CLAUDE CODE
유출로 알아보는
harness engineering

DeepShark Lab
학부생 전범기

26.05.08

목 차

유출 경위

어디서 **유출**되었는가?

-2026년 3월 31일
@anthropic-ai/claude-code v2.1.88 npm 배포 중 배포자의 실수로 인해 **.map** 파일이 같이 올라갔습니다.
그것을 한 유저가 발견을 하고, **SNS**에 올려 확산이 시작됐습니다.

문제의 근원 **.map**이란?

.map 파일은 **난독화된 파일을 복원**할 수 있게 만들어주는 파일입니다.
Claude Code는 **CLI 코딩 에이전트**이기 때문에 **npm** 배포에서 **난독화 과정**을 거칩니다. 그 과정에서 난독화된 파일과 .map파일이 나오게 됩니다.

왜 **확산**을 막지 못 하였는가?

유출되어 버전 삭제까지 총 3시간.
npm삭제 및 Github에 확산 방지 요청을 하였지만, Github에 **삭제되는 속도보다 확산되는 속도가 빨라 막을 수 없었습니다.**
Claude-code가 유출된 소스 코드를 다루는 github 중 가장 인기 많은 곳은 최단기간 내 github 100,000 스타를 넘겼습니다.

유출된 핵심 내용

모델 웨이트가 유출된 것이 아닌 개발 도구(Claude code)의 “소스 코드” 유출

Claude code란?

: **터미널에서 Claude AI와 직접 상호작용하여 코딩 작업을 수행하는 CLI 코딩 에이전트**

개발 도구(Claude code)의 **규모 및 개요**

총 파일 수

: 약 1,900 개

총 코드 라인

: 약 510,000 줄

주요 언어

: TypeScript: JavaScript에 type을 추가한 언어 (확장자 : ts / tsx)

유출된 핵심 내용

유출된 51만 줄의 본질은 단순한 코드가 아니라

LLM이라는 엔진을 실무에 즉시 투입하기 위해
정교하게 설계된 제어 틀

즉 '하네스(Harness)' 그 자체입니다

하네스 엔지니어링

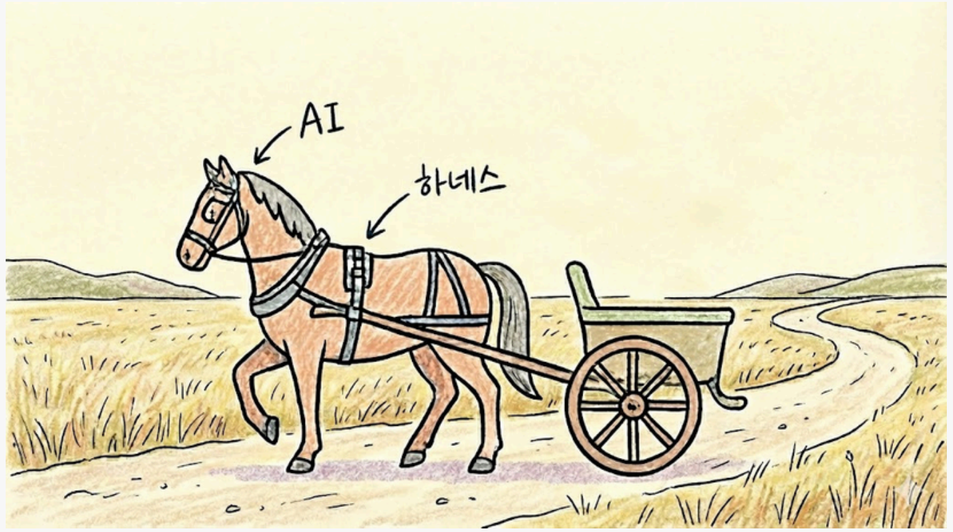
harness란 무엇인가?
:harness의 어원

본래 의미: 말의 힘을 제어하여 원하는 방향으로 이끄는
'말을 부리는데 쓰는 기구'

AI 분야에서의 의미

말 → AI

하네스 → AI 에이전트를 원하는 방향으로 이끄는 구조

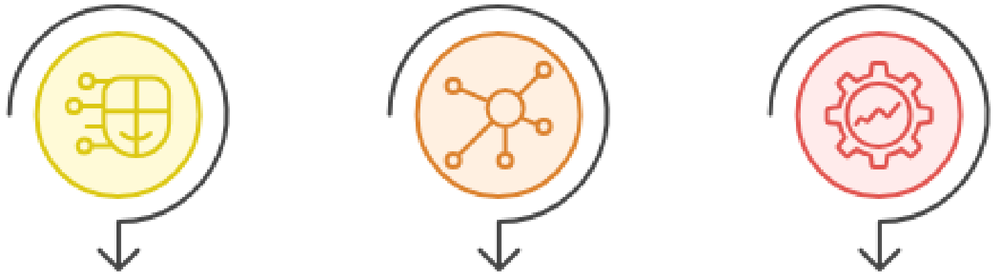


하네스 엔지니어링

harness가 중요한 이유

모델 그 자체로는 코딩 에이전트를 수행하기에 부족합니다. 모델에 하네스를 추가함으로써 모델이 실제 환경과 상호작용 할 수 있도록 돕습니다. 특히 AI가 어떤 도구를 사용하고, 흐름을 통제하는지에 따라 성능이 크게 달라집니다.

하네스의 역할



통제

하네스는 AI의 작동 범위를 정의하고 결과의 신뢰성을 담보합니다.

연결

하네스는 기술들을 연결하여, AI가 실제 업무 시스템 내에서 작동하도록 돕는 중추적인 역할을 수행합니다

최적화

컨텍스트 관리, 토큰 비용 최적화, 그리고 오류 복구 메커니즘을 포함하여 에이전트의 효율성을 극대화합니다.

하네스 엔지니어링

AI 선두주자가 말하는 harness engineering의 중요성

- 2026년 2월 11일
<하네스 엔지니어링: 에이전트 우선 세계에서 Codex 활용하기>
: AI 하네스 엔지니어링의 중요성에 대해 이야기.

-OpenAI 기술 블로그

- 2026년 3월 24일

<Harness design for long-running application development>
'Harness design is key to performance at the frontier of agentic coding.'

-Anthropic 기술 블로그

분석 방법

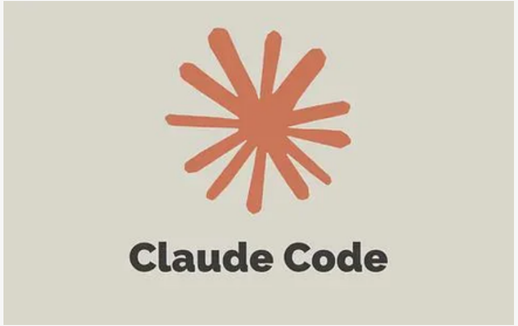
코드 분석 방법

- 1차 분석: 코드 에이전트 활용

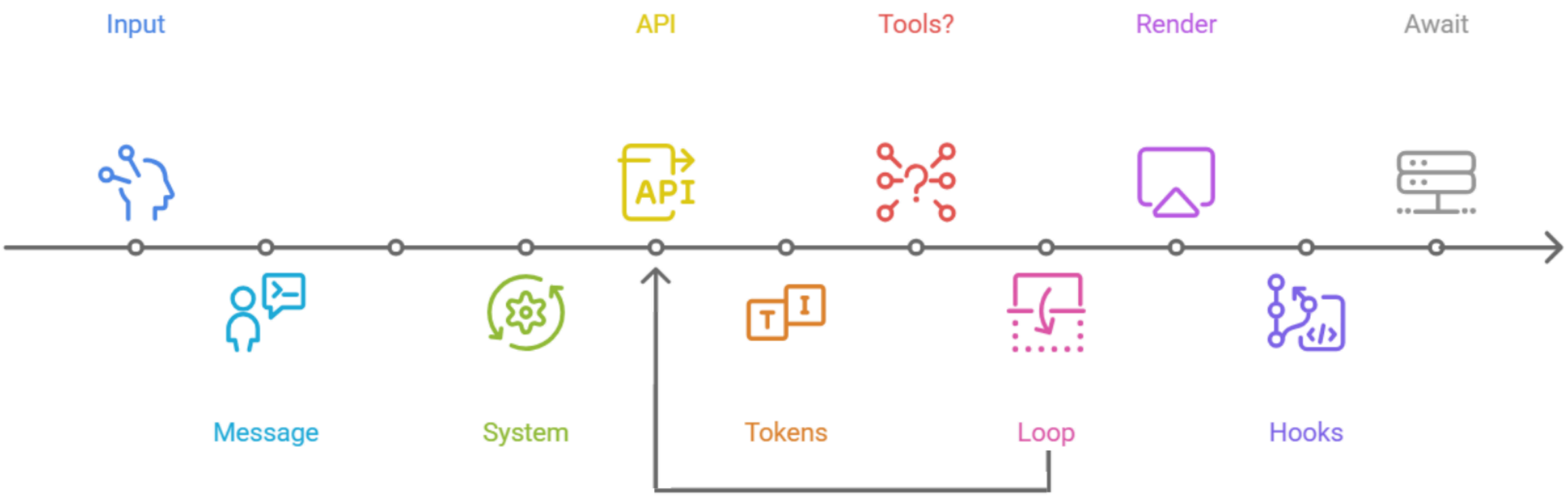
본 분석은 코딩 에이전트인 Claude Code와 Codex 기반의 OMX를 활용하여 수행되었습니다. AI를 통해 유출 데이터의 패턴 매칭 및 로직 분석을 자동화하였습니다.

- 2차 분석 : Human 검증 및 AI 재검증

Claude Code와 OMX로 생성한 문구의 파일 출처를 표시 한 뒤 Human 검증 및 AI(gpt5.5, gemini 3)로 신뢰성을 확보했습니다.



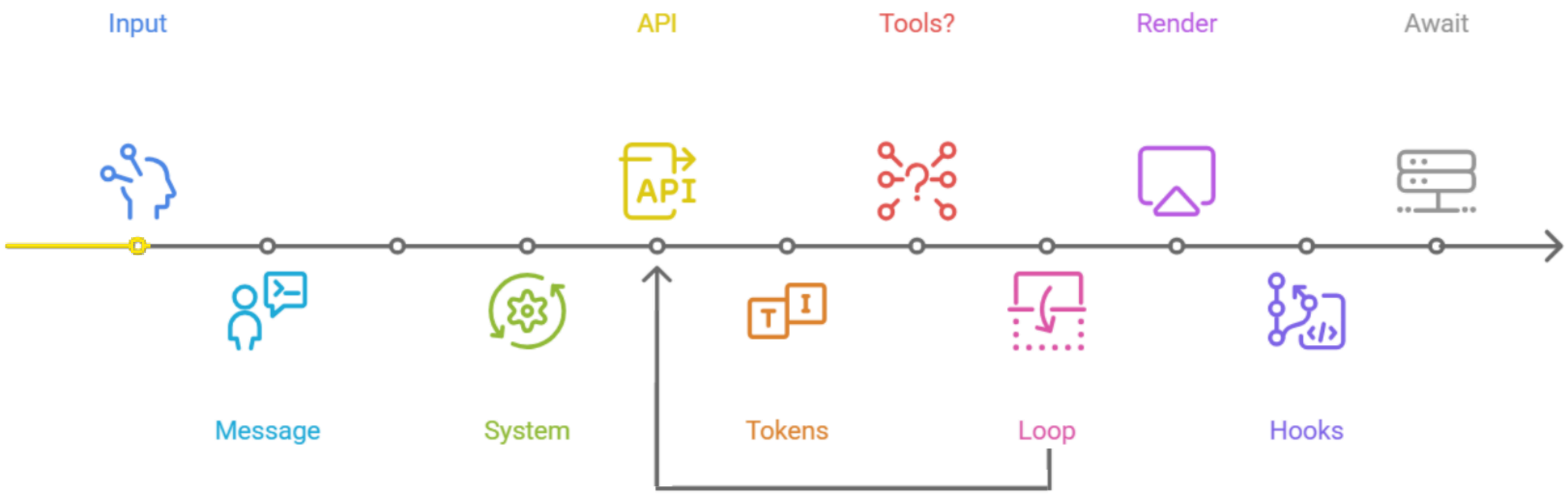
클로드 코드 분석


Input
API
Tools?
Render
Await
API
Message
System
Tokens
Loop
Hooks

클로드 코드 분석

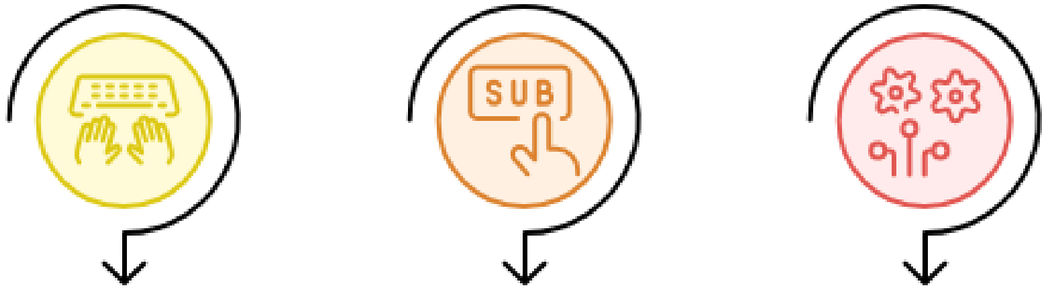
Input

사용자의 Input을 받고 처리합니다.



클로드 코드 분석

Input



사용자 타이핑

사용자가 텍스트를 입력합니다

PromptInput.onChange() 가 호출되고 REPL.setInputValue()가 호출됩니다.

사용자 제출

사용자가 Enter 또는 submit을 누릅니다

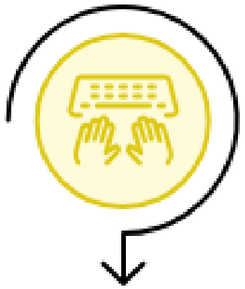
PromptInput.onSubmit() 이 호출되고 REPL.onSubmit()이 호출됩니다.

프롬프트 처리

handlePromptSubmit() 함수가 호출되어 프롬프트를 처리합니다.

클로드 코드 분석

Input



사용자 타이핑

사용자가 텍스트를
입력합니다

PromptInput.onChange()
가 호출되고
REPL.setInputValue()가
호출됩니다.

REPL 란?

Claude Code UI의 최상위 컨트롤러

PromptInput.onChange() 란?

실시간 state 동기화용 함수

.onChange() 효과

[자동 완성 기능]

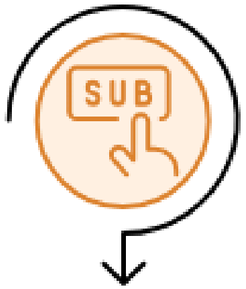
- slash command 자동완성 (/ 치는 순간 목록 표시)
- mention 자동완성 (@ 치는 순간)

[speculation 기능 (실험 중)]

- 사용자가 타이핑하는 동안 Claude가 미리 답변 생성을 시작 (process.env.USER_TYPE === 'ant')

클로드 코드 분석

Input



사용자 제출

사용자가 Enter 또는
submit을 누릅니다

PromptInput.onSubmit()
이 호출되고
REPL.onSubmit()이 호출
됩니다.

PromptInput.onSubmit() 이란?

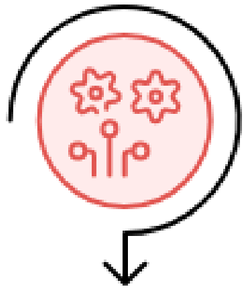
사용자의 Input을 submit 시 REPL에 넘겨주는 역할

REPL.onSubmit() 이란?

REPL.onSubmit() 호출 시
handlePromptSubmit()이 호출됩니다.

클로드 코드 분석

Input



프롬프트 처리

handlePromptSubmit()
함수가 호출되어 프롬프
트를 처리합니다.

handlePromptSubmit()이란?

REPL의 교통정리 함수(6개의 if문)

1. queuedCommands 있으면 바로 실행
2. input 비어있으면 → 무시
3. exit/quit/:q 입력 → 종료 처리
4. /slash 명령 → 슬래시 커맨드 실행
5. 이미 로딩 중 → 큐에 넣기(enqueue)
6. 1 ~ 5 실행 후 executeUserInput() 호출

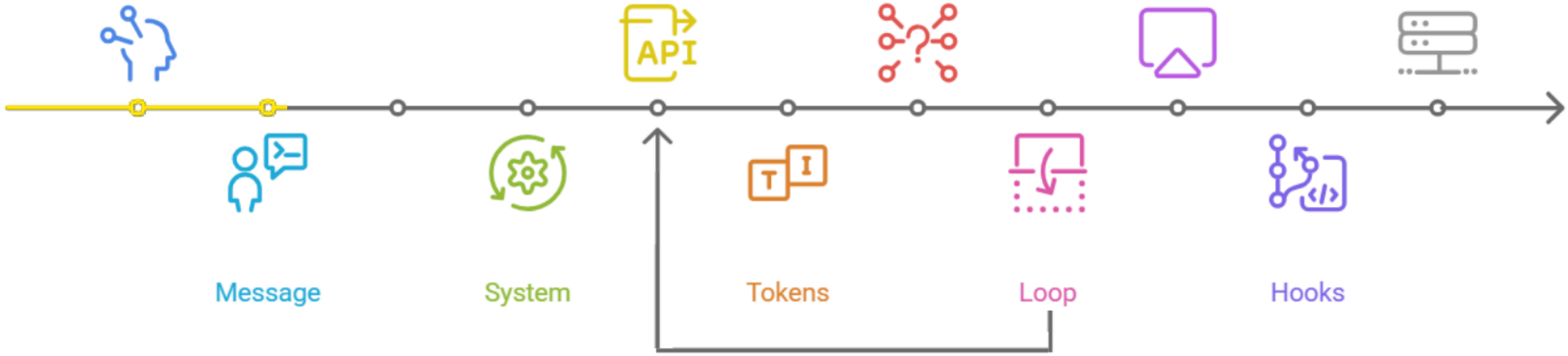
executeUserInput()

→ 입력을 메시지로 만들고
Claude한테 보낼지 말지 결정해서 보내는 함수

클로드 코드 분석

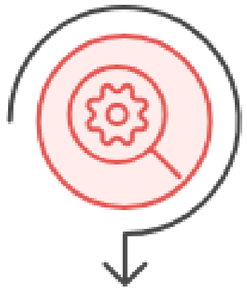
Message

API에 전달할 사용자의
입력을 준비합니다



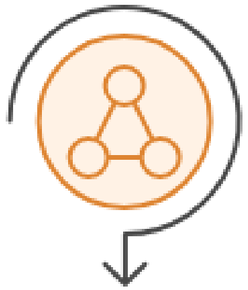
클로드 코드 분석

Message



전처리

handlePromptSubmit()
에게 받은 텍스트를 전처
리를 수행합니다.



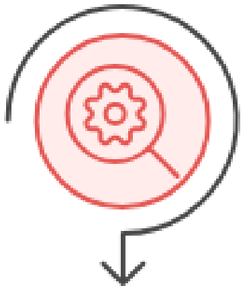
Anthropic's
Message format

JSON 형태로 format
정렬 시킵니다

클로드 코드 분석

/

Message



전처리

handlePromptSubmit()
에게 받은 텍스트를 전처리를 수행합니다.

어떤 것을 전처리하나요?

1. slash command (' / ... ') 인지
2. 일반 프롬프트인지
3. 이미지를 포함하는지

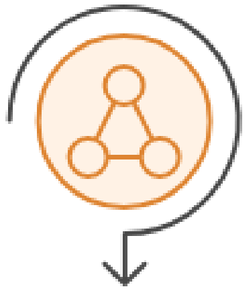
전처리 하는 이유

역할이 다르기 때문.

1. slash command (' / ... ') → onQuery 실행 X
   (Claude API 호출 없이 사용) ex) /config , /help
2. 일반 프롬프트 → onQuery 실행 O (Claude API 호출 사용)
3. 이미지를 포함 → 이후 JSON 형태에 text가 아니라 img로 기입

클로드 코드 분석

Message



Anthropic's
Message format

JSON 형태로 format
정렬 시킵니다

- [입력 텍스트 / 압축된 요약 메시지]를 전달할 준비를 마칩니다.

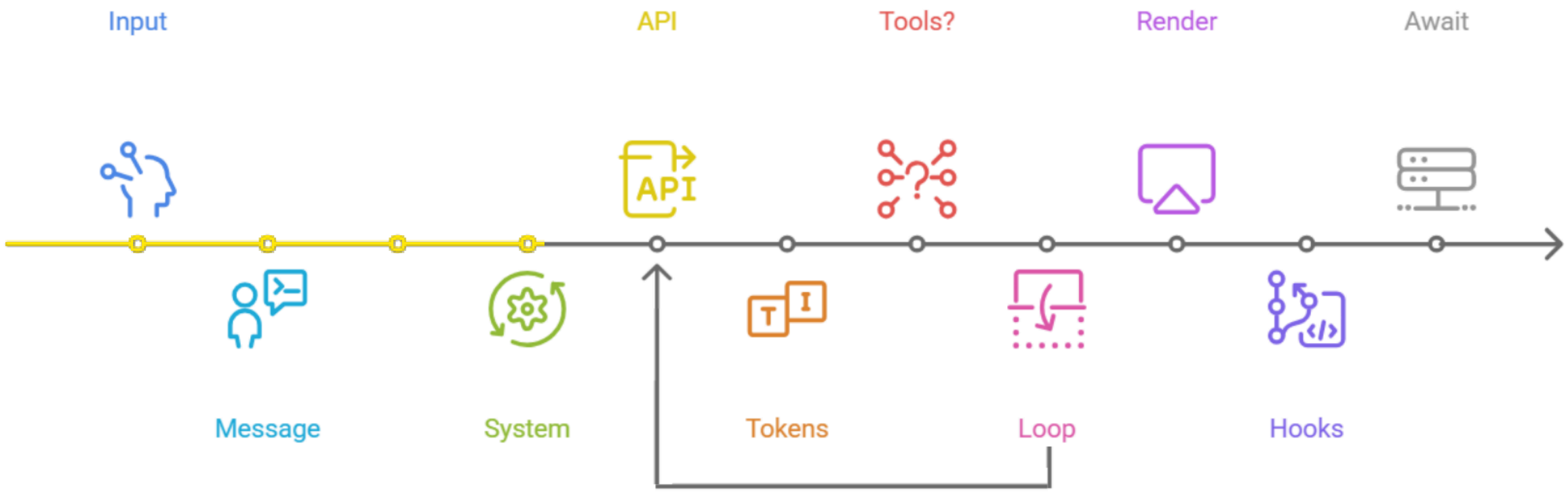
압축된 요약 메시지란?

대화가 길어지거나, 사용자가 /Compact 명령어를 사용시 Claude가 요약문을 생성합니다. 이로인해, Claude가 이해해야하는 Context양을 줄이고 계속 대화하도록 만듭니다.

클로드 코드 분석

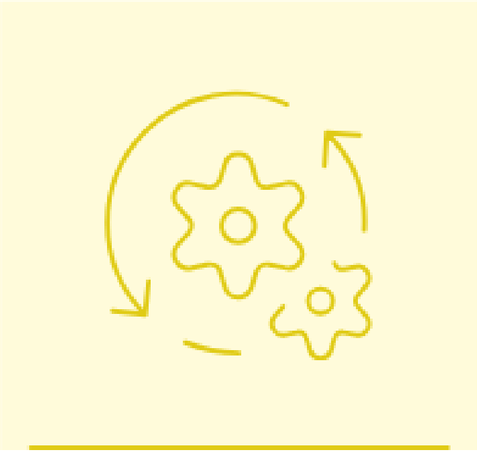
System

System이 API에 추가적으로
전달할 자료를 준비합니다.



클로드 코드 분석

System

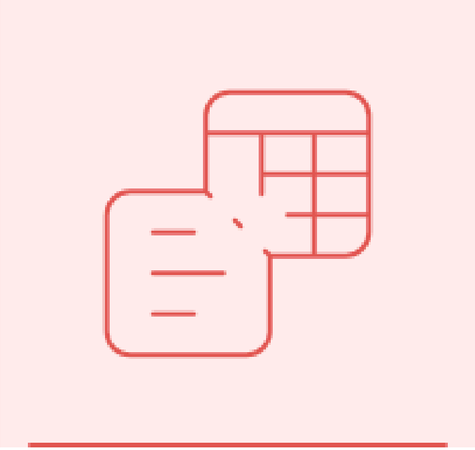


System Prompt

시스템 프롬프트
도구 사용 컨텍스트

System Context

git status
날짜



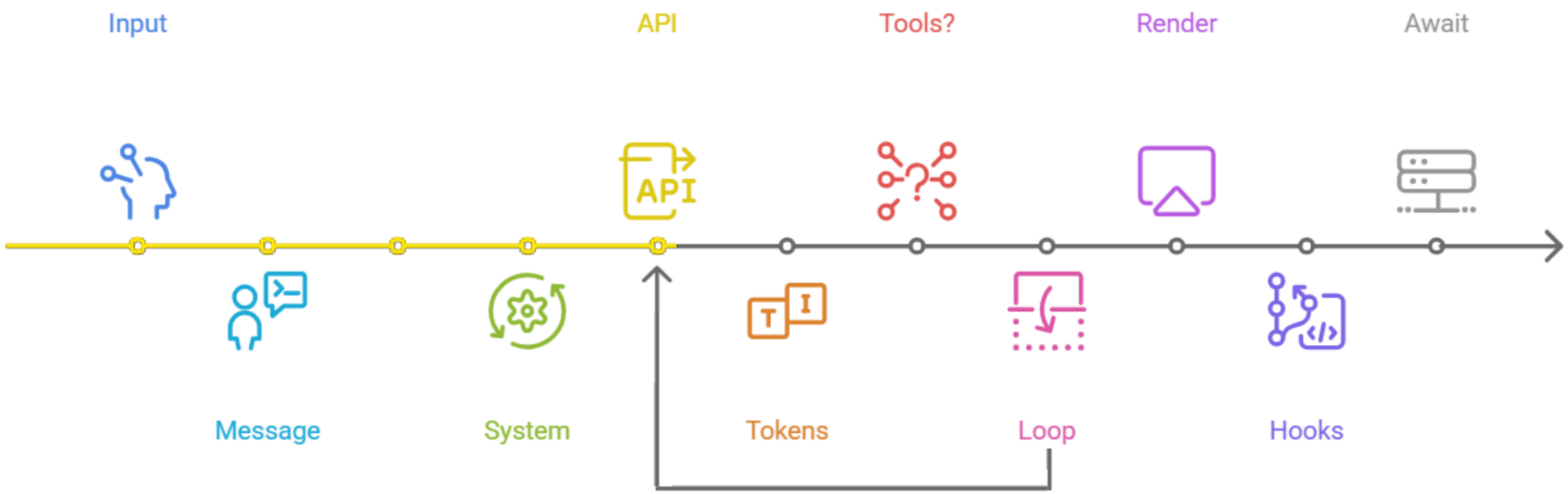
User Context

claude.md 파일
날짜

클로드 코드 분석

API

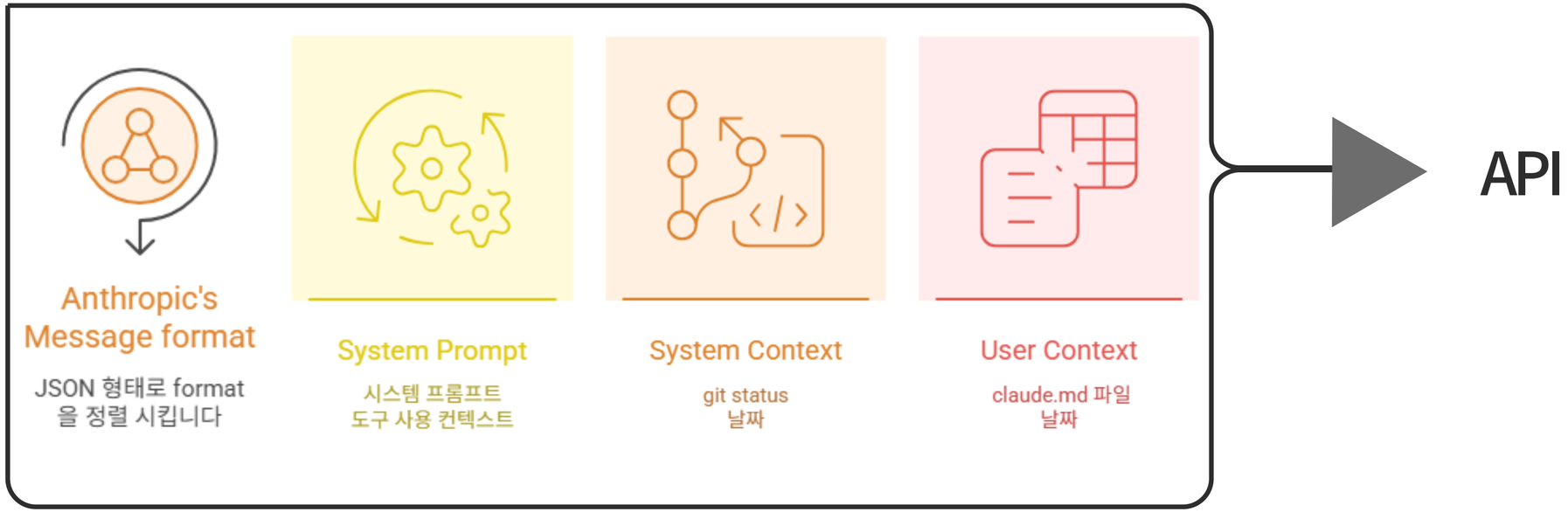
Message와 System을
더하여 API에 전달합니다.



클로드 코드 분석

/

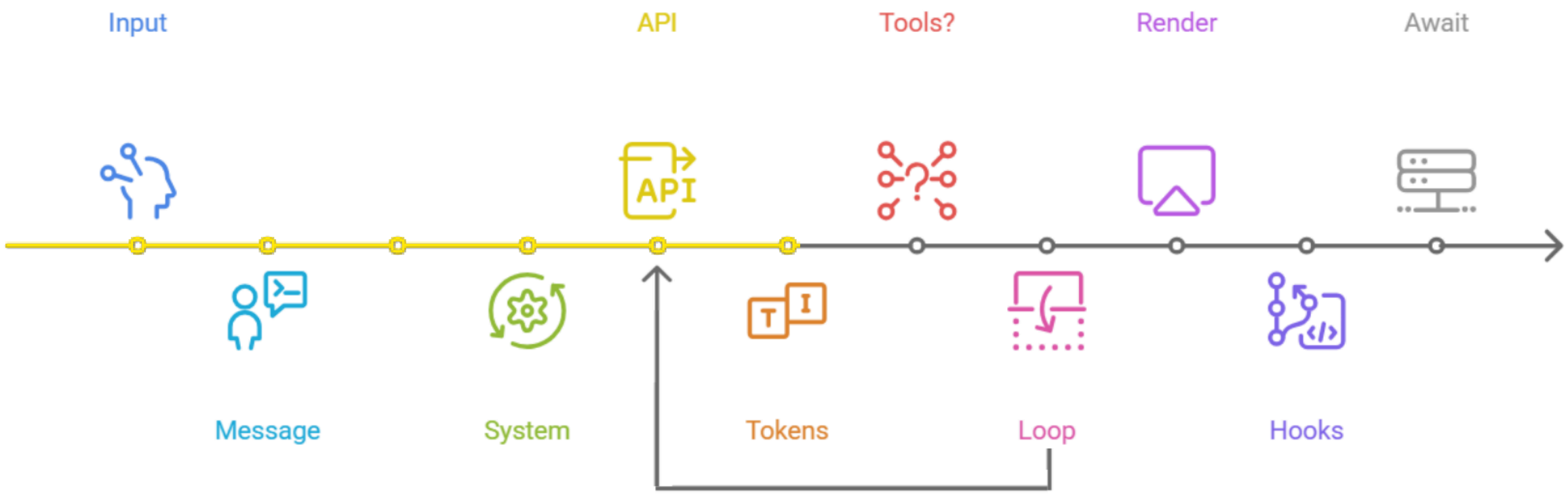
API


Anthropic's
Message format
JSON 형태로 format
을 정렬 시킵니다
System Prompt
시스템 프롬프트
도구 사용 컨텍스트
System Context
git status
날짜
User Context
claude.md 파일
날짜
API

클로드 코드 분석

Tokens

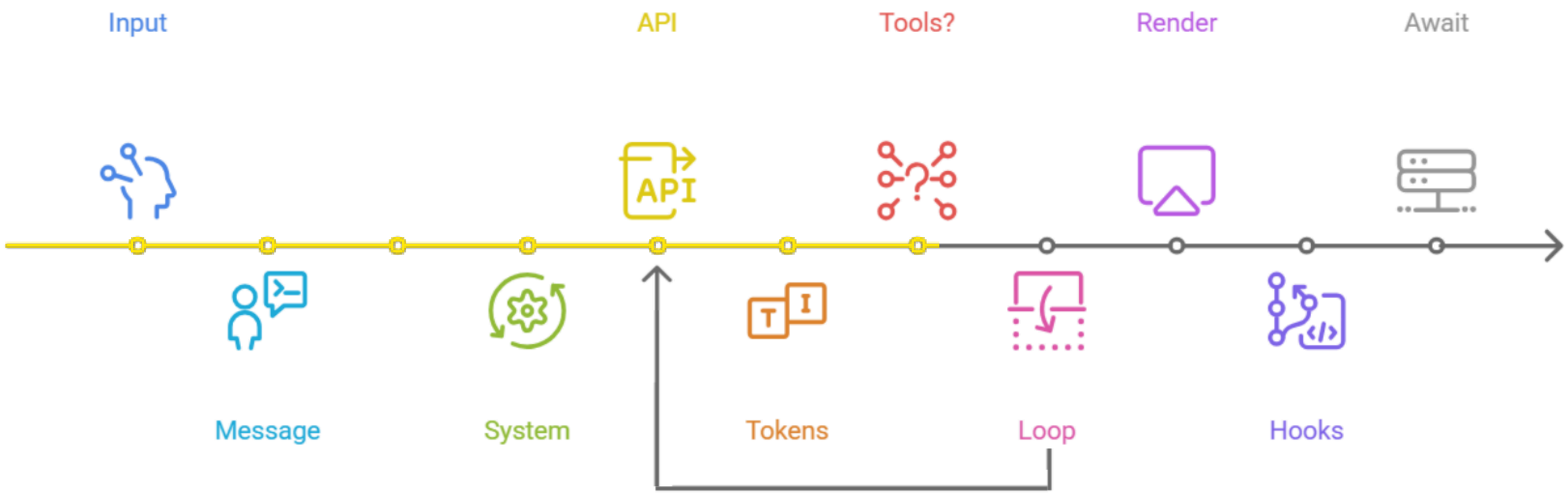
API가 주는 응답을 받는 즉시
화면에 표시합니다.



클로드 코드 분석

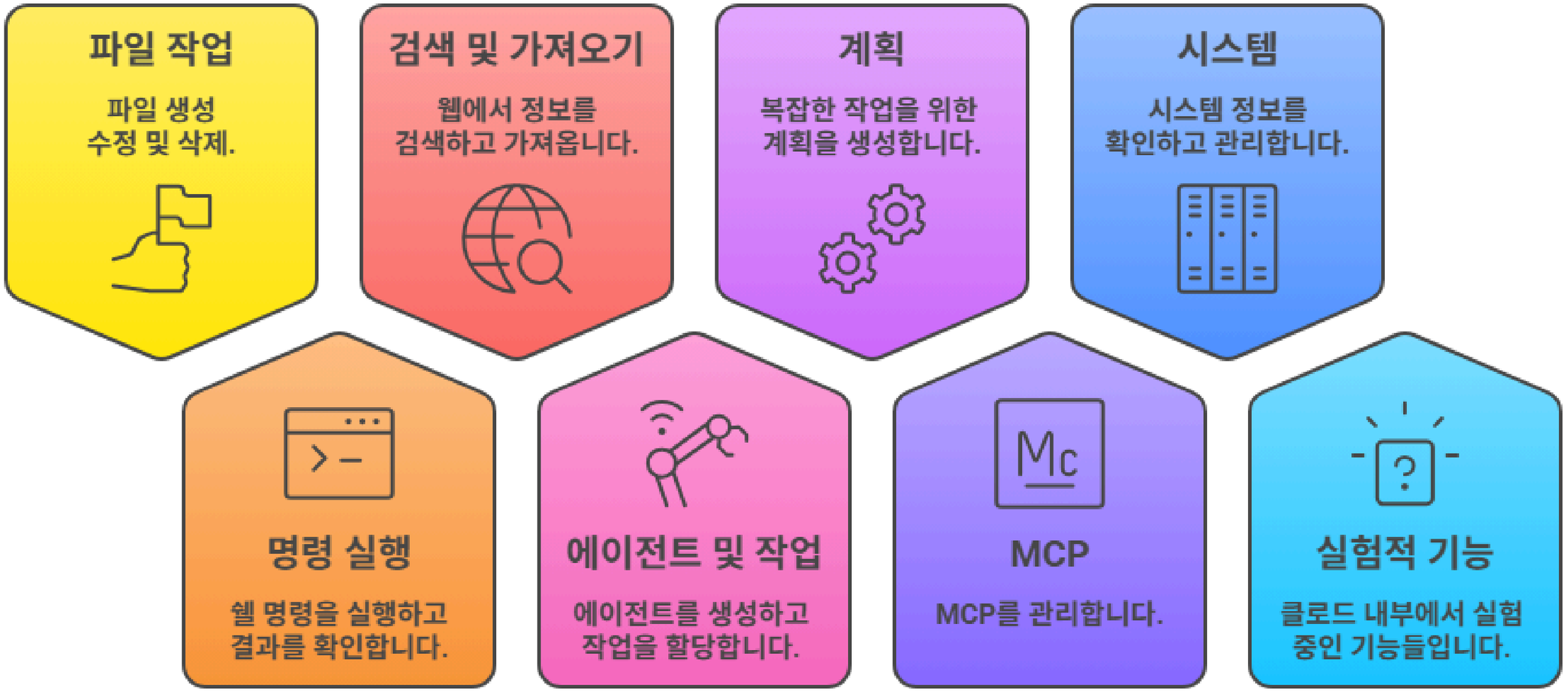
Tools?

Token들 중 tool_use 블럭이
있으면 도구를 실행합니다.



클로드 코드 분석

Tools



클로드 코드 분석

도구	설명	활성화 조건
FileRead	파일 읽기	항상 활성
FileEdit	파일 부분 수정	항상 활성
FileWrite	파일 전체 쓰기	항상 활성
NotebookEdit	Jupyter 노트북 셀 편집	항상 활성
Glob	파일 패턴 검색	EMBEDDED_SEARCH_TOOLS 환경변수가 없을 때만
Grep	파일 내용 검색	EMBEDDED_SEARCH_TOOLS 환경변수가 없을 때만

클로드 코드 분석

도구	설명	활성화 조건
Bash	셸 명령 실행	항상 활성
PowerShell 🔒	Windows PowerShell 명령 실행	Windows 플랫폼 + 조건부 환경변수
REPL 🔒	격리된 VM 안에서 Bash/Read/Edit 실행	USER_TYPE=ant 전용

클로드 코드 분석

도구	설명	활성화 조건
WebFetch	URL에서 내용 가져오기	항상 활성
WebSearch	웹 검색	항상 활성
ToolSearch 🔒	비활성 도구를 이름으로 검색해 스키마 로드	isToolSearchEnabledOptimistic()

클로드 코드 분석

도구	설명	활성화 조건
Agent	서브에이전트 생성 및 실행	항상 활성
SendMessage	에이전트 간 메시지 전송	항상 활성
TaskCreate 🔒	비동기 태스크 생성	isTodoV2Enabled()
TaskGet 🔒	태스크 상태 조회	isTodoV2Enabled()
TaskList 🔒	태스크 목록 조회	isTodoV2Enabled()
TaskUpdate 🔒	태스크 상태 업데이트	isTodoV2Enabled()
TaskStop	태스크 중단	항상 활성
TaskOutput	태스크 결과 출력	항상 활성
TeamCreate 🔒	에이전트 팀 생성	isAgentSwarmsEnabled()
TeamDelete 🔒	에이전트 팀 삭제	isAgentSwarmsEnabled()
ListPeers 🔒	연결된 피어 에이전트 목록 조회	feature('UDS_INBOX') ON

클로드 코드 분석

도구	설명	활성화 조건
EnterPlanMode	플랜 모드 진입	항상 활성
ExitPlanMode	플랜 모드 종료	항상 활성
EnterWorktree	git worktree 격리 환경 진입	isWorktreeModeEnabled()
ExitWorktree	worktree 환경 종료	isWorktreeModeEnabled()
VerifyPlanExecution 🔒	계획 실행 검증	CLAUDE_CODE_VERIFY_PLAN=true

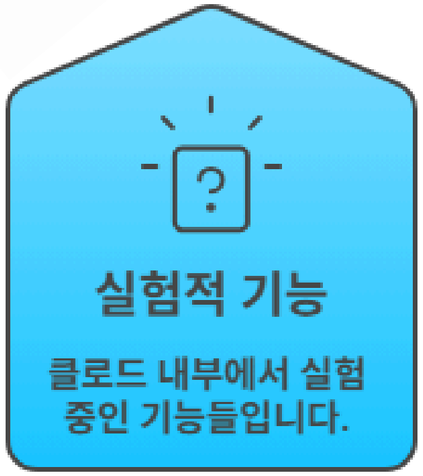
클로드 코드 분석

도구	설명	활성화 조건
mcp	MCP 서버가 노출하는 외부 도구들	MCP 서버 연결 시 동적 추가
ListMcpResources	MCP 서버 리소스 목록 조회	항상 활성
ReadMcpResource	MCP 서버 리소스 읽기	항상 활성
McpAuth	MCP 서버 인증	항상 활성

클로드 코드 분석

도구	설명	활성화 조건
AskUserQuestion	사용자에게 선택지 질문	항상 활성
TodoWrite	현재 세션 할 일 목록 관리	항상 활성
Skill	슬래시 커맨드(/skill) 실행	항상 활성
Config 🔒	Claude Code 설정 변경	USER_TYPE=ant 전용
RemoteTrigger 🔒	원격 에이전트 트리거	feature('AGENT_TRIGGERS_REMOTE') ON
CronCreate 🔒	반복 스케줄 생성	feature('AGENT_TRIGGERS') ON
CronDelete 🔒	반복 스케줄 삭제	feature('AGENT_TRIGGERS') ON
CronList 🔒	반복 스케줄 목록 조회	feature('AGENT_TRIGGERS') ON
Snip 🔒(Compact의 경량 버전)	대화 히스토리 구간 잘라내기	feature('HISTORY_SNIP') ON
TerminalCapture 🔒	터미널 패널 캡처	feature('TERMINAL_PANEL') ON

클로드 코드 분석

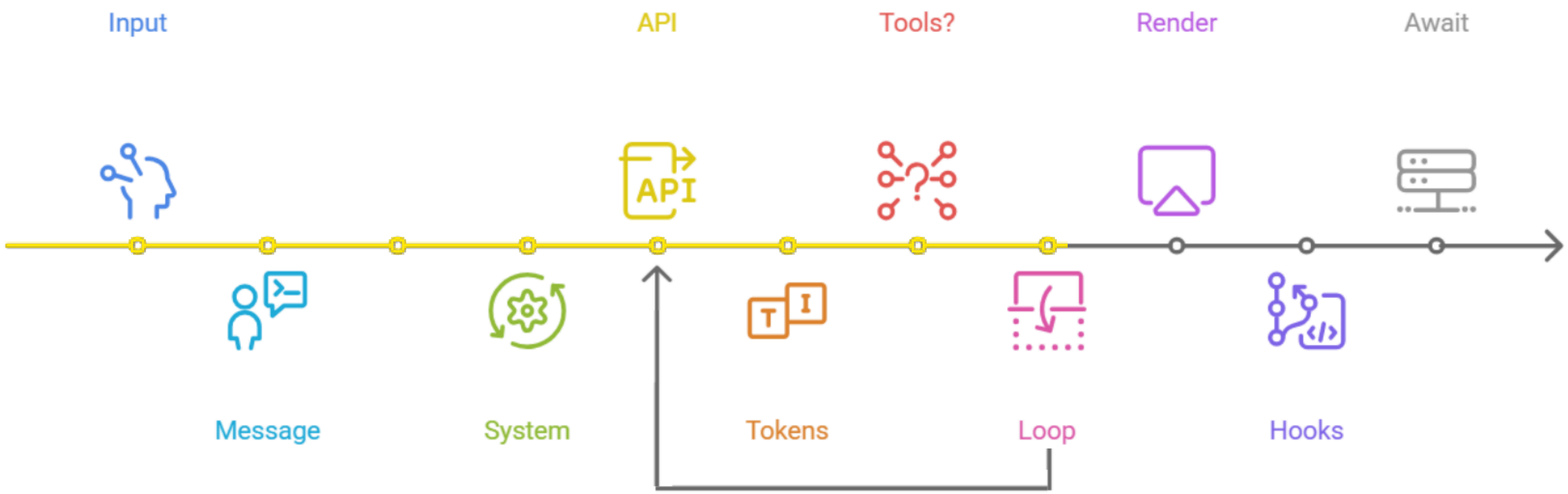


도구	설명	활성화 조건
Sleep 🔒	지정 시간 대기	feature('PROACTIVE') 또는 feature('KAIROS') ON
SendUserMessage	사용자에게 메시지 전송 (자율 에이전트 모드)	feature('KAIROS') ON
StructuredOutput 🔒	구조화된 출력 생성	확인 불가
LSP 🔒	Language Server Protocol 연동	ENABLE_LSP_TOOL=true
SendUserFile 🔒	사용자에게 파일 전송	확인 불가
PushNotification 🔒	푸시 알림 전송	확인 불가
Monitor 🔒	백그라운드 프로세스 모니터링	확인 불가

클로드 코드 분석

Loop

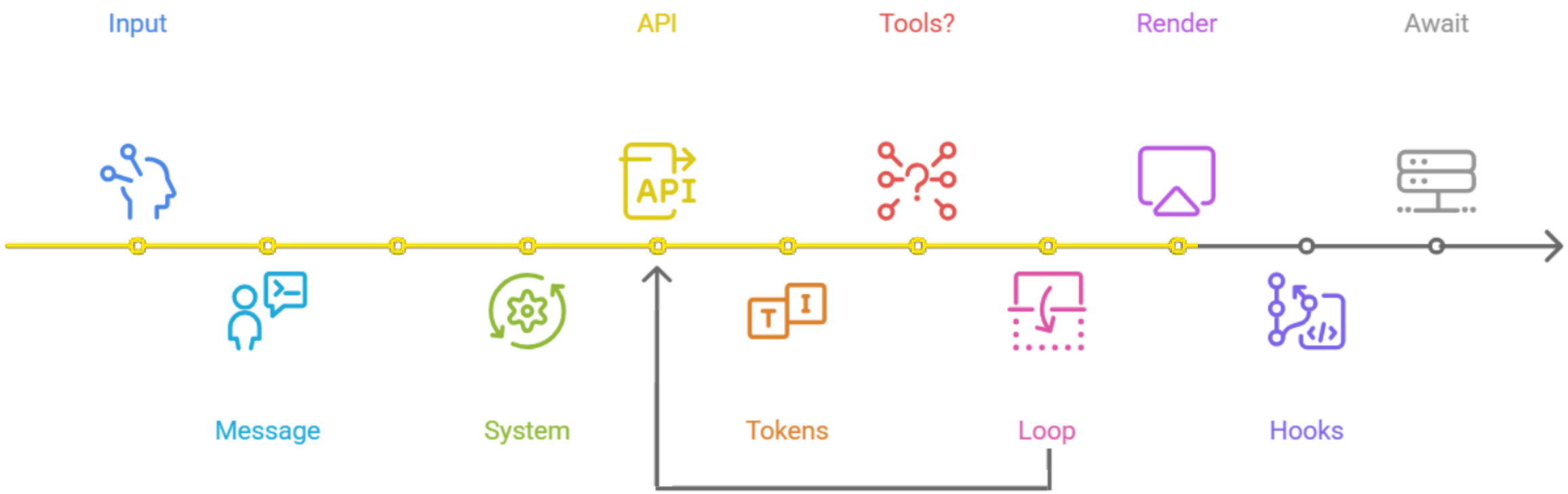
도구 실행 결과를 대화 메시지 배열에 다시 추가하고,
모델이 추가 판단을 해야 한다면 API를 재호출합니다.



클로드 코드 분석

Render

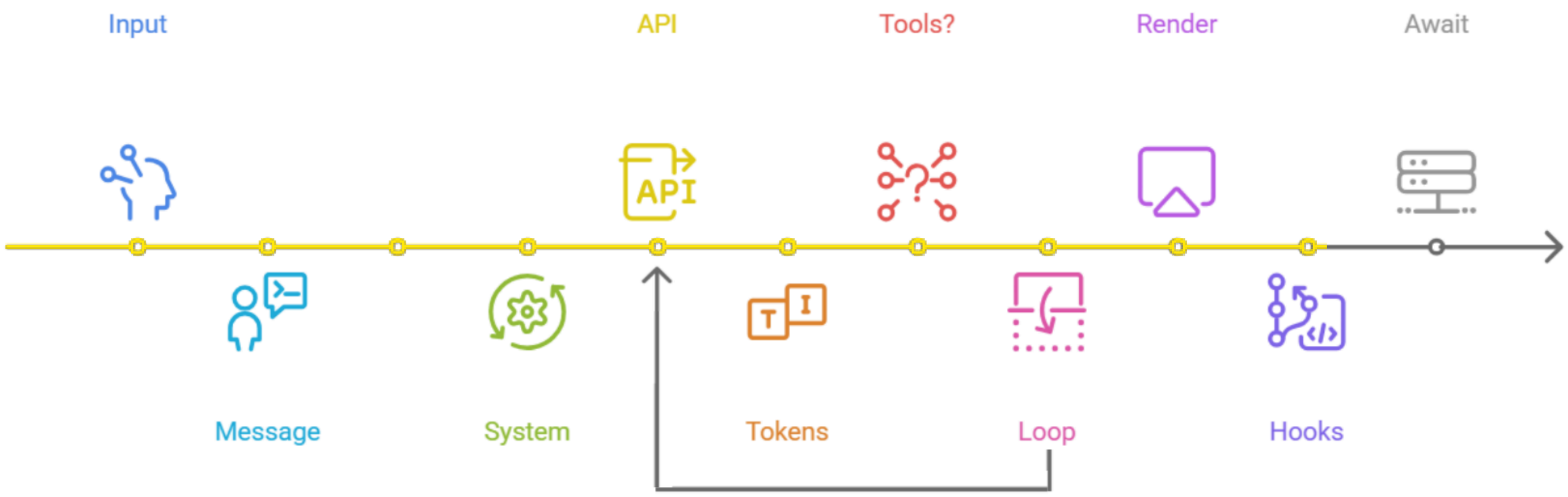
최종 메시지 상태를 터미널 UI에 표시합니다.



클로드 코드 분석

Hooks

답이 끝난 후 실행되는 작업들



클로드 코드 분석

Hooks

Auto Compact

대화 토큰 한계 확인 후 Compact 결정

메모리 추출

나중에 쓸 내용 골라서 기억 파일 저장

대화 스냅샷 저장

현재 대화 상태를 파일로 저장

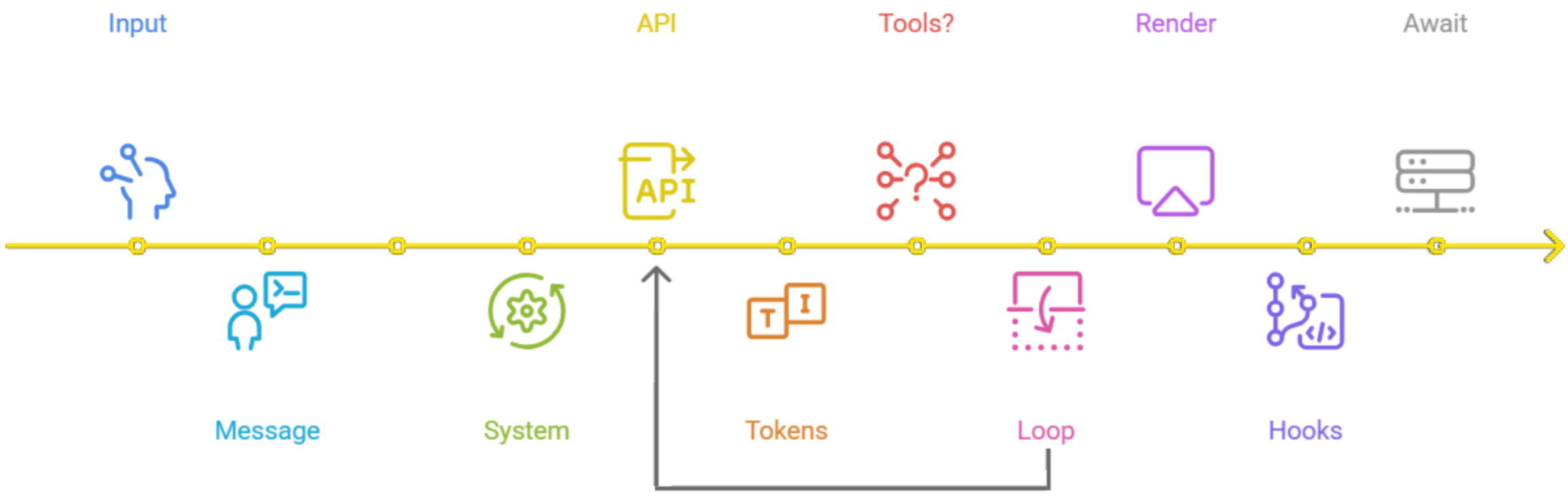
드림 모드(실험)

유휴 시간에 자율적으로 메모리 정리

클로드 코드 분석

Await

idle 상태로 돌아가,
사용자의 다음 입력을 기다립니다.



감사합니다.