PROJECT SYNAPSE

Agentic Last-Mile Coordinator

A solution proposal by

Team LangLow

Devansh Varshney

Saarthak Sabharwal

Vansh Agarwal

THE CHALLENGE

Last-mile delivery faces unpredictable real-time disruptions

Sudden road closures & traffic congestion

Incorrect addresses & unavailable recipients

Merchant delays & inventory issues

Dynamic scheduling conflicts

Traditional rule-based systems cannot handle these complex scenarios requiring human-like reasoning

OUR SOLUTION

MAIN ORCHESTRATING AGENT (LLM)

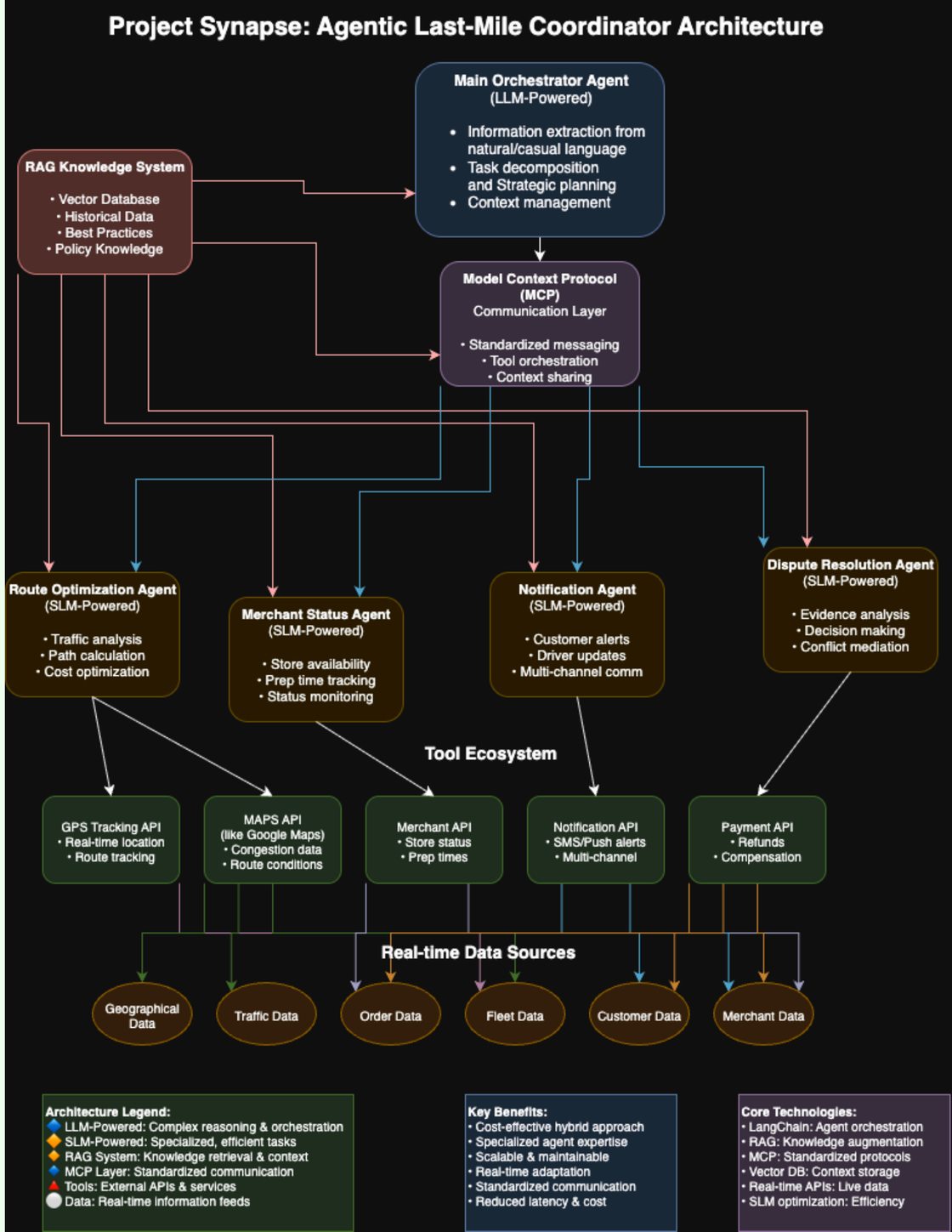
- **Purpose:** Information extraction from natural language.
- **Advantage:** Allows for human-like interaction with the end-user; provides sanitized input to MCP.

RAG KNOWLEDGE SYSTEM

- **Purpose:** Contains policy knowledge in the form of system prompts for all language models (LLMs/SLMs), to be concatenated with user input.
- **Advantage:** Consolidated data and context extension for all models.

MODEL CONTEXT PROTOCOL (MCP)

- **Purpose:** Intakes a set of flags and parameters; acts as a task mapper for SLM agents.
- **Advantage:** Integrations are simplified and makes the system more scalable and robust (reduces code length).



SLM-POWERED SPECIALIZED SUB-AGENTS

- **Purpose:** Reasoning for tasks defined in their subdomain, data extraction from API endpoints and final decision making for task execution.
- **Advantage:** They can work as per the policies defined by the system; less chances of hallucination due to specified domains.
    - More specialized agents can be created as more tasks are added for the system, which makes the system scalable and flexible.

TOOL ECOSYSTEM

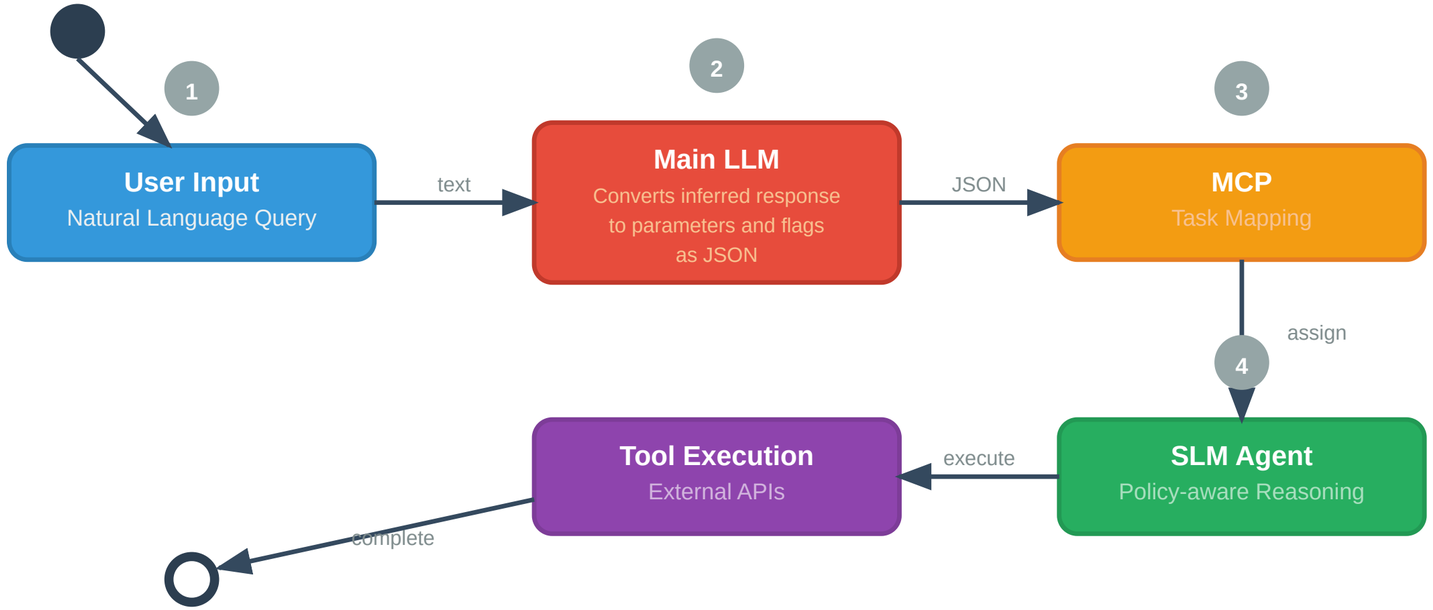
- **Purpose:** Final algorithmic layer for task execution using external APIs, for example, Google Maps API.
- **Advantage:** Improves confidence.

KEY VALUE PROPOSITION

- **Cost-effectiveness:** Cost reduction using SLMs for specialized tasks.
- **Scalability:** Non-monolithic architecture supports growth.
- **Real-time:** Sub-second response times for critical decisions.
- **Privacy:** Offline hosted SLMs can make the system work for sensitive data as well.
- **Guardrails:** Existence of an MCP layer protects the system against prompt injections and provides protection against other bad actors.
- **Flexibility:** Due to RAG, the system can adapt to evolving policies instead of redesigning the entire architecture.

TL;DR

Project Synapse - Process Flow



Process Flow:

1. User provides natural language input → 2. Main LLM converts to JSON parameters
3. MCP algorithmically assigns to relevant SLM agent → 4. SLM executes tasks using external APIs

THANK YOU