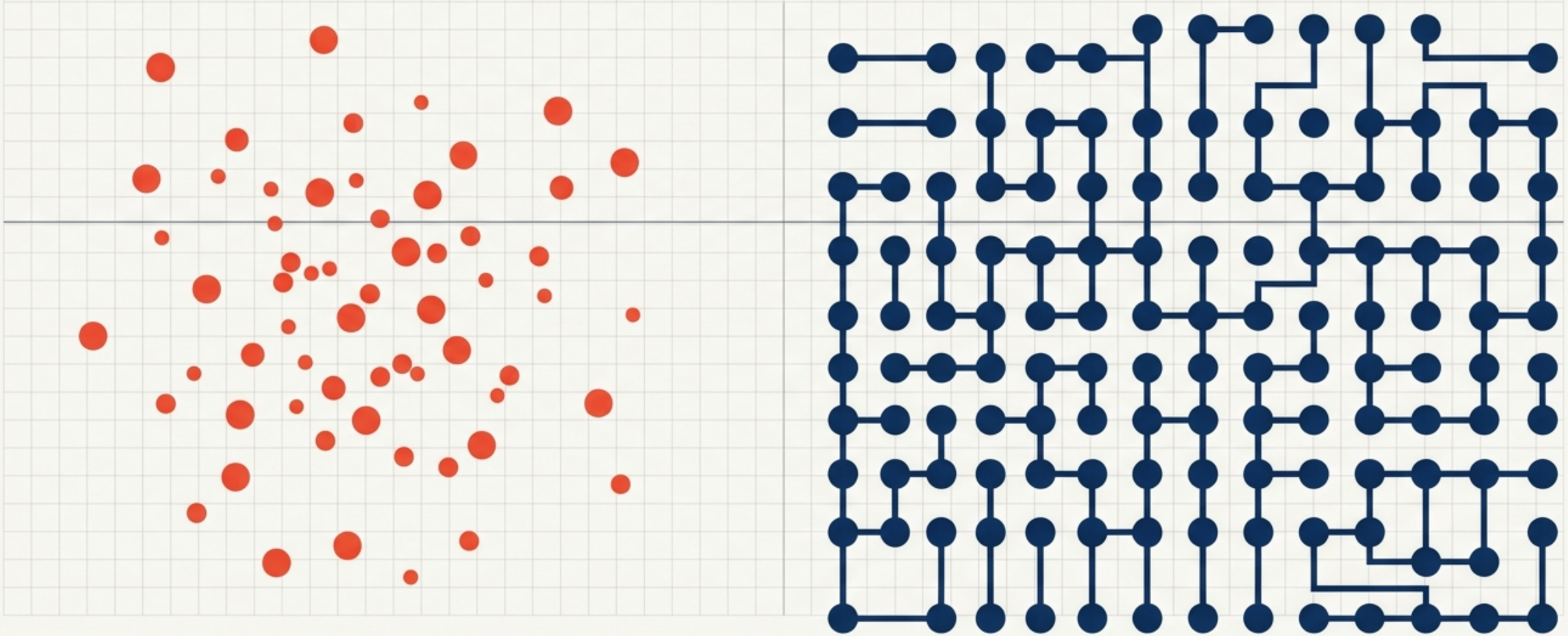
Structure Beats Search

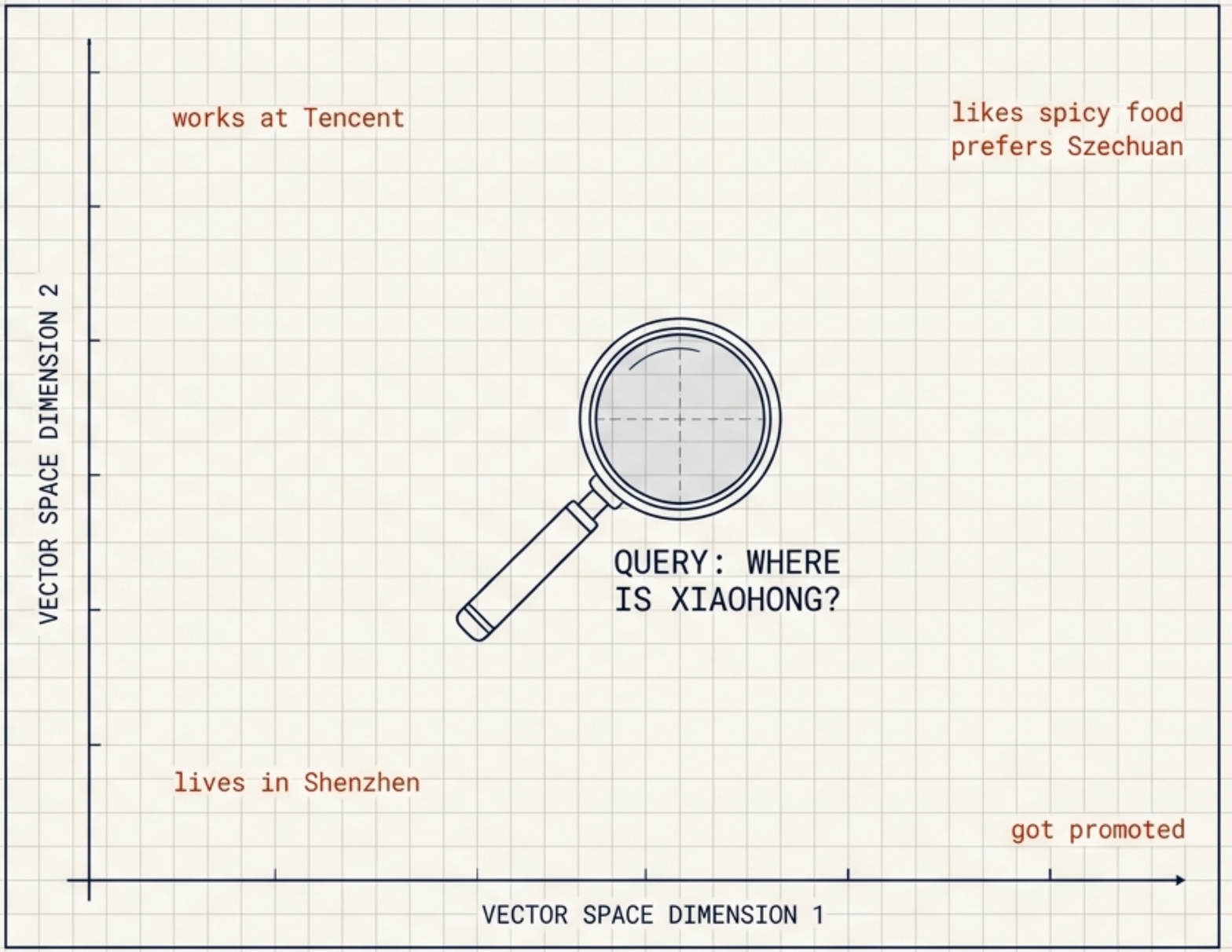
Architecting Contact-Based Memory for LLMs



The Flat Row Illusion

Mio's V1 memory system dumped everything into a single table. Each row contained content, metadata, and a vector embedding.

For preference retrieval, semantic clustering worked. But for human relationships, the model developed immediate amnesia. The structural links simply did not exist.

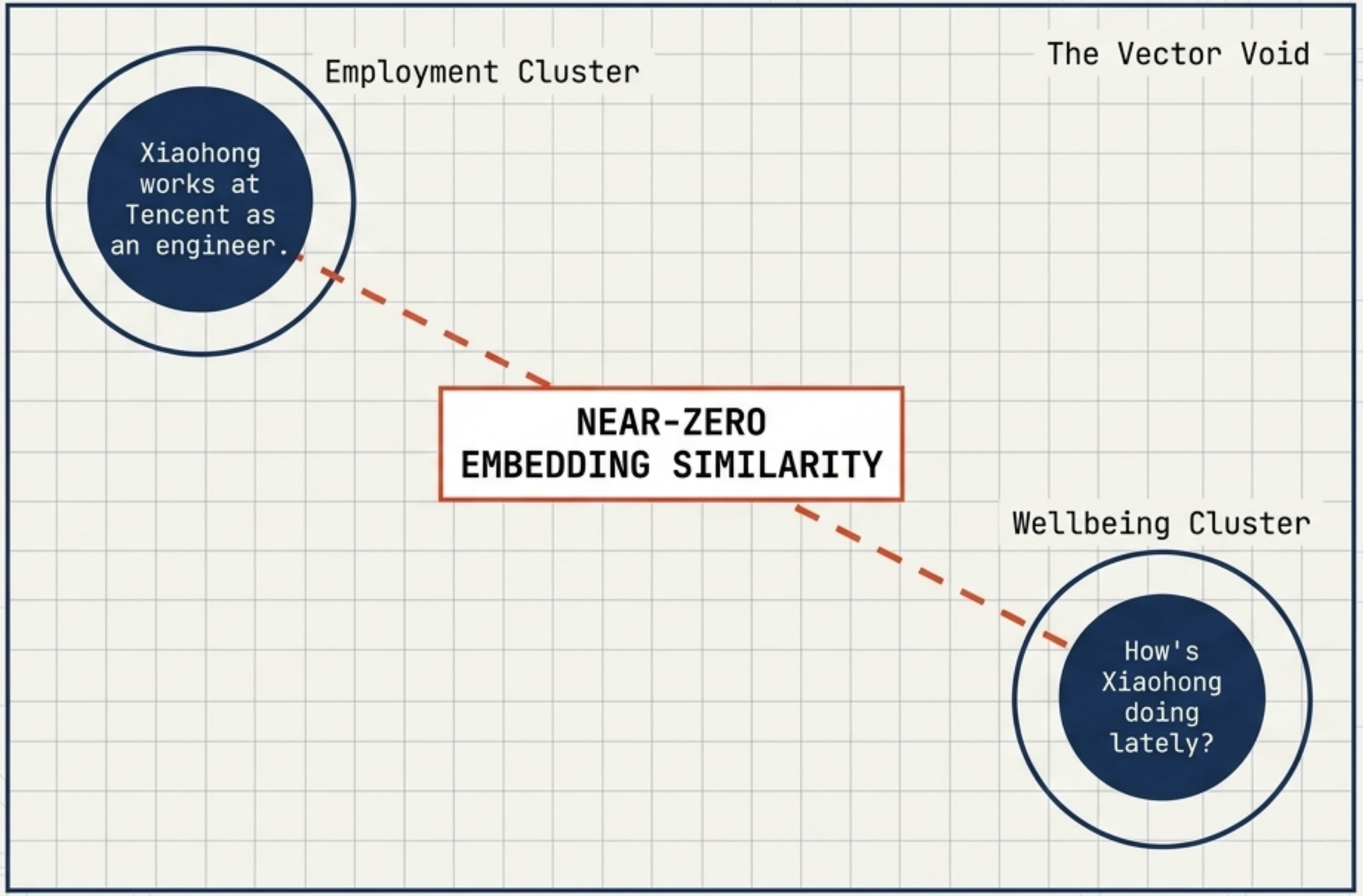


The Semantic Disconnect

Vectors group concepts, not entities. A factual statement and a relational question completely different neighborhoods of vector space.

When asked about a friend, the model literally cannot recall what it knows.

The memories exist, but they are mathematically invisible.



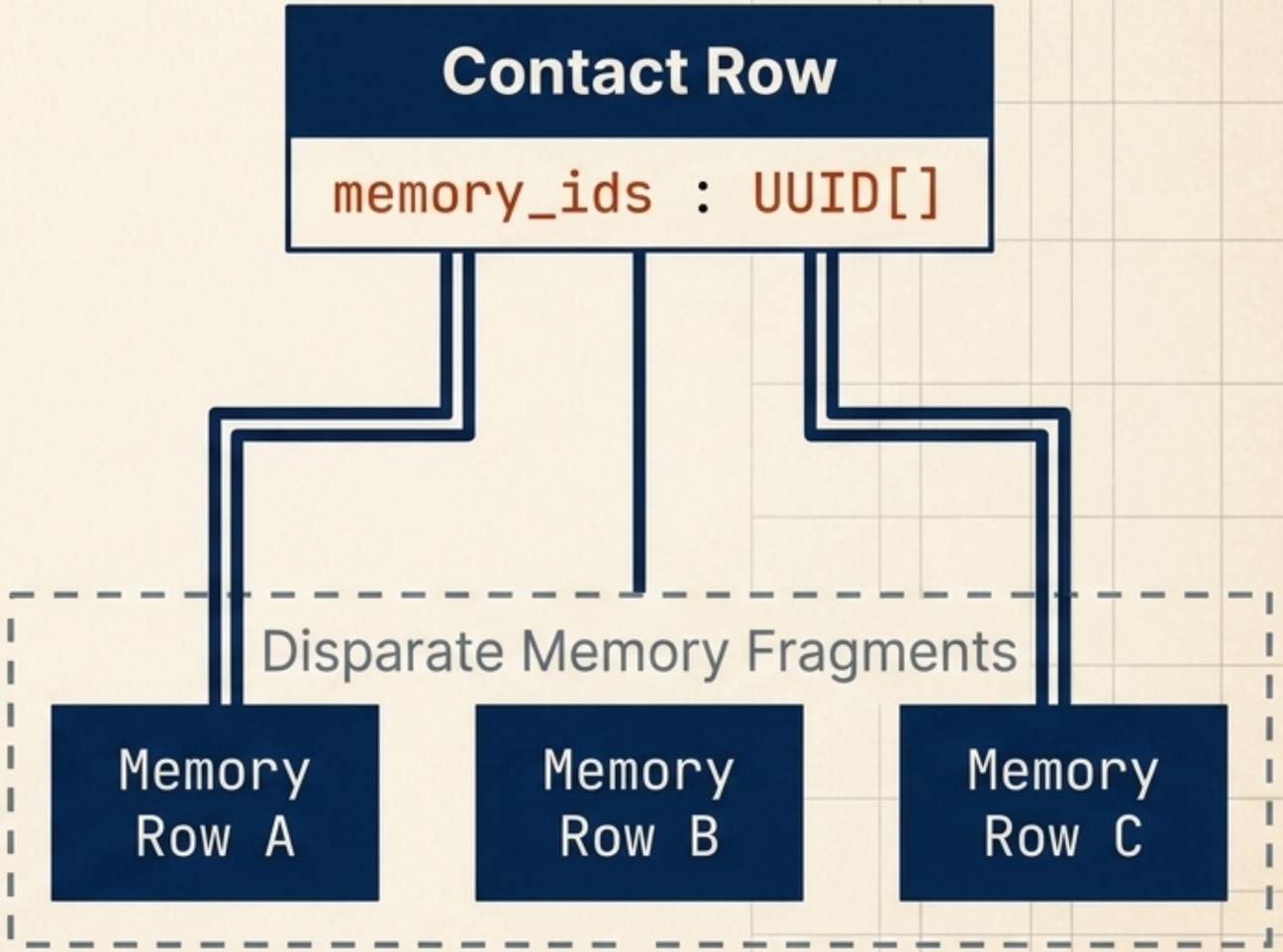
Migrating to Entity-First Architecture

Feature	V1: Flat Memory Rows	V2: First-Class Entities
Data Model	Scattered strings & embeddings	**Centralized contacts table**
Retrieval Strategy	Semantic search + ILIKE %name%	**Direct Prompt Injection**
Lookup Complexity	O(N) Full-Table Scans	**O(1) Linked UUID Lookup**
State Storage	None (Recalculated on fly)	**Synthesized attributes JSONB**

The Relational Fix

The solution relies on relational database fundamentals: a dedicated table where every person in the user's life gets their own row.

The critical component is `memory_ids`—a UUID array securely linking the contact to every disparate memory fragment. Look up the contact, fetch the IDs. No searching required.



The Anatomy of a Contact Card

Instead of raw fragments, the LLM is fed a consolidated attributes JSONB column, automatically built and refined from unstructured chats.

System Prompt Injection

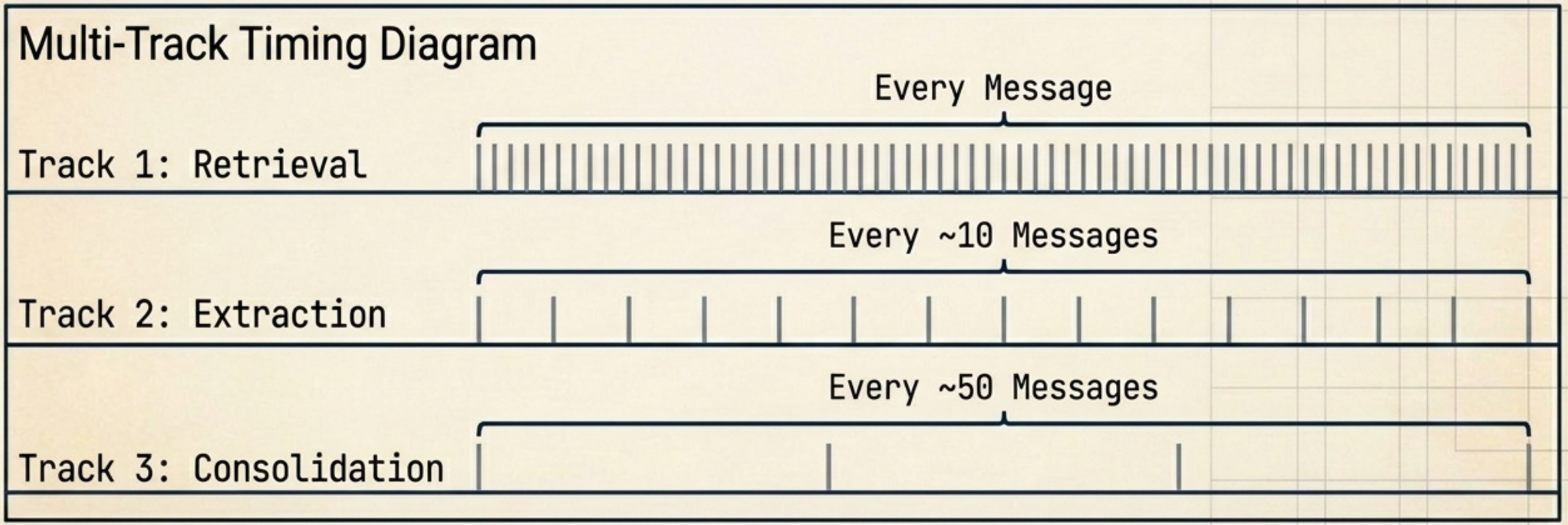
```
Name: Xiaohong

Job Title: Engineer at Tencent

Timeline:
 -  Met in college
 -  Got promoted
 -  Stressed at work lately
```

attributes (JSONB)

```
{
  "name": "Xiaohong",
  "job_title": "Engineer at Tencent",
  "timeline": [
    "Met in college",
    "Got promoted",
    "Stressed at work lately"
  ],
  "memory_ids": ["uuid-1", "uuid-2", "uuid-3"]
}
```

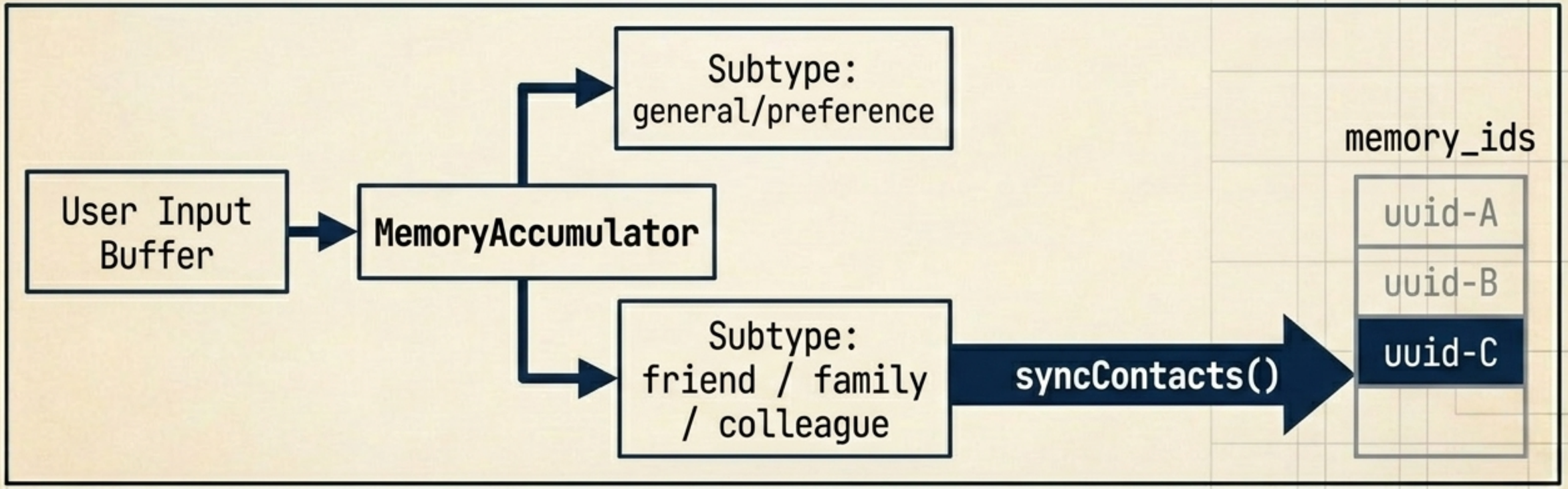
The Rhythm of Memory

Entity memory cannot be processed in real-time without destroying latency. The Mio V2 pipeline decouples processing into three asynchronous tracks, running continuously at different cadences.



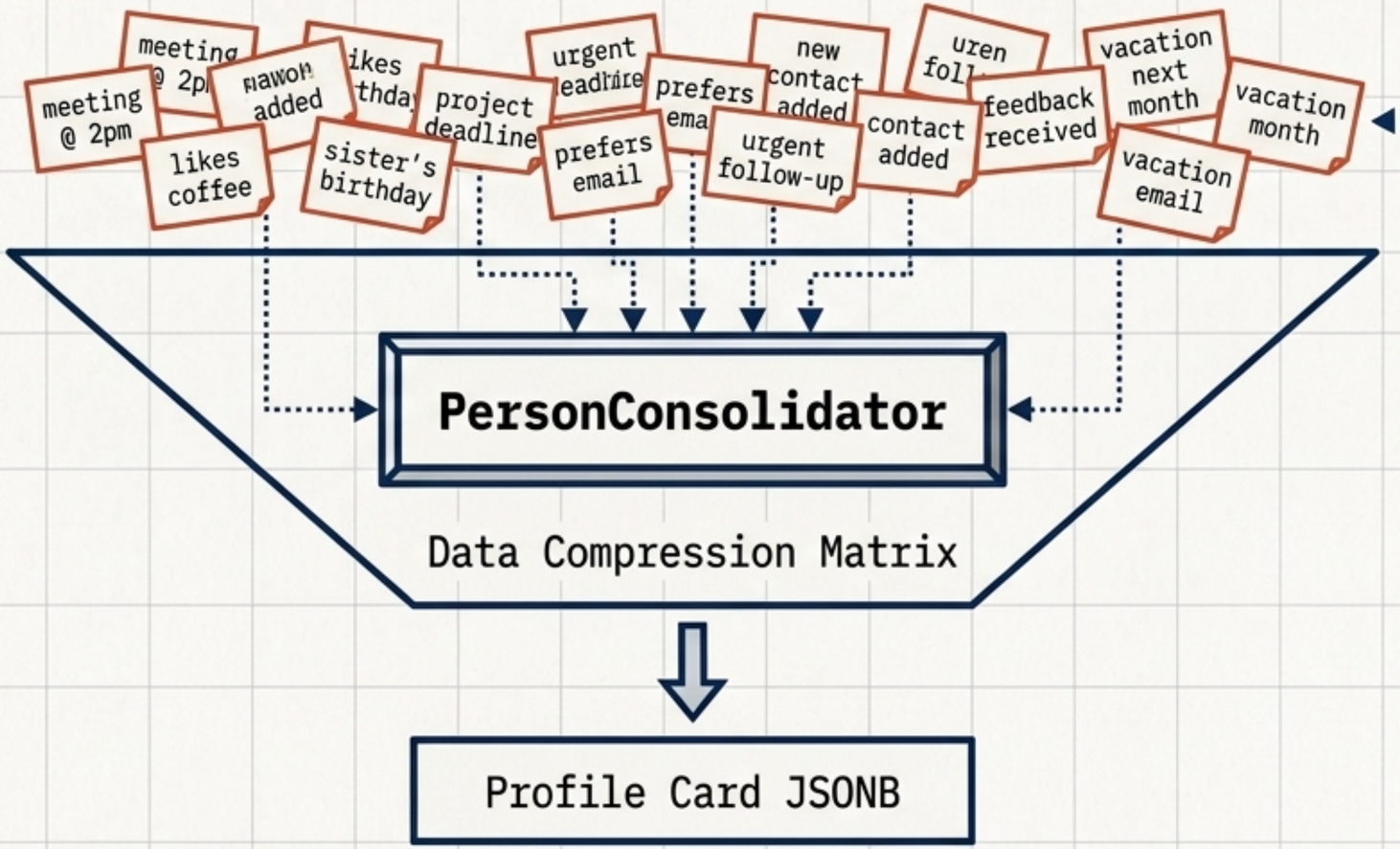
Track 1: Extraction & Linking (Every 10 Msgs)

The `MemoryAccumulator` monitors the conversation buffer. When it identifies an extractable memory with a person-related subtype (friend, family, colleague), it immediately fires **`syncContacts()`**. The contact row is upserted, and the new memory's UUID is locked into the array.



Track 2: The Consolidation Funnel (Every 50 Msgs)

The **PersonConsolidator** gathers all linked memories and commands an LLM to forge a structured profile. Source fragments have their importance scores dropped, eliminating context window clutter during retrieval.



IMPORTANCE SCORING

RAW FRAGMENTS (HIGH)

meeting @ 2pm [Score: 0.92]
likes coffee [Score: 0.85]
project deadline [Score: 0.88]
urgent follow-up [Score: 0.90]

CONSOLIDATED (LOW)

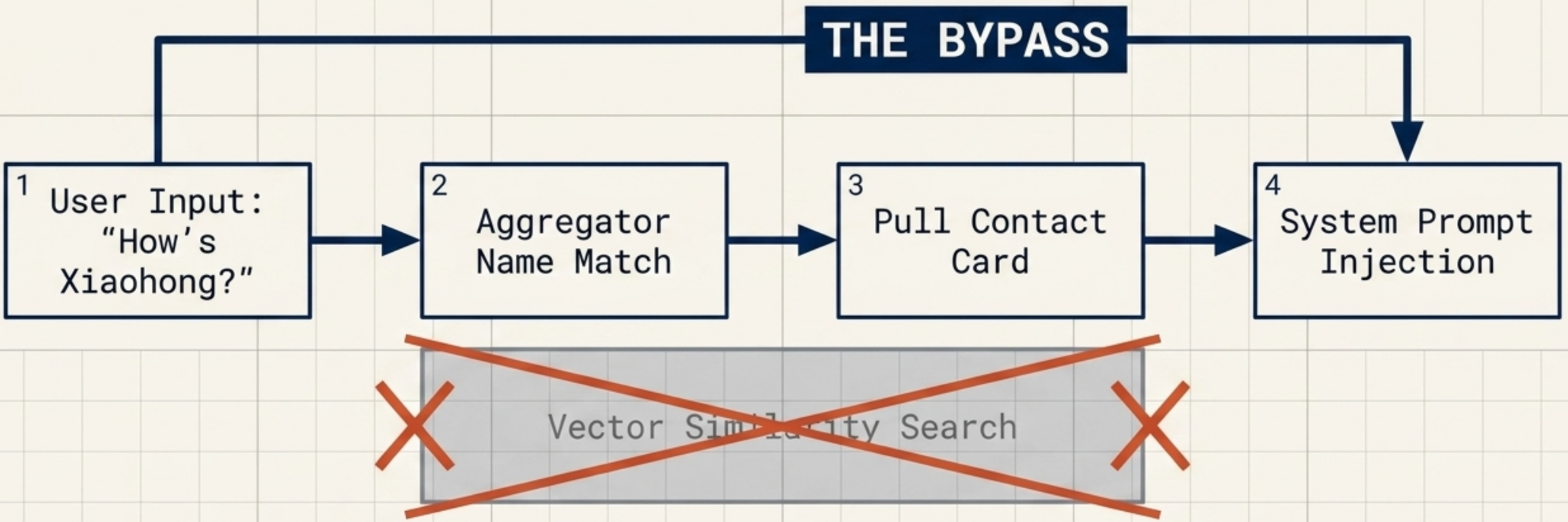
meeting @ 2pm [Score: ~~0.92~~]
likes coffee [Score: ~~0.85~~]
project deadline [Score: ~~0.88~~]
urgent follow-up [Score: ~~0.90~~]

0.05

Importance Score Floor Applied

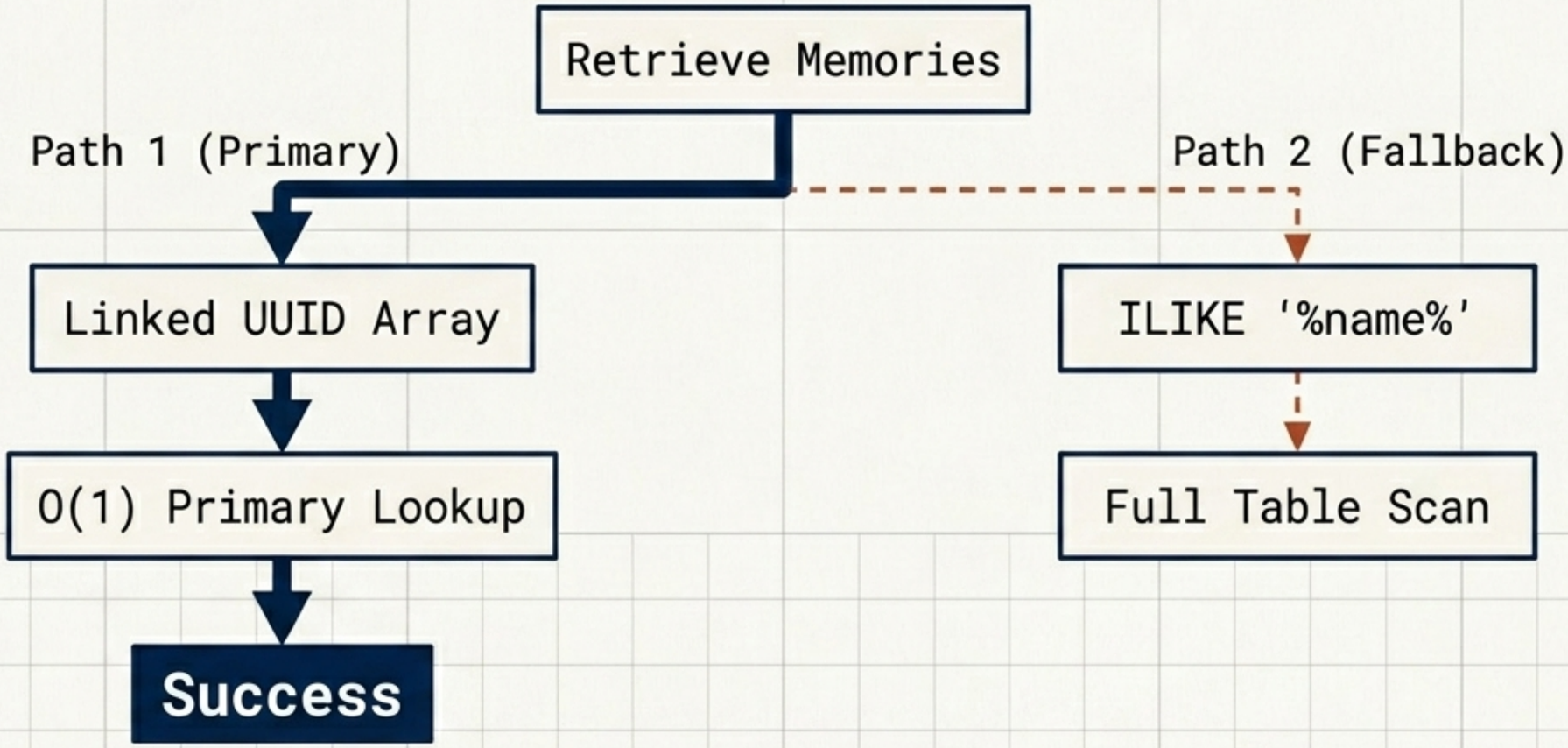
Track 3: Direct Prompt Injection (Every Msg)

If a contact name is matched, the system pulls the pre-compiled contact card and injects it straight into the system prompt. The modelinstantly possesses full context. No embedding search is invoked.



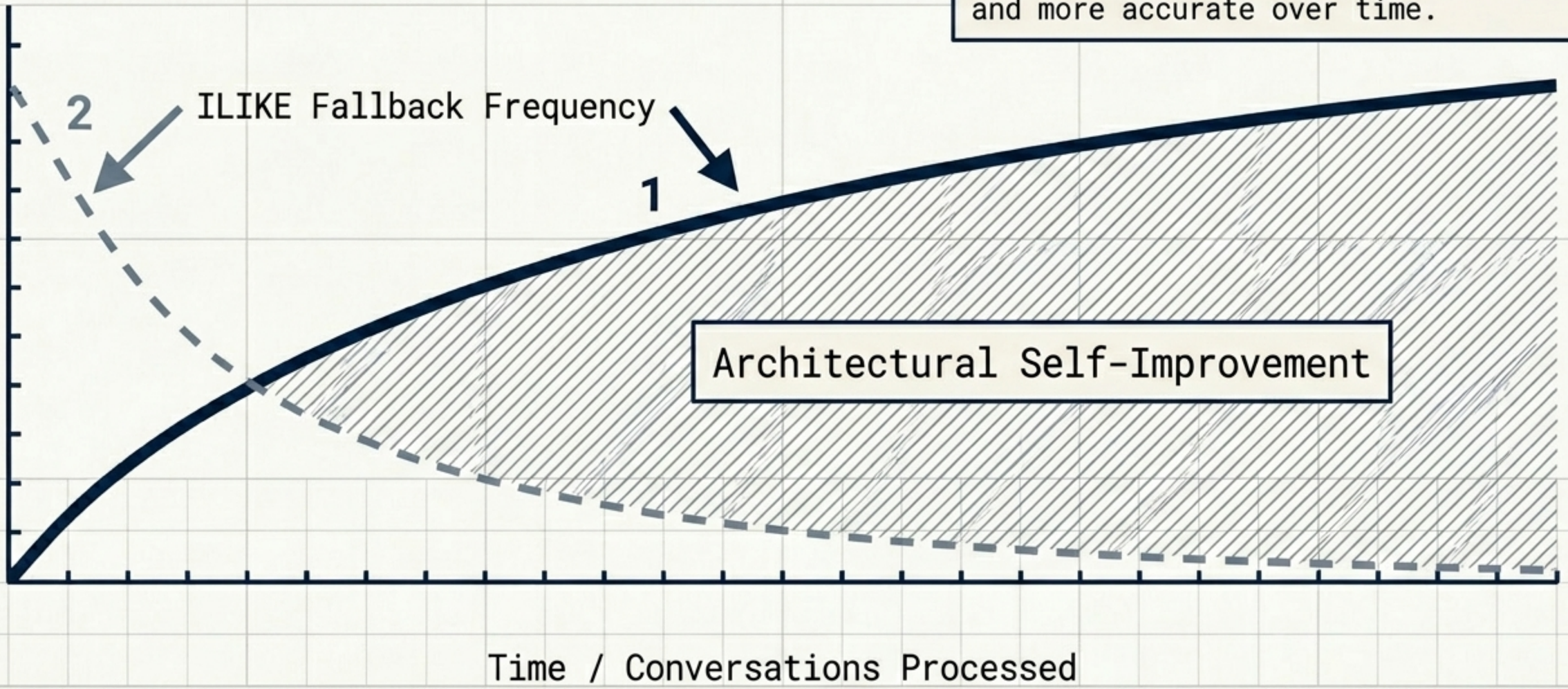
Two-Phase Retrieval Strategy

Because syncs occasionally fail silently, and legacy memories existed before contacts were mapped, retrieval requires a text fallback strategy.



Systemic Self-Improvement

As more conversations happen, the MemoryAccumulator retroactively links more fragments. The slow ILIKE fallback fires less and less. The architecture organically gets faster and more accurate over time.



The Engineering Reality: Production Scars

⚠️ Symptom	Root Cause	✓ The Fix
Wrong memories linked (Mom's bday attached to coworker)	Array index misalignment. storedIds mixed ADD/UPDATE operations, corrupting batch order.	Query DB for direct content→ID mappings instead of trusting array positions.
System crashes on new contacts	Postgres Type Error: Passing empty '{}' into ANY() fails on certain query plans.	Split linked-ID paths and fallback paths into separate, isolated queries.

The Engineering Reality: Edge Cases

⚠️ Symptom	Root Cause	✓ The Fix
Raw fragments still appearing in Top-K retrieval	Importance floor of GREATEST(0.1, imp * 0.5) was too high.	Dropped the consolidation floor dramatically to 0.05.
Garbage text in Chinese logs	CJK Truncation. .slice(0, 100) splits multi-byte characters in half.	Spread into an array of code points first: [...text].slice(0, 100).join('')

What’s Still Broken

Pronoun Amnesia

她

How's 'she' doing? breaks the aggregator. If a name isn't explicitly used, the card isn't injected. Memory search alone cannot resolve pronouns.

Rigid String Matching

小红 != 红红

Zero fuzzy matching. If a user types 'Xiaohong' today and and 'Honghong' tomorrow, the system spawns two isolated entities.

PostgreSQL Limits

ts_vector
vs
pg_jieba

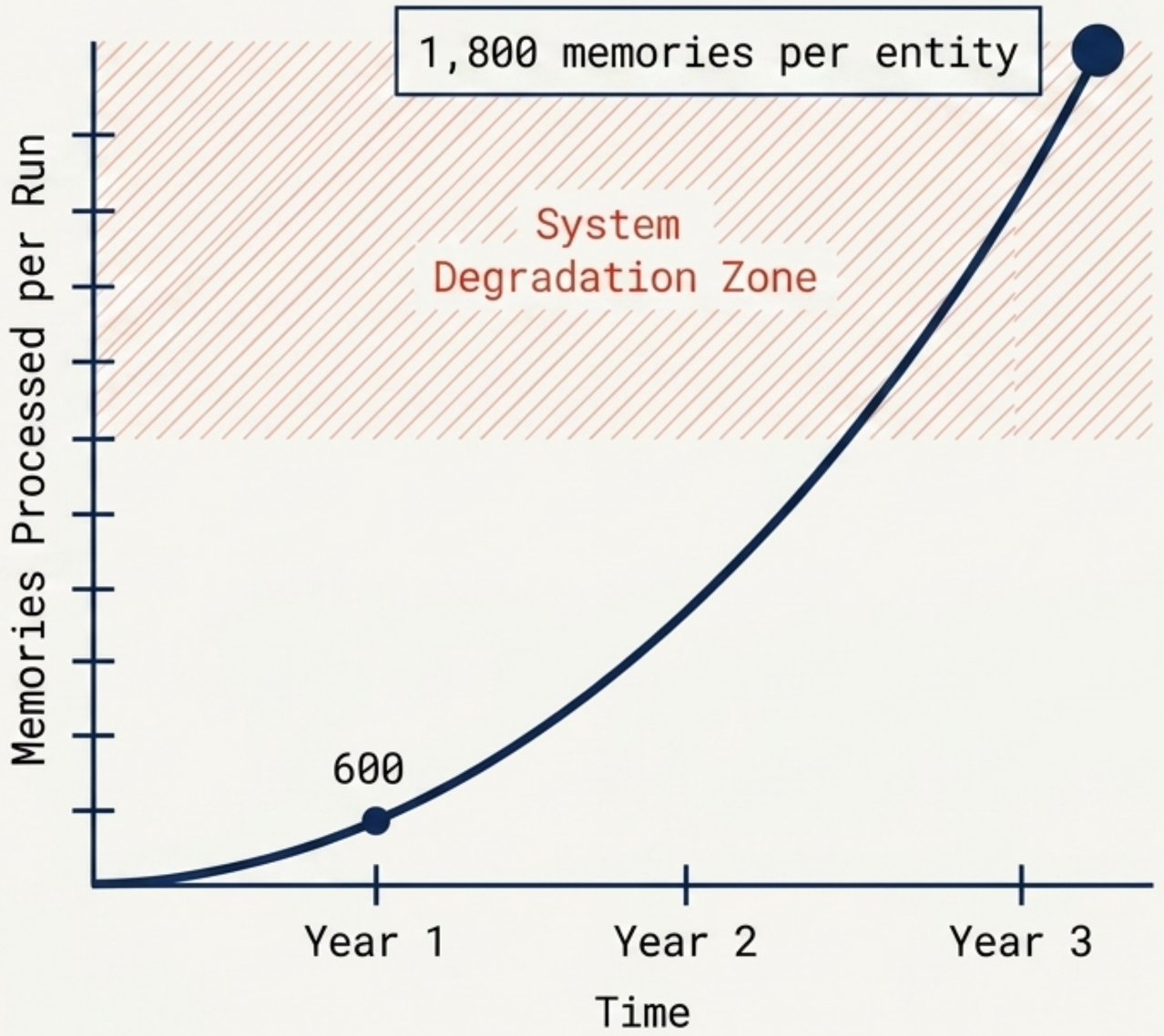
Built-in word segmentation for Chinese is functionally useless without heavy external plugins.

The $O(N^2)$ Time Bomb

Currently, the Consolidator re-reads every linked memory for a contact, every time it runs.

At 20 messages per day, a user generates ~600 memories a year.

Survival requires building incremental consolidation.



Vectors provide proximity. Only graphs provide relationships.

Semantic search is a powerful tool for discovering general concepts, but human memory is inherently relational. To cure LLM amnesia for the people who matter most, we must shift from calculating distance to engineering structure.