Blog Post 4: The Nervous System – Workflows and Real-Time Kafka Streaming

From Static Storage to Dynamic Flows Static databases are insufficient for real-time crisis management. PANTHEON utilizes Apache Kafka to create an event-driven "nervous system" that connects data storage to processing components.

The Workflow Architecture All scenarios follow a standardized five-layer workflow:

1. UI Trigger: A user initiates a scenario.
2. Simulation Processing: Core engines (e.g., Fire Simulator) ingest data from the repository.
3. Kafka Messaging: Results are published to specific topics.
4. Decision Support: Downstream components (e.g., Routing Engine) consume messages instantly.
5. Output: Final actionable intelligence is stored and visualized.

Unified Topic Naming Convention To manage complexity, we enforce a strict naming schema for Kafka topics: `ds-{city}-{scenario}-{component}`.

- `ds-ath-b-fire-propagations`: Carries 15-minute fire spread updates (Processing time: <10s).
- `ds-vie-b-smoke-propagations`: Carries gas plume dispersion data (Processing time: ~1s).
- `ds-ath-a-blocked-roads`: Carries vectors of debris-blocked streets.

Interdependencies & Synchronization The repository manages complex dependencies. For example, in the Wildfire scenario, the Self-Adaptation/Traffic component listens to both the `fire-propagations` topic and the `uav-monitoring` topic. If the UAV detects fire crossing a road, the Traffic component instantly recalculates evacuation routes, updating the `weighted-routes` dataset in real-time.