LangGraph

LangGraph 소개

[Recap] 수많은 에이전트 구조

대부분의 에이전트 구조가 복잡하게 구성되었는데..

- LangChain만으로는 만드는 것에 한계가 있는 것 같다
- 멀티 에이전트 같은 구조는 어떻게 구현할 수 있을까?

이론적으로 구현이 불가능한 것은 아님

- LangChain의 방식이 효과적이지 않은 구성
- 이 때 우리는 LangGraph를 사용할 수 있다

[Recap] LangChain의 장점

직관적이고 간결한 코드 구성

- 파이프 구조 기반의 효과적인 서브모듈 처리
- Invoke(), Batch()만으로 실행되는 구조

높은 추상화

- Vector DB, File Loader, Parser 등의 다양한 모듈 지원

LangChain의 개선점은?

중간 상태의 명시적 관리 어려움

- 파이프 구조는 직관적이나, 이전 단계를 보존하지 않음
- 조건부 로직이나, 중간 값 변화 등을 처리하려면 랭체인 문법을 이탈해야 함

디버깅과 모니터링의 어려움

- 중간에 오류가 생기면 이를 추적하기 어려움
- 모니터링/테스팅을 위한 기능 부족

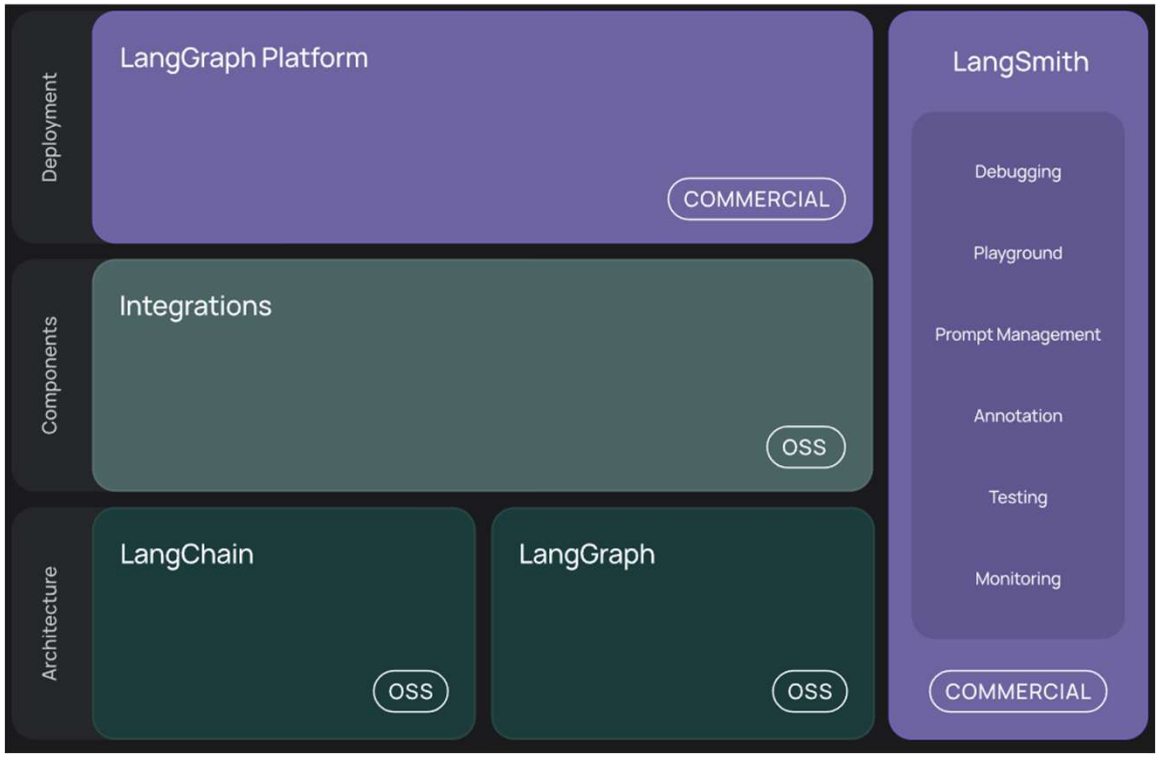
Beyond LangChain

LangGraph (Agentic)

- 상태(State) 기반의 그래프 구조
- 복잡한 Agent와 Workflow를 쉽게 구현

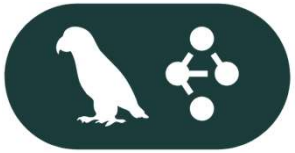
LangSmith (LLMOps)

- 랭체인의 중간 실행 결과를 트래킹
- 평가 및 모니터링 도구를 지원



Source: https://python.langchain.com/docs/introduction/

LangGraph(드디어!)



LangGraph

그래프 기반의 LLM Workflow

- 프로그램의 Flow Chart와 같은 구조로, LLM이 연결된 Workflow를 구성할 수 있음
- LangChain보다 적용 범위가 넓고 구현이 간결함

LangGraph + LangChain

- 단일 기능(입력 to 출력)은 LangChain Chain으로 구성하고,
- 이들을 조직하고 연결하는 형태의 전체 플로우를 LangGraph로 구성

LangGraph의 기본 구조

State Diagram

- State가 그래프를 통과하며 값이 저장/추가/수정되는 구조
- 예시) {질문} → {질문/검색 결과} → {질문/검색 결과/ 답변}

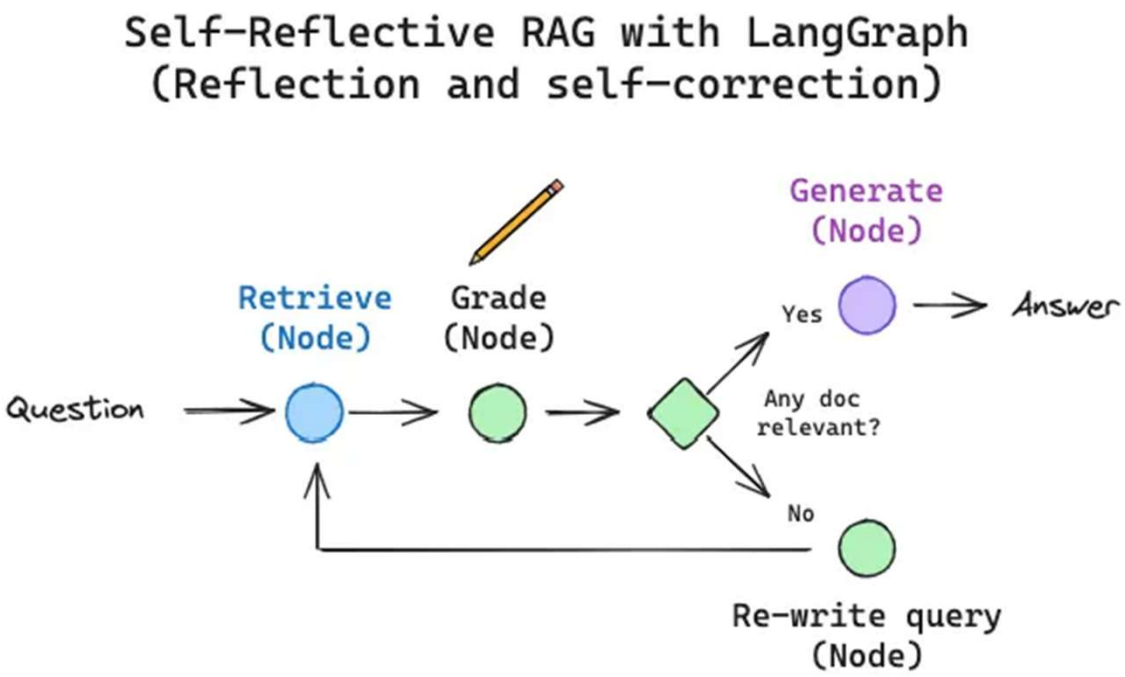
그래프: Node와 Edge의 구성

- 각 노드는 주로 단일 기능을 수행하며(랭체인의 체인과 주로 대응)
- 각 엣지는 다음 단계의 Workflow를 다룸

LangGraph 예시: Self-Corrective RAG

State

- {Question, Context, Answer} 구조가
- 처음에는 {Question, - , -}에서
- {Question, Context, Answer}로 업데이트



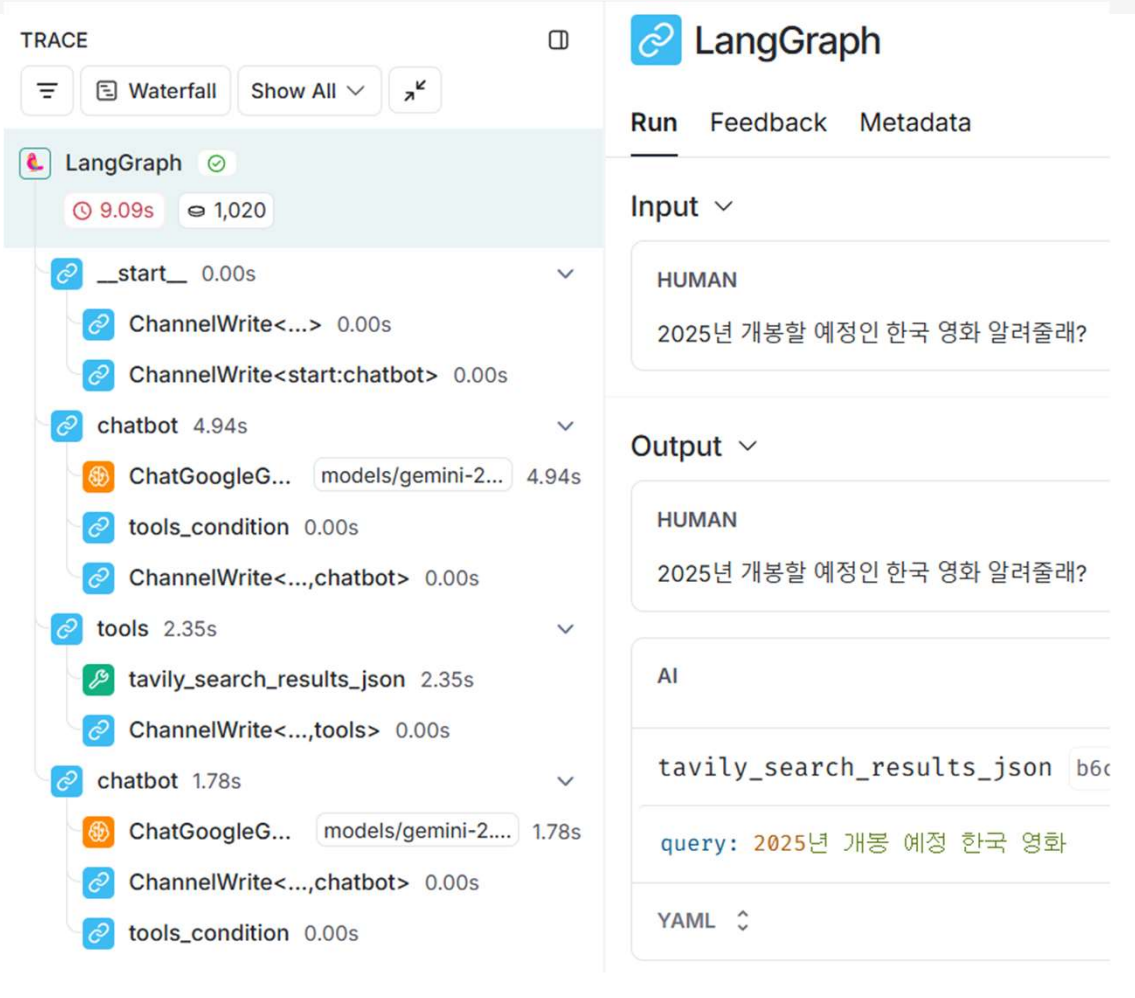
Source: https://blog.langchain.dev/agentic-rag-with-langgraph

Tracking

LangSmith **기반의** Tracking **용이**

- LangChain과 동일하게,
  각 그래프의 세세한 작동 과정 트래킹

- 피드백/평가 구조를 통한 모니터링도 가능
  → 단, LangSmith는 월 5,000회 무료 제한

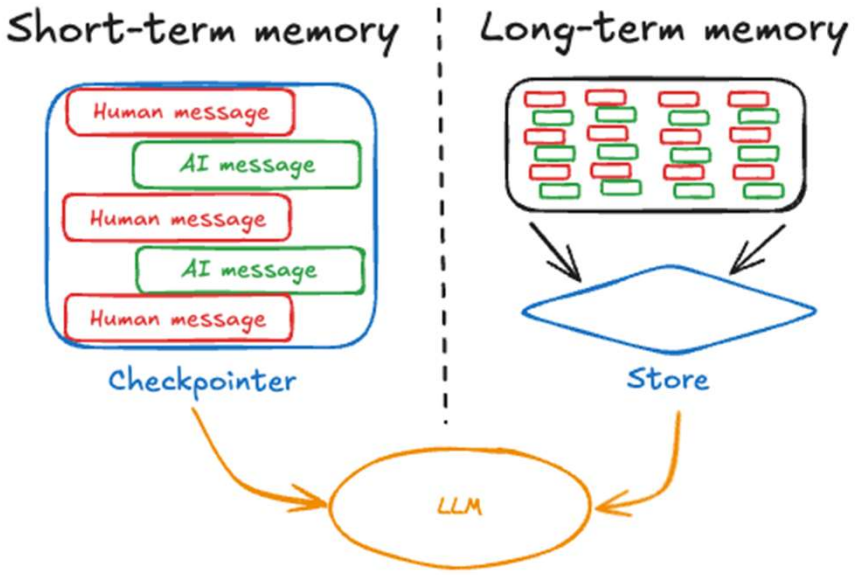


Memory

데이터를 기억하는 세션 단위 동작

- LangGraph Thread와 같은 세션 관리 지원
- 기존의 랭체인과 연동하여, 다양한 디자인 구조 지원
- 필요한 부분만 효과적으로 보존하기 위한 압축

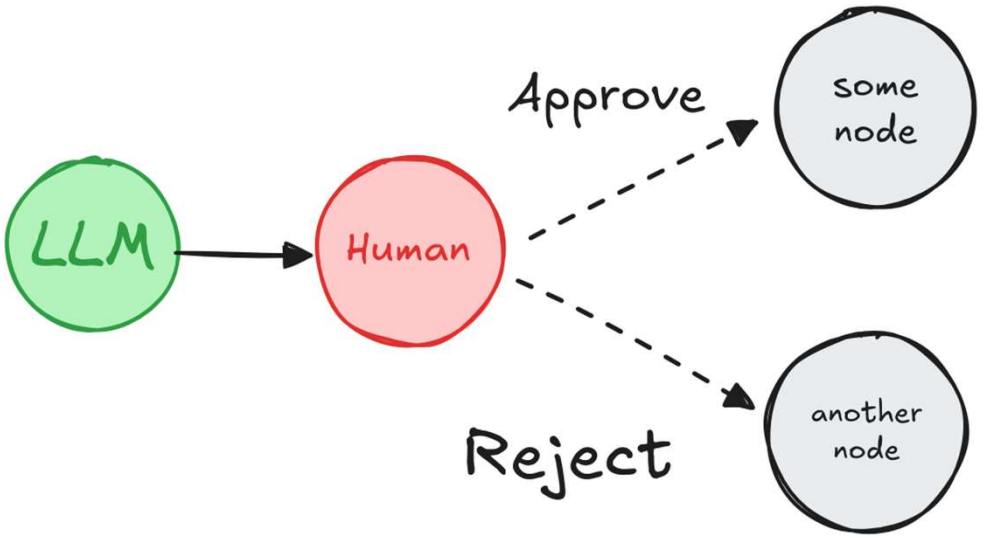
→ 이전 세션의 일부만 보존하는 Window 기반 디자인
과거 세션을 요약하는 Summary 기능 연동



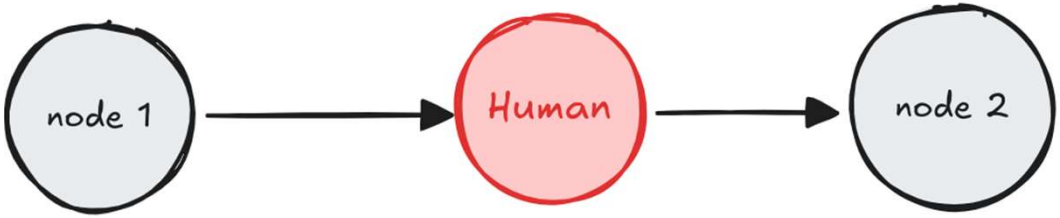
Human-in-the-loop

인간의 개입이 필요한 경우

- LLM 기반의 어플리케이션은 다양한 변수 존재
- 중요한 경우에는 인간이 개입해야 할 수 있음
  → 툴/API 실행 전 검증, 최종 결과물 평가 등



- LangChain Agent에서 LangGraph로 변화한 주요 이유



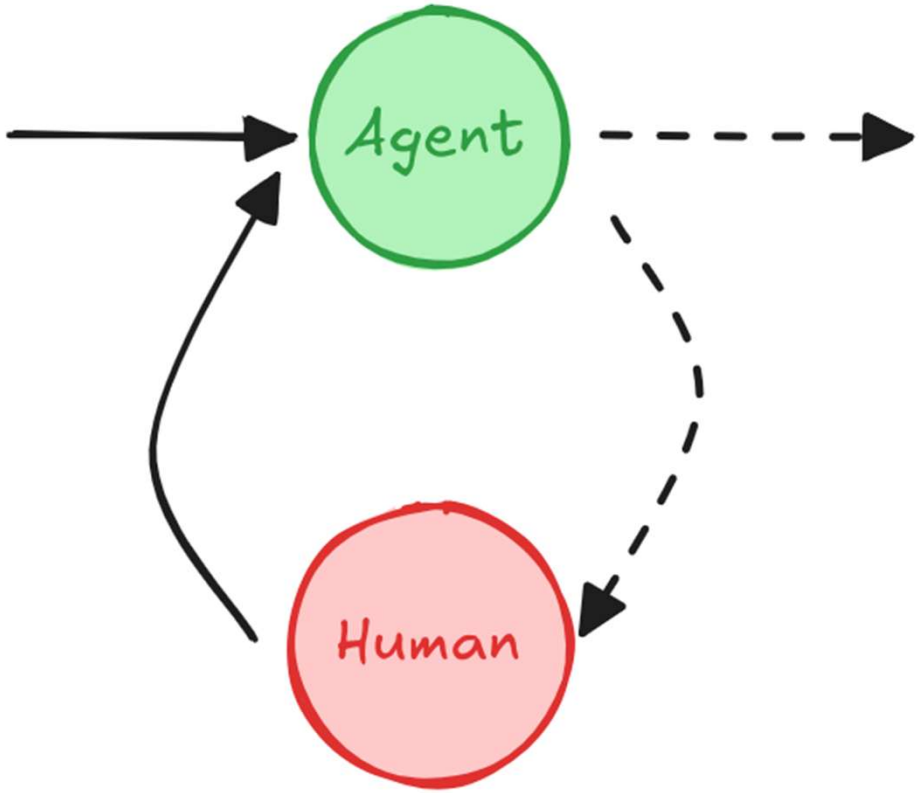
Human-in-the-loop

인간에게 더 물어봐야 하는 경우

- 첫 요청이 불명확하여 추가 정보가 필요한 경우

  →충분한 정보가 모이면 Agent 작동

- Ex) OpenAI Deep Research



Summary

LangGraph

- 그래프 기반의 Workflow 구성을 바탕으로
  LangChain보다 넓은 범위의 Orchestration 가능

- 메모리/휴먼인더루프와 같은 효과적인 모듈 지원

- 모니터링/ 중간 개입의 차원에서 LangChain의 구조를 개선