

CivicSense

AI on Data in Motion — Turning live city data streams
into human-friendly guidance, as events happen.



Confluent Cloud

Gemini



MongoDB Atlas

City Alerts are Fragmented, Delayed, and Confusing

CHAOS



Emergency Alert:

Emergency Alert: Code Red
for Sector 7.



Transit Notification:

Service Change: Line B
experiencing signal problems.



City Update:

Public Works Notice: Paving
on Main St.

ANXIETY



The Parent:

Is it safe to send my children to
school today?



The Commuter:

My train is stopped. How do I get
to work on time?



The Senior:

What does a 'Code Red' alert
mean for me?

The problem isn't a lack of data; it's a lack of real-time interpretation.

Our Cities are Overwhelmed by Disconnected, Delayed Signals

Every day, thousands of critical alerts are generated across our cities. But for citizens, this information is often fragmented, generic, confusing, and arrives too late.

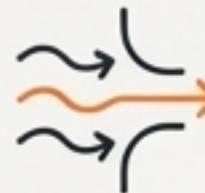
- **Fragmented:** Alerts are spread across countless disconnected systems.
- **Impersonal:** Generic warnings lack personal relevance or clear actions.
- **Confusing:** Technical jargon is inaccessible to vulnerable populations.
- **Delayed:** Batch processing means information is often outdated on arrival.



The result: Parents struggle with safety decisions, commuters are stranded, and seniors miss vital warnings.

The Solution is a Central Nervous System for the City

CivicSense ingests **every live event stream**, processes it in real-time, and delivers clear, actionable guidance to the people who need it most. We transform data chaos into civic clarity.



Streaming Platform:
Confluent Cloud Kafka



Real-Time Processing:
Apache Flink SQL



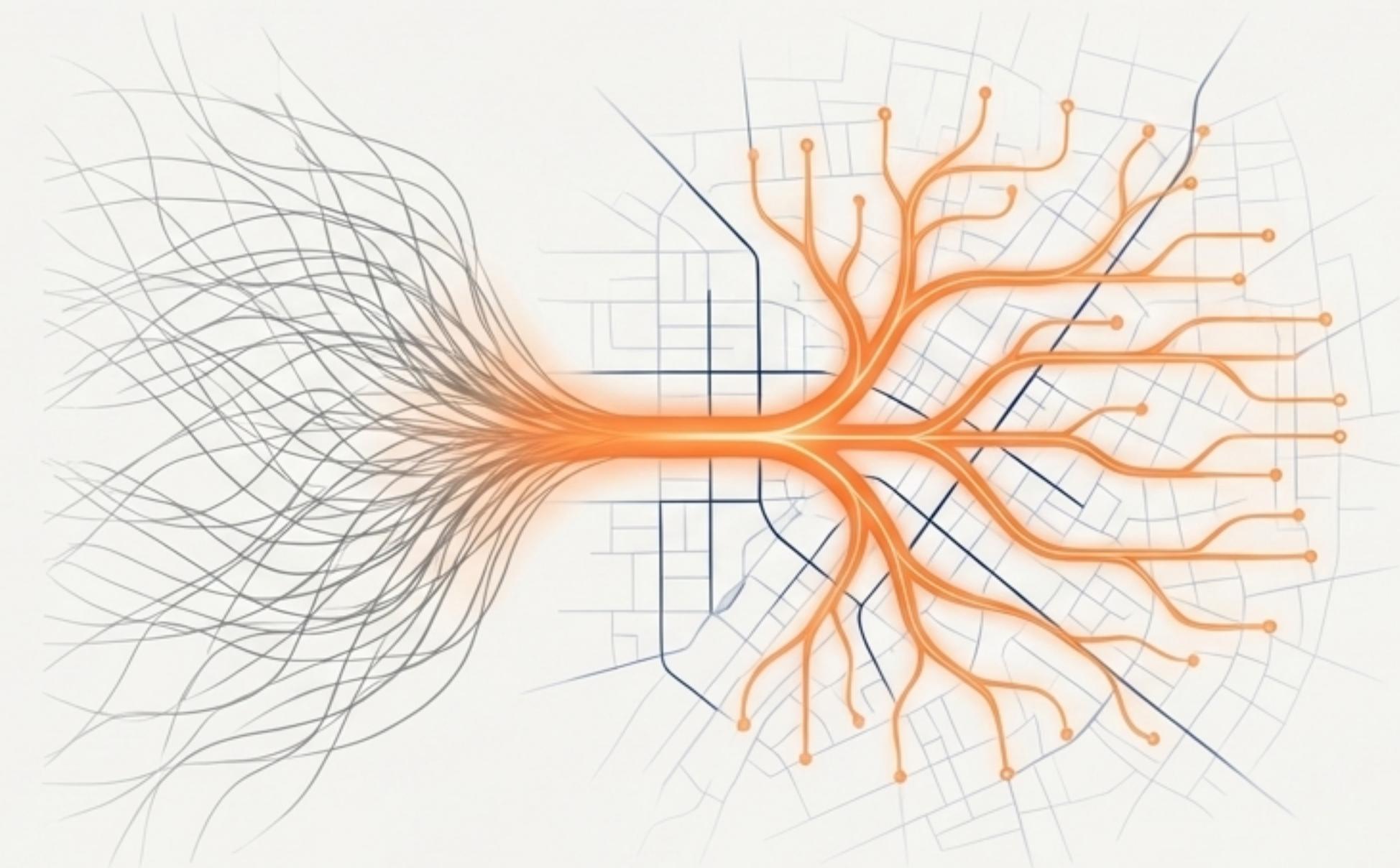
Multi-Agent Reasoning:
Google Gemini AI



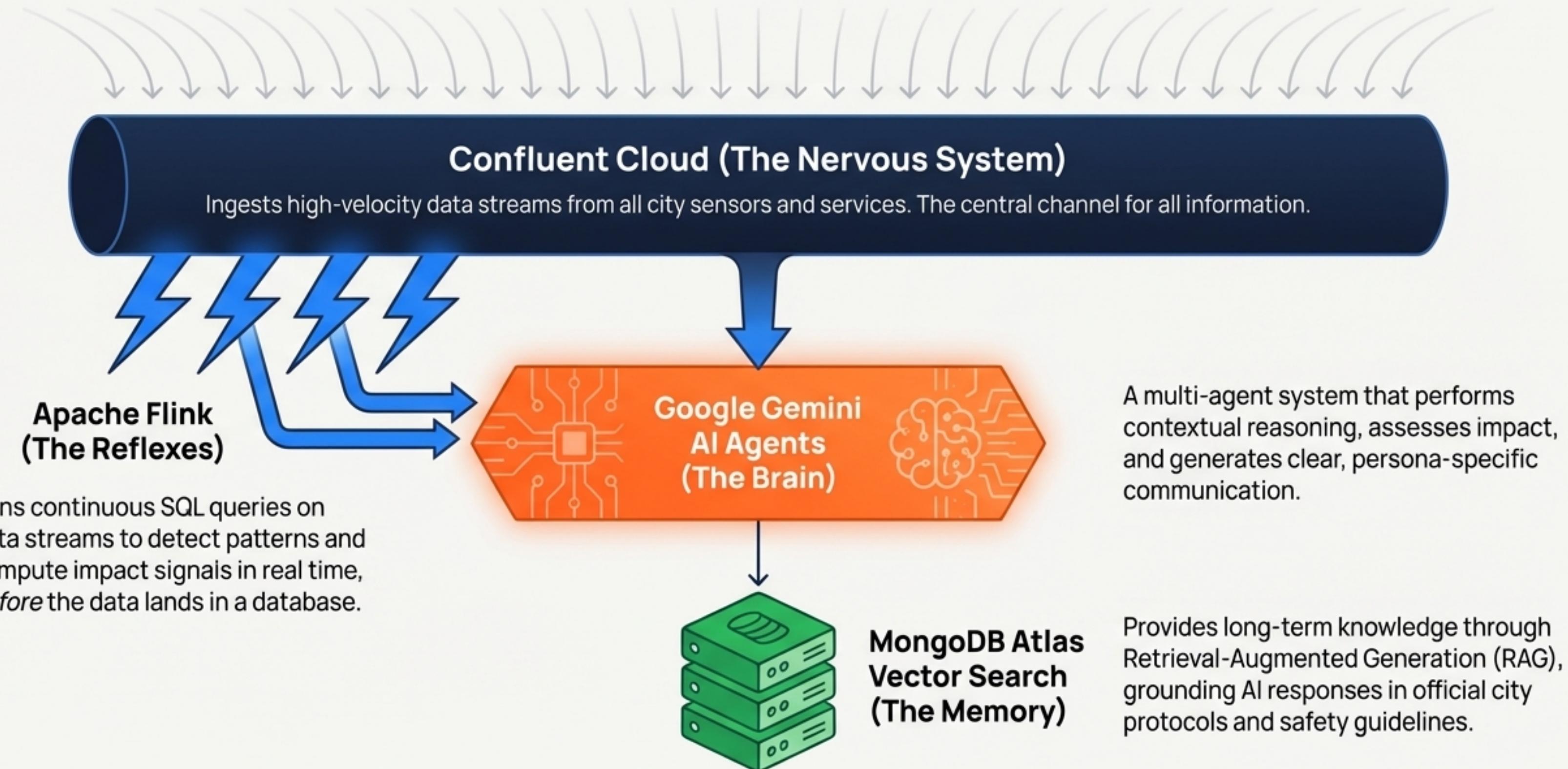
Contextual Memory:
MongoDB Atlas Vector Search



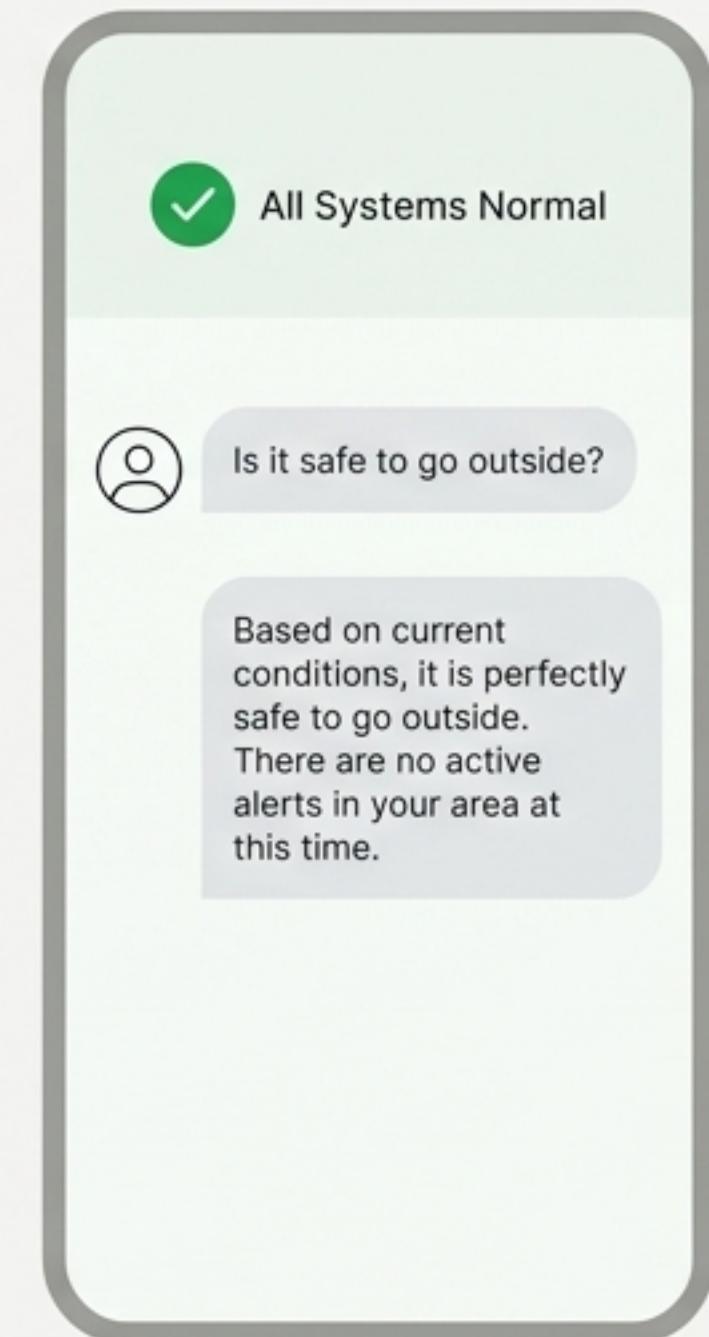
Instant Delivery:
WebSocket Protocol



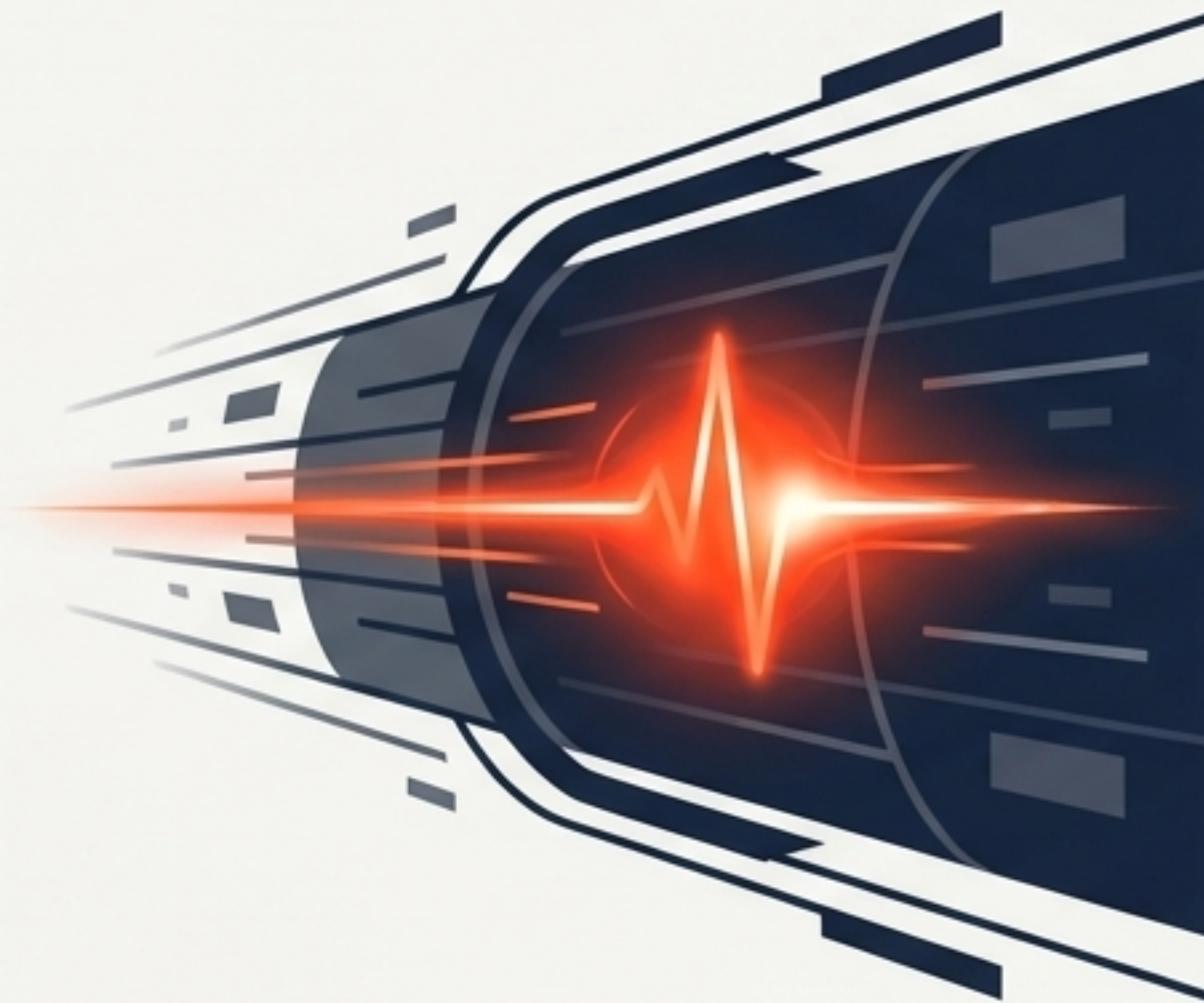
An Architecture Modelled on a Central Nervous System



The ‘WOW’ Moment, Part 1: A Question in a Calm City



The ‘WOW’ Moment, Part 2: A Critical Event Streams In

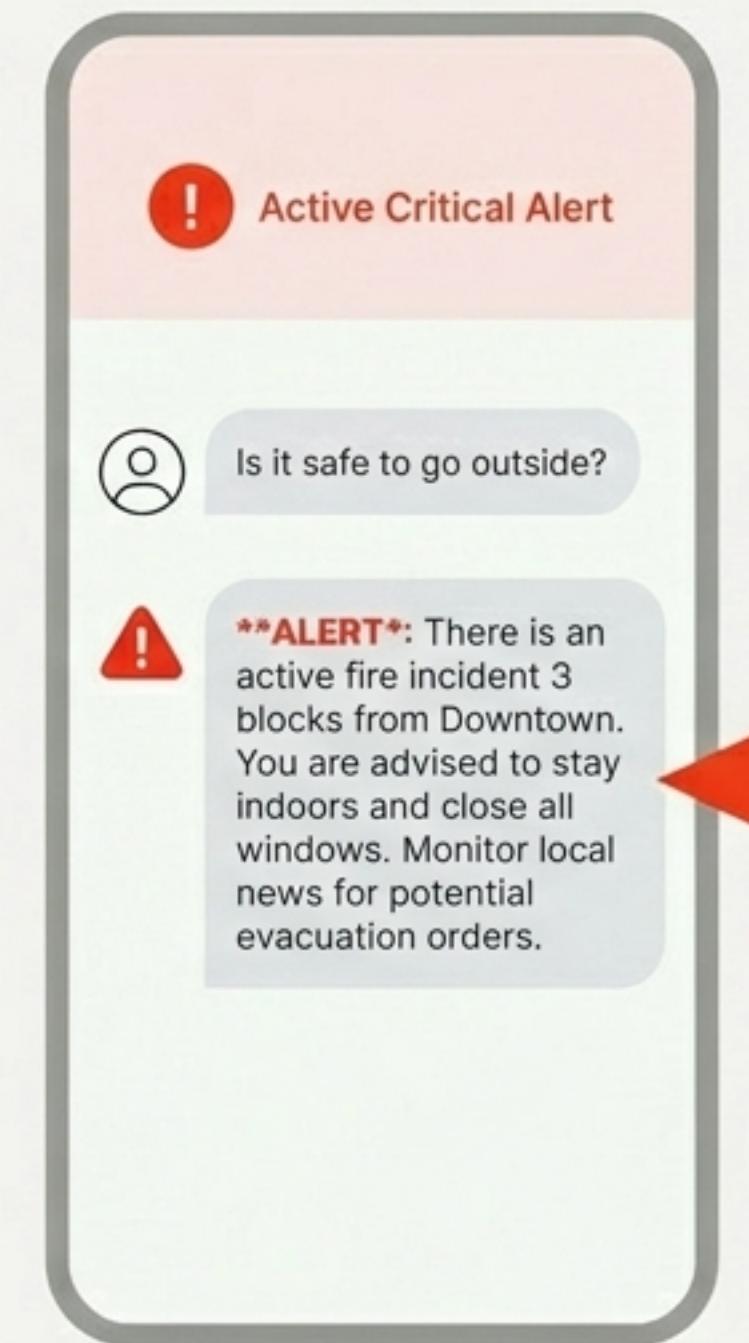


emergency_events

```
{  
  "event_id": "F789-2024",  
  "event_type": "fire",  
  "location": "Downtown",  
  "severity": "critical",  
  "source": "911_dispatch"  
}
```

A new event hits the Kafka topic. Flink processes it. The AI is now aware.

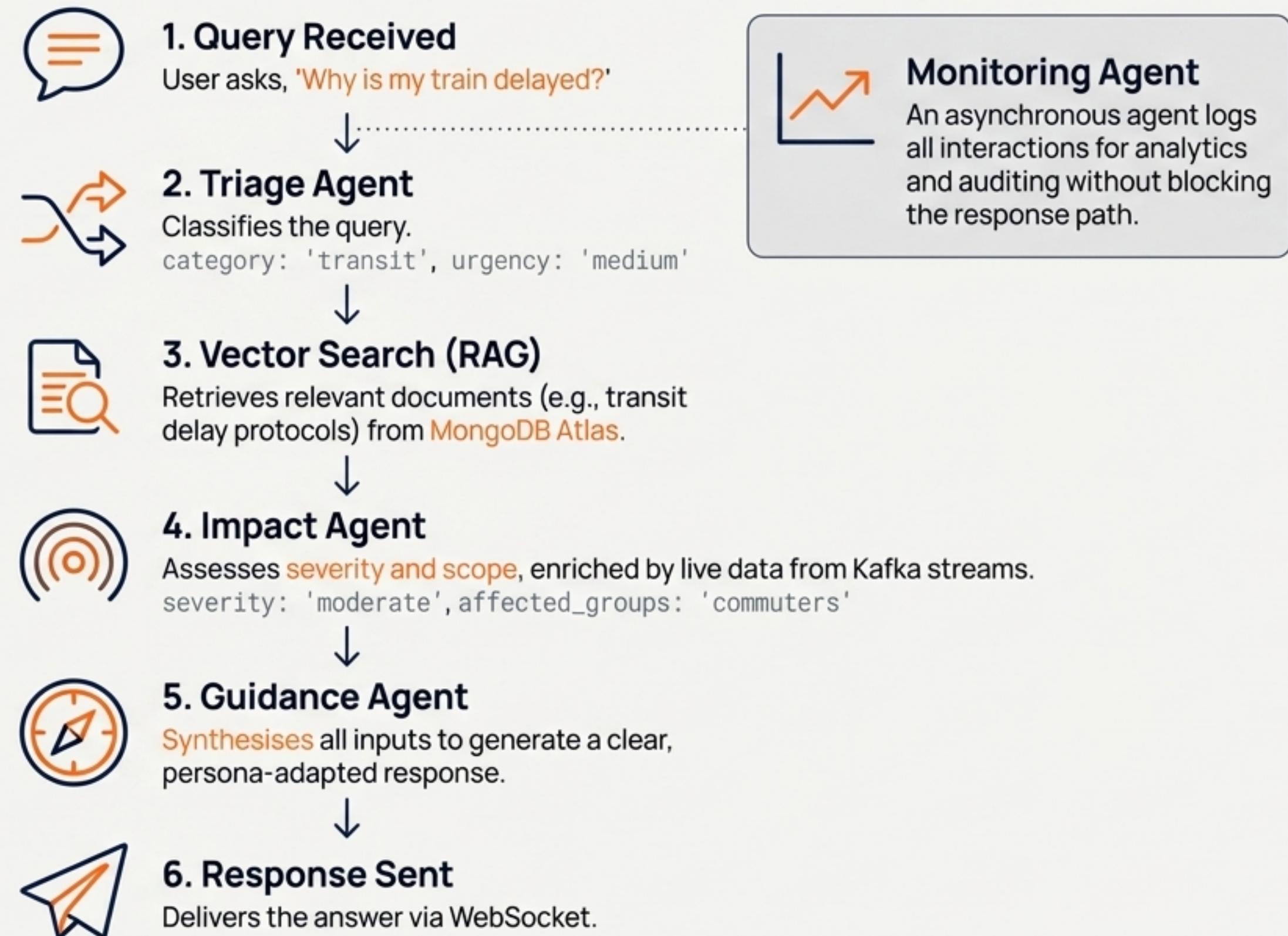
The ‘WOW’ Moment, Part 3: The Same Question, A New Reality



This is AI reacting to data in motion.

End-to-end latency from event to notification:
< 5 seconds.

Inside the Brain: A Multi-Agent AI Processing Pipeline



The Reflexes: Flink SQL Detects Patterns in the Stream

```
1 SELECT
2   TUMBLE_START(event_ts, INTERVAL '5'
3 MINUTE) AS window_start,
4   area,
5   COUNT(*) AS event_count,
6   MAX(severity) AS max_severity
7 FROM civic_events
8 GROUP BY
9   TUMBLE(event_ts, INTERVAL '5'
10 MINUTE),
11   area;
```



True Stream Processing

We run continuous queries on live Kafka streams, not periodic batch jobs.



Real-Time Aggregation

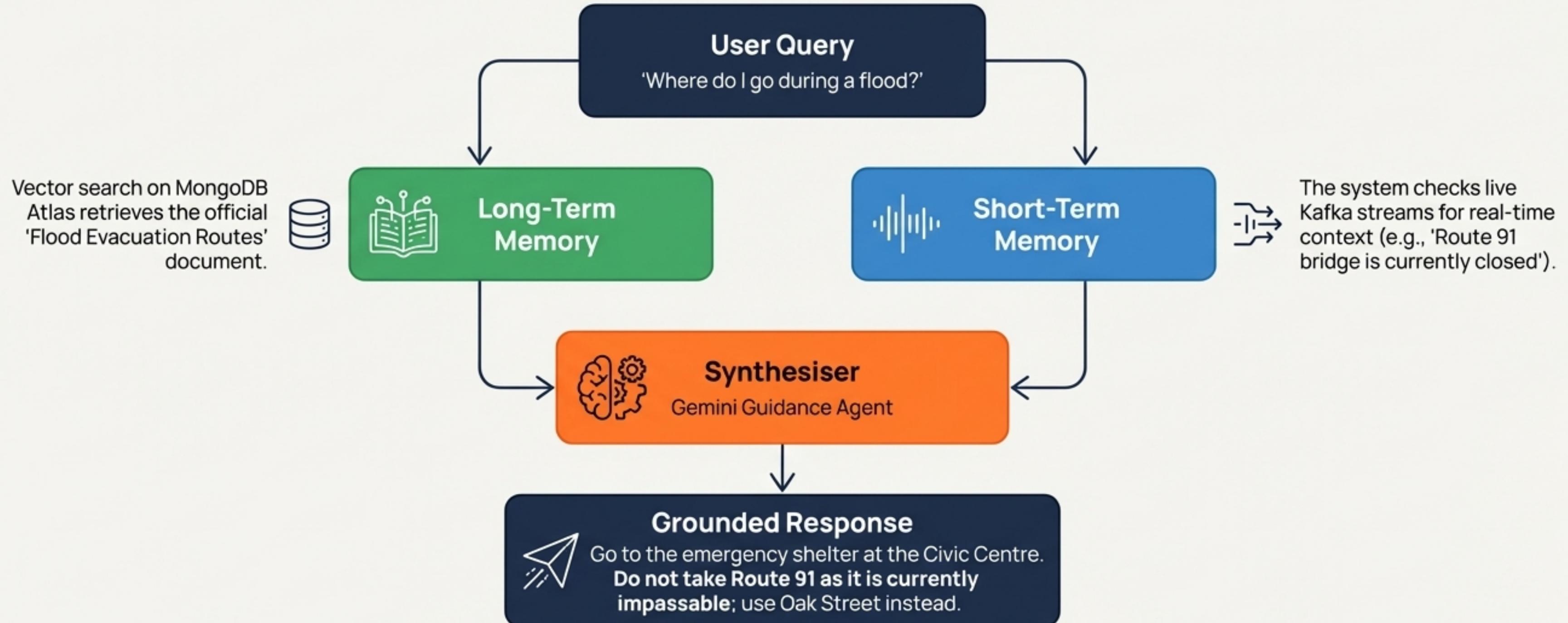
5-minute tumbling windows aggregate thousands of raw events into meaningful 'Severity Signals'.



Proactive Detection

This allows us to identify event clusters and severity escalations instantly, powering proactive alerts.

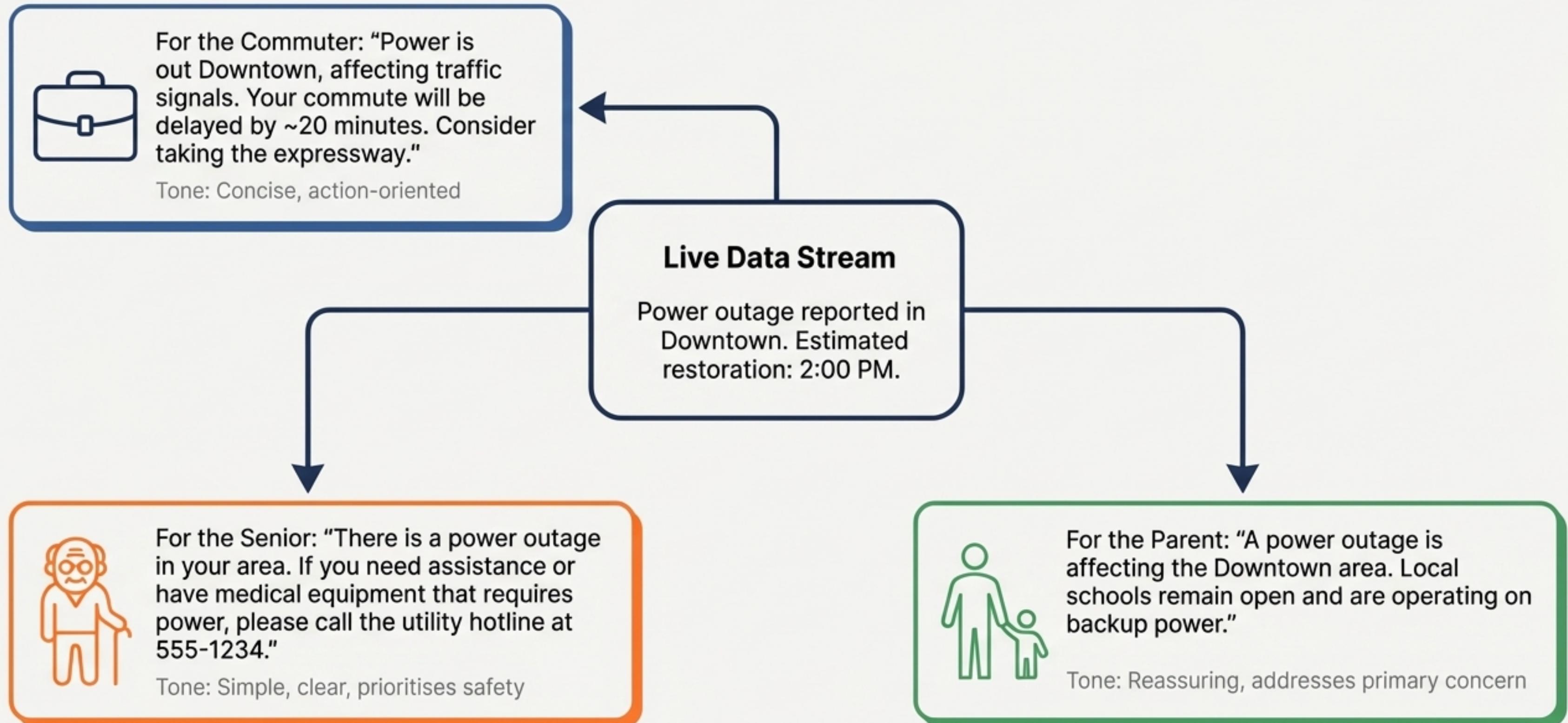
The Memory: Grounding AI in Truth with RAG on Streaming Data



Key Insight

Traditional RAG uses static knowledge. We innovate by augmenting it with real-time context from data in motion, ensuring guidance is always relevant.

Designed for Everyone: Guidance Tailored to Your Needs



Engineered for Production, Not Just a Prototype



Scalability

Built on a serverless, cloud-native stack (GCP Cloud Run, Confluent Cloud, MongoDB Atlas) that auto-scales from zero to millions of users.



Performance

Key performance metrics:

- < 5s Event-to-Notification Latency
- ~2.5s Average Query Response Time
- 1000+ Events/Second Processed



Reliability

Comprehensive error handling, graceful degradation, and automatic WebSocket/Kafka reconnection logic ensure system uptime and availability.



Code Quality

2,000+ lines of production-quality Python and TypeScript, fully containerised with Docker, and supported by 40+ pages of documentation.

Why CivicSense Wins: A Synthesis of Innovation and Impact



AI on Data in Motion

Our entire architecture—from Kafka ingestion and Flink processing to the AI agents and WebSocket delivery—is a live demonstration of this core theme. We don't simulate; we stream.



Novel Technical Innovation

We introduce a Multi-Agent Streaming Architecture and pioneer RAG on Streaming Data, a significant step beyond standard implementations.



Broad Real-World Impact

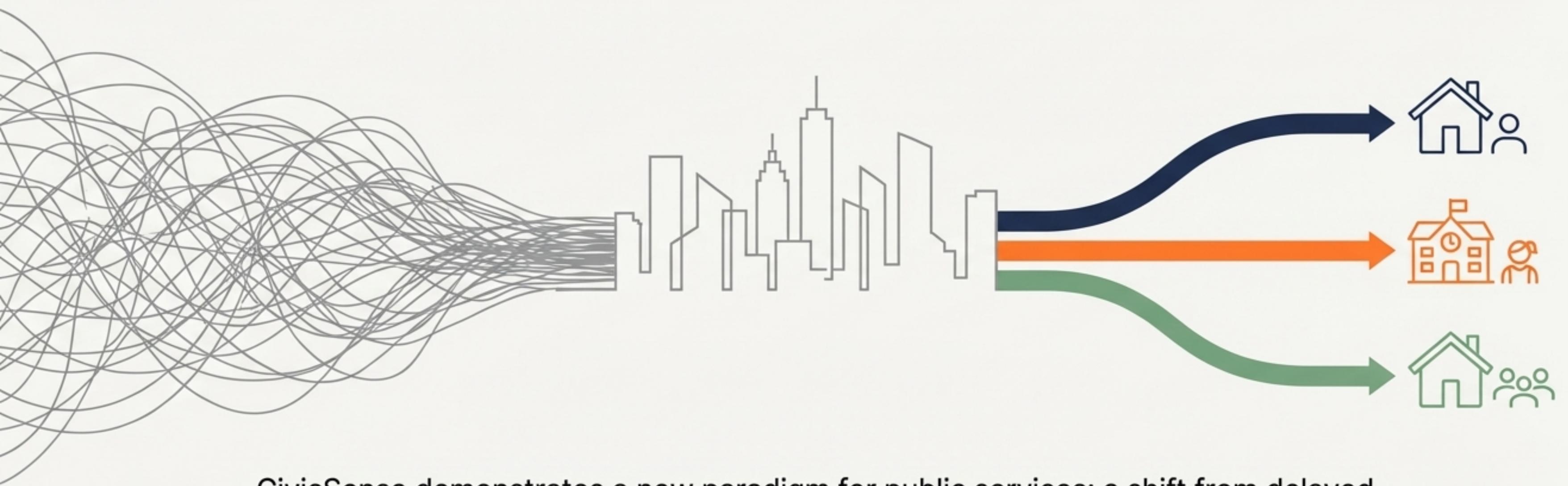
This is a scalable 'public good' platform. It solves a universal problem for entire populations, with a focus on helping the most vulnerable community members.



Production-Ready Implementation

This is not a concept. It is a fully documented, tested, and deployable system built with enterprise-grade practices, ready to create value immediately.

“The Future of Civic Intelligence is Streaming”



CivicSense demonstrates a new paradigm for public services: a shift from delayed, generic broadcasts to instant, intelligent, and personalised guidance.

By placing AI directly on data in motion, we can build safer, smarter, and more resilient cities for everyone.