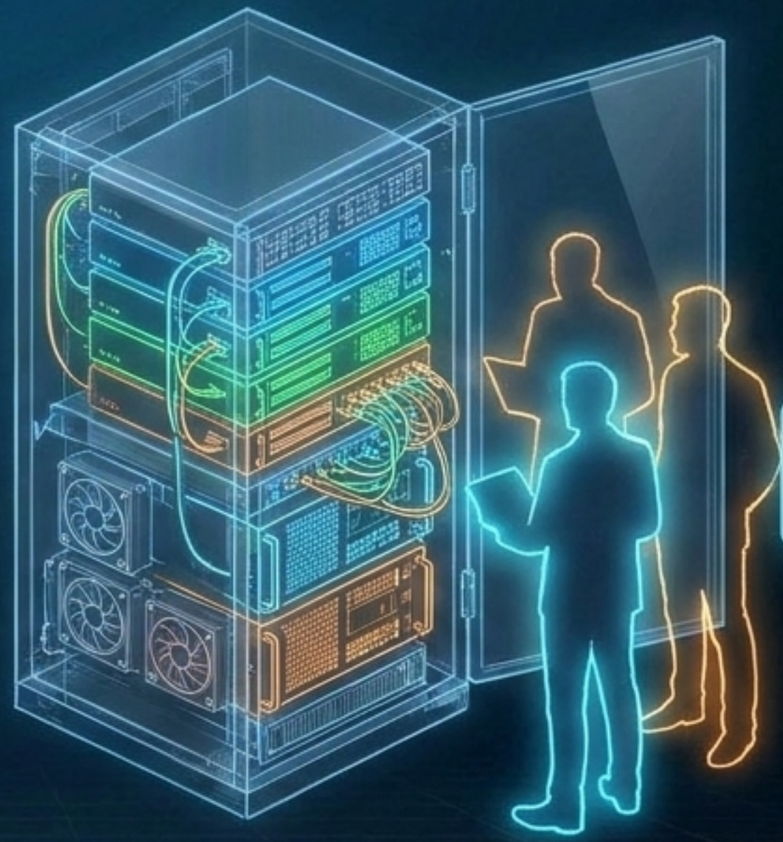
Q6: Disaster Recovery (RPO/RTO). Q7: SRE Headcount.

RPO
(Recovery Point Objective)



RTO
(Recovery Time Objective)





SRE Headcount

005

> “The long-term costs of keeping a system working reliably vastly exceed any inconveniences encountered while building it.”

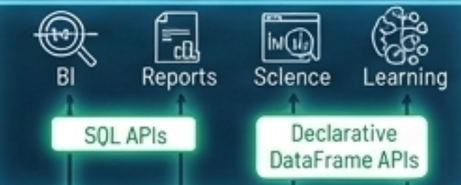
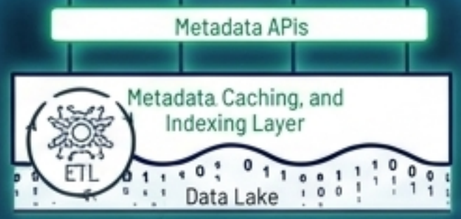
Gemini Notebook

	Ideal Workload Triggers	Scale Thresholds	Operational Burden (Low/Med/High)	Do NOT Use When
SQLite	Single-user, embedded applications. Low QPS requirements.	<1TB DB size. Low write concurrency.	Low (No server, minimal tuning).	Disqualified for concurrent network writers.
PostgreSQL (+ jsonb + pgvector)	Complex relational queries, JSON ops, vector search. Transactional workloads.	Mid-high QPS. Terabyte scale databases.	Med (Requires DBA, tuning, replication).	Disqualified at 32TB single table.
Redis	Caching, real-time analytics, pub/sub, session storage. Sub-ms latency.	High QPS. Data must fit in memory (up to few hundred GB).	Med (Memory management, clustering, persistence).	Dataset exceeds available RAM.
DynamoDB	Predictable K/V access, massive scale, serverless apps. High availability.	Extremely high QPS, PB scale. Auto-scaling.	Low (Fully managed, serverless, auto-scaling).	Complex ad-hoc queries or JOINs.

Gemini Notebook

	Ideal Workload Triggers	Scale Thresholds	Operational Burden (Low/Med/High)	Do NOT Use When
CockroachDB / Vitess	Distributed SQL, high availability, multi-region transactions. Elastic scale.	PB scale. Extremely high QPS. Global distribution.	Med-High (Cluster management, schema design, multi-region ops).	Disqualified for low-latency single-node applications.
ScyllaDB / Cassandra	Ideal for write-heavy IoT/telemetry. Time-series, wide-column. High availability.	PB scale. Massive write throughput. Linearly scalable.	Med (Node management, data modeling, consistency tuning).	Disqualified for complex relational queries, joins, ACID transactions.
Neo4j	Graph data models, relationship traversal, knowledge graphs. Deep queries.	Billions of nodes/relationships. High query complexity.	Med (Graph data modeling, query optimization, clustering).	Disqualified for massive historical analytical scans.
DynamoDB	Predictable K/V access, massive scale, serverless apps. High availability.	Extremely high QPS, PB scale. Auto-scaling.	Low (Fully managed, serverless, auto-scaling).	Complex ad-hoc queries or JOINs.

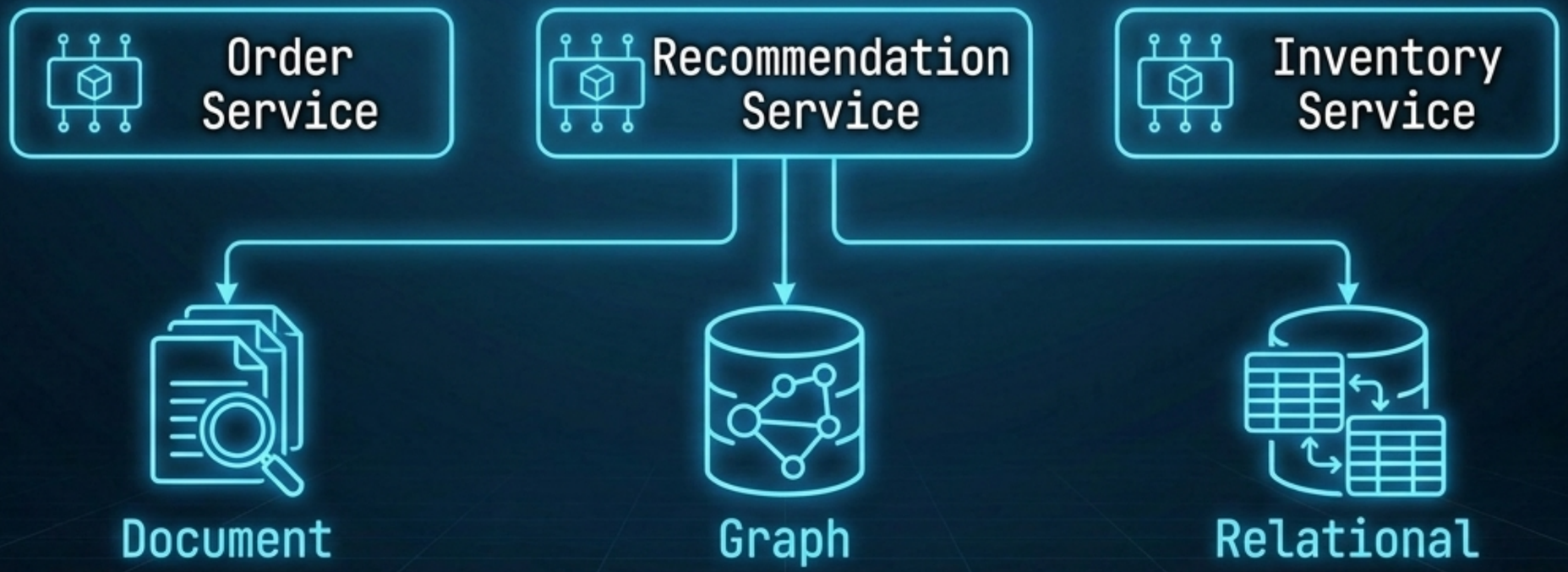
Gemini Notebook

	Ideal Workload Triggers	Scale Thresholds	Operational Burden (Low/Med/High)	Do NOT Use When
DuckDB	Embedded analytics, high-performance OLAP, local data processing. Column-	<1TB dataset. Single-node, scale-up architecture.	Low (Embeddable, serverless option, minimal overhead).	Disqualified for highly concurrent multi-writer OLTP.
ClickHouse	Real-time analytics, high-throughput ingestion, massive log data analysis.	PB scale. Trillions of rows. High concurrency reads.	Med (Cluster management, sharding, tuning for concurrency).	Disqualified for frequent updates/deletes or complex transactions.
Snowflake / BigQuery	Cloud data warehousing, massive-scale OLAP, multi-cluster compute. Decoupled storage.		Low (Fully managed, serverless, automated maintenance).	Disqualified for low-latency operational transactions or single-row lookups.
Apache Iceberg	Ideal for massive historical analytical scans.		Med (Requires query engines, governance, compaction).	Disqualified for low-latency OLTP or real-time point lookups.
Kafka / Debezium	Real-time data streaming, event sourcing, CDC. Decoupling services.	Extremely high throughput. PB scale. Distributed logs.	Med-High (Cluster management, topic configuration, offsets).	Disqualified for complex ad-hoc queries, joins, or random access.
Qdrant / Milvus	Vector similarity search, AI/ML applications, semantic search. Specialized vector DB.	Billions of vectors. High QPS for vector search.	Med (Index tuning, resource management, scaling).	Disqualified for general purpose relational or document workloads.

Gemini Notebook

Polyglot Persistence & Database-per-Service

Fowler’s premise of wrapping specific storage technologies within bounded service contexts.



The benefit is perfect fit. The cost is cognitive overhead and the loss of the transactional single point of truth.

Gemini Notebook

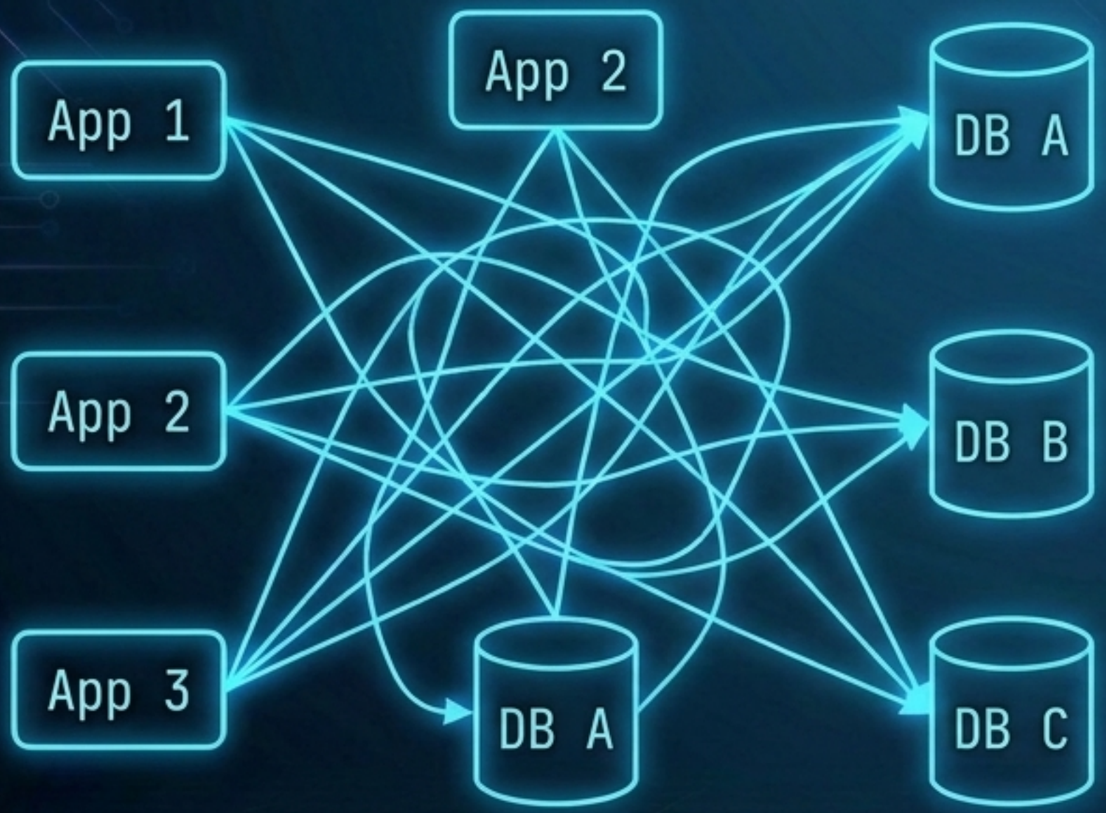
The Polyglot Tax: Dual Write Corruption



Why dual writes inherently corrupt state. Without distributed transactions (which kill availability via CAP theorem), partial failures leave systems permanently out of sync.

Gemini Notebook

BEFORE



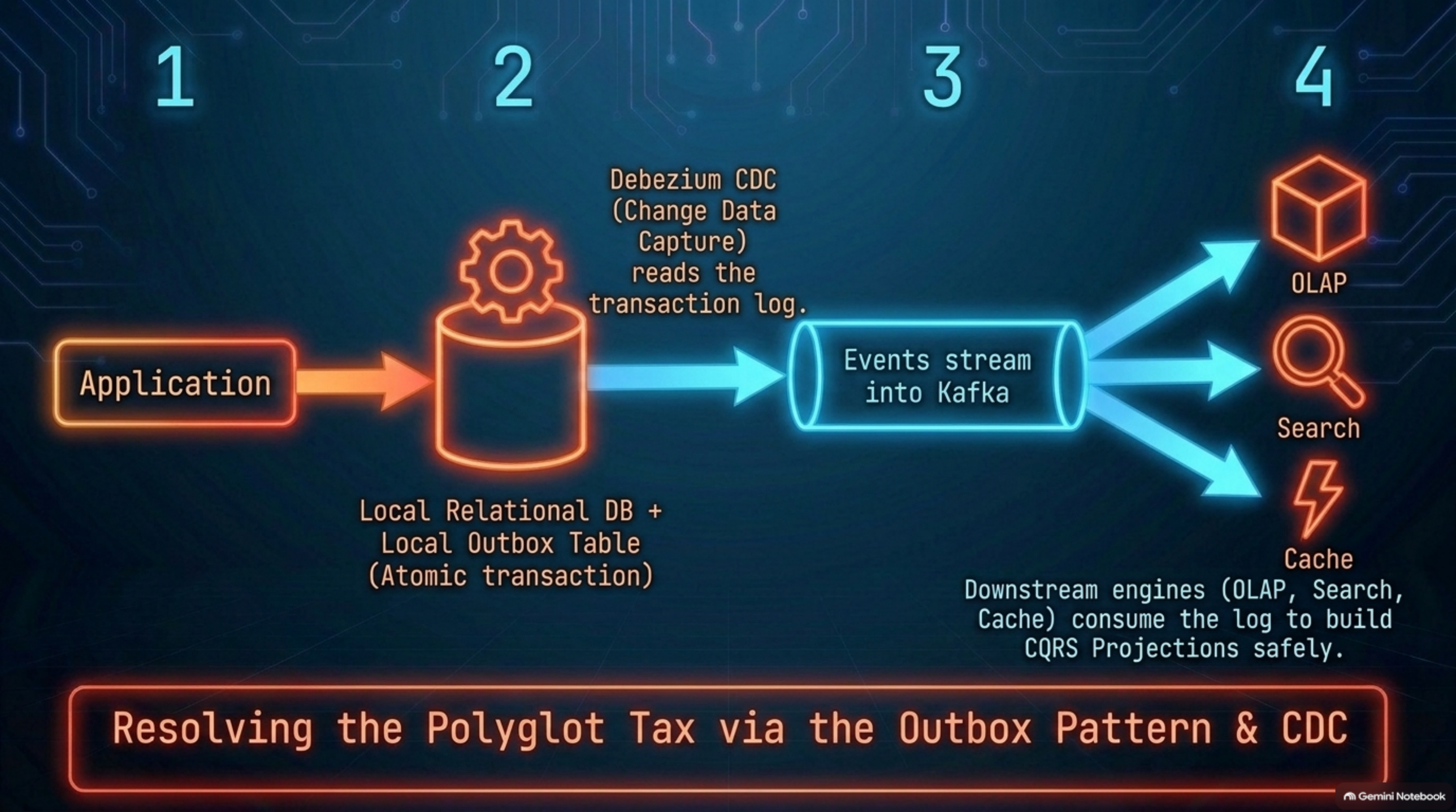
AFTER

Turning the Database Inside Out.



The transaction log becomes the central nervous system. Databases become unbundled modular components (caches, indexes, materialized views) purely derived from the immutable event stream.

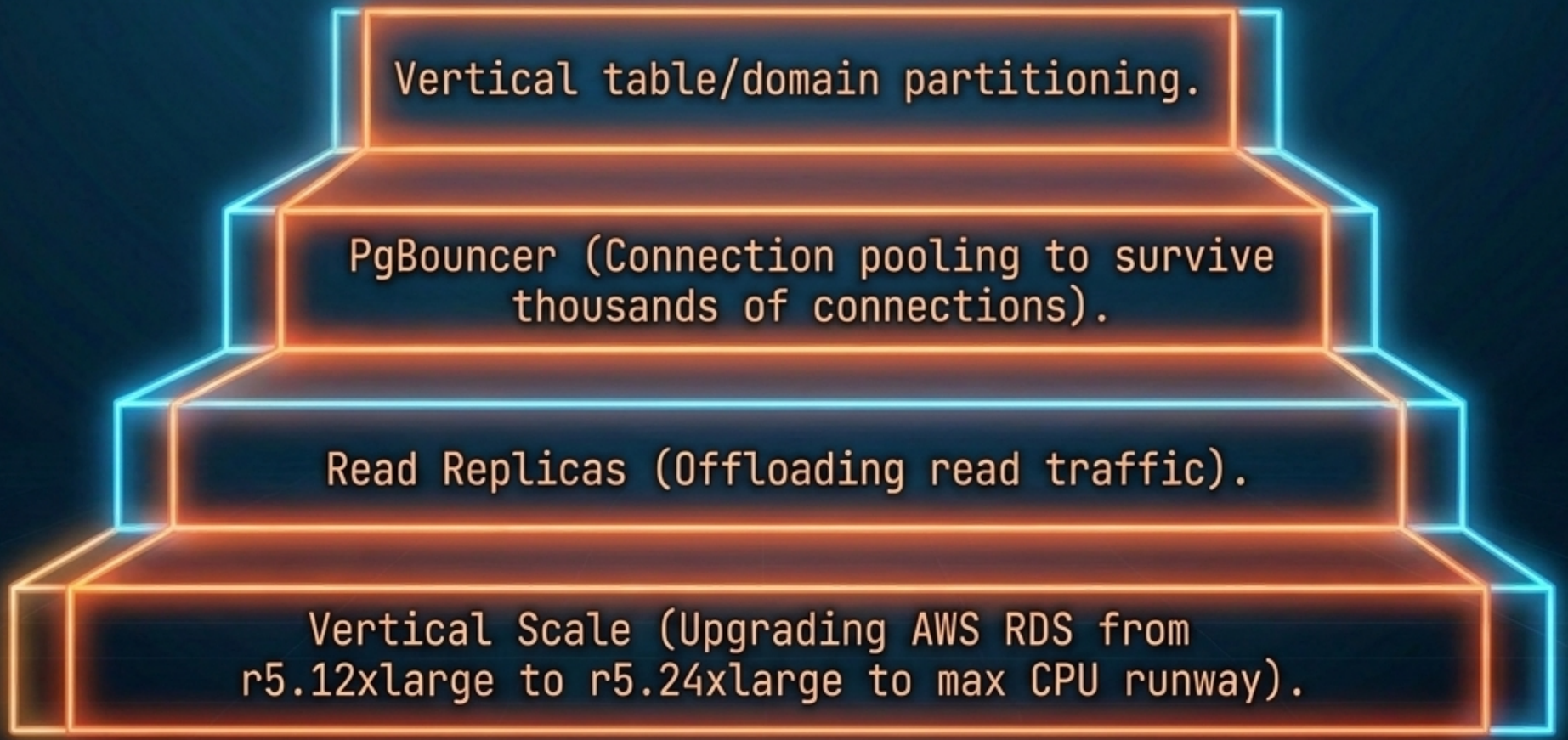
Gemini Notebook



Gemini Notebook

Case Study 1 (Figma) – Relational Evolution

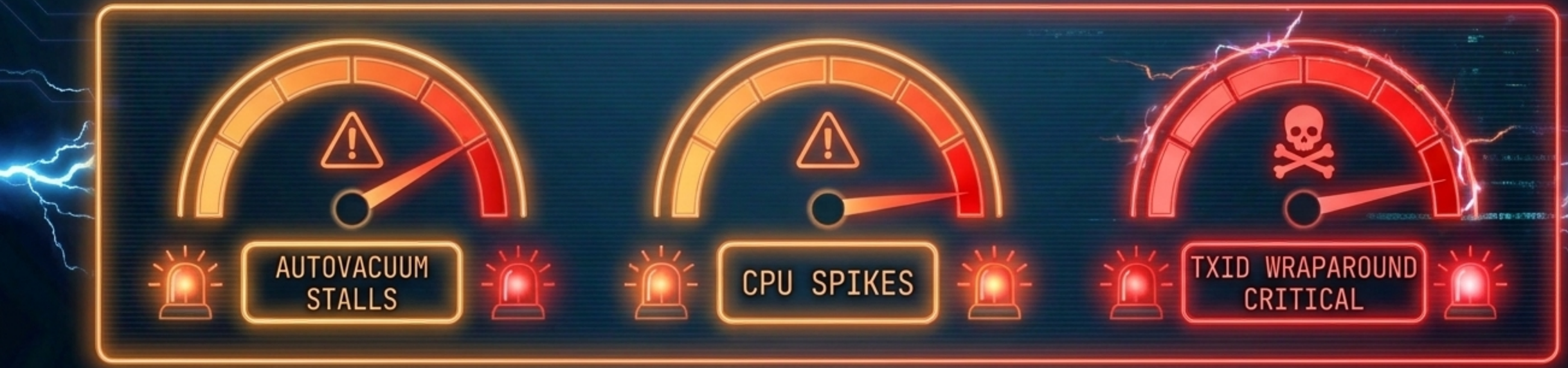
Exhaust standard relational scaling capabilities before adopting distributed architectures.



Gemini Notebook

Case Study 1 (Notion) – Hitting the Ceiling

When catalog-only migrations become unsafe and the monolith breaks.



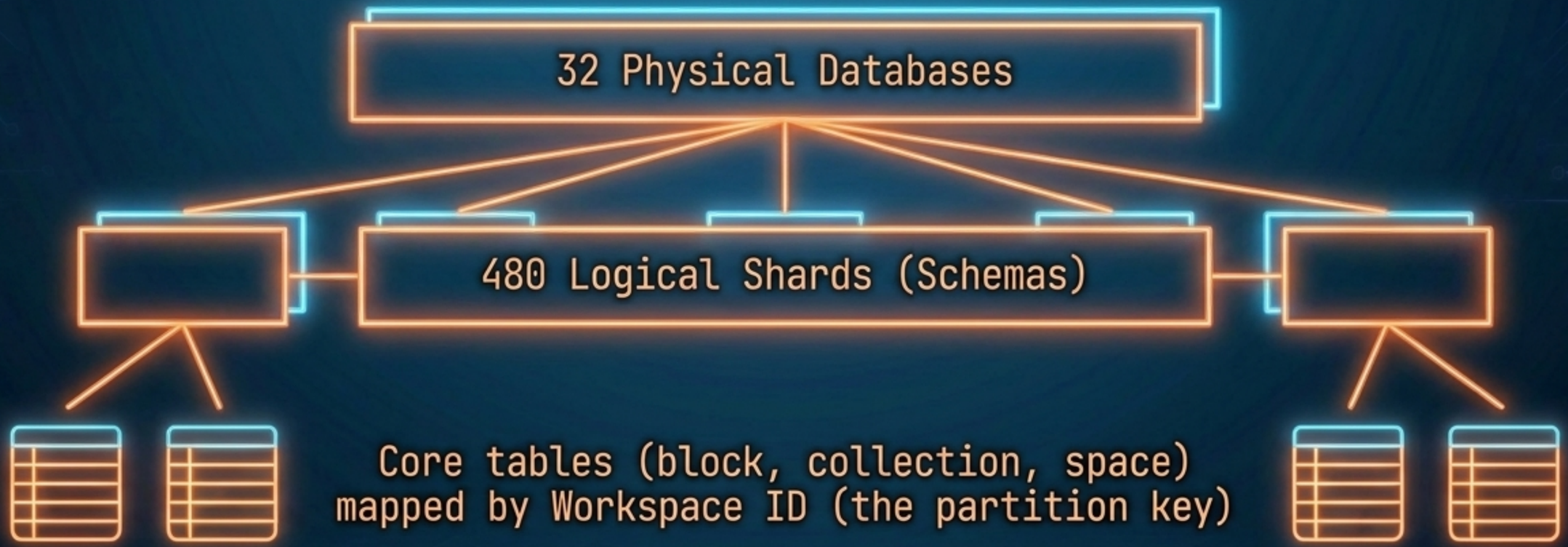
AUTOVACUUM Stalls – unable to reclaim disk space from dead tuples.

Constant CPU spikes and thrashing.

Transaction ID (TXID) Wraparound – The existential threat where Postgres stops all writes to prevent data clobbering.

Gemini Notebook

The Last Resort: Application-Level Sharding



Why not Vitess or CockroachDB? Application context dictated tight control over partition keys to ensure all block data for a workspace lived on a single host, avoiding distributed joins.

Gemini Notebook

Case Study 2 (Discord) – Trillions of Messages



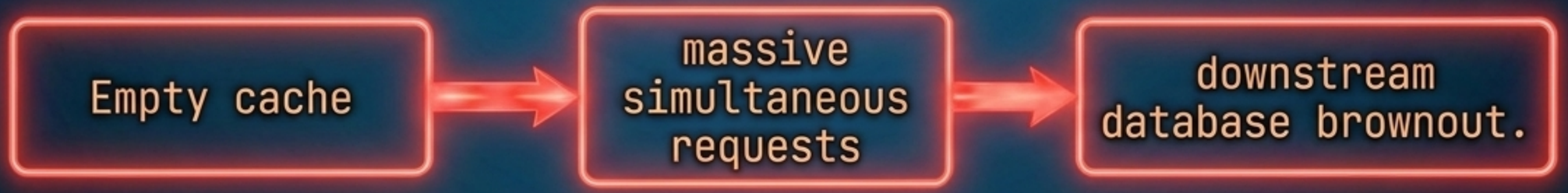
(Hit JVM Garbage Collection
pauses & hot-partition cliffs)

The metrics of success: Reduced fleet from 177 Cassandra nodes to just 72 ScyllaDB nodes (9TB disk each). p99 latency dropped from 40-125ms down to a stable 15ms.

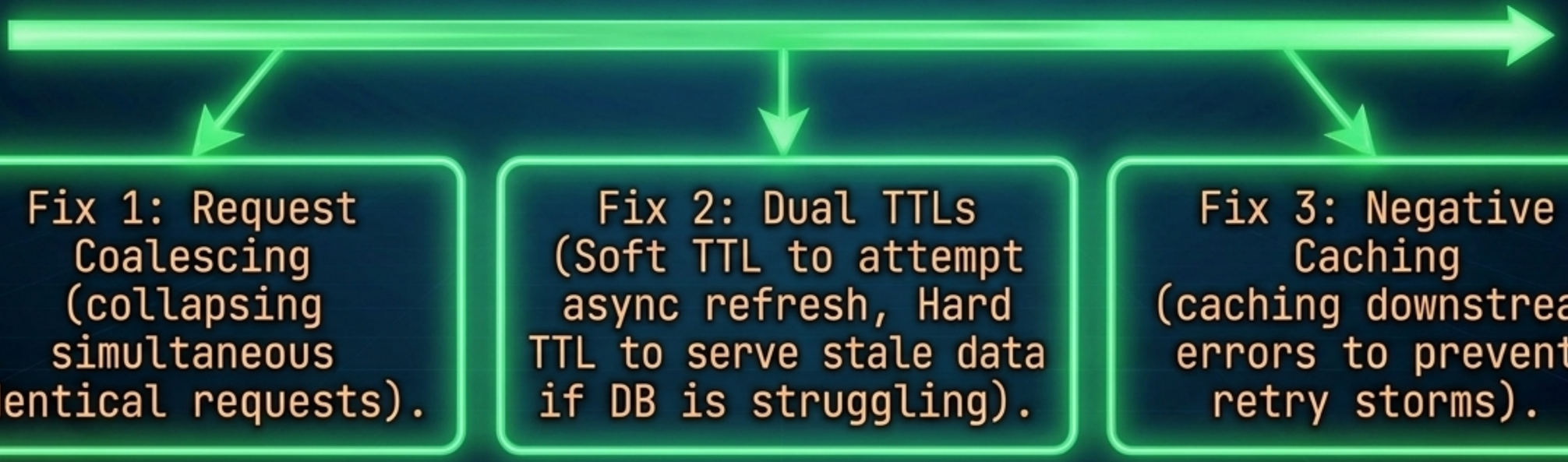
Gemini Notebook

The Caching Trap

Thundering Herd Anti-Pattern



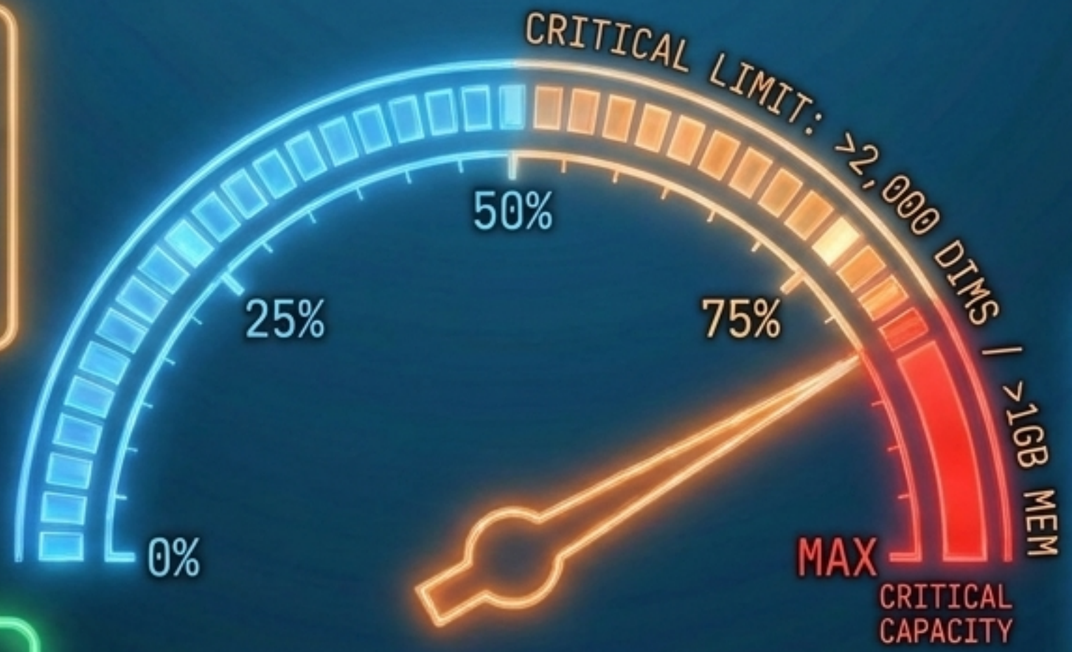
Mitigation strategies applied.



Gemini Notebook

Graduating from pgvector

Max Dimensions:
2,000 for standard vector,
4,000 for halfvec,
64,000 for bit with
binary quantization.



CRITICAL LIMIT REACHED:
>2,000 DIMENSIONS / HNSW MEMORY WALL HIT

The Memory Wall:
HNSW graph indexes
build drastically
slower if they do not
fit entirely in
maintenance_work_mem.

Graduation Trigger:
When IVFFlat recall drops
unacceptably, or HNSW
memory limits are
breached at scale, spend
an innovation token on a
dedicated vector engine.

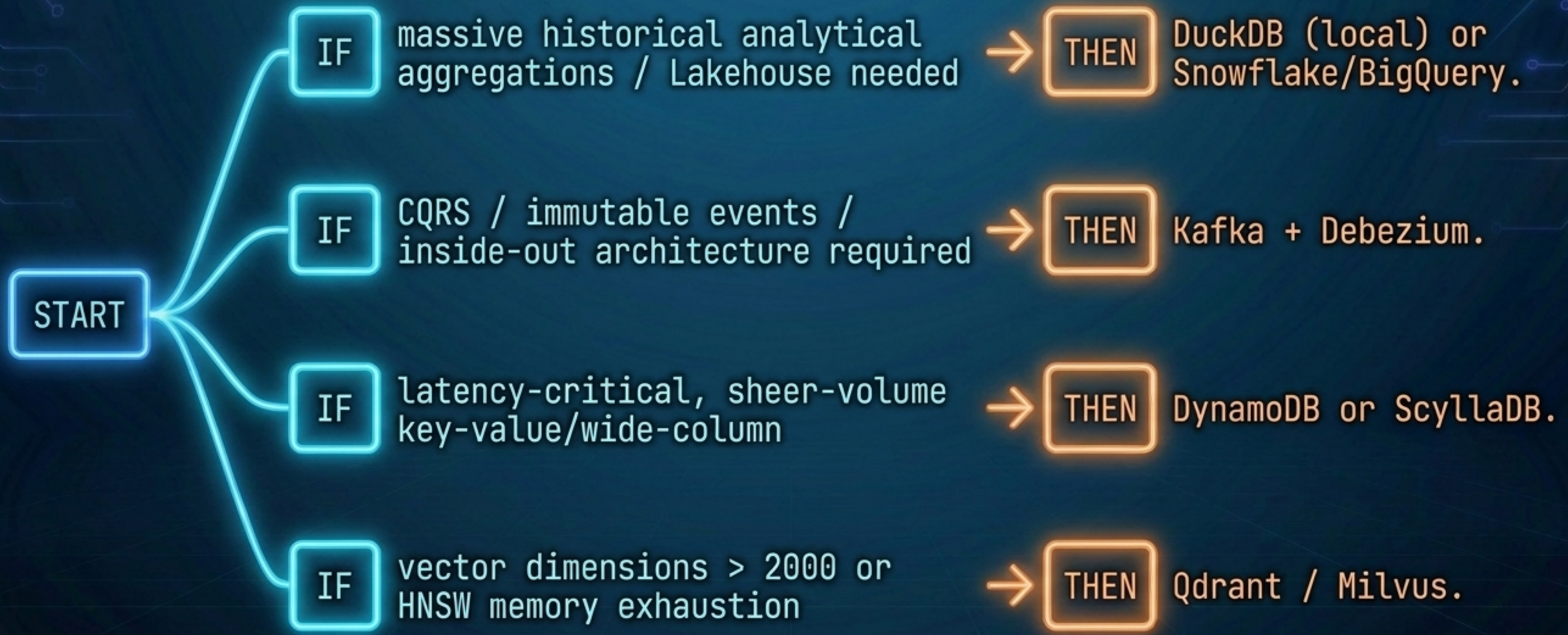
Gemini Notebook

Capstone Rubric – Stay on Postgres Until…

- [x] You approach 32 TB single-table limits or TXID wraparound risks.
- [x] AUTOVACUUM cannot physically keep up with update/delete churn.
- [x] Read/Write QPS exceeds max vertical hardware (even with PgBouncer & Replicas).
- [x] The operational tax of polyglot persistence is mathematically lower than the cost of monolithic CPU spikes.

Gemini Notebook

Capstone Rubric – Add Specialised Stores When…



Gemini Notebook

The Architect's Blueprint: Final Synthesis

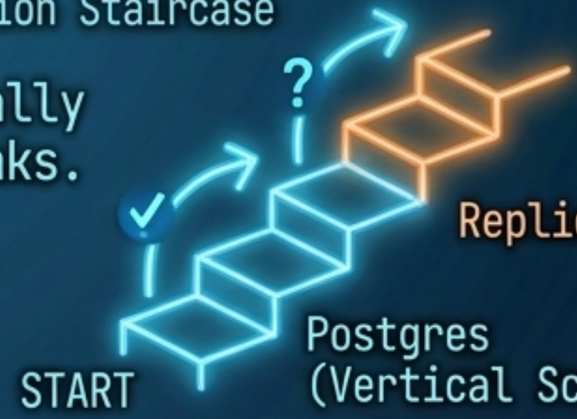
Zone 1:



Spend tokens wisely.

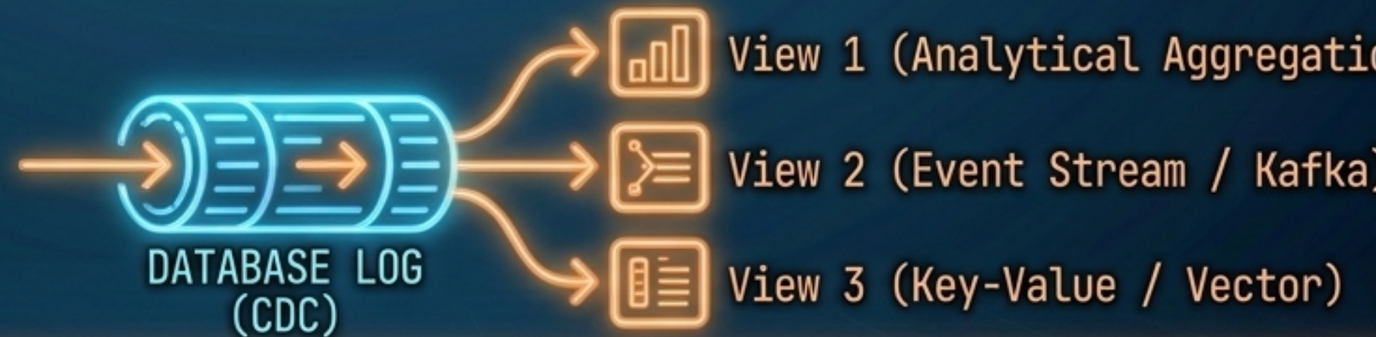
Zone 2: Escalation Staircase

Scale vertically until it breaks.



Zone 3: Inside-Out CDC Pipeline

Turn the database inside-out for safe Polyglot persistence.



Data engineering is not about finding the perfect database. It is about designing robust, boring, purpose-driven data flows that respect physical hardware limits and operational human limits. Spend tokens wisely. Scale vertically until it breaks. Turn the database inside-out for safe Polyglot persistence.

Gemini Notebook