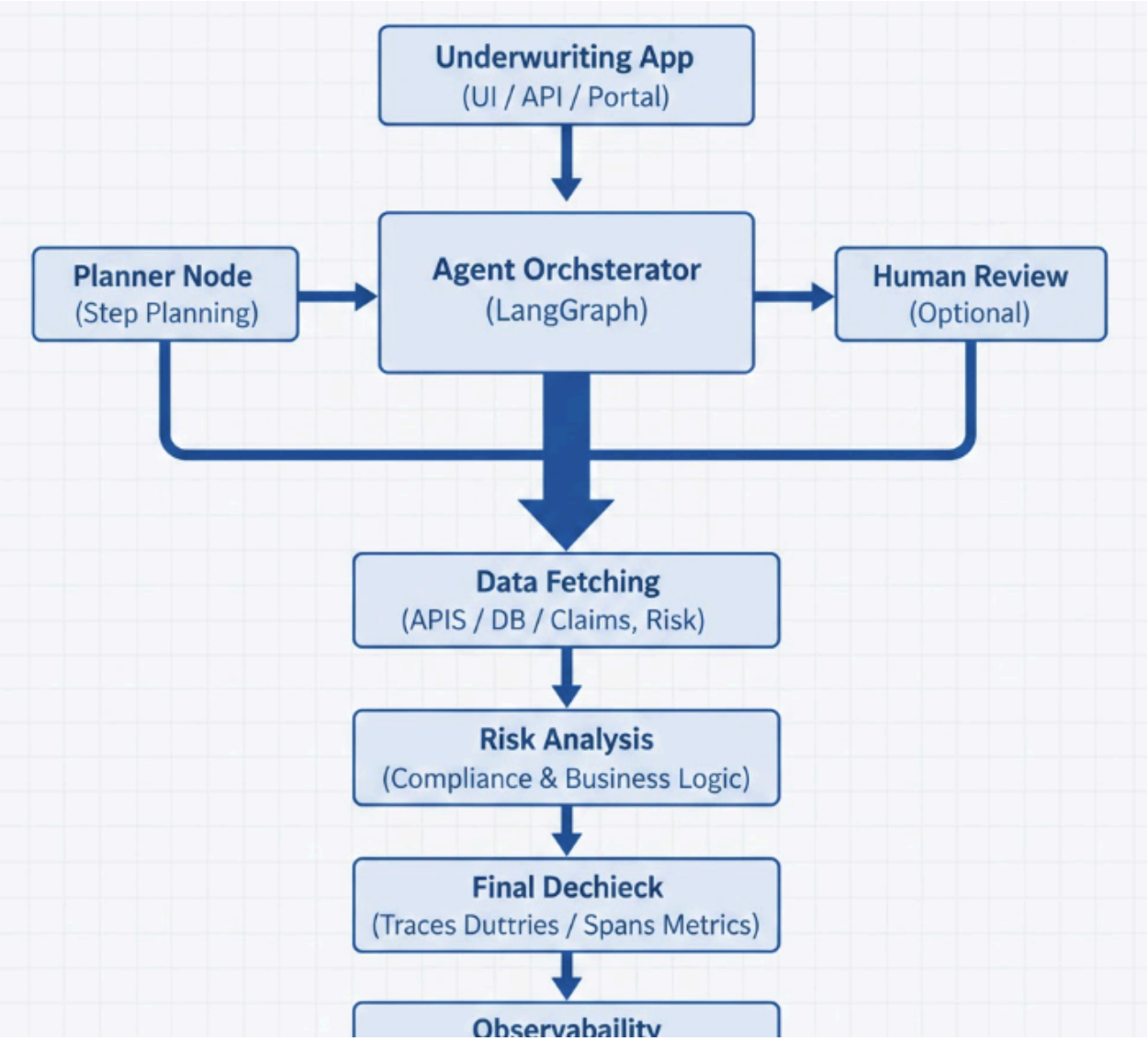
Design Artifact of the Agent Pipeline (LLM API)

Tools

- Agent AI Framework: LangGraph, LangChain
- LLM AI: AWS Bedrock, some LLM API key
- Observability: OpenTelementary
- Server: VM (AWS EC2)

Design Architecture



Architecture Components Explained

1. Underwriting App

- Entry point for applications (internal portal, API, or automated ingestion)
- Sends applicant data to the agentic pipeline

2. Agent Orchestrator (LangGraph)

- Core **state machine**
- Breaks underwriting into **nodes** (planner, data fetch, risk analysis, rules check, decision)
- Supports **conditional routing** (e.g., refer to human if borderline)
- Maintains **state** for each underwriting case

3. Nodes

- **Planner Node**: Creates step-by-step plan for each application
- **Data Fetching Node**: Pulls historical claims, financial data, risk scores
- **Risk Analysis Node**: Uses LLMs + rules to identify risk factors
- **Rules Check Node**: Applies underwriting rules and compliance logic
- **Final Decision Node**: Generates decision and human-readable explanation

4. Human Review Node (optional)

- For edge cases or high-value policies
- Fully auditable, integrated with trace

5. Observability (OpenTelemetry)

- Each node execution → **span**
- Each underwriting workflow → **trace**
- Captures:
    - Input metadata (without sensitive PII)
    - Decisions and rationale
    - Latency per step
    - Errors or retries

6. Monitoring & Audit Backend

- Collects all traces & metrics
- Examples: **Jaeger**, **Honeycomb**, **Tempo**
- Enables:
    - Regulatory audit

- Debugging
- Workflow optimization
- Reporting