



# Coding with AI



**[https://github.com/up1/  
course-ai-coding-2026](https://github.com/up1/course-ai-coding-2026)**



# Coding with AI



# Topics

Programming/Coding workflow  
Specification-Driven Development (SDD)  
Model-Driven Development (MDD)  
Coding with AI Agent workshop

Requirement

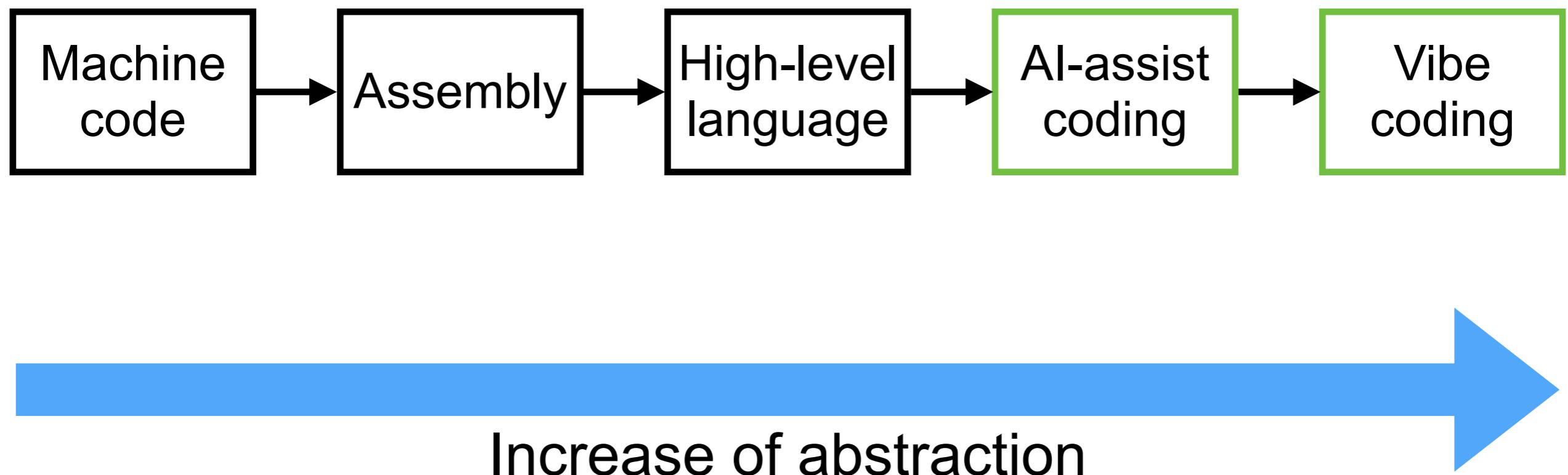
Planning

Coding

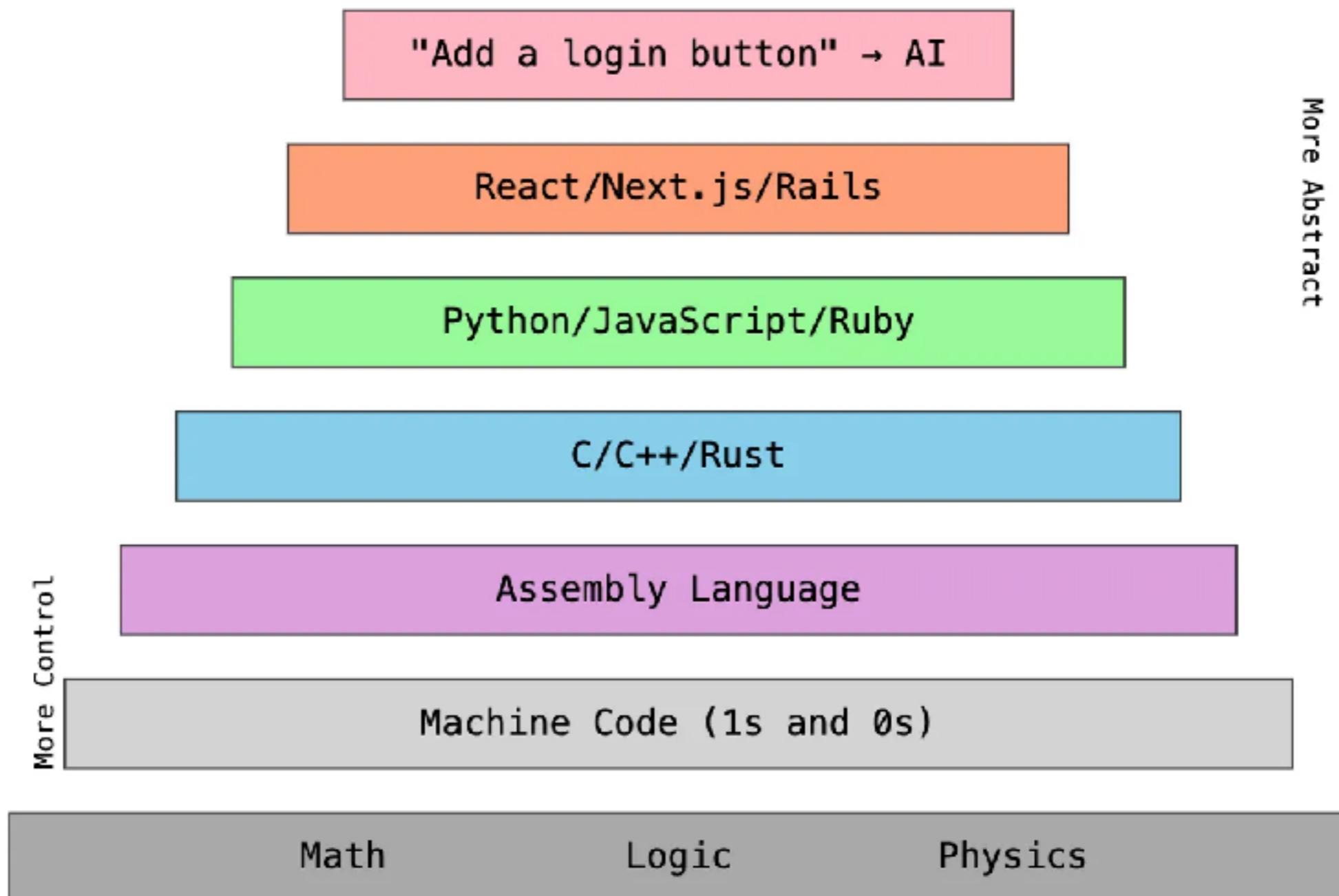
Validate and Testing



# History of Programming



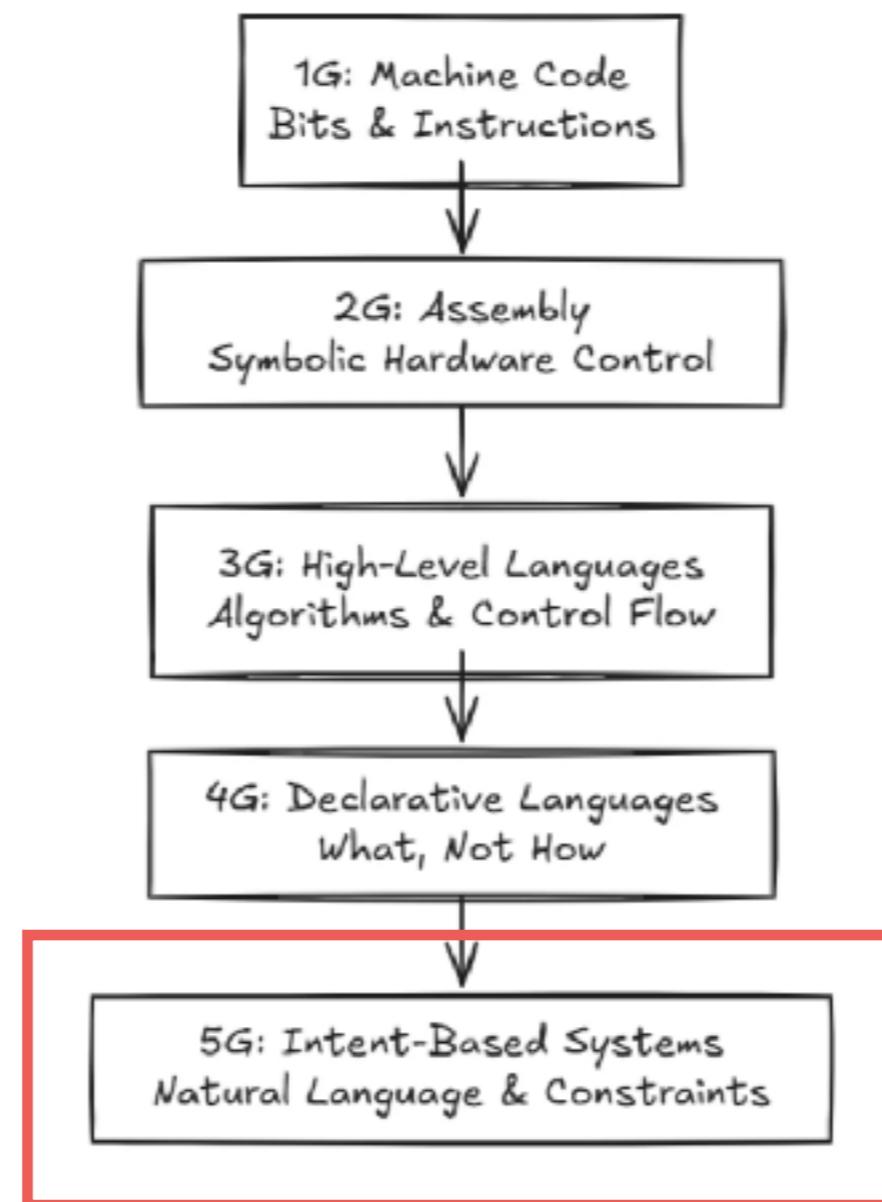
# The Modern Tower of Abstraction



<https://cline.bot/blog/from-assembly-to-ai-why-vibe-coding-is-just-another-chapter-in-our-abstraction-story>



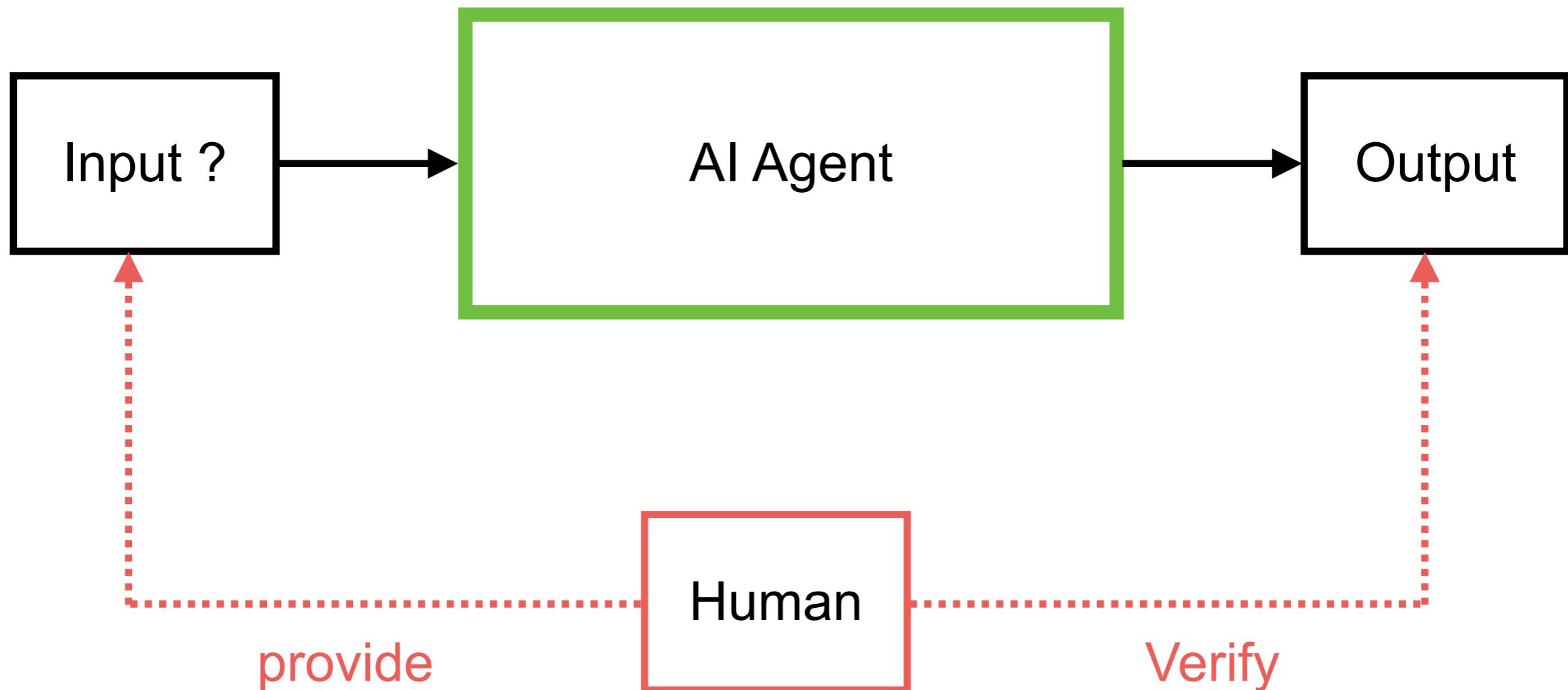
# Generation of Programming languages



<https://medium.com/@ajuatahcodingarena/generations-of-programming-languages-bed30d19ea8e>



# Coding with AI Agent



# AI Agent for Coding

AI Agent

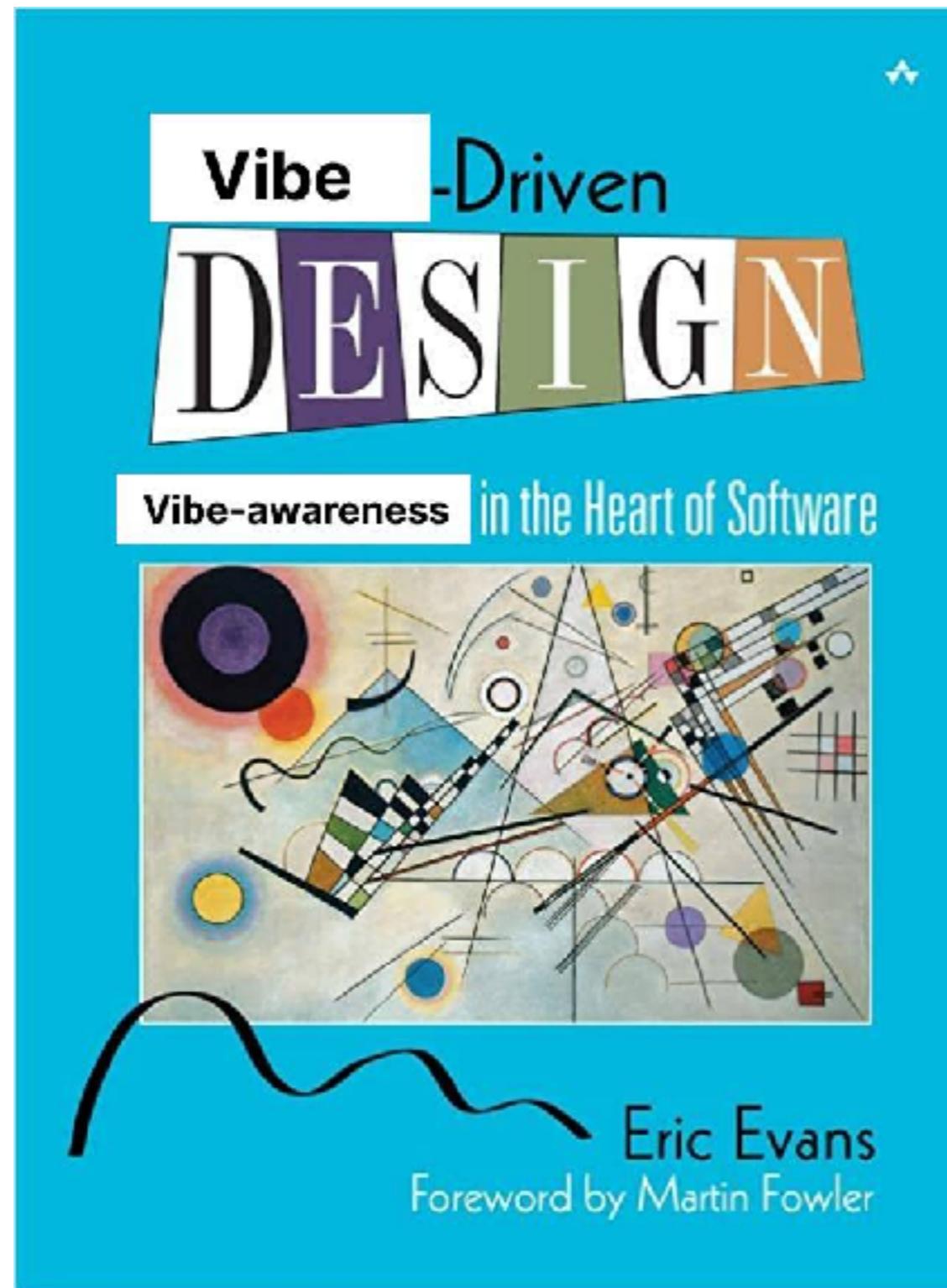
LLM

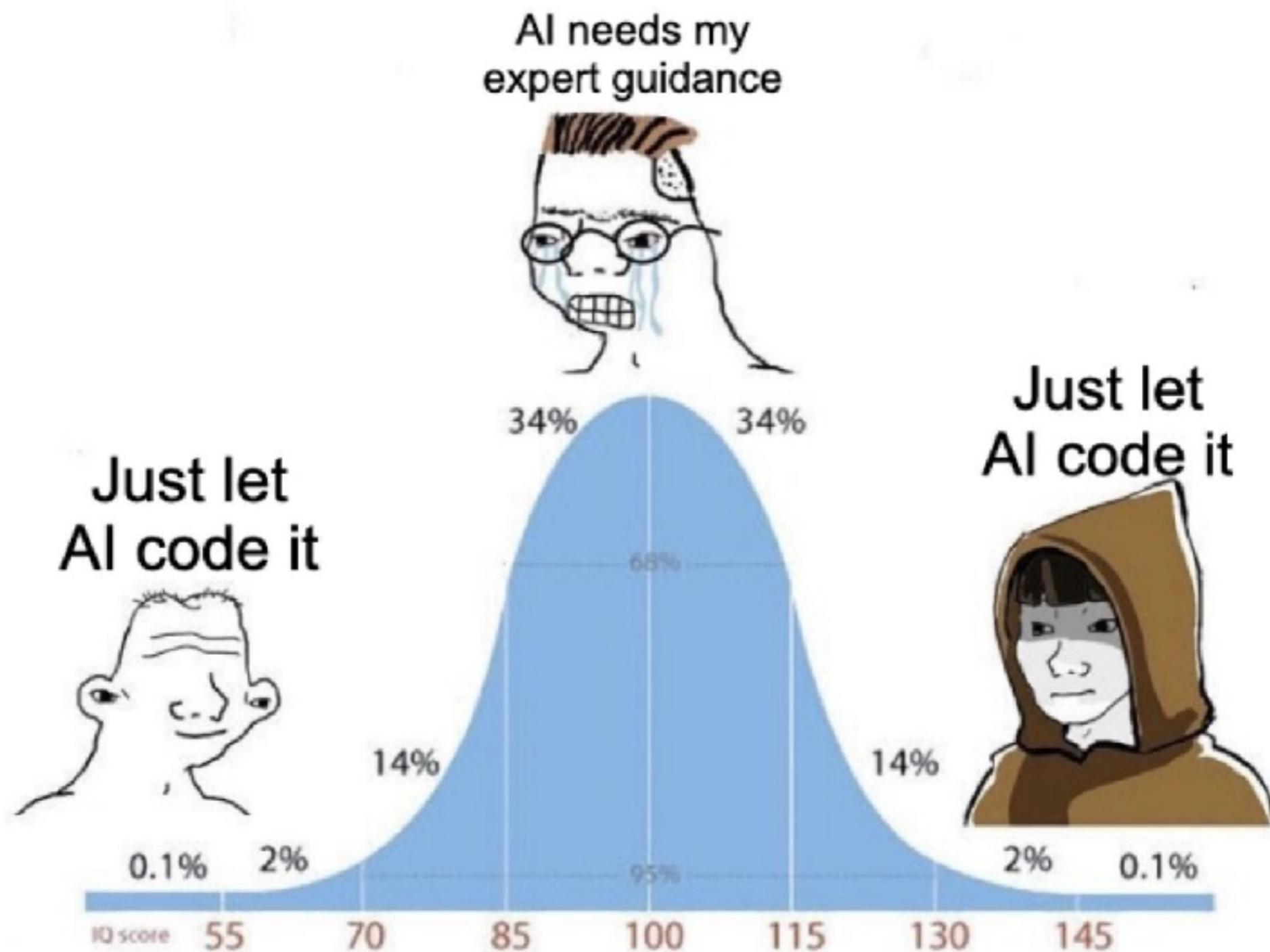
Code writing  
tools

Memory and  
Context

Other tools









Andrej Karpathy ✅

@karpathy

...

There's a new kind of coding I call "vibe coding", where you fully give in to the vibes, embrace exponentials, and forget that the code even exists. It's possible because the LLMs (e.g. Cursor Composer w Sonnet) are getting too good. Also I just talk to Composer with SuperWhisper so I barely even touch the keyboard. I ask for the dumbest things like "decrease the padding on the sidebar by half" because I'm too lazy to find it. I "Accept All" always, I don't read the diffs anymore. When I get error messages I just copy paste them in with no comment, usually that fixes it. The code grows beyond my usual comprehension, I'd have to really read through it for a while. Sometimes the LLMs can't fix a bug so I just work around it or ask for random changes until it goes away. It's not too bad for throwaway weekend projects, but still quite amusing. I'm building a project or webapp, but it's not really coding - I just see stuff, say stuff, run stuff, and copy paste stuff, and it mostly works.

6:17 AM · Feb 3, 2025 · 5.3M Views

<https://x.com/karpathy/status/1886192184808149383>



Development

© 2020 - 2026 Siam Chamnankit Company Limited. All rights reserved.

# Vibe Coding

**Software development practice** that uses artificial intelligence (AI) to generate functional code from **natural language prompts**, accelerating development, and making app building more accessible, especially for those with limited programming experience

<https://cloud.google.com/discover/what-is-vibe-coding>



# Types of Vibe coding ?

Pure vibe coding

**Responsible AI-assist development**



## PAIR PROGRAMMING



Reviews, Tests and Understands  
with experience



# Workflows ?

Application level

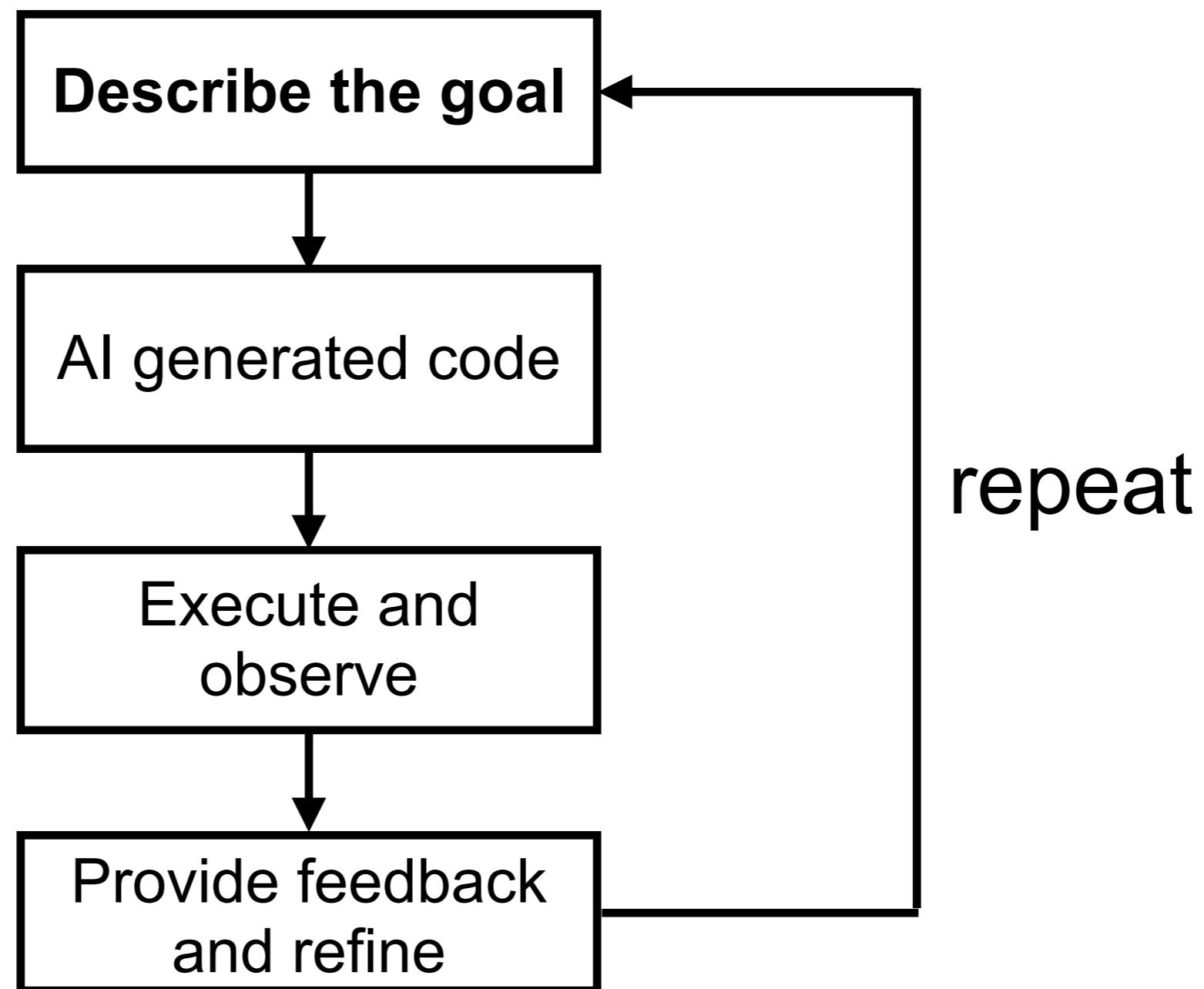
Code level

<https://cloud.google.com/discover/what-is-vibe-coding>



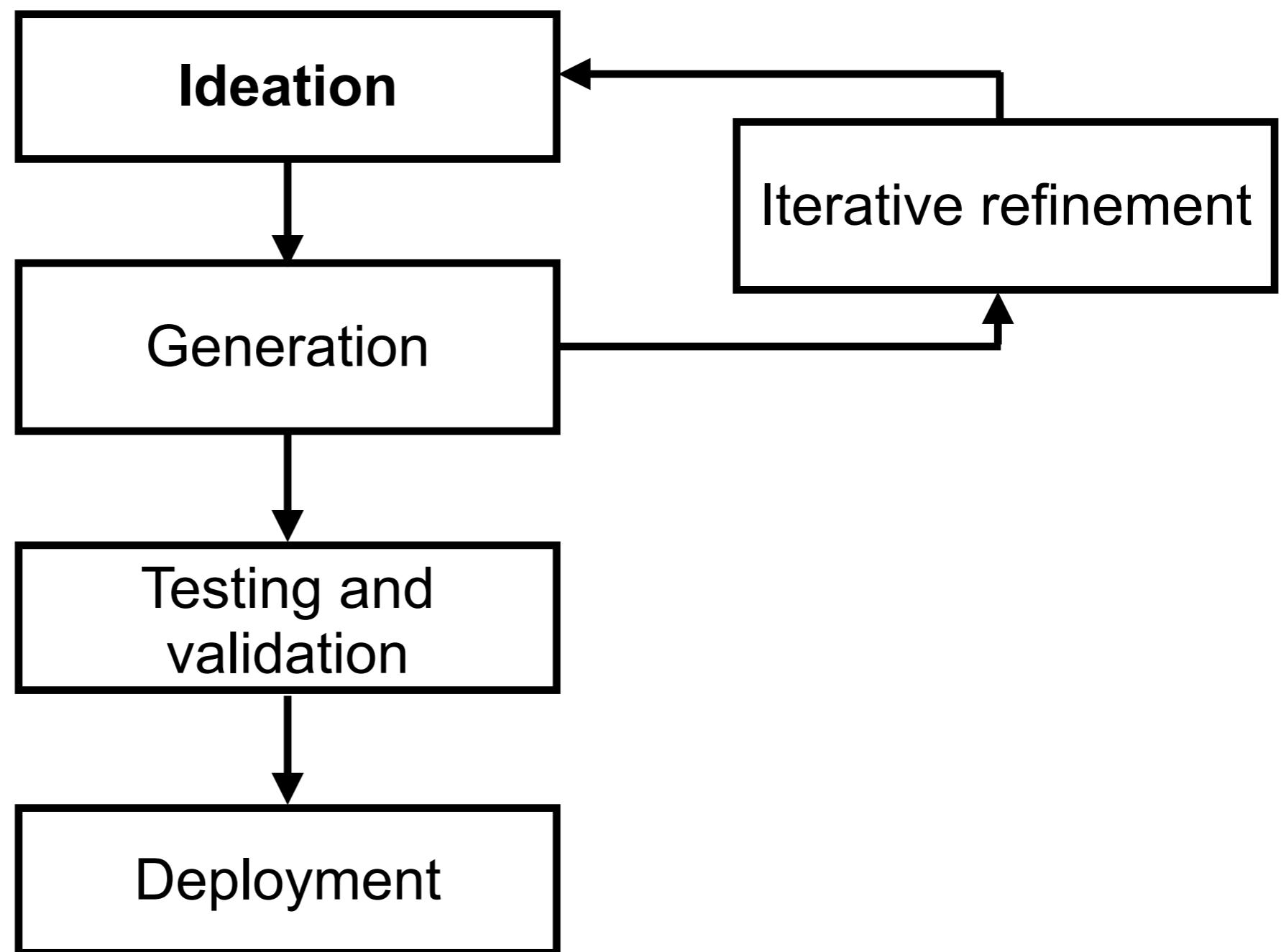
# Code level workflow

Conversational loop to create a specific code



# Application level workflow

Talking about high-level idea from concept to deployment



# AI Agent for Coding

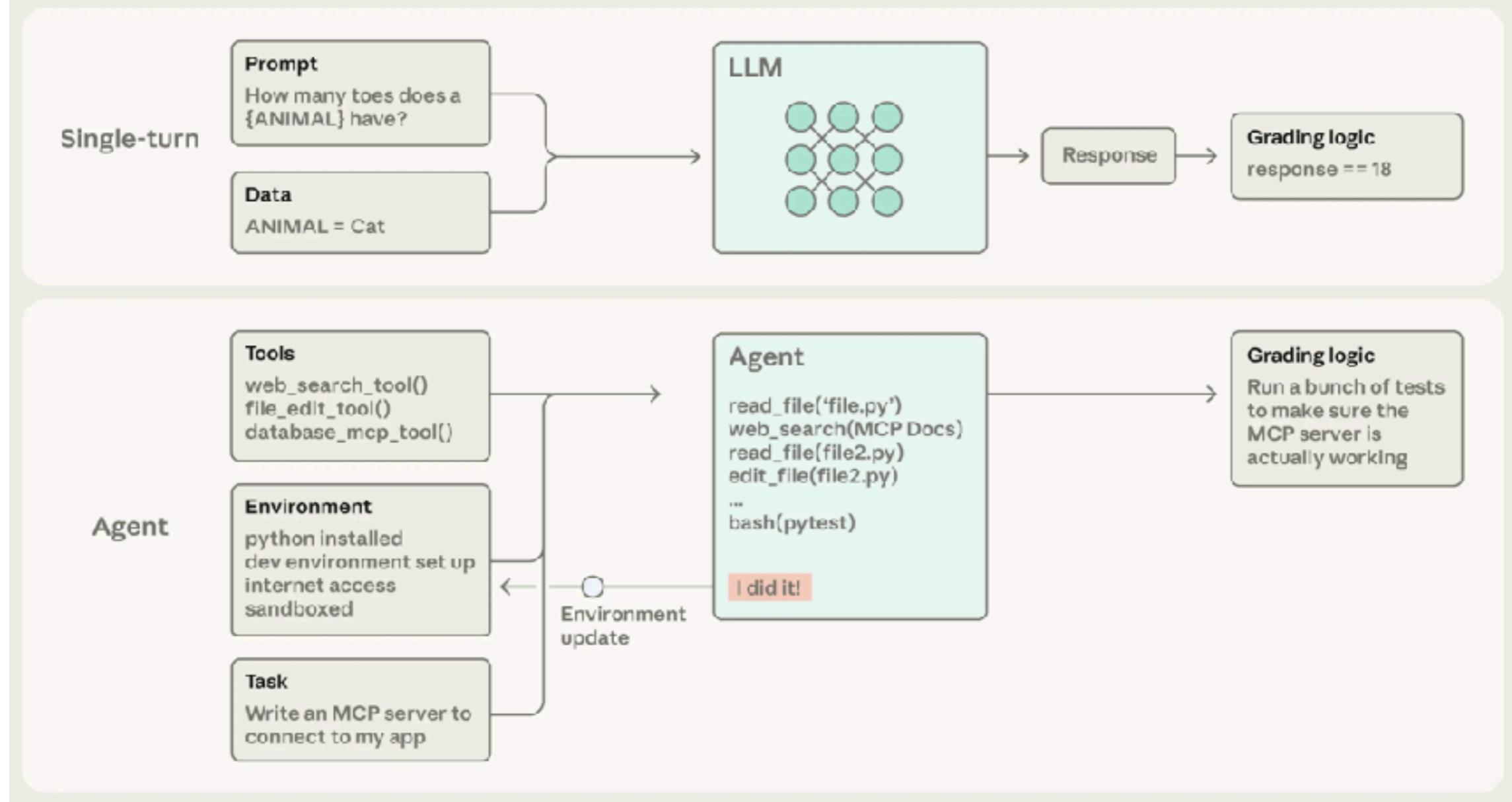


# AI Agent for Coding



# AI Agent for Coding

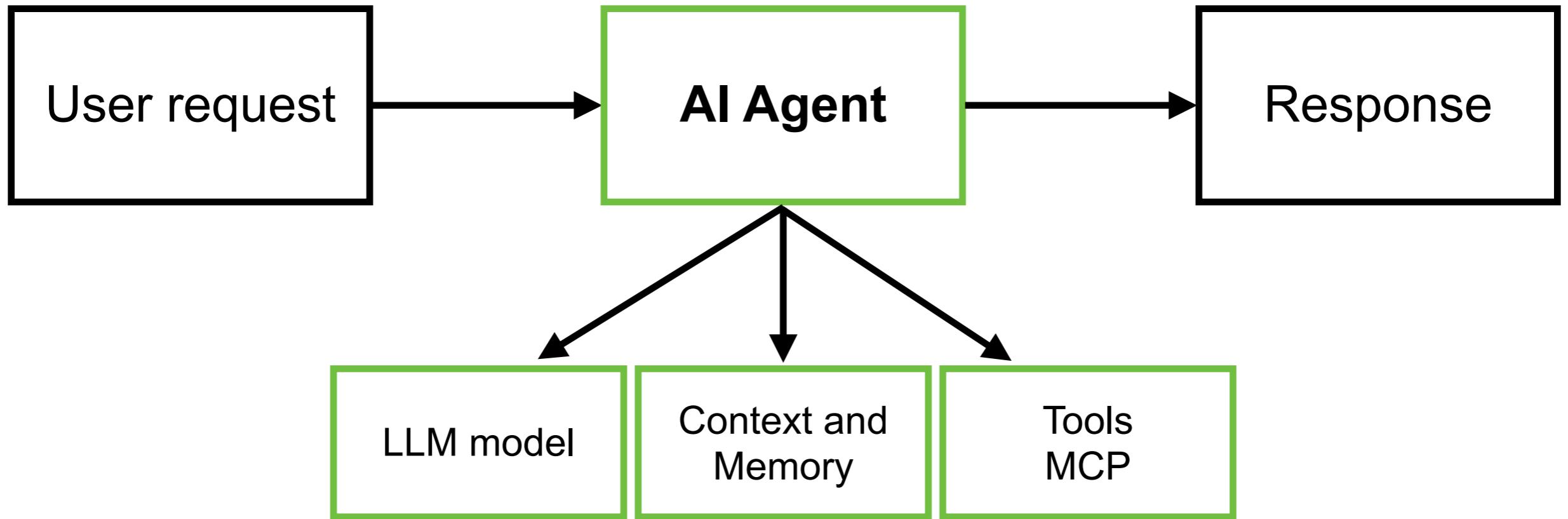
## Comparison: Single-Turn vs Agent Evaluations



<https://www.anthropic.com/engineering/demystifying-evals-for-ai-agents>



# Simple AI Agent



How to manage context or knowledge  
for Agent ?



# Types of Knowledge/Context ?

## Shared context

Static Knowledge

Prompt engineering

Global Instructions

AGENTS.md

CLUDE.md

Rules

Sub-agent and Modes

## Task context

Dynamic Knowledge

Memory

Graph/Vector database

MCP tools

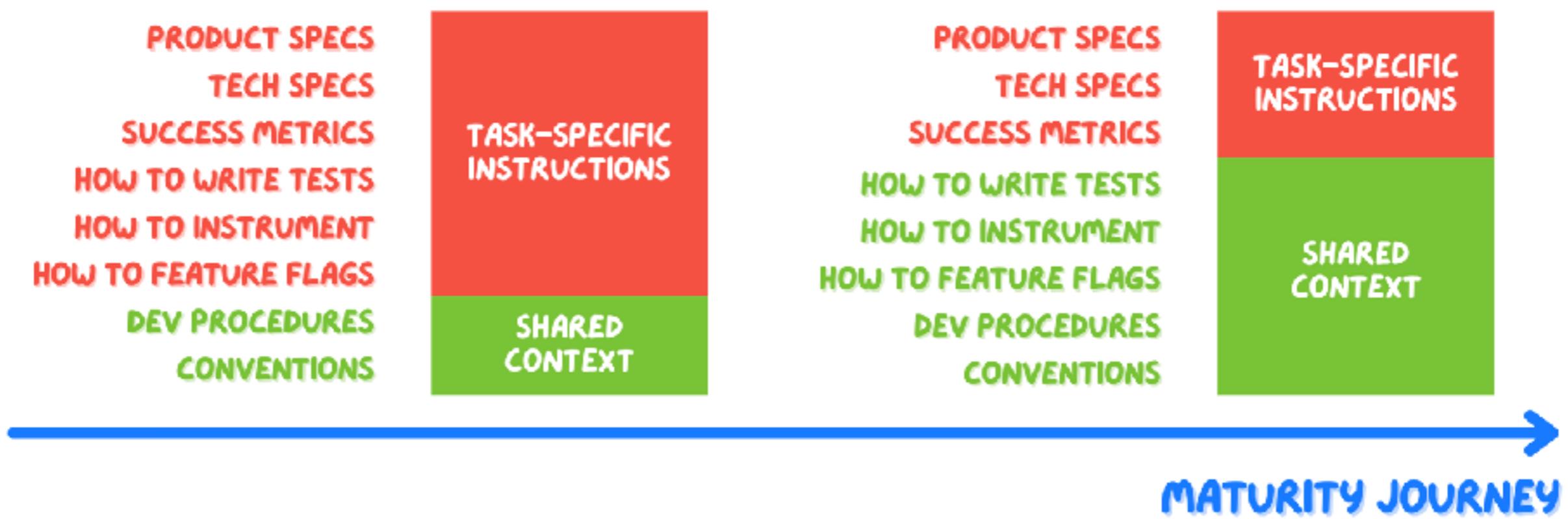
Specific Instructions

Agent SKILLS

Context engineering ?



# Types of Knowledge/Context ?



<https://refactoring.fm/p/context-capabilities-and-tech-hubs>



# **Vocabulary before start !!**



# Vocabulary before start !!

**Rules**

**Prompts**

**MCP servers**

**Instructions**

**Commands**

**Subagents**

**Modes**

**Hooks**

**Agent Skills**



# Rules and instructions

Static context

Rules are files that contain essential context  
Codebase, best practices, common pitfalls

Business requirements

Prevent LLM models from hallucinating  
and to ensure specific standards



# Commands and Prompts

Static context

Text instructions sent to the agent

Repeat workflow that can be triggered on demand

Allow teams to share standardized workflows  
And run complex tasks conditionally

**git add -> generate commit message -> git commit**



# Subagents and Modes

Specific persona or task to specialized agent  
Agent, Plan, Ask, Edit mode  
Work AI as a team

Make the agent's behavior more deterministic, reliable  
and discoverable



# Agent Skills

Dynamic context

Create workflow + resources + script

Skills do not bloat the initial context window

Only loaded when the agent determines (relevant with task)

Provide an open standard for sharing capabilities with a team without hitting context limits

<https://agentskills.io/>



# Hooks

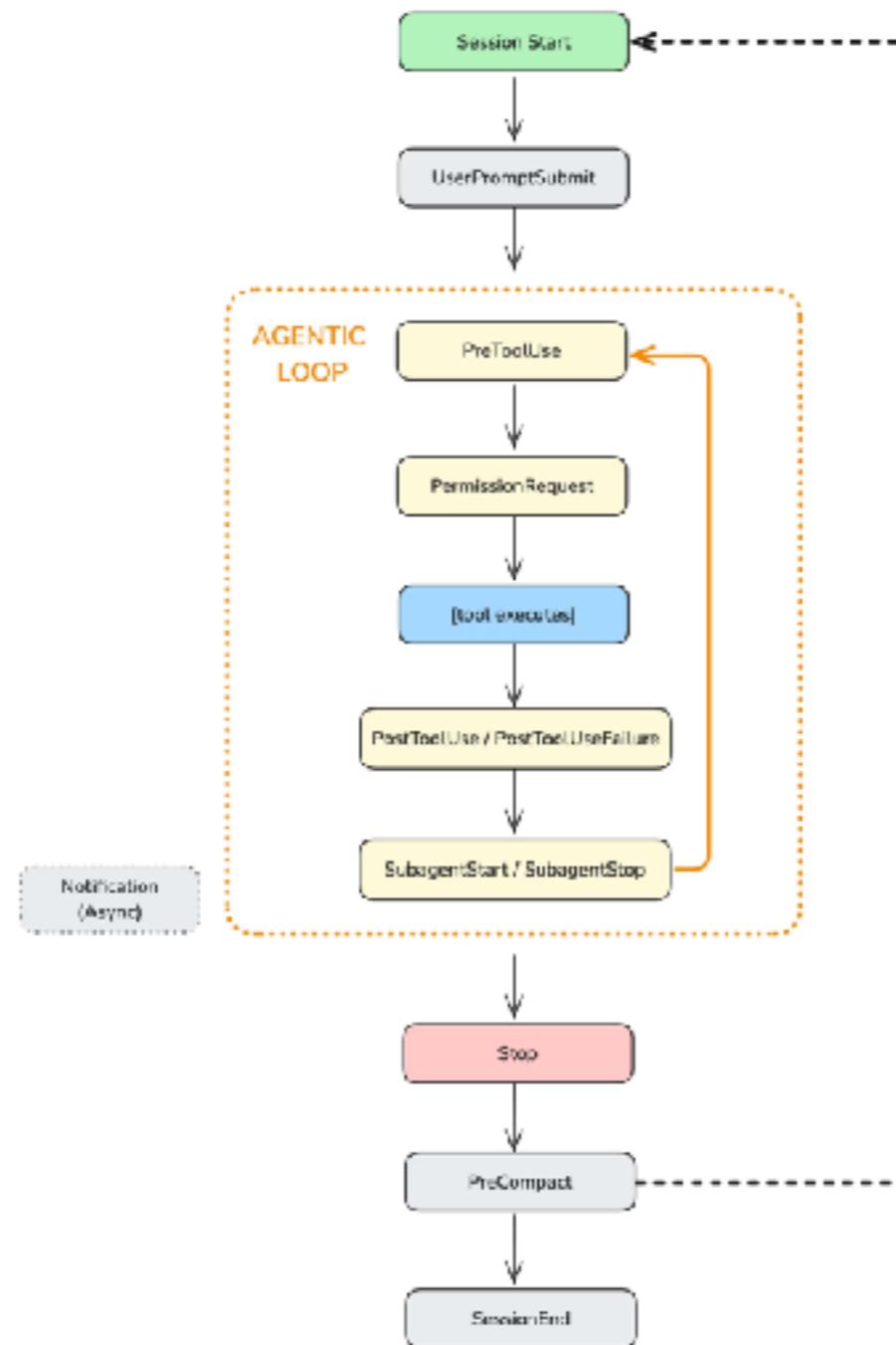
Customize and extent behavior of agent  
Triggers for actions before/after conversation

Setup environment, run test, cleanup resources,  
Notification, automated pre-commit check ...

<https://code.claude.com/docs/en/hooks>



# Hooks in Agent's Lifecycle

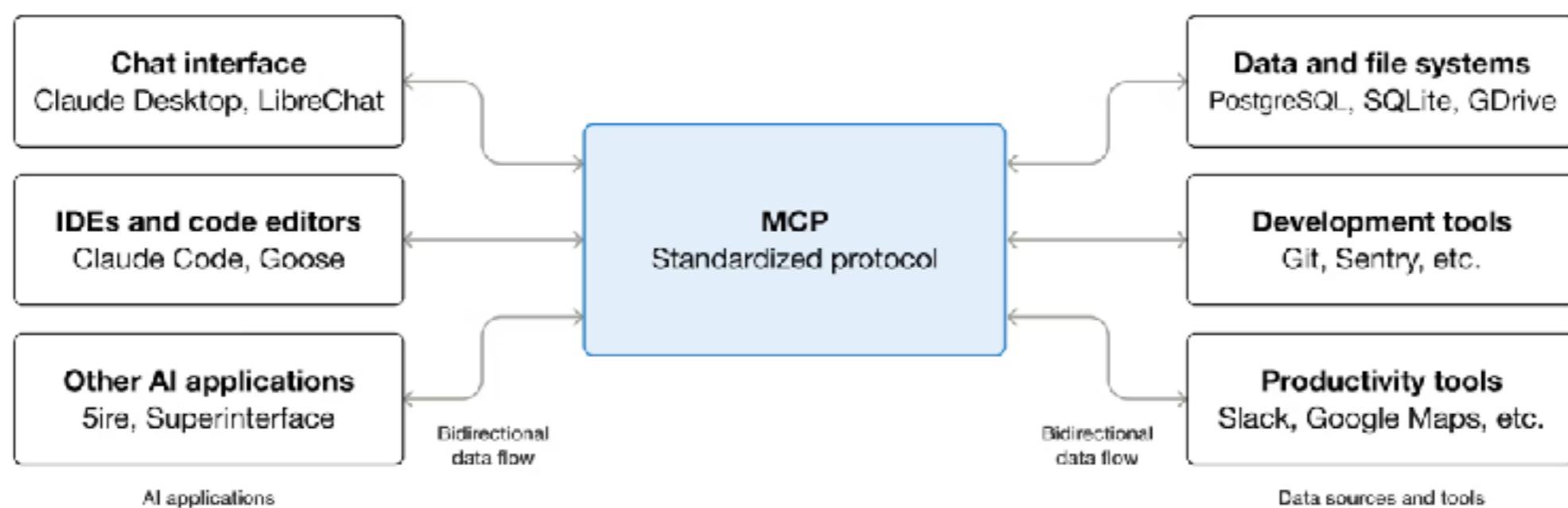


<https://code.claude.com/docs/en/hooks>



# MCP (Model Context Protocol)

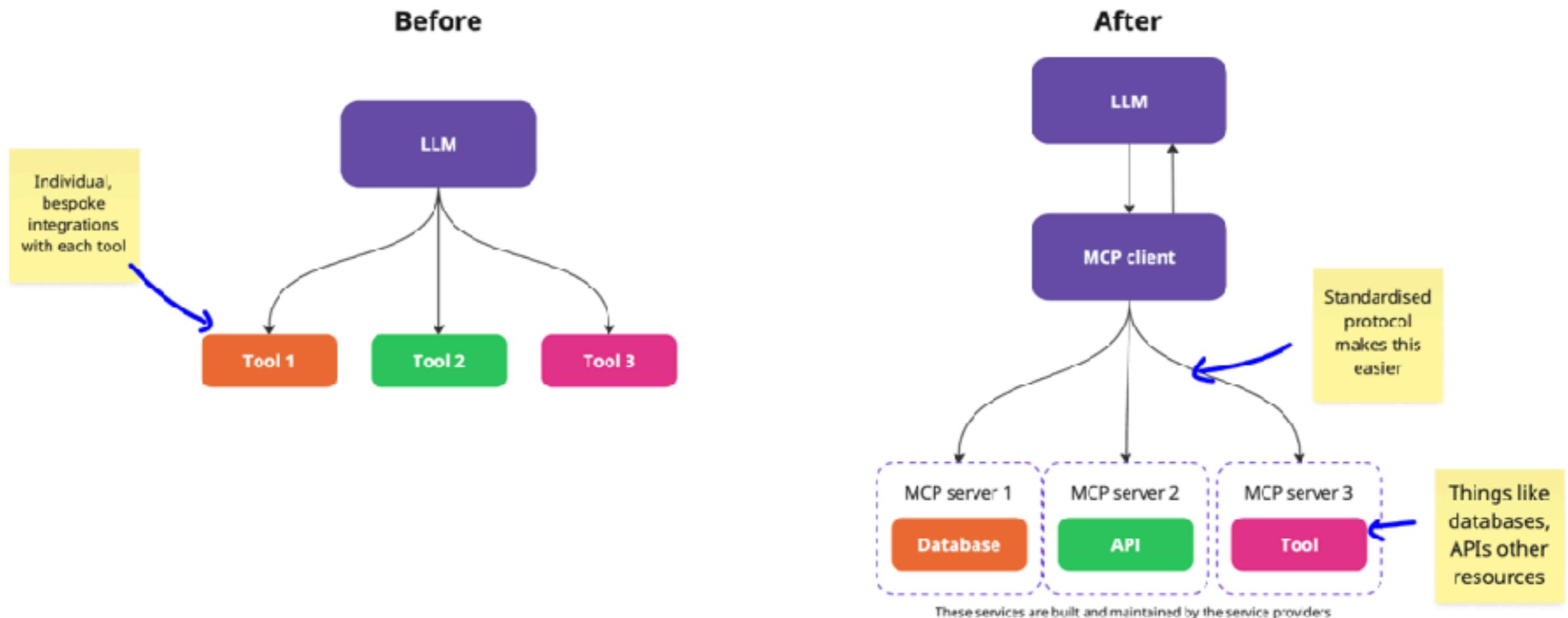
Open standard for connecting AI app to external system



<https://modelcontextprotocol.io/>



# MCP (Model Context Protocol)



<https://departmentofproduct.substack.com/p/mcp-explained-a-simple-guide-for>

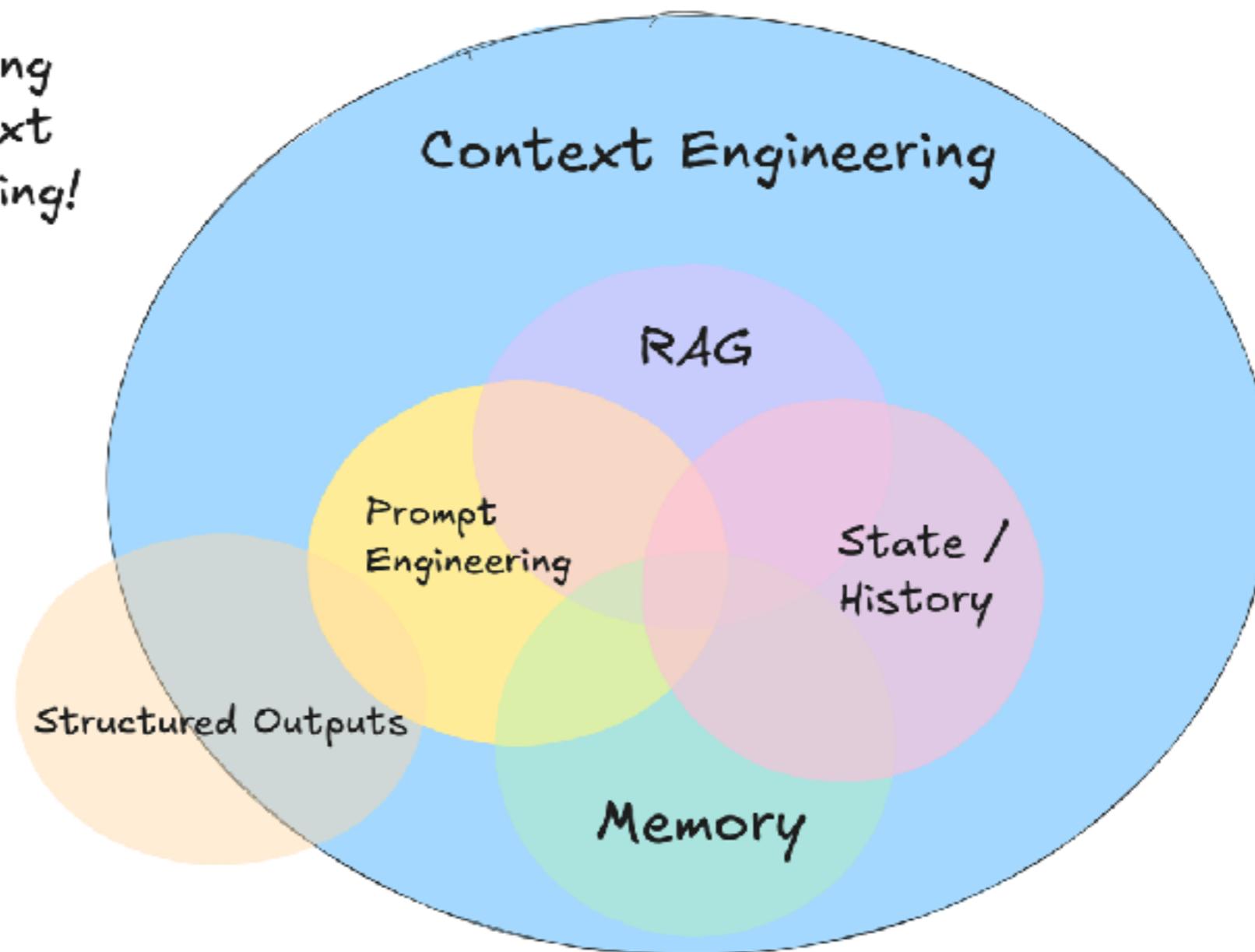


# How to manage context ?



# Context Engineering

Everything  
is Context  
Engineering!



<https://www.promptingguide.ai/guides/context-engineering-guide>



# Context Engineering

Right information

Right tools

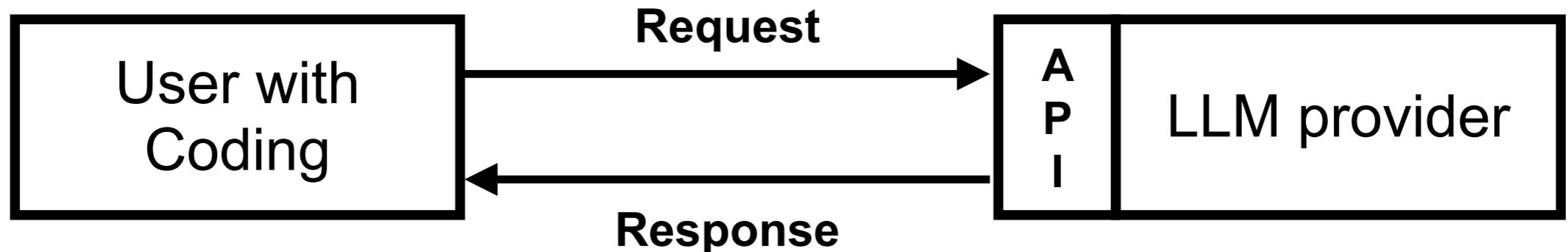
Structured  
context



# Context window size ?

Amount of text a model can process and remember at once time

Measure with **token**



# Context window size of models

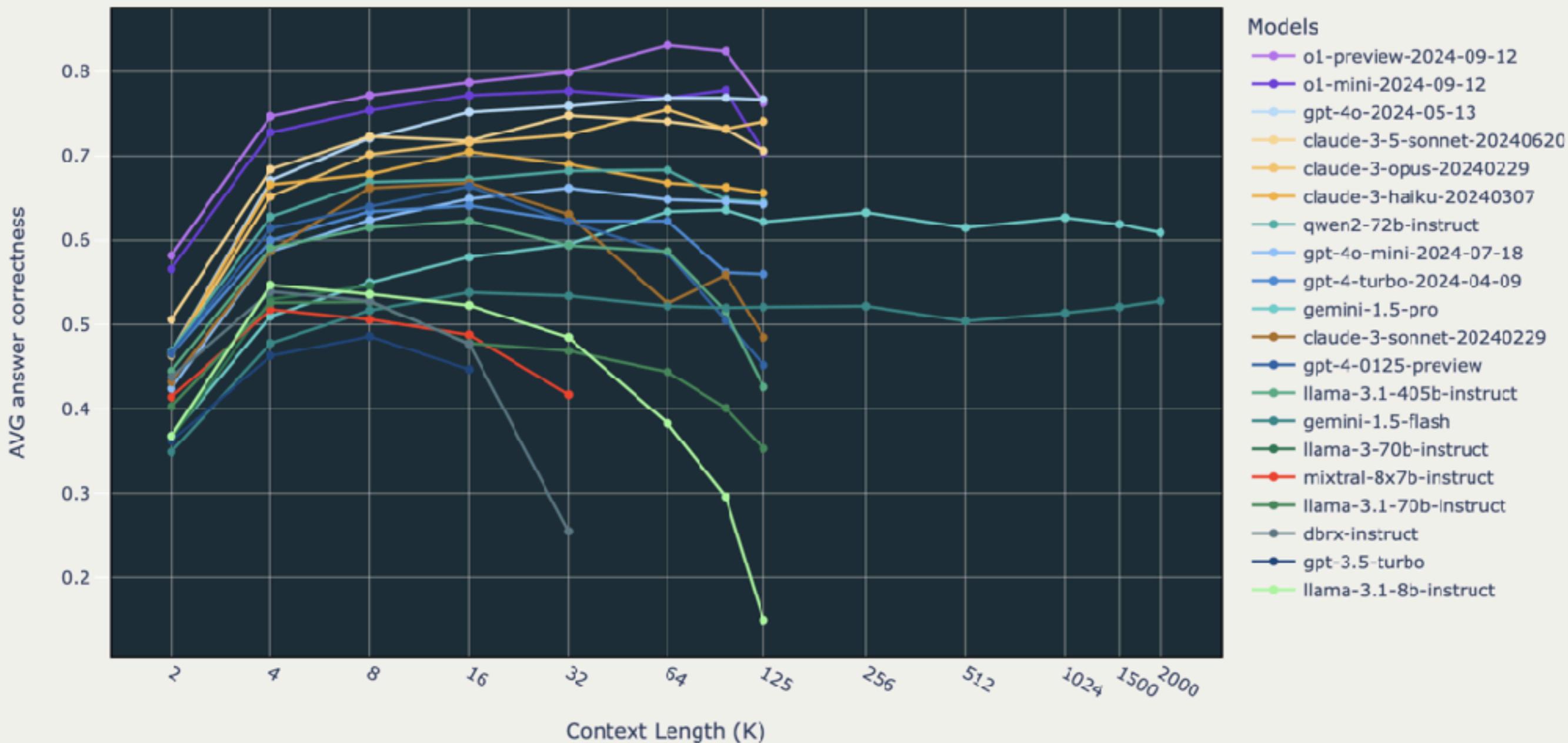
Different models may have different window size

Model name	Context window size
GPT 4.1	1M
Gemini 1.5, 2.5	1M
Llama 4	1M
Claude Sonnet 3.7	200,000
o4	200,000



# Long context LLM is better !!

Long Context RAG Performance of LLMs



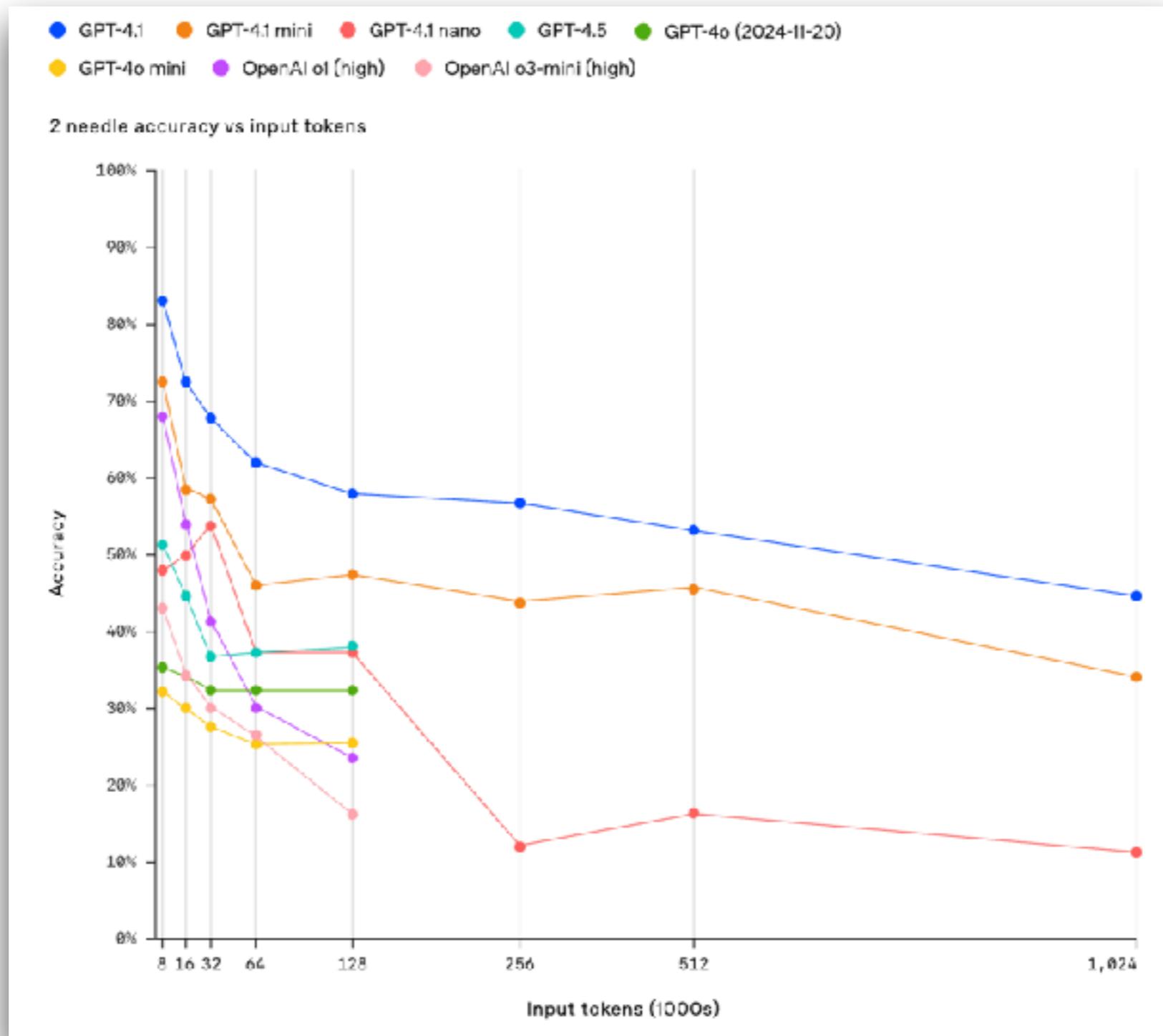
<https://arxiv.org/abs/2411.03538>



Development

© 2020 - 2026 Siam Chamnkit Company Limited. All rights reserved.

# Long context LLM is better !!



<https://openai.com/index/gpt-4-1/>



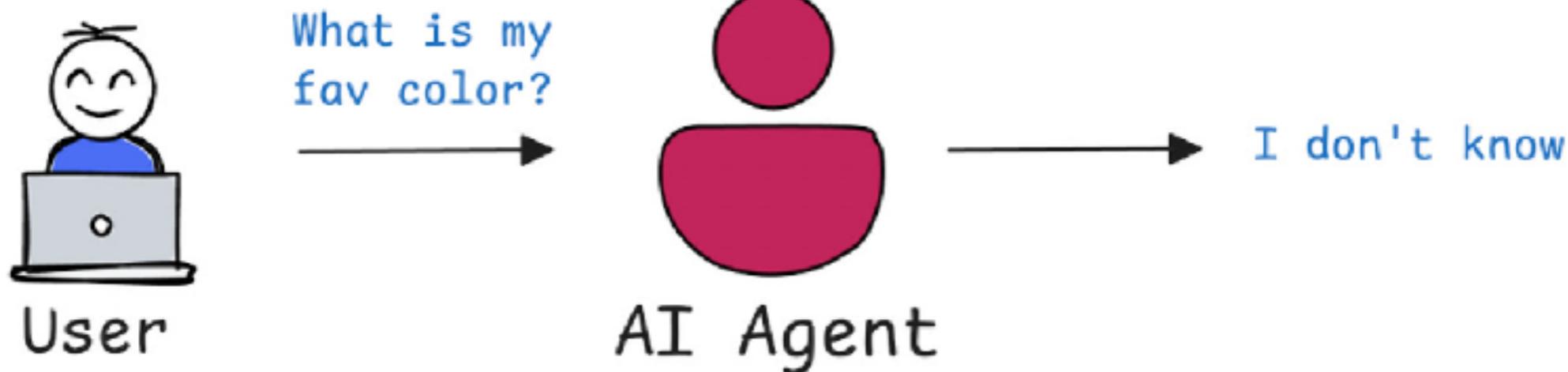
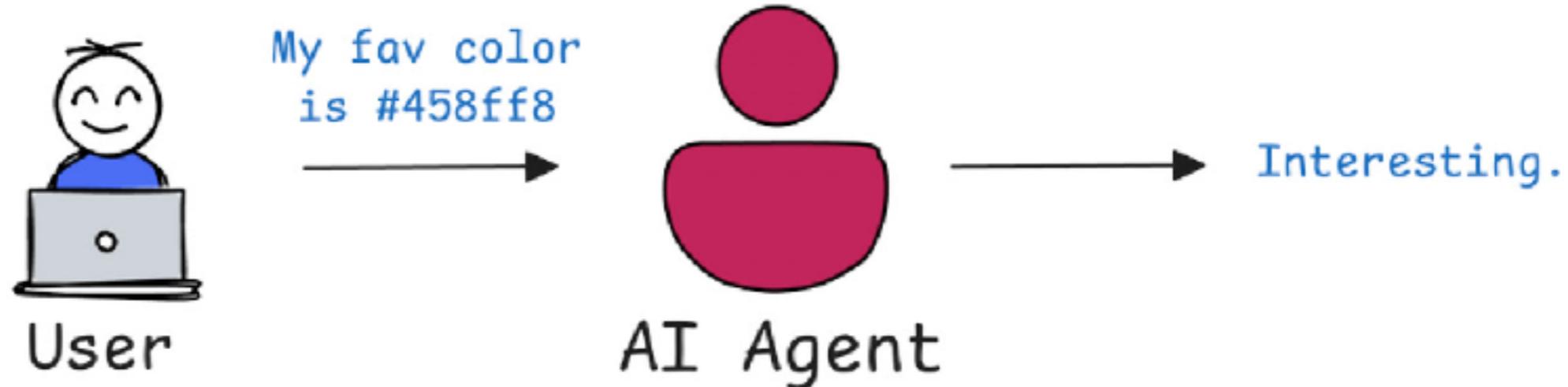
# **Better context == Better result**



# **Agent Memory Memory Bank**



# Interaction without memory

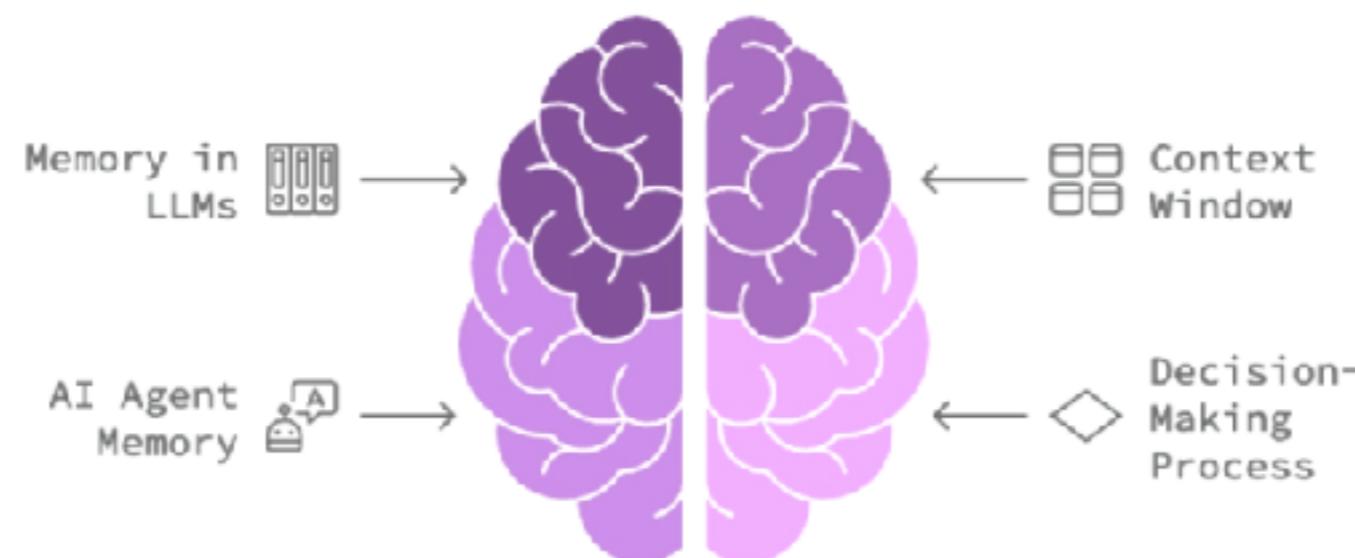


# Why AI Agent memory important ?

Enable personalized and context-aware interactions

Improve decision-making and adaptability

Enhance LLM's performance and response

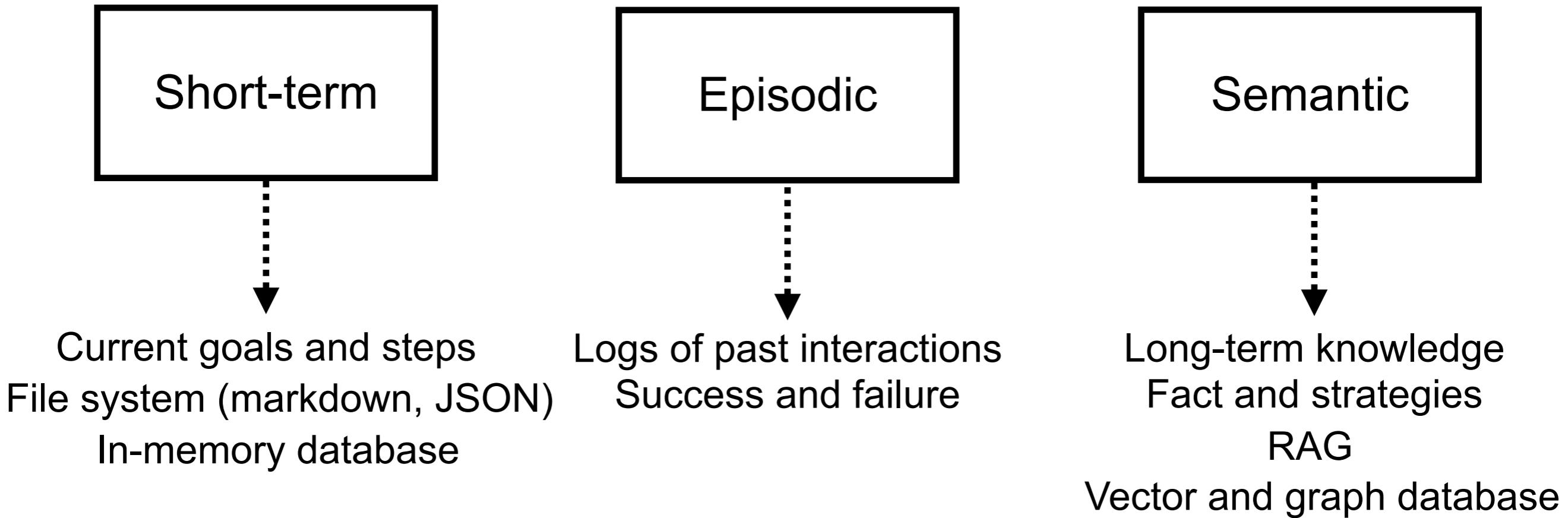


<https://www.falkordb.com/blog/ai-agents-memory-systems/>



# Types of Memory

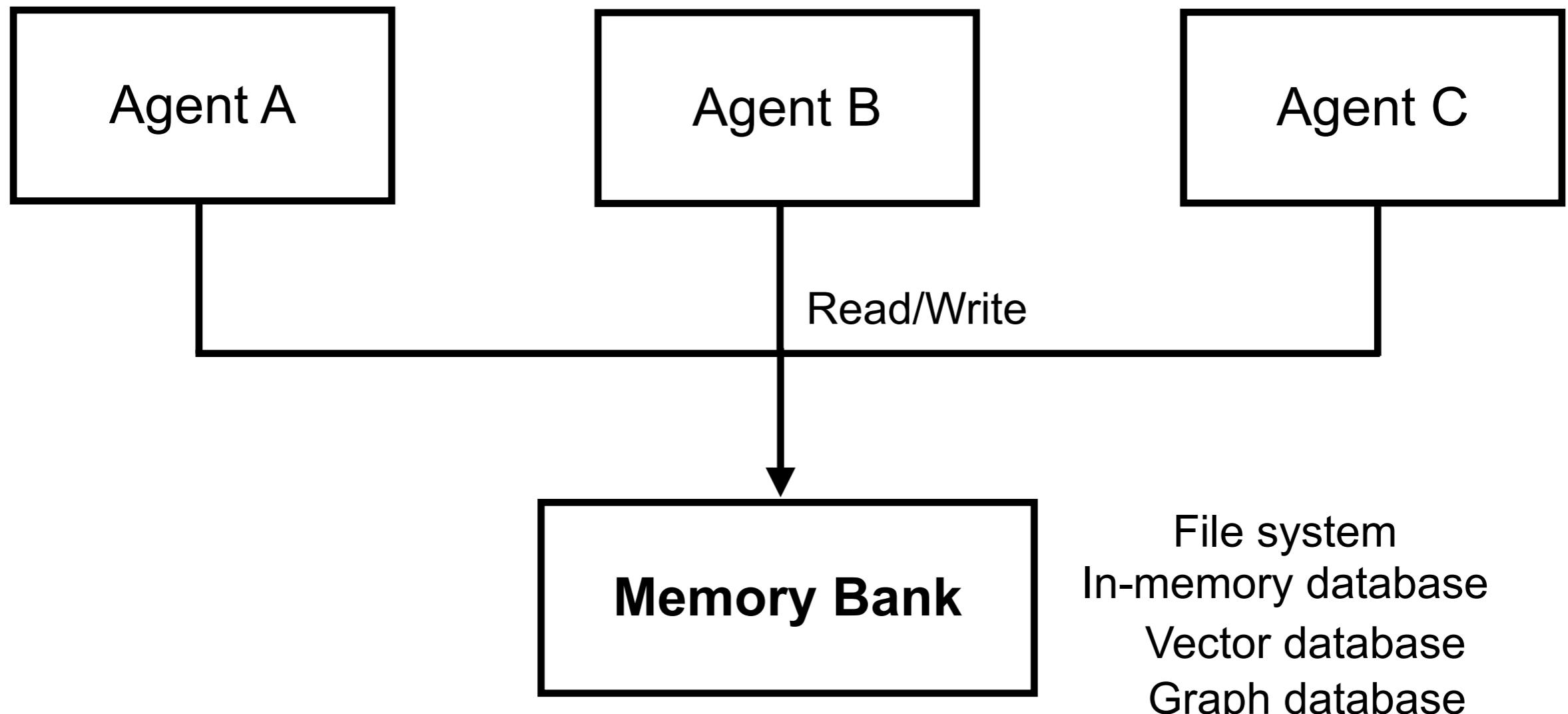
To make intelligent decisions  
Agents combine different types of memory



<https://bhavishyapandit9.substack.com/p/how-memory-works-in-agentic-ai-a>

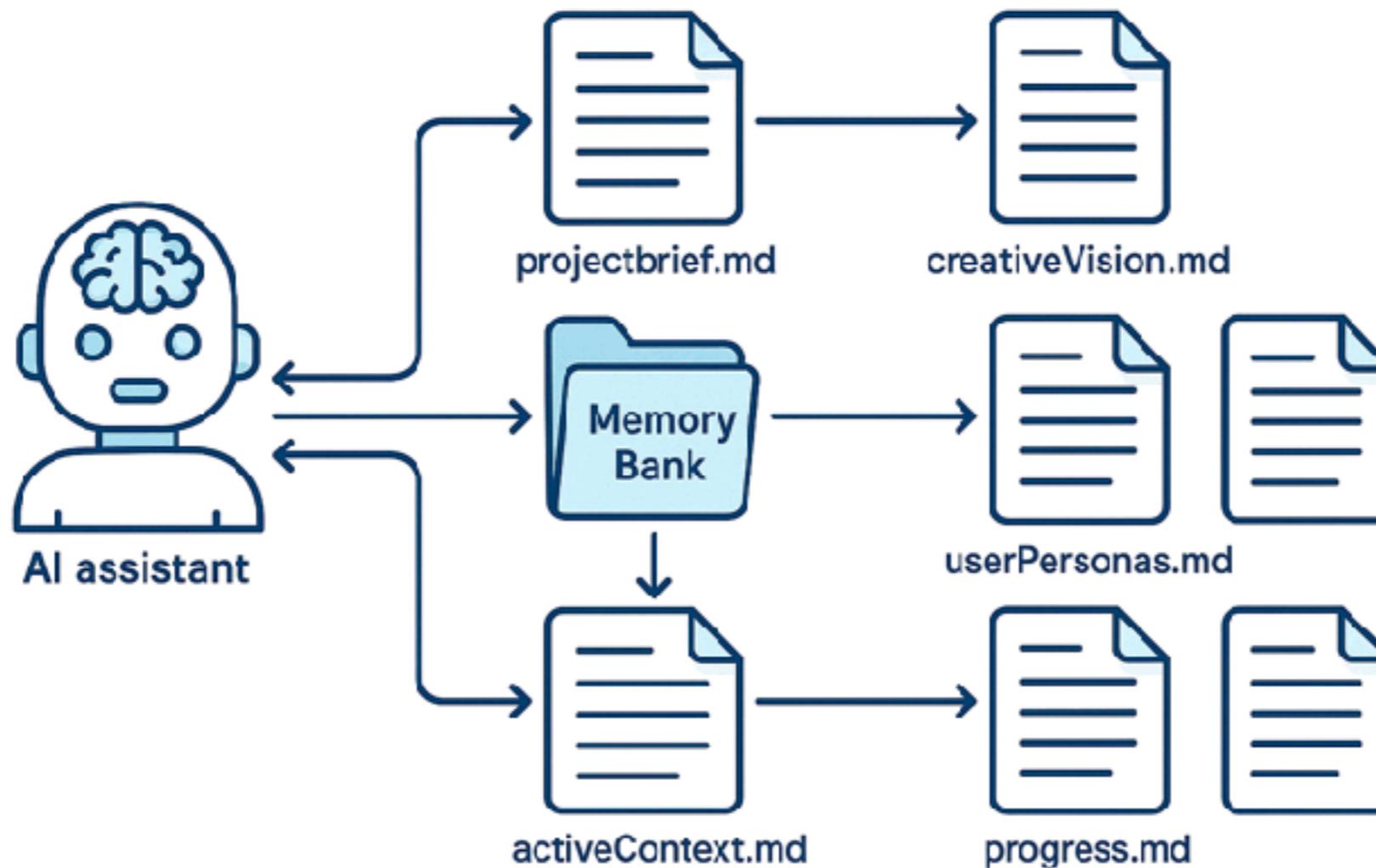


# Shared memory between Agent



# Cline Memory Bank

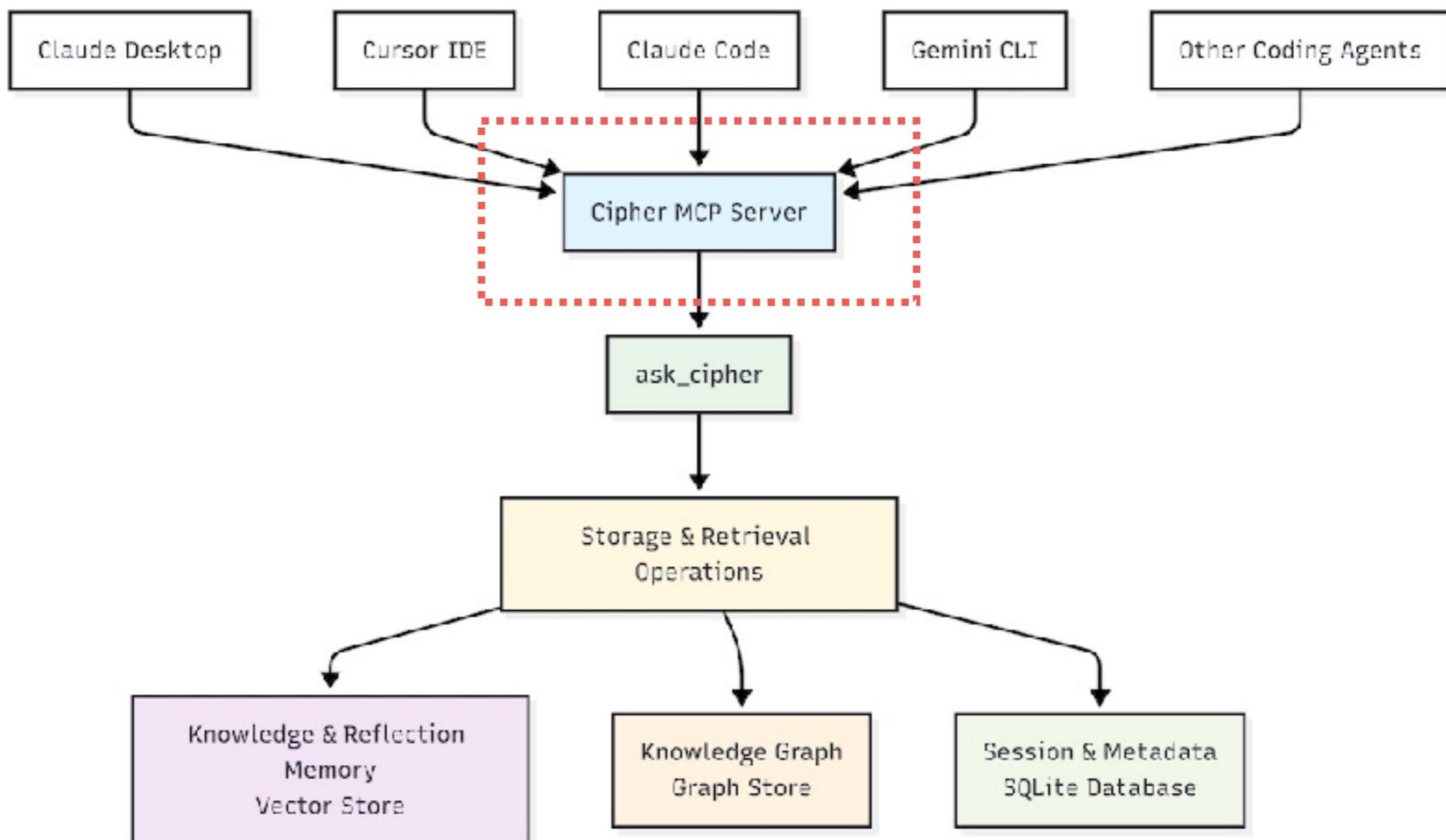
## Working with file system



<https://docs.cline.bot/prompting/cline-memory-bank>



# Memory Bank via MCP



<https://www.somkiat.cc/share-memory-for-ai-ide/>



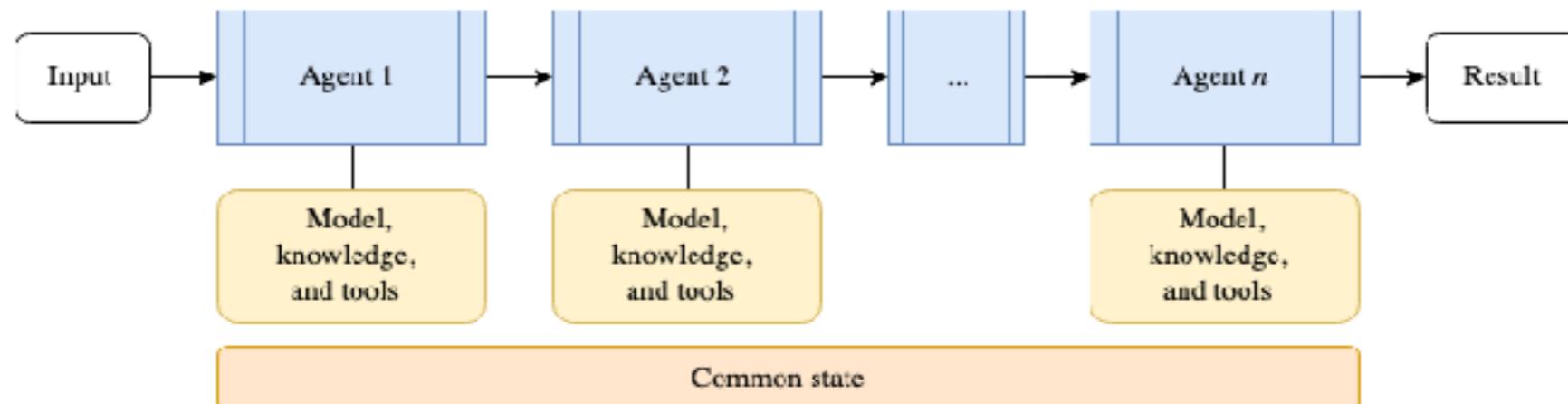
# **Sub-agent and Multi-agent !!**

**Reduce size of knowledge !!**



# Monolith to Multi-agent

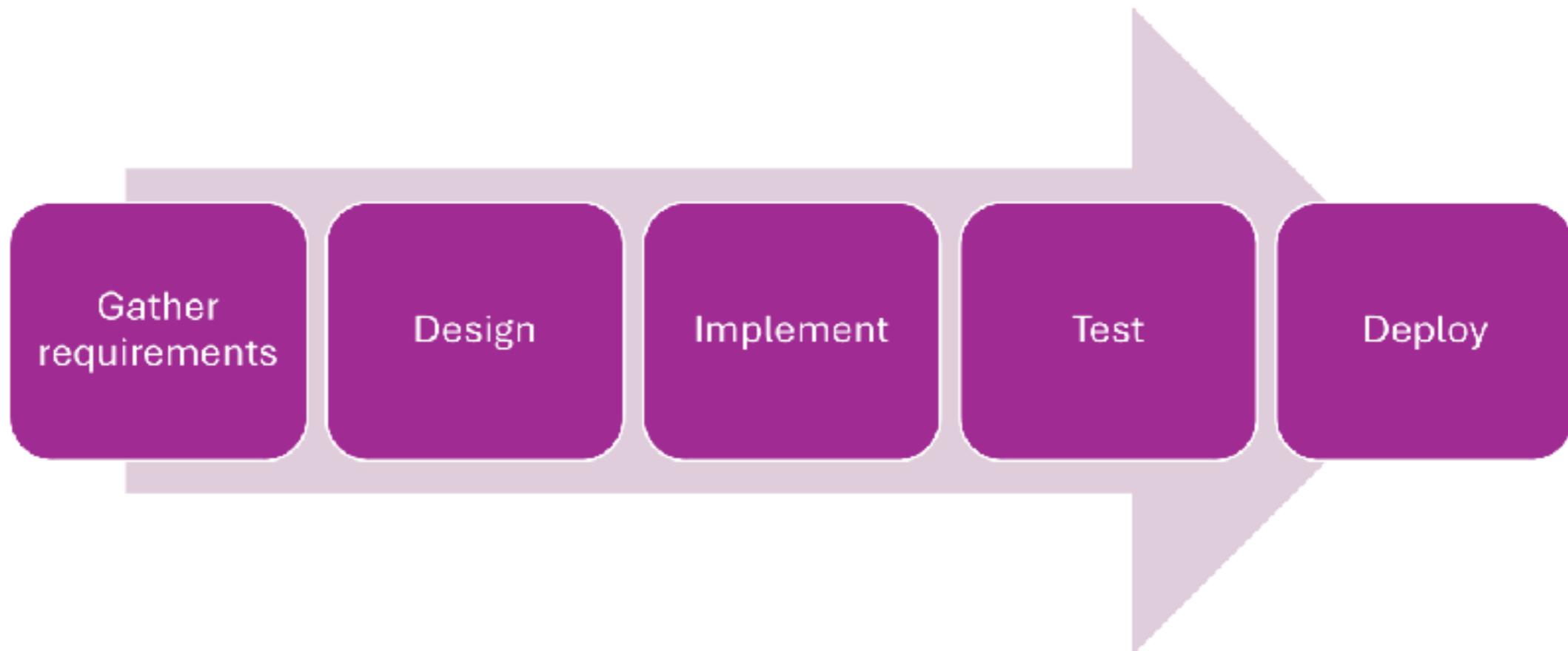
Specialization  
Scalability  
Maintainability  
Optimization



<https://learn.microsoft.com/en-us/azure/architecture/ai-ml/guide/ai-agent-design-patterns>



# Design your Agents ?



<https://devblogs.microsoft.com/dotnet/introducing-microsoft-agent-framework-preview/>



# Design your Agents ?

Each step may contain subtasks

Different specialists have different steps

**Progress isn't always linear**

Bugs found during testing may send back to implementation



# Design your Agents ?

Full Stack Agent !!

Frontend

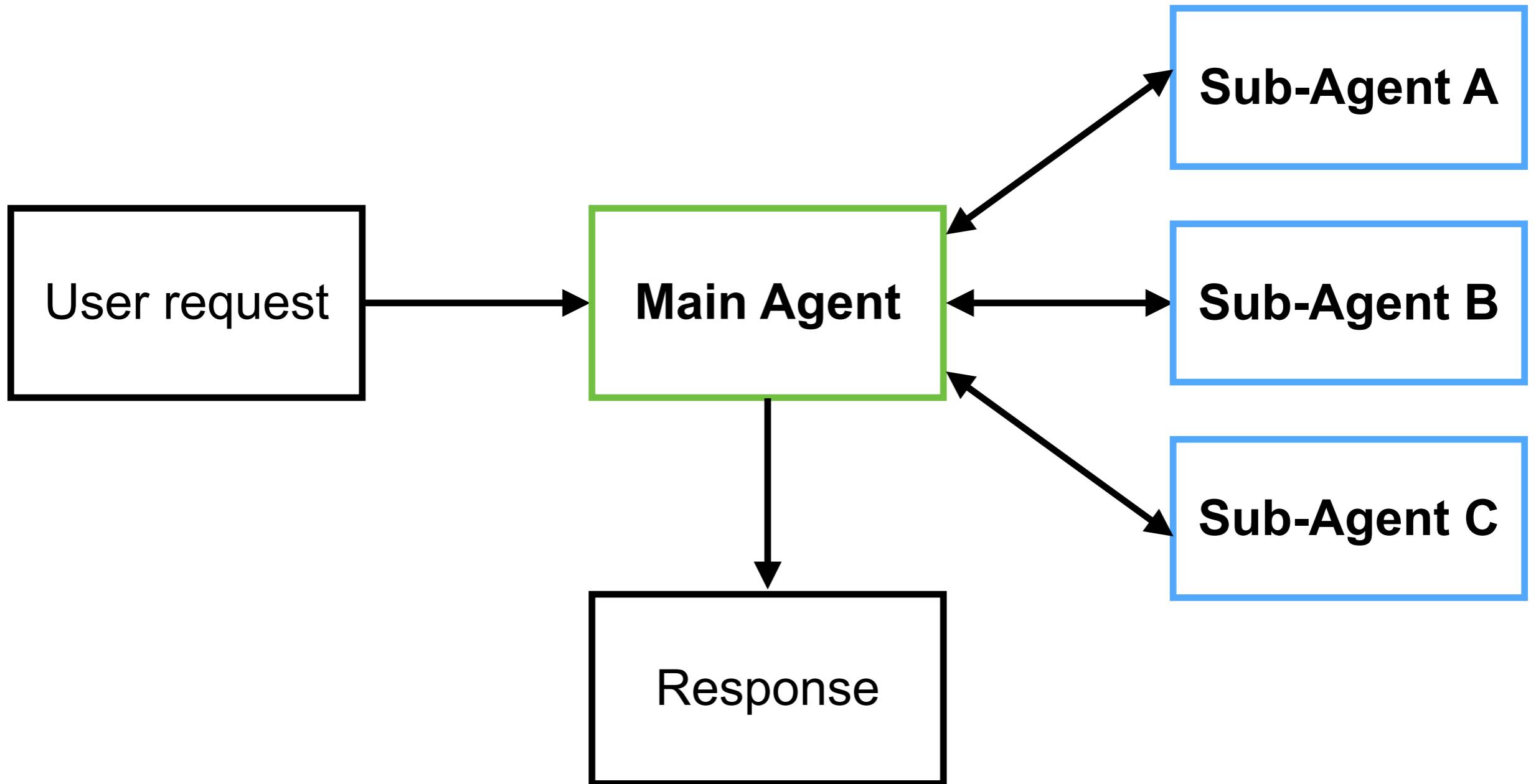
Backend

Database



# Multi AI Agents !!

Sub-agent with centralized orchestration

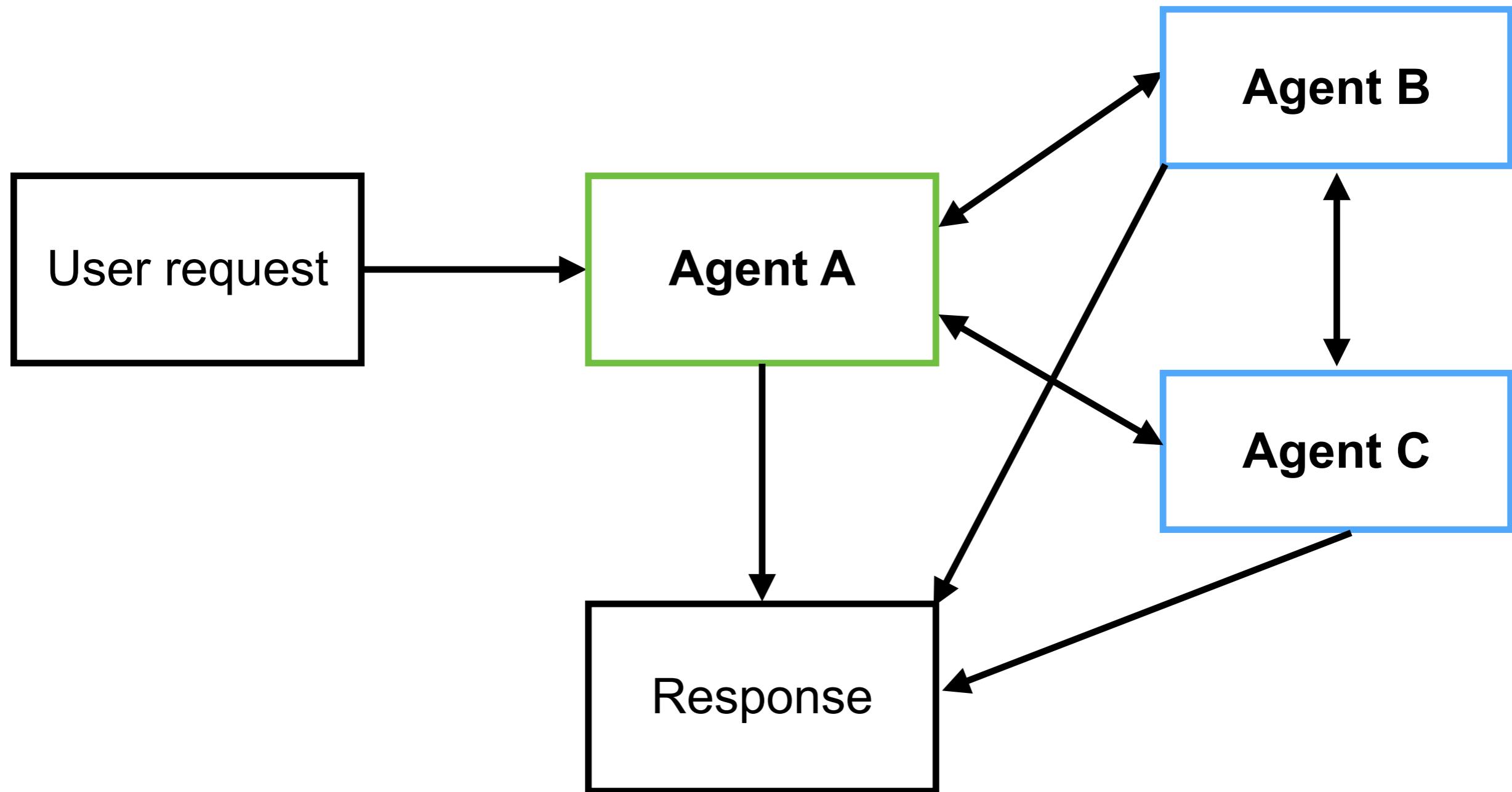


<https://www.blog.langchain.com/choosing-the-right-multi-agent-architecture/>



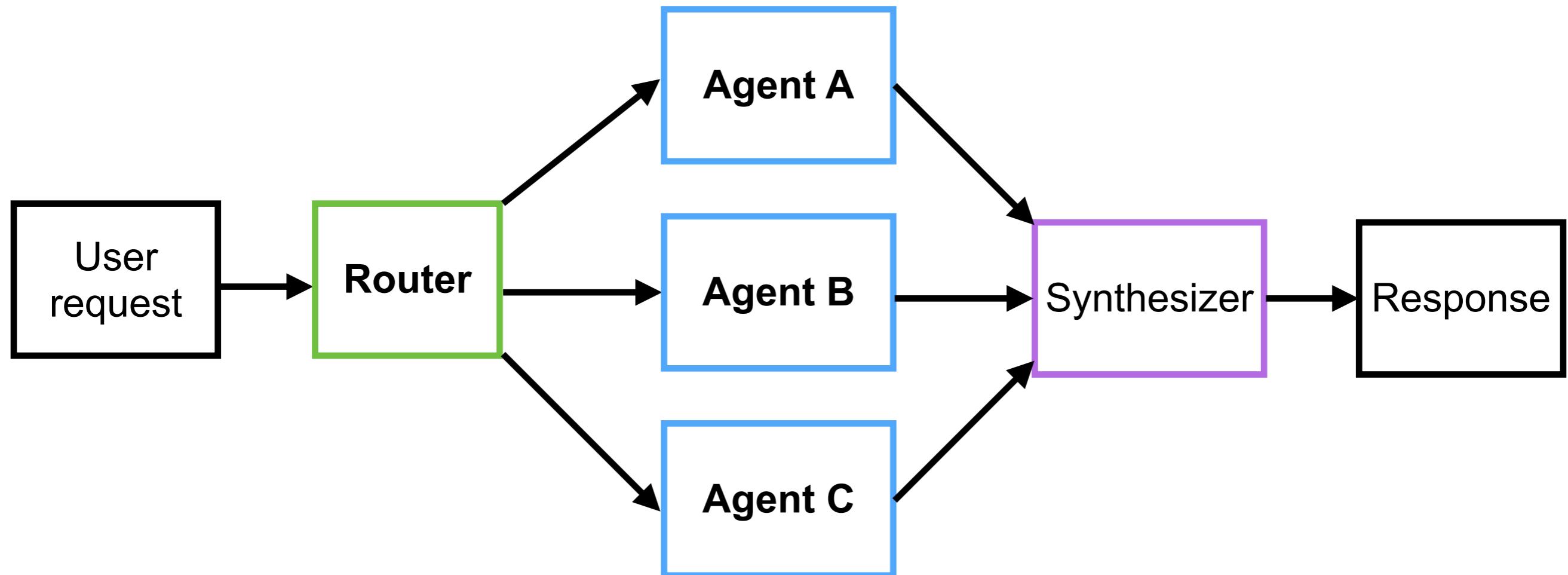
# Multi AI Agents !!

State-driven transitions based-on conversation context



# Multi AI Agents !!

Router and parallel run



# Prompt vs Agent Skills ?



Command

<https://agentskills.io/>



# Agent Skills

Moving from a paradigm of  
“prompt” to “programming by instruction”

Agent loads specialized prompts and knowledge on-demand

Specialized  
task

Reduce  
repetition

Compose  
capabilities

Efficient  
loading

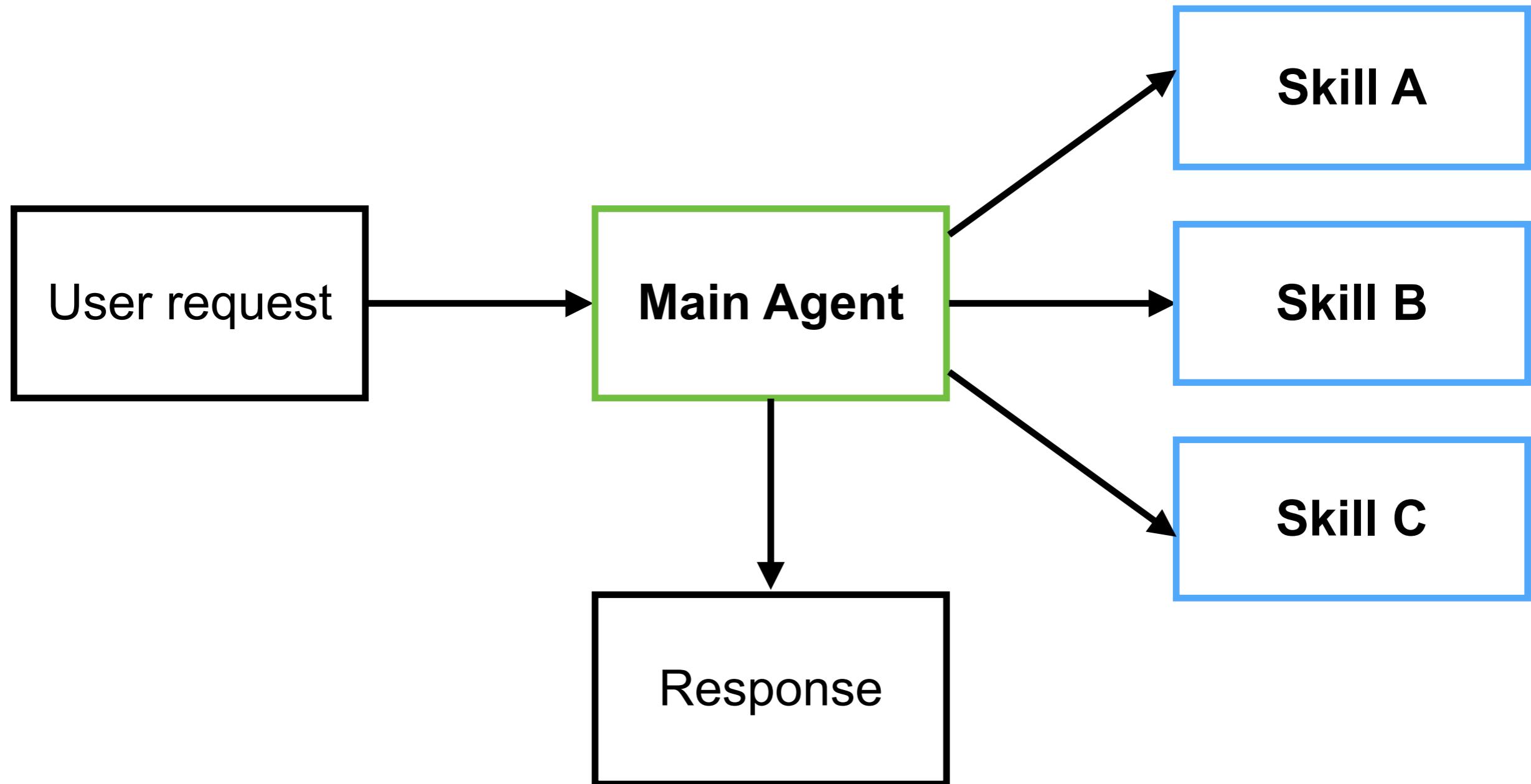
Interoperability

<https://agentskills.io/>



# Agent with Skills

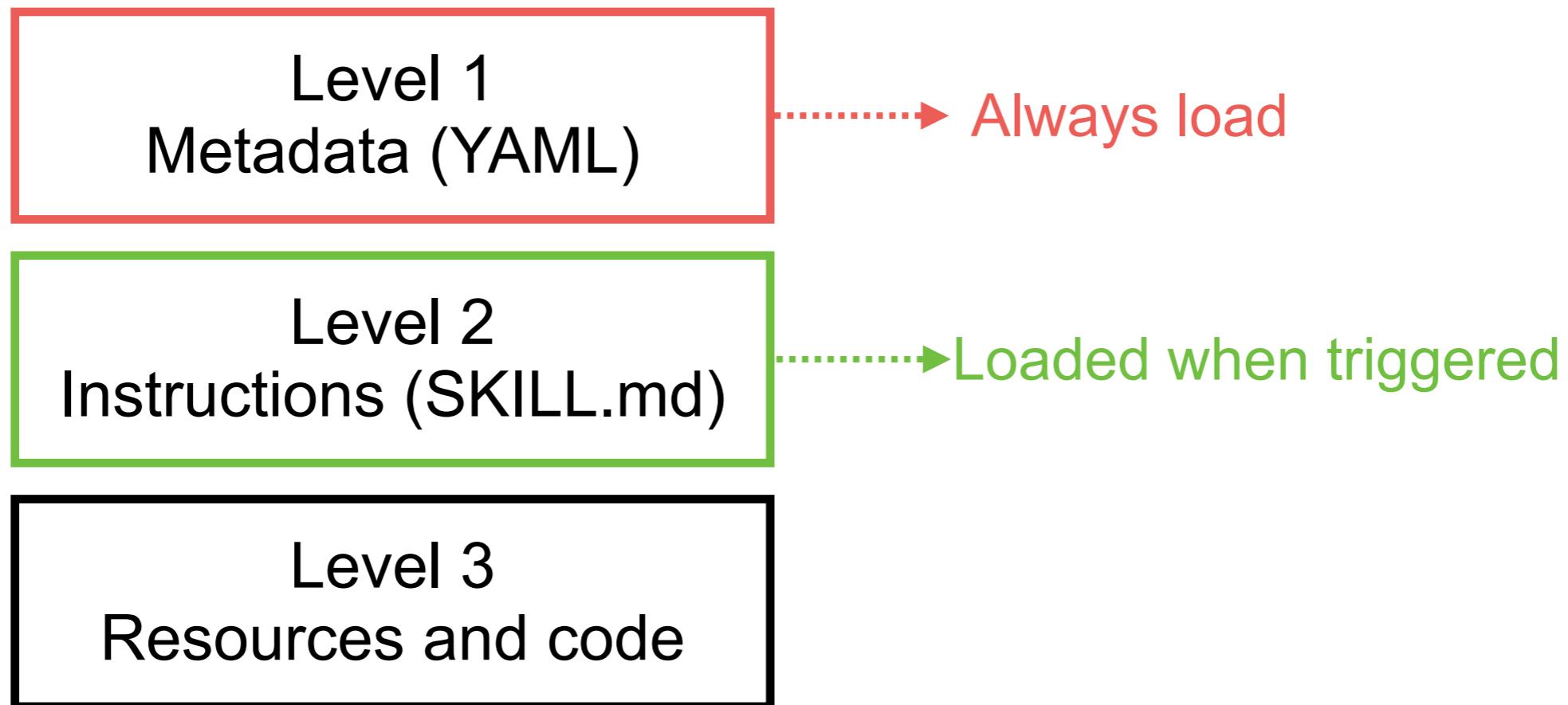
Load skills on-demand from context



<https://www.blog.langchain.com/choosing-the-right-multi-agent-architecture/>



# Structure of Agent Skills



<https://agentskills.io/>



# Prompt vs Skill

Features	Prompt	Agent Skills
Persistence	Ephemeral (chat-specific)	Persistent (Useable cross conversation)
Activation	Manual by user	Automatic (Triggered by context)
Structure	Unstructured text/ instruction	Structured markdown File system-based
Context load	Full instruction	Load only when needed
Token cost	Higher (Repeated usage)	Lower (Load on-demand)
Use case	One-time, creative quick query	Repeatable workflows Specialized tasks

<https://agentskills.io/>



# Awesome Agent Skills

The screenshot shows a web application interface for a skills marketplace. At the top, there's a header with a file icon and the text "Agent-Skills.md". On the right side of the header are "Submit" and a refresh/cog icon. Below the header, the main title "Find awesome Agent Skills" is displayed in large, bold, black and teal text. A subtitle below it reads "Agent-Skills.md is a agent skills marketplace, to find the right agent skills for you." A search bar at the bottom left contains the query "c#".

**csharp-developer**  
Use when building C# applications with .NET 8+, ASP.NET Core APIs, or Blazor web apps. Invoke for Entity Framework Core, minimal APIs, async patterns, CQRS with MediatR. Keywords: C#, .NET, ASP.NET...  
c# .net aspnet-core blazor  
Jeffallan ⚡ 24 🌟 5

**async-expert**  
Expert in asynchronous programming patterns across languages (Python asyncio, JavaScript/TypeScript promises, C# async/await, Rust futures). Use for concurrent programming, event loops, async patterns...  
asynchronous-programming concurrency  
performance-optimization error-handling  
martinholovsky ⚡ 9 🌟 2

**xUnit Test Framework**  
Execute and generate xUnit tests for C#/NET projects with FluentAssertions and Moq support.  
.NET xunit test-framework moq  
FortiumPartners ⚡ 5 🌟 1

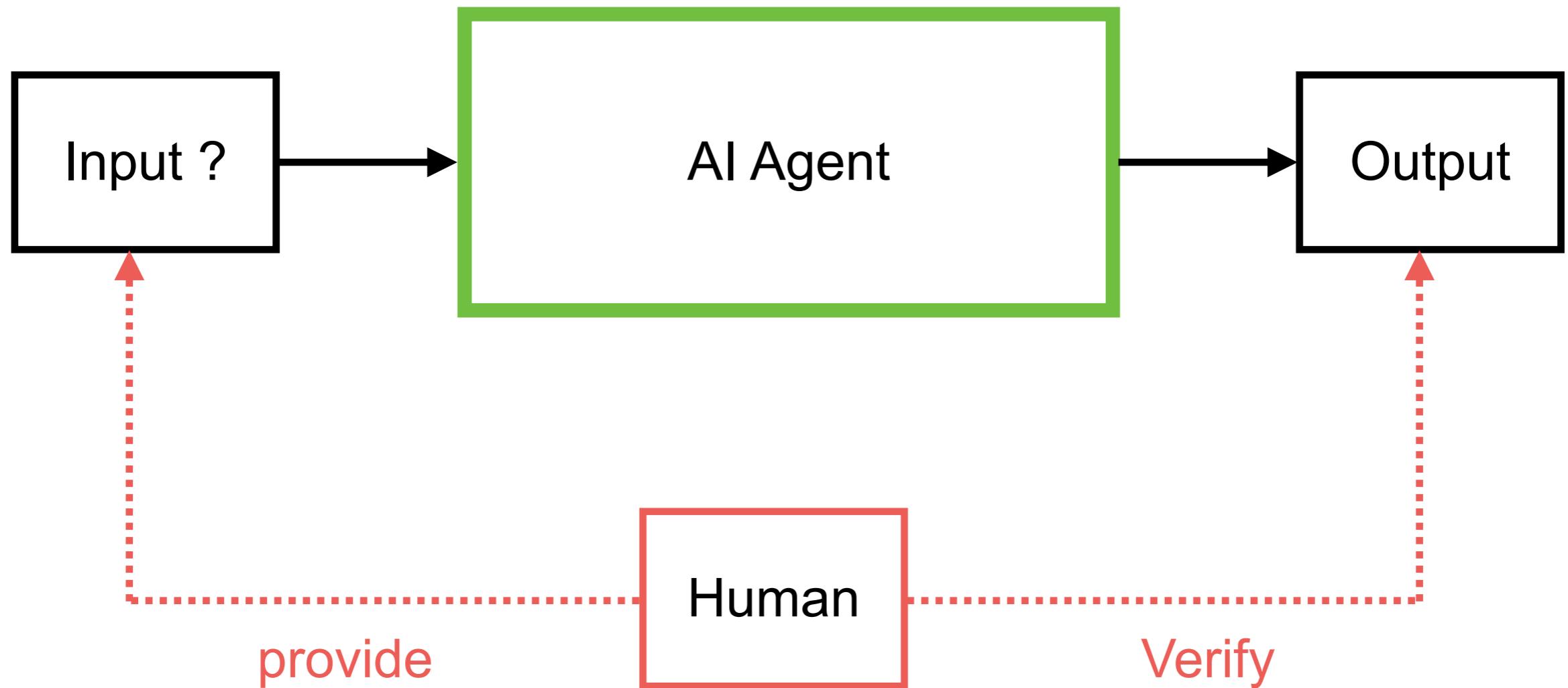
<https://agent-skills.md/>



# How to provide inputs ?



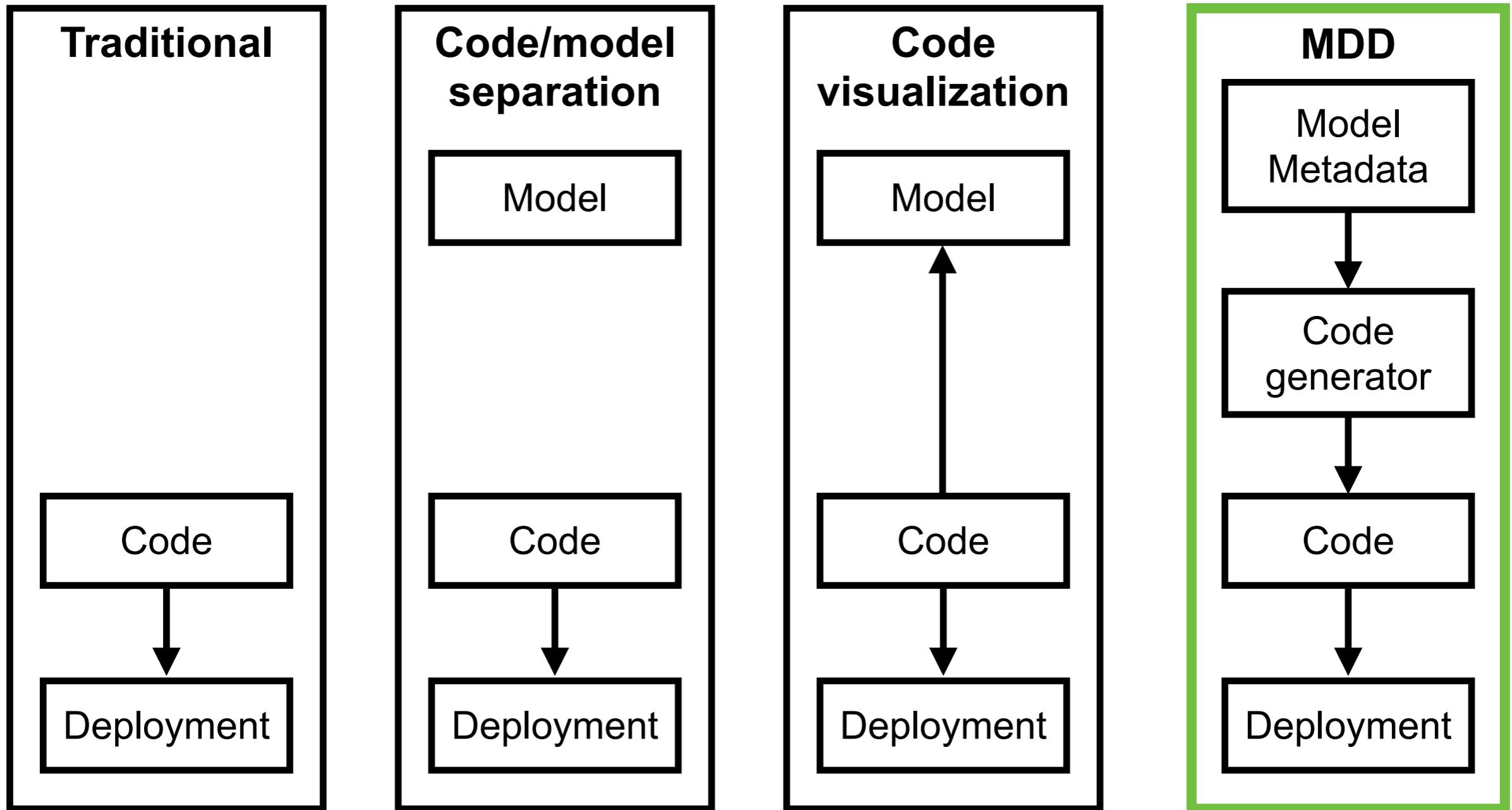
# How to provide inputs ?



# **Model-Driven Development (MDD)**



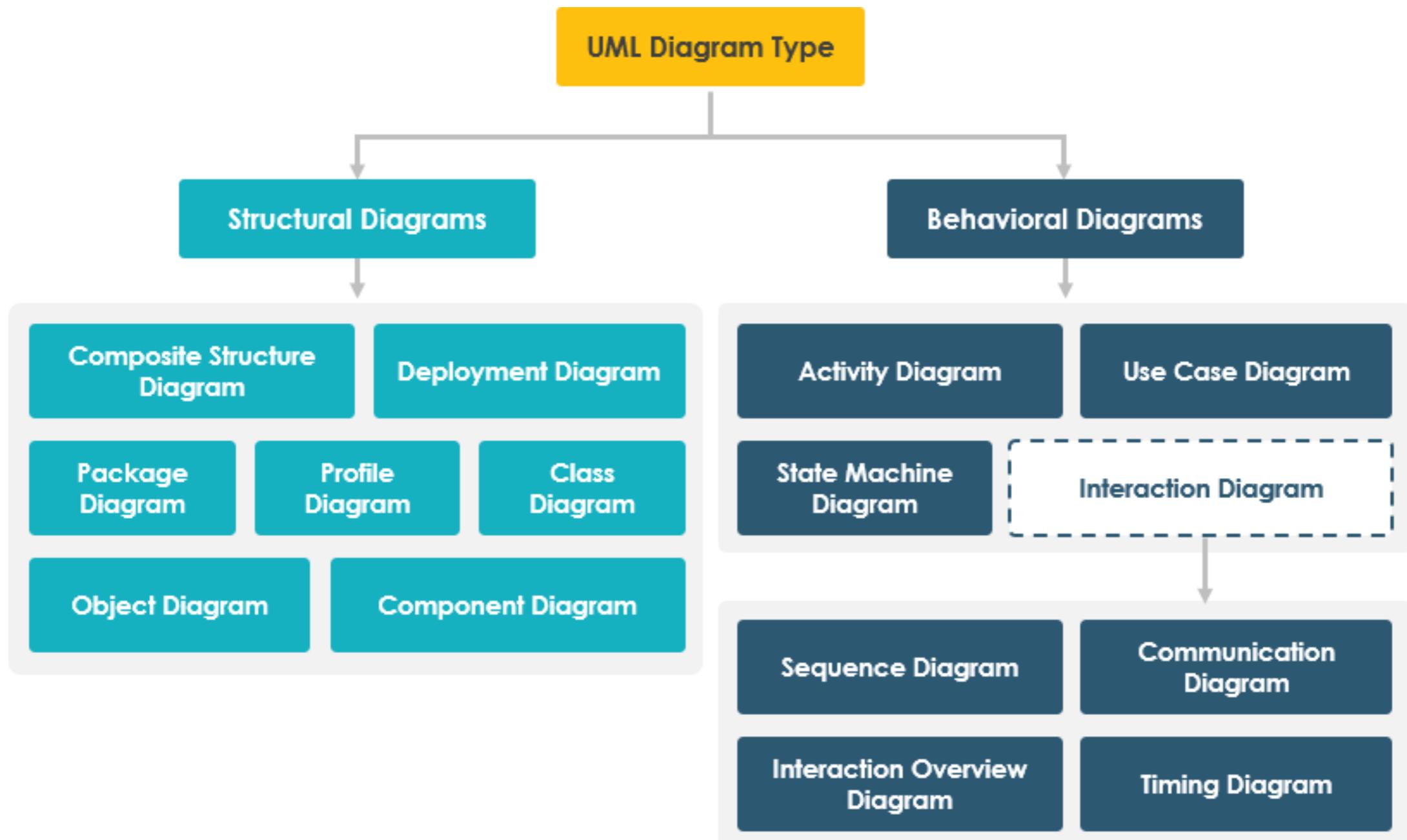
# Traditional coding to MDD



<https://arxiv.org/abs/2410.18489>



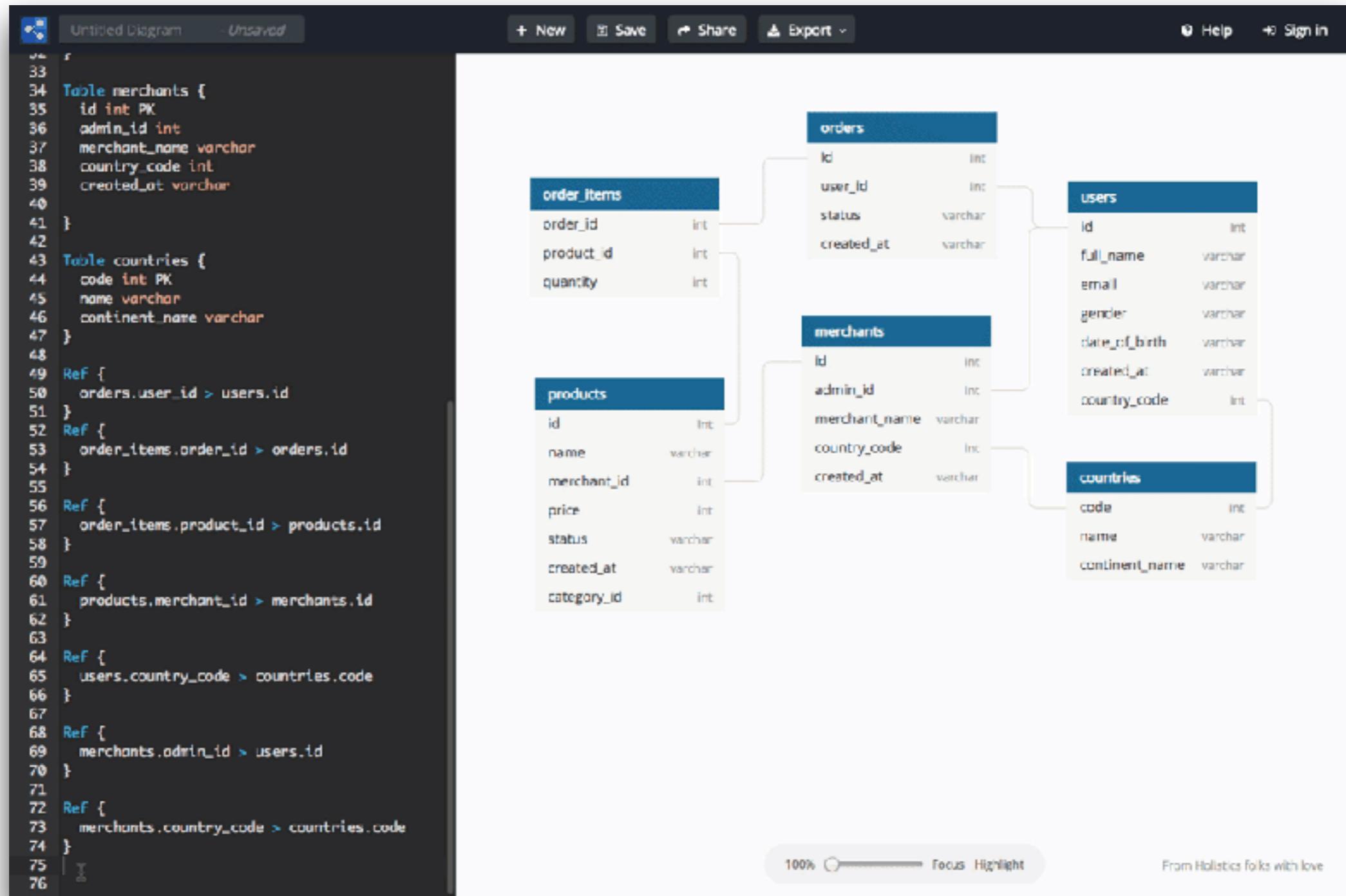
# UML Diagrams !!



<https://www.visual-paradigm.com/guide/uml-unified-modeling-language/behavior-vs-structural-diagram/>



# ER Diagrams for Database !!



<https://dbml.dbdiagram.io/home/>



# Mermaid Diagram

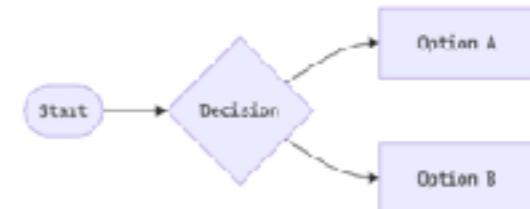
Diagram as a Code (AI-friendly)

## Mermaid Diagramming and charting tool

JavaScript based diagramming and charting tool  
that renders Markdown-inspired text definitions to  
create and modify diagrams dynamically.

[Try Editor](#)[Get started](#)

```
1 flowchart LR
2 | A["Start"]
3 | A --> B{"Decision"}
4 | B --> C["Option A"]
5 | B --> D["Option B"]
```

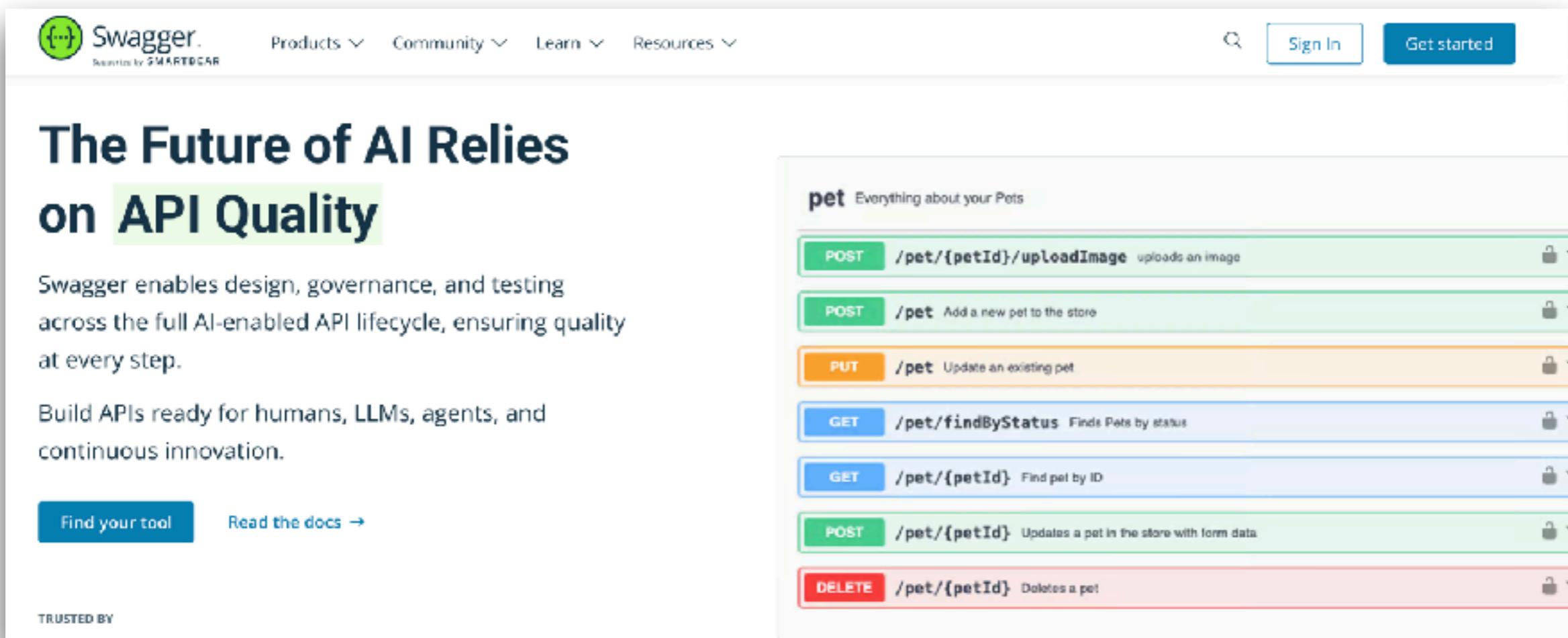


<https://mermaid.js.org/>



# Swagger or OpenAPI

Standardized specification for describe REST APIs



The screenshot shows the Swagger homepage with a banner about AI and API quality, followed by a detailed API specification for managing pets.

**Banner:** The Future of AI Relies on API Quality

**API Documentation:**

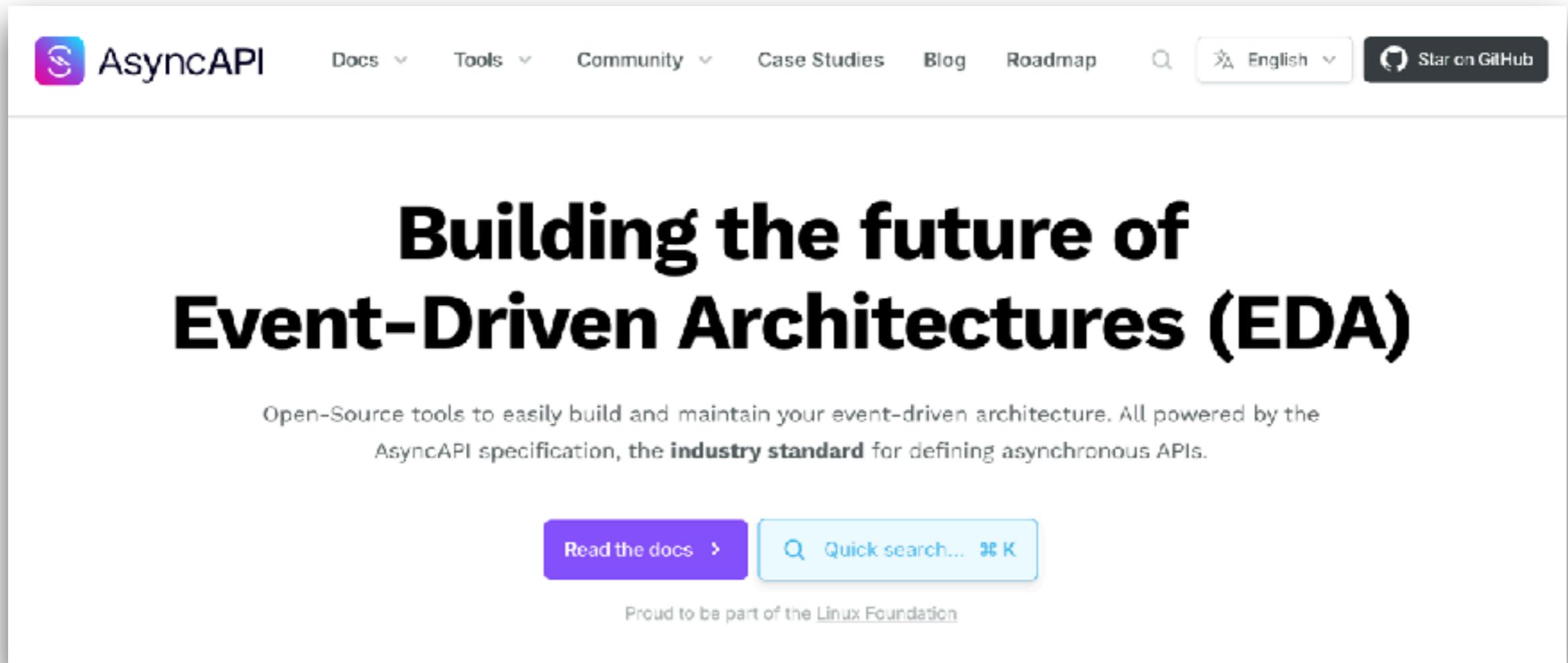
- Resource:** pet Everything about your Pets
- Operations:**
  - POST /pet/{petId}/uploadImage** uploads an image
  - POST /pet** Add a new pet to the store
  - PUT /pet** Update an existing pet
  - GET /pet/findByStatus** Finds Pets by status
  - GET /pet/{petId}** Find pet by ID
  - POST /pet/{petId}** Updates a pet in the store with form data
  - DELETE /pet/{petId}** Deletes a pet

<https://swagger.io/>



# AsyncAPI

Standardized specification for describe asynchronous and event-driven APIs (Kafka, AMQP, WebSocket)



The screenshot shows the AsyncAPI website homepage. At the top, there is a navigation bar with links for Docs, Tools, Community, Case Studies, Blog, and Roadmap. There is also a search icon, a language selector set to English, and a GitHub star button. The main title "Building the future of Event-Driven Architectures (EDA)" is prominently displayed in large, bold, black font. Below the title, a subtitle reads: "Open-Source tools to easily build and maintain your event-driven architecture. All powered by the AsyncAPI specification, the **industry standard** for defining asynchronous APIs." At the bottom of the main content area, there are two buttons: "Read the docs >" and "Quick search... 36 K". A small note at the bottom says "Proud to be part of the [Linux Foundation](#)".

<https://www.asyncapi.com/>



# JSONSchema

Declarative language used to annotate  
and validate the structure, constraints and data type of  
JSON documents

Data validation

Documentation

Automated  
testing

<https://json-schema.org/>



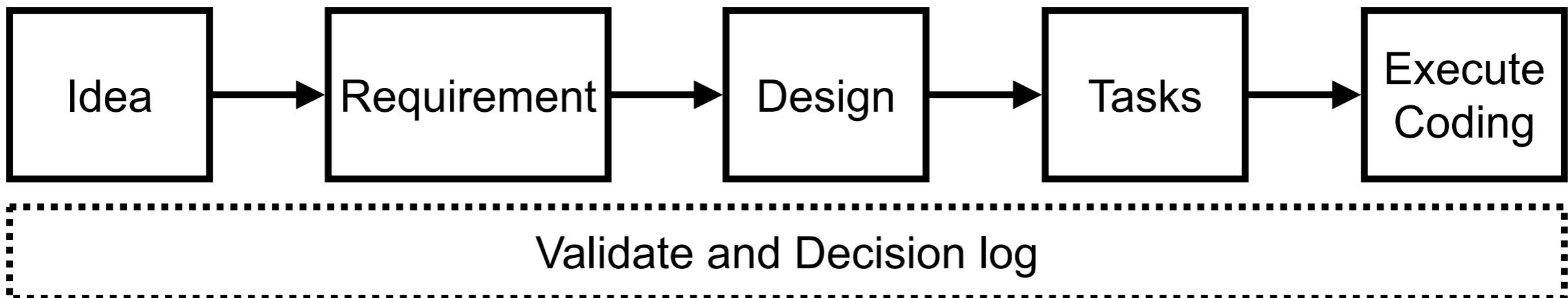
# **Specification-Driven Development (SDD)**



# SDD ?

## Specification or Documentation-First

Modern software development approach where detailed, structure specifications are created before writing code

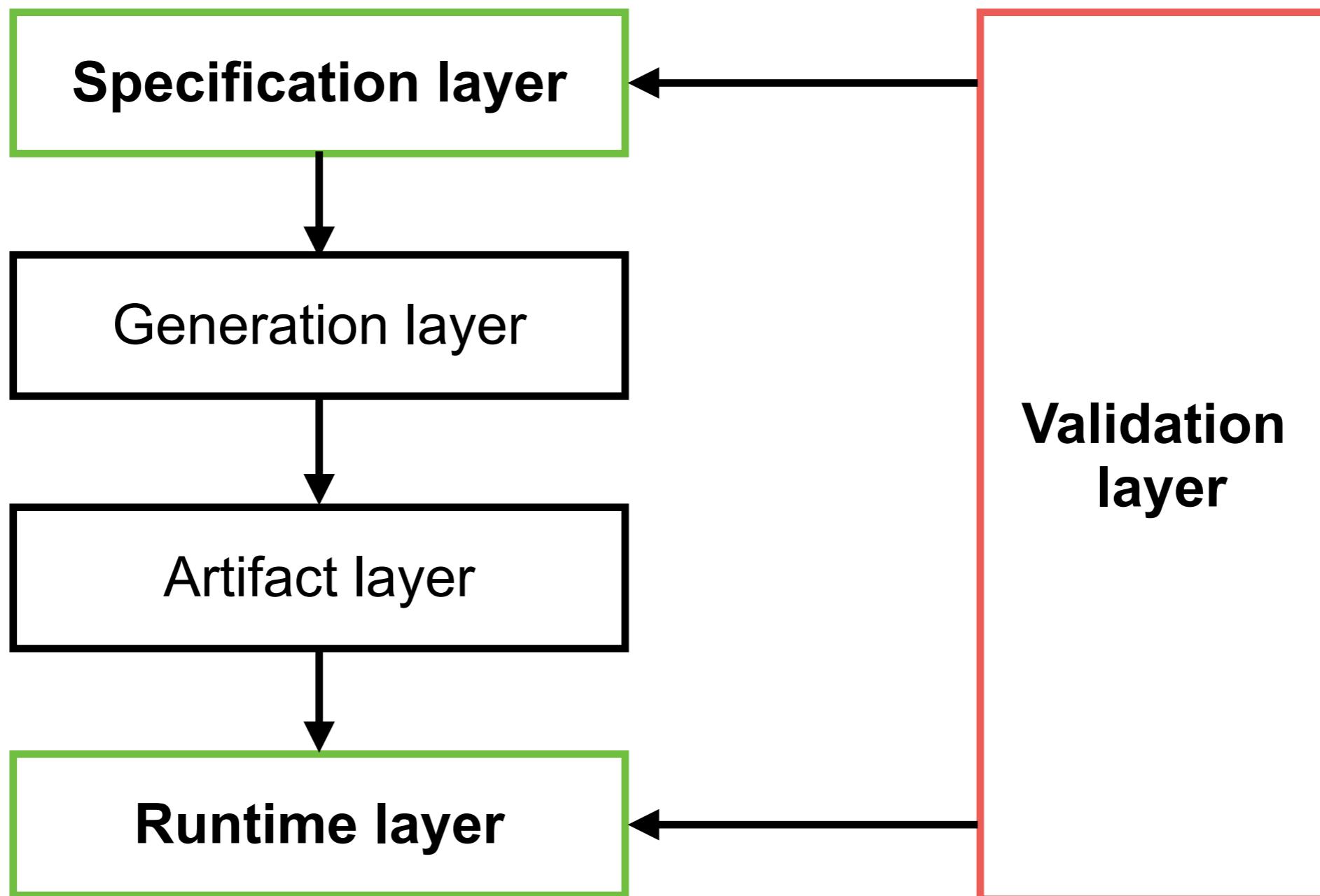


# Types of Specification ?

User requirements  
Design documents  
Interface contracts (UI, API)  
Data schemas  
Validation rules  
Data/business flows  
Security policy and constraints  
Resources and performance constraints



# SSD Architecture



<https://www.infoq.com/articles/spec-driven-development>



# **MDD + SDD == Better result**



# When to use SDD ?

No one-size-fit-all !!

**Low complexity**

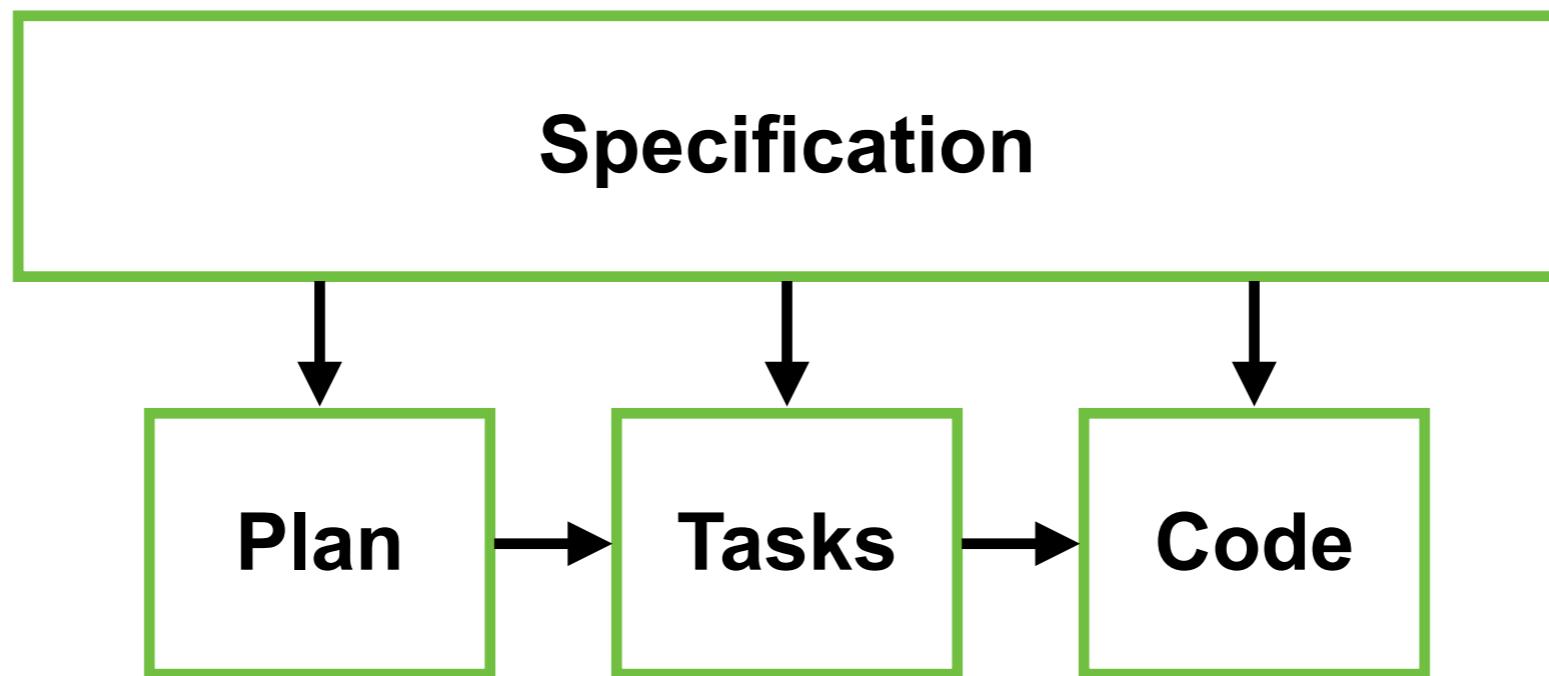
**High complexity**

<https://medium.com/google-cloud/benefits-and-challenges-of-spec-driven-development-and-how-antigravity-is-changing-the-game-3343a6942330>



# SDD Tools ?

GitHub Spec-kit  
Open Spec  
Kiro.dev  
Tessl



# Try by yourself !!





dot cline



**Frontend**  
Web application

**Backend**  
RESTful API +  
Database

Write specification

Coding

Run and Test !!



**Frontend**  
Web application

**Backend**  
RESTful API +  
Database

Vercel React Skill

Write spec or feature

Prompt or command

Working with MCP



**Frontend**  
Web application

**Backend**  
RESTful API +  
Database

Vercel React Skill

Global instruction

Write spec or feature

Write spec or feature

Prompt or command

Prompt or command

Working with MCP

Working with MCP

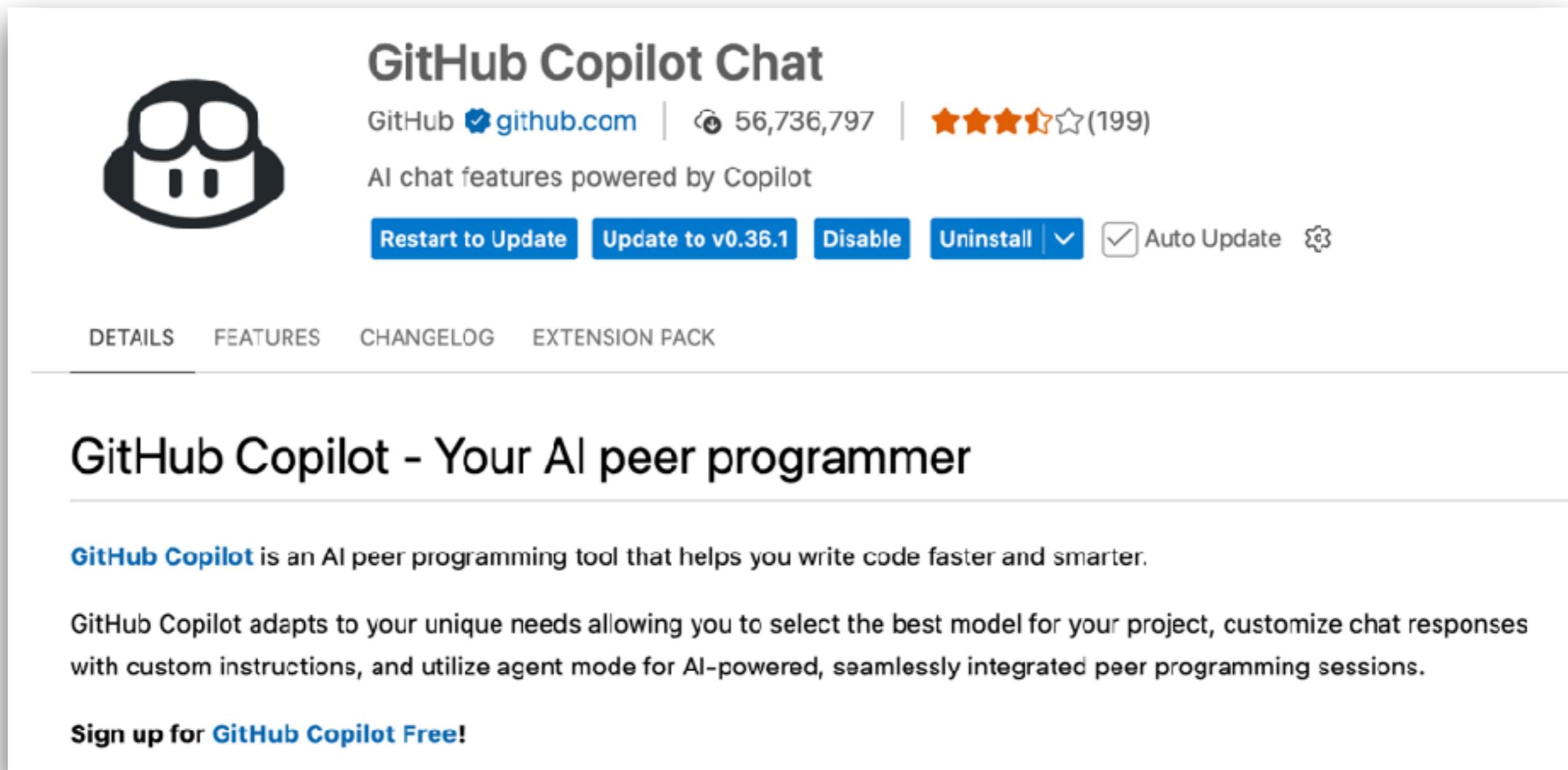


# **Workshop with vs Code + GitHub Copilot Chat**

<https://code.visualstudio.com/docs/copilot/chat/getting-started-chat>



# Install extension in VS Code

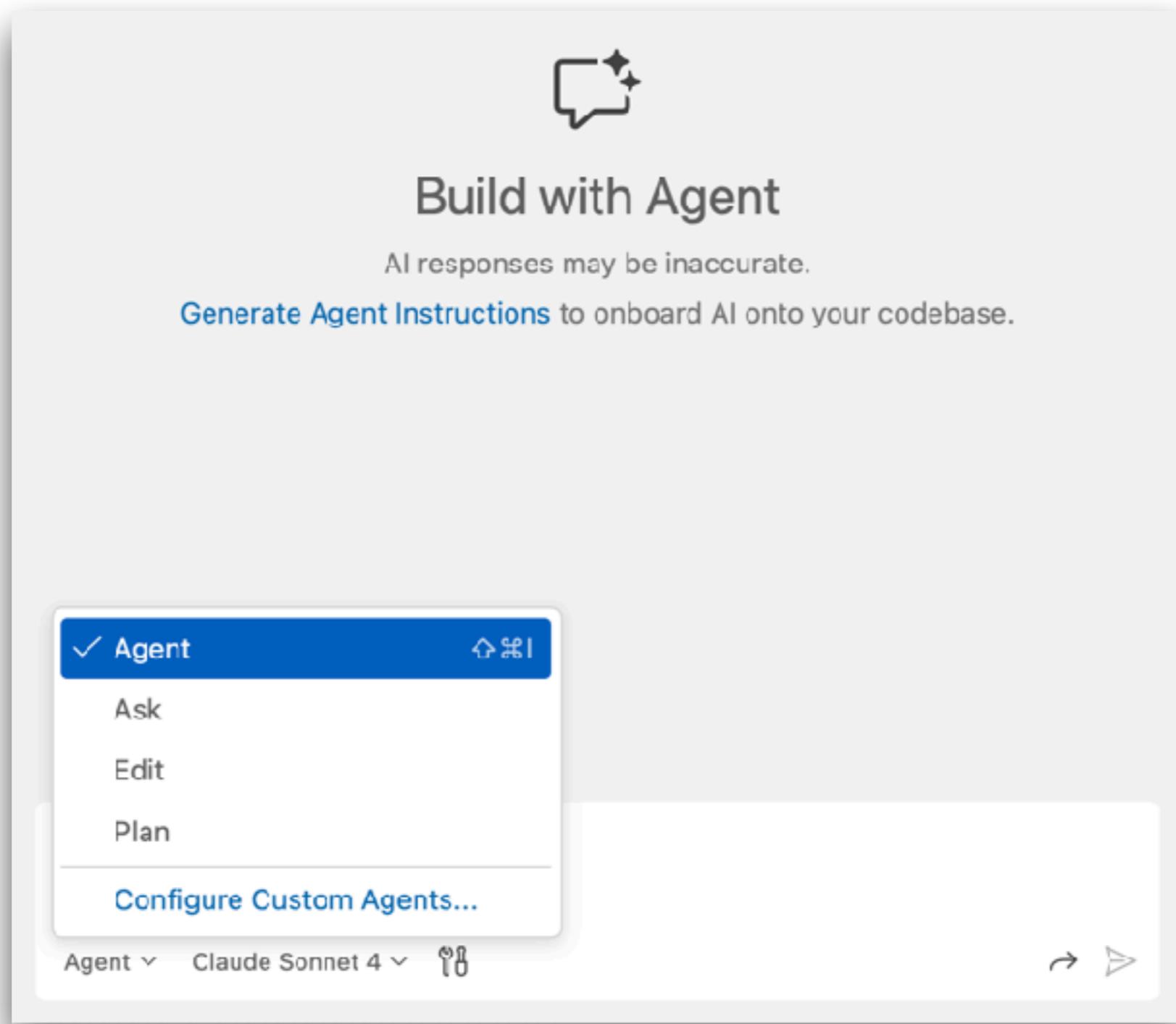


The screenshot shows the GitHub Copilot Chat extension page in the Visual Studio Code Marketplace. At the top, there's a large icon of a white owl wearing headphones. To the right of the icon, the extension name "GitHub Copilot Chat" is displayed in bold. Below the name, it says "GitHub [github.com](#)" with a blue checkmark, "56,736,797" with a download icon, and a star rating of "★★★★★ (199)". A subtitle reads "AI chat features powered by Copilot". Below this, there are several buttons: "Restart to Update", "Update to v0.36.1", "Disable", "Uninstall", a dropdown menu, and an "Auto Update" checkbox which is checked. At the bottom of the main section, there are links for "DETAILS", "FEATURES", "CHANGELOG", and "EXTENSION PACK". The main content area below has a heading "GitHub Copilot - Your AI peer programmer" and a paragraph describing the tool as an AI peer programming tool that helps you write code faster and smarter. It also mentions that GitHub Copilot adapts to unique needs, allows selecting models, customizing chat responses, and utilizing agent mode. A "Sign up for GitHub Copilot Free!" button is at the bottom.

<https://marketplace.visualstudio.com/items?itemName=GitHub.copilot-chat>



# Hello Copilot Chat in VS Code



# Copilot Chat Modes

Agent

Ask

Edit

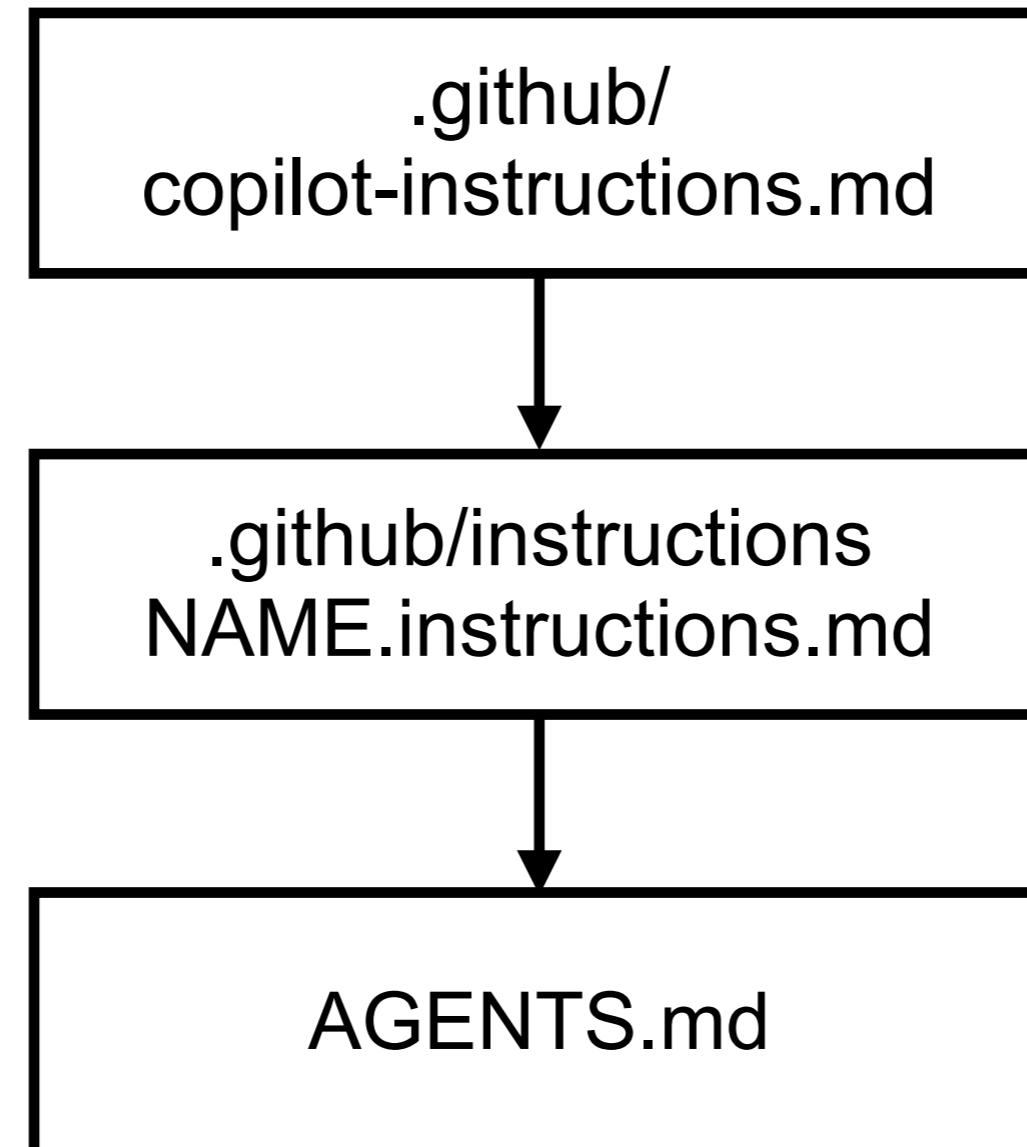
Plan

Custom  
Agent

<https://code.visualstudio.com/docs/copilot/customization/custom-agents>



# Custom instructions !!



Global

Specific by types

Disable by default in VS Code

<https://docs.github.com/en/copilot/how-tos/configure-custom-instructions/add-repository-instructions>



# Customization !!

Agent

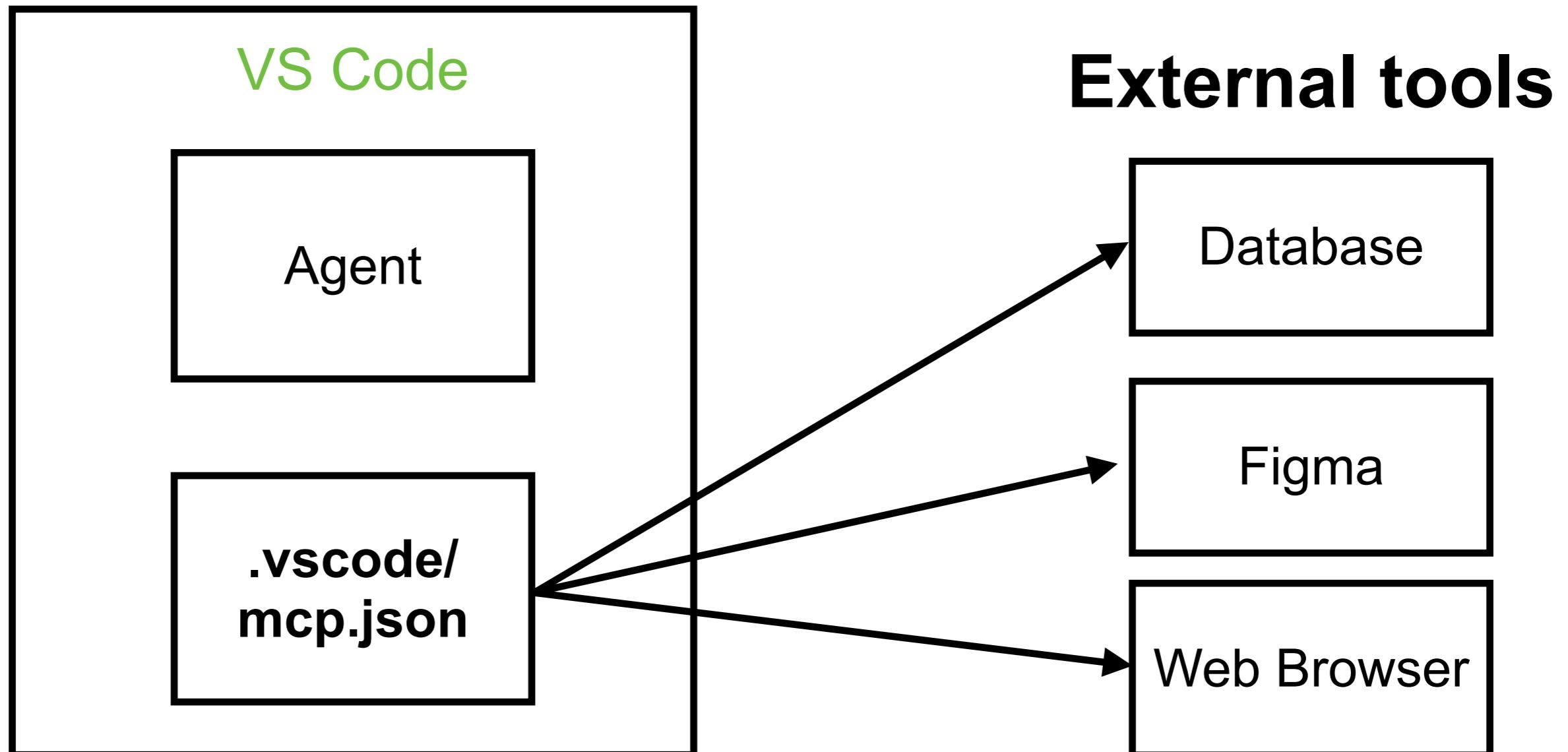
Instruction

Prompts

Skills



# Working with MCP

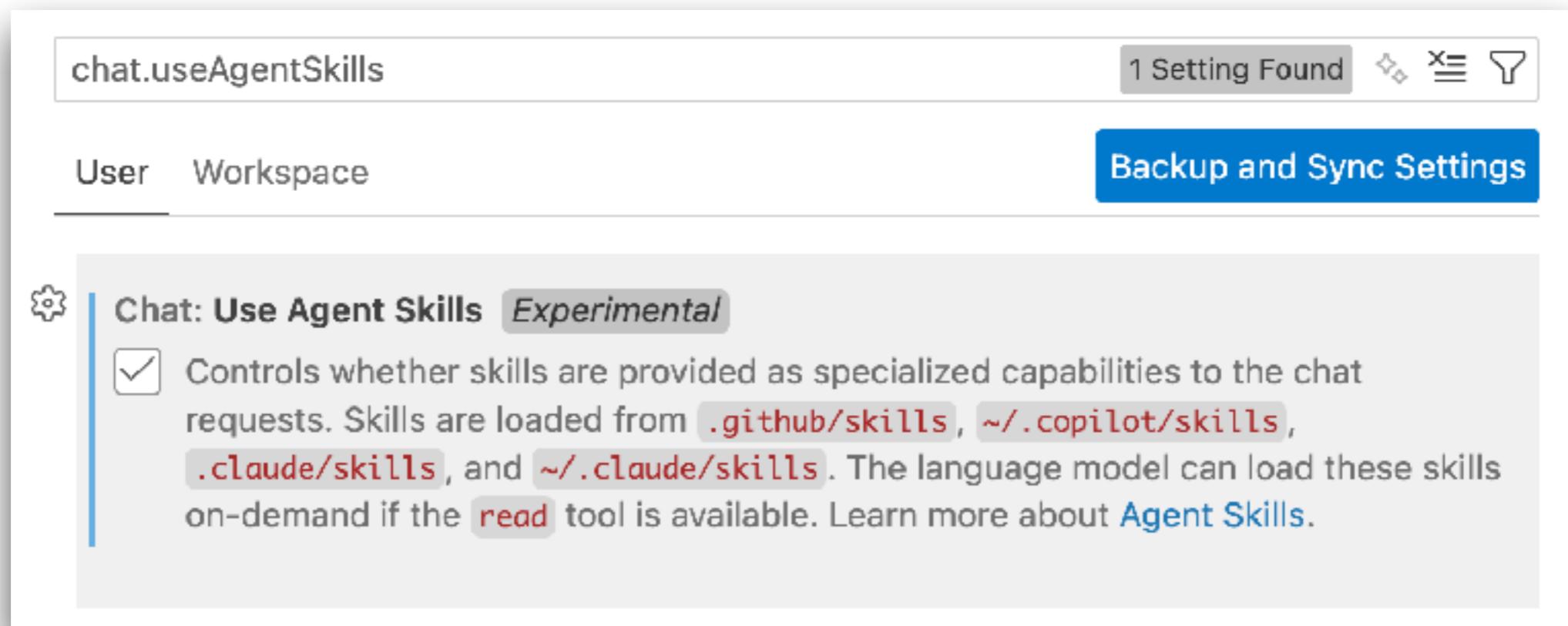


<https://code.visualstudio.com/docs/copilot/customization/custom-agents>



# Enable Agent Skills in VS Code

Setting => “chat.useAgentSkills”  
Create skills in “.github/skills/name/SKILL.md”



<https://code.visualstudio.com/docs/copilot/customization/agent-skills>



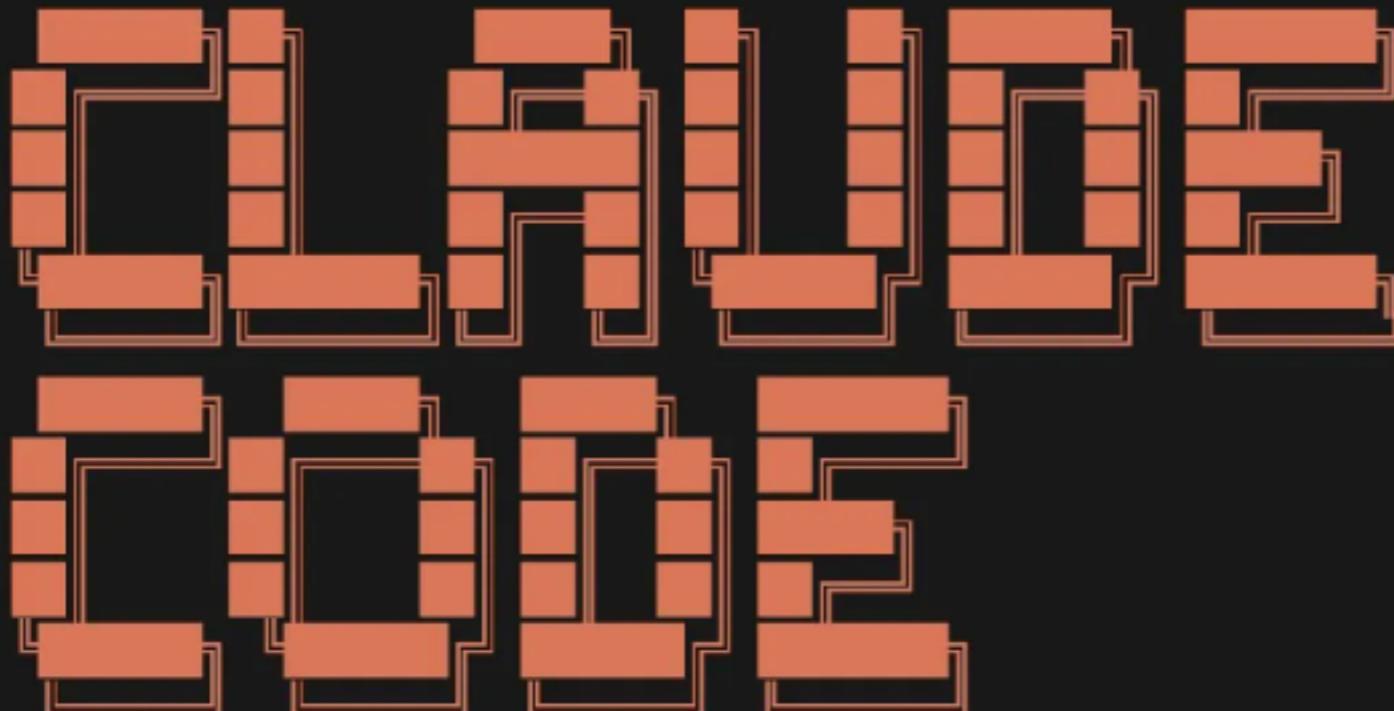
# Workshop with Claude Code

<https://claude.com/product/claude-code>





\* Welcome to Claude Code



Press **Enter** to continue

**npm install -g @anthropic-ai/clause-code**



# Workshop with VS Code + Cline

<https://cline.bot/>



# Cline ?

Opensource AI coding agent  
Development partner

Create/edit  
files

Run in  
terminal

Browse  
website

Multi-step  
tasks

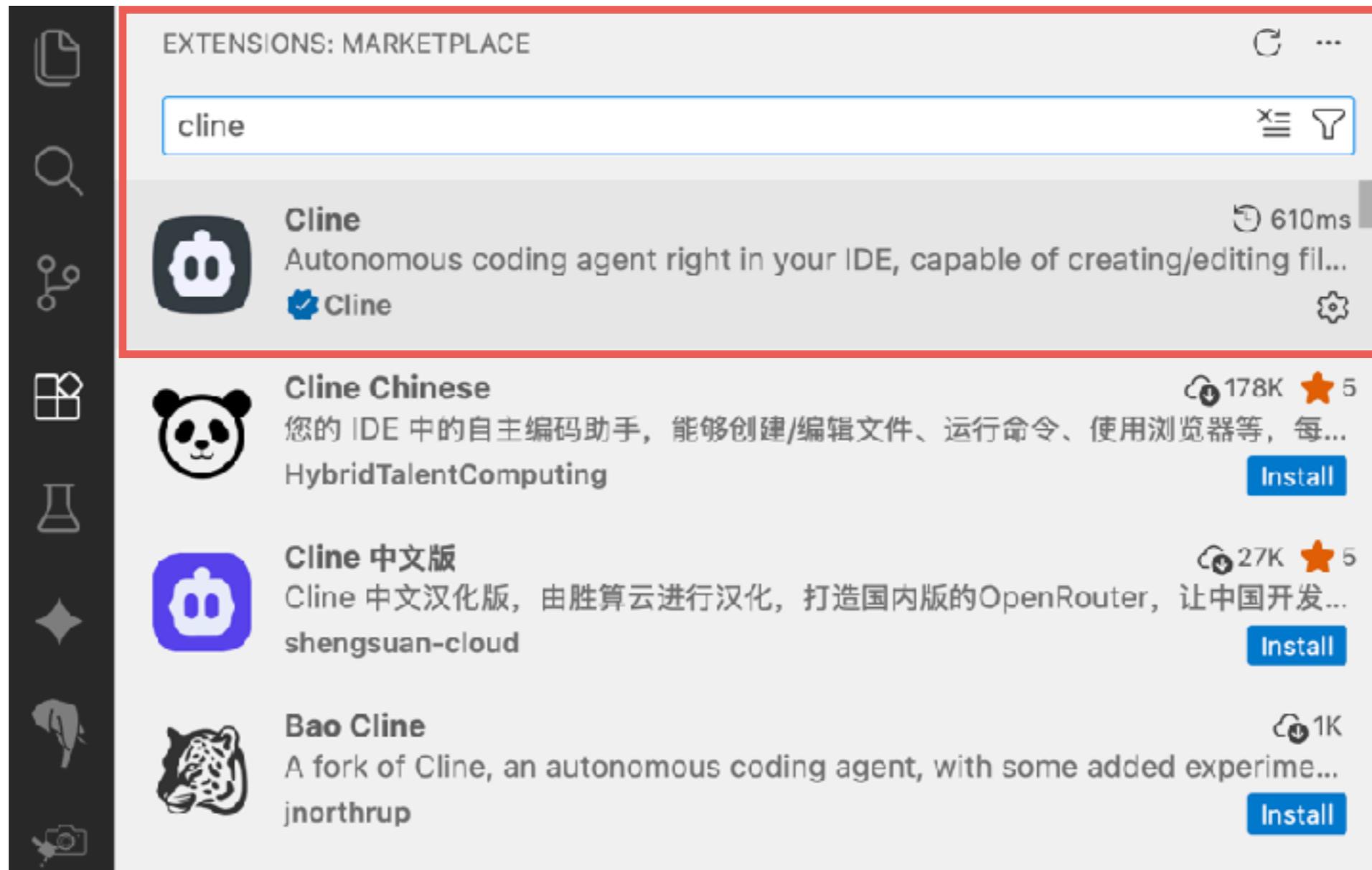
Human-in-the  
loop

Support tools

<https://cline.bot/>



# Install extension in VS Code



<https://marketplace.visualstudio.com/items?itemName=saoudrizwan.claude-dev>



# Install extension in VS Code



**Cline**

Cline ✅ cline.bot | ⚡ 2,902,099 | ★★★★★ (264)

Autonomous coding agent right in your IDE, capable of creating/editing files, running commands,

[Disable](#) [Uninstall](#) |  Auto Update 

[DETAILS](#) [FEATURES](#) [CHANGELOG](#)

---

Meet Cline, an AI assistant that can use your **CLI aNd Editor**.

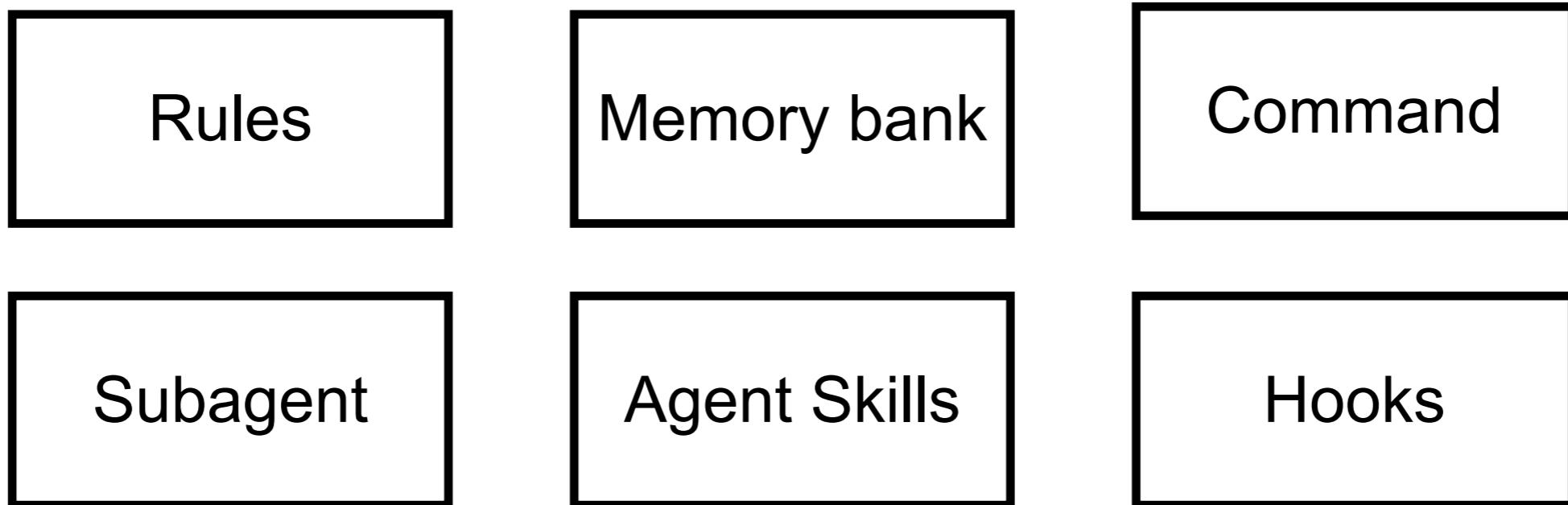
Thanks to [Claude Sonnet's agentic coding capabilities](#), Cline can handle complex software development tasks step-by-step. With tools that let him create & edit files, explore large projects, use the browser, and execute terminal commands (after you grant permission), he can assist you in ways that go beyond code completion or tech support. Cline can even use the Model Context Protocol (MCP) to create new tools and extend his own capabilities. While autonomous AI scripts traditionally run in sandboxed environments, this extension provides a human-in-the-loop GUI to approve every file change and terminal command, providing a safe and accessible way to explore the potential of agentic AI.

<https://marketplace.visualstudio.com/items?itemName=saoudrizwan.claude-dev>

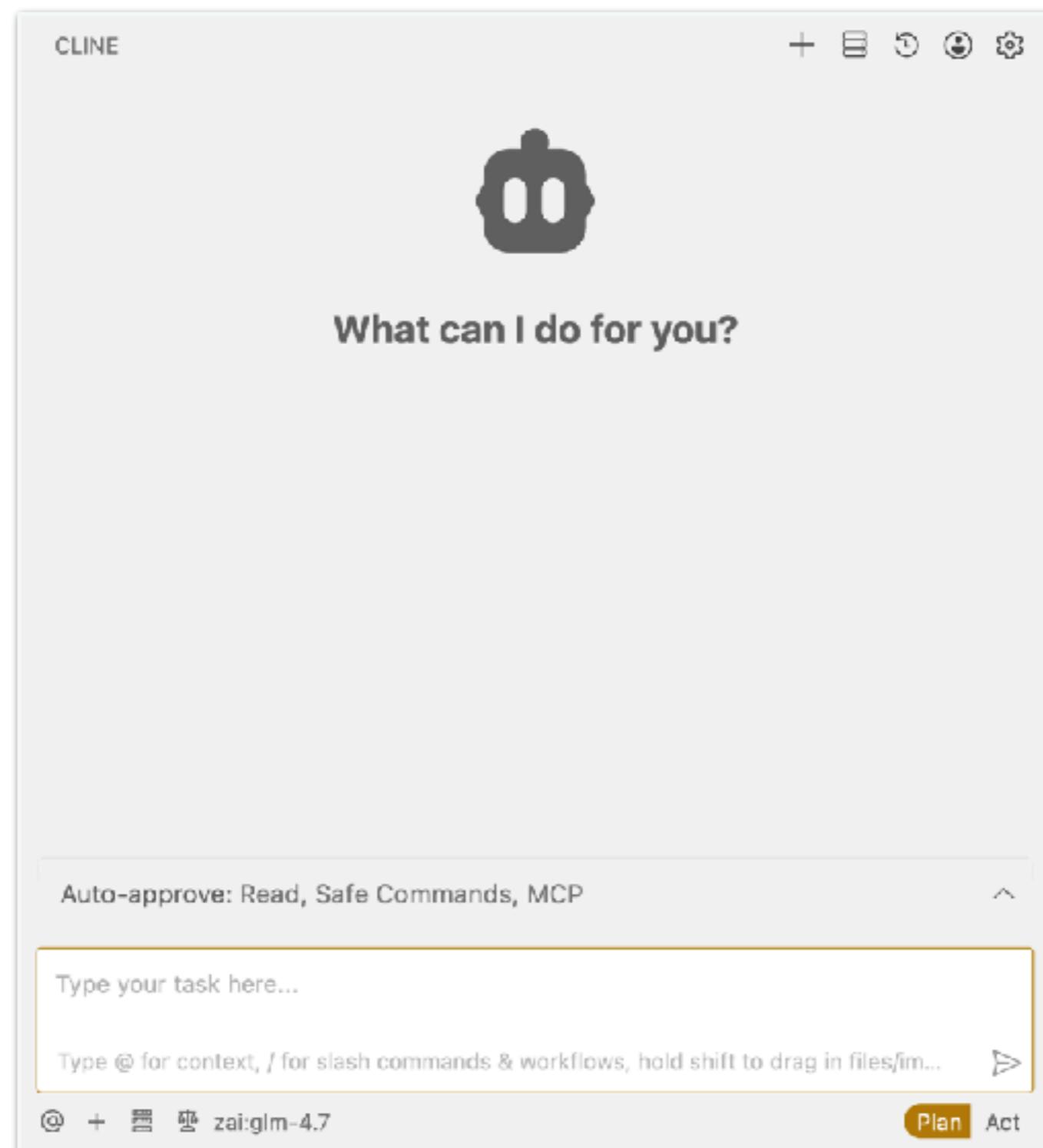


# Working with Cline

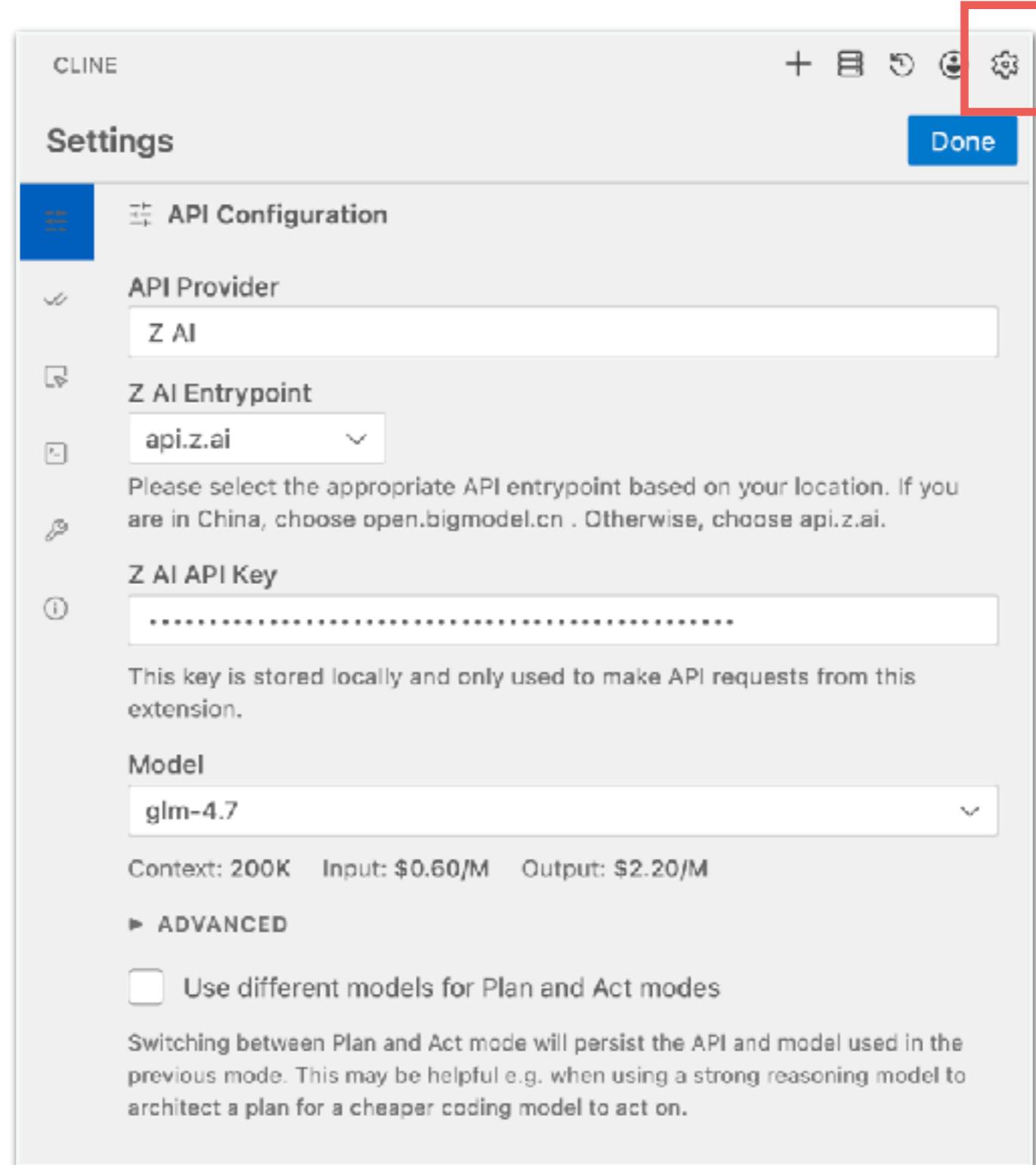
Plan and Act modes  
Toggleable rules (.clinerules)  
Checkpoint management  
Memory bank system



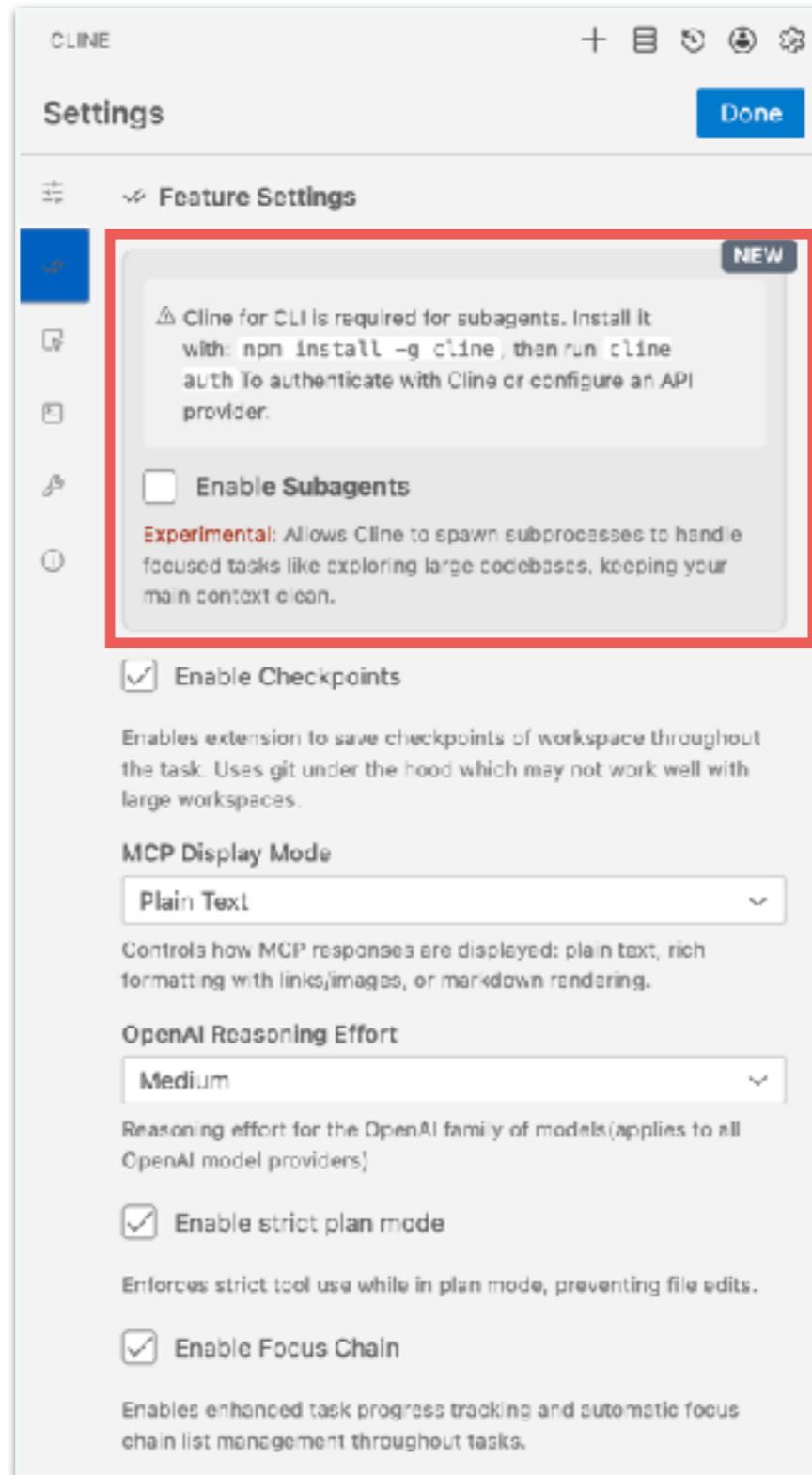
# Chat with Cline



# Select your model provider



# Feature Settings



Enable subagents  
with Cline CLI



# Cline CLI

```
npm install -g cline  
client --version
```

```
Cline CLI  
Cline CLI Version: 1.0.10  
Cline Core Version: 3.52.0  
Commit: 66d7664  
Built: 2026-01-22T20:32:21Z  
Built by: runner  
Go version: go1.24.12  
OS/Arch: darwin/arm64
```

Support only Mac and Linux !!

<https://docs.cline.bot/cline-cli/overview>

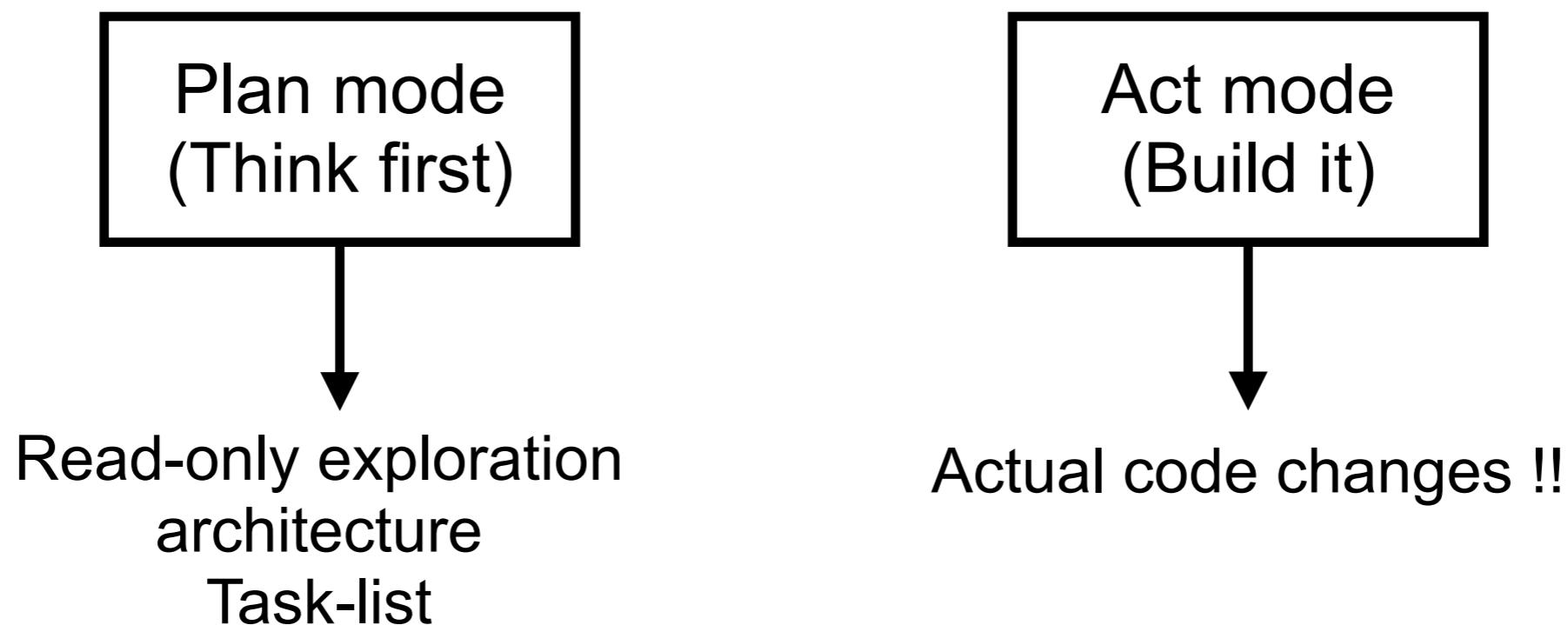


# How to use Cline ?



# Plan and Act mode

Separate strategic thinking from implementation

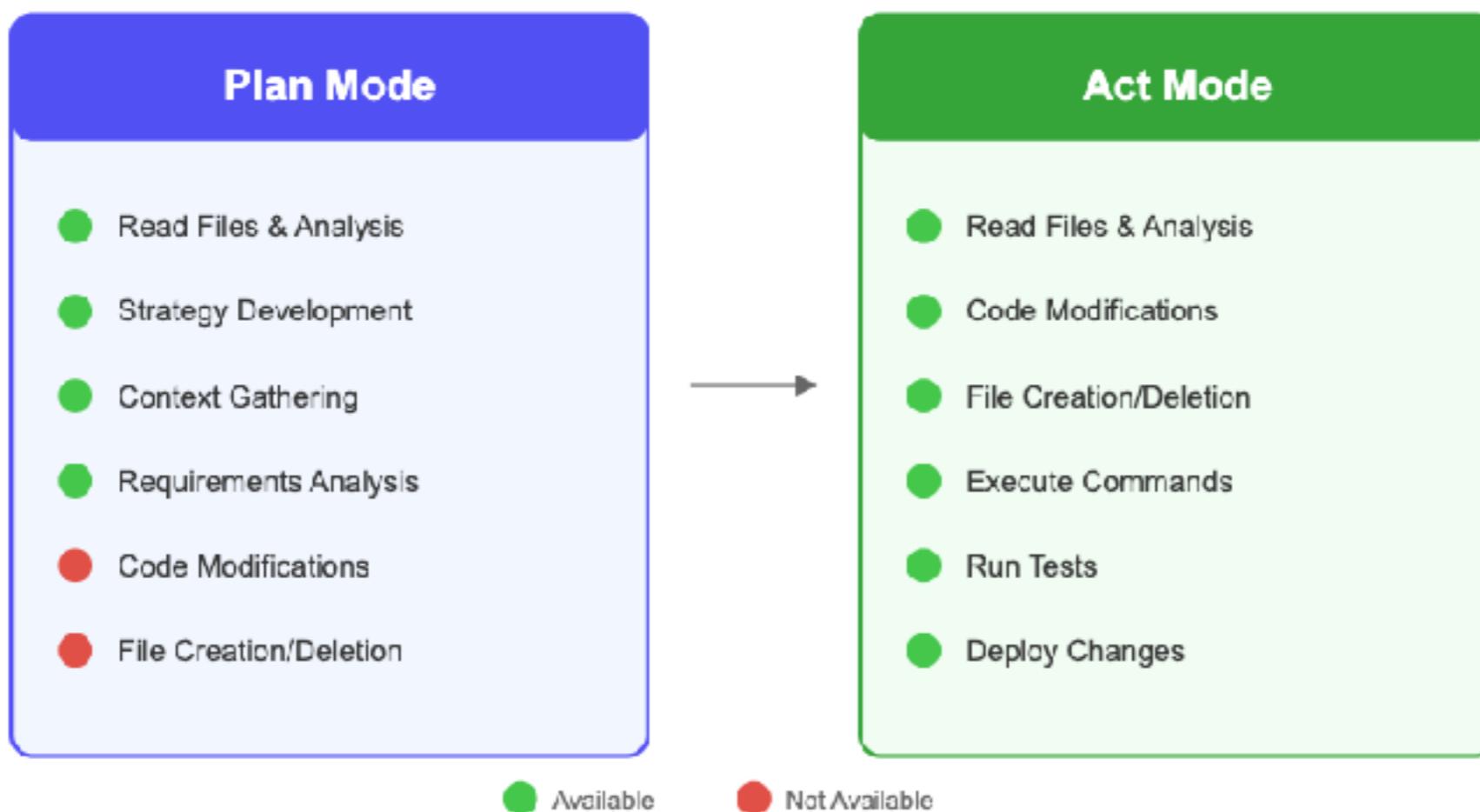


<https://docs.cline.bot/features/plan-and-act>



# Plan and Act mode

## Plan vs Act Mode Capabilities



<https://docs.cline.bot/features/plan-and-act>



# Manage rules/workflow/hooks

The screenshot shows a user interface for managing system-level configurations. At the top, there are three tabs: "Rules", "Workflows", and "Hooks". The "Rules" tab is selected, indicated by a horizontal line underneath it.

Below the tabs, there is a descriptive text block:

Rules allow you to provide Cline with system-level guidance. Think of them as a persistent way to include context and preferences for your projects or globally for every conversation. [Docs](#)

Under the "Global Rules" section, there is a button labeled "New rule file..." and a plus sign (+) icon.

Under the "Workspace Rules" section, there is a list item named "rules.mdc". To its right is a blue toggle switch, a pencil icon for editing, and a trash can icon for deleting.

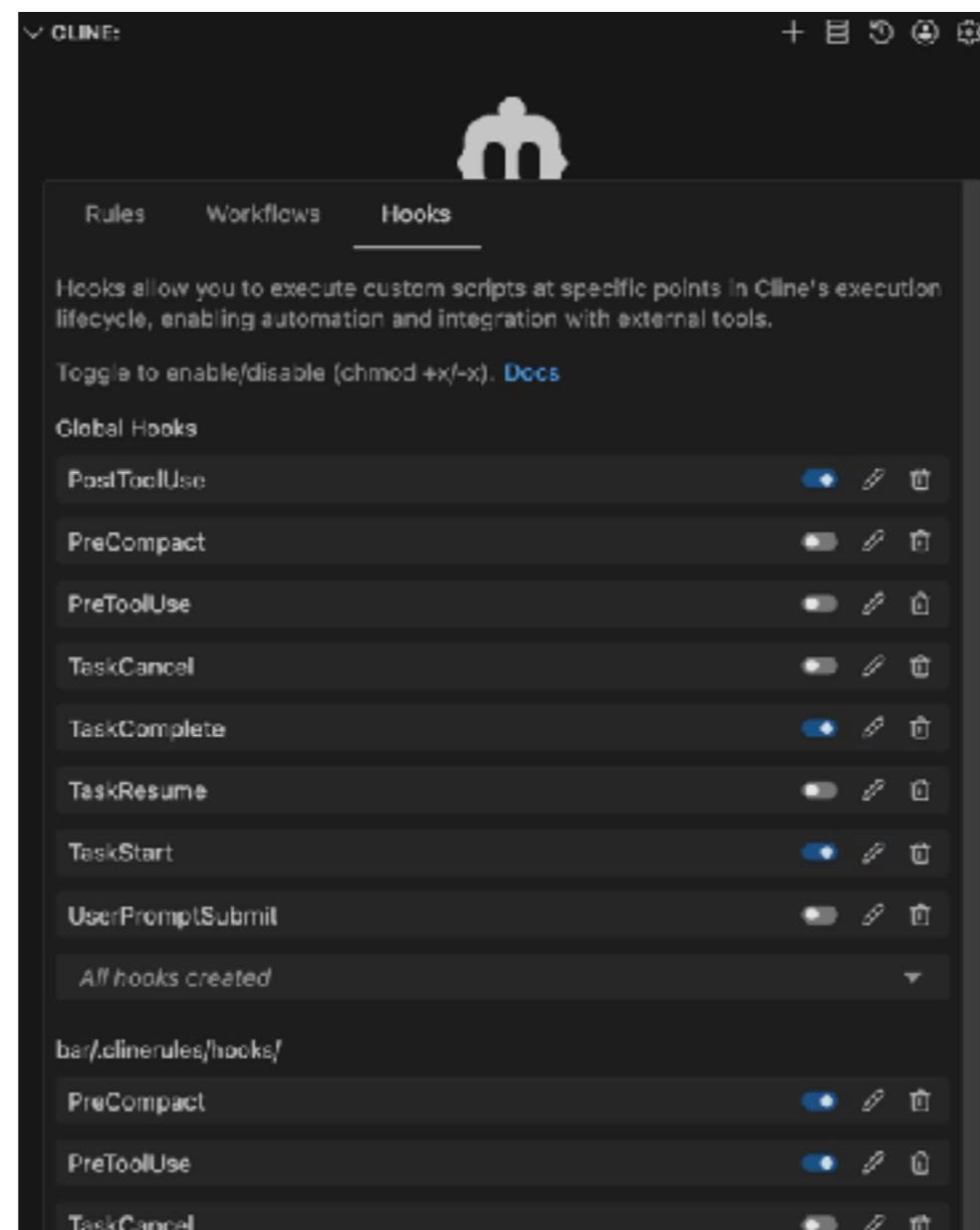
Below the workspace rules list, there is another "New rule file..." button and a plus sign (+) icon.

At the bottom of the interface, there is a navigation bar with icons for "@", "+", "≡", and "zai:glm-4.6". To the right of these icons are two buttons: "Plan" (highlighted in orange) and "Act".



# Hooks

Inject custom logics into Cline's workflow



<https://docs.cline.bot/features/hooks>



# Auto-approve !!

Read and write files  
Execute safe commands

Auto-approve: Read (all), Edit (all), All Commands, MCP

Let Cline take these actions without asking for approval. [Docs](#)

-  Read project files
-  Read all files
-  Edit project files
-  Edit all files
-  Execute safe commands
-  Execute all commands
-  Use the browser
-  Use MCP servers
-  Enable notifications



# Enable MCP in workspace

The screenshot shows the MCP Servers interface in the CLINE application. At the top, there's a toolbar with icons for adding, deleting, and configuring, with a red box highlighting the 'Add' icon. Below it is a search bar labeled 'Search MCPs...'. A dropdown menu for 'FILTER' is set to 'All Categories'. Under 'SORT', the 'Newest' option is selected. Two add-ons are listed:

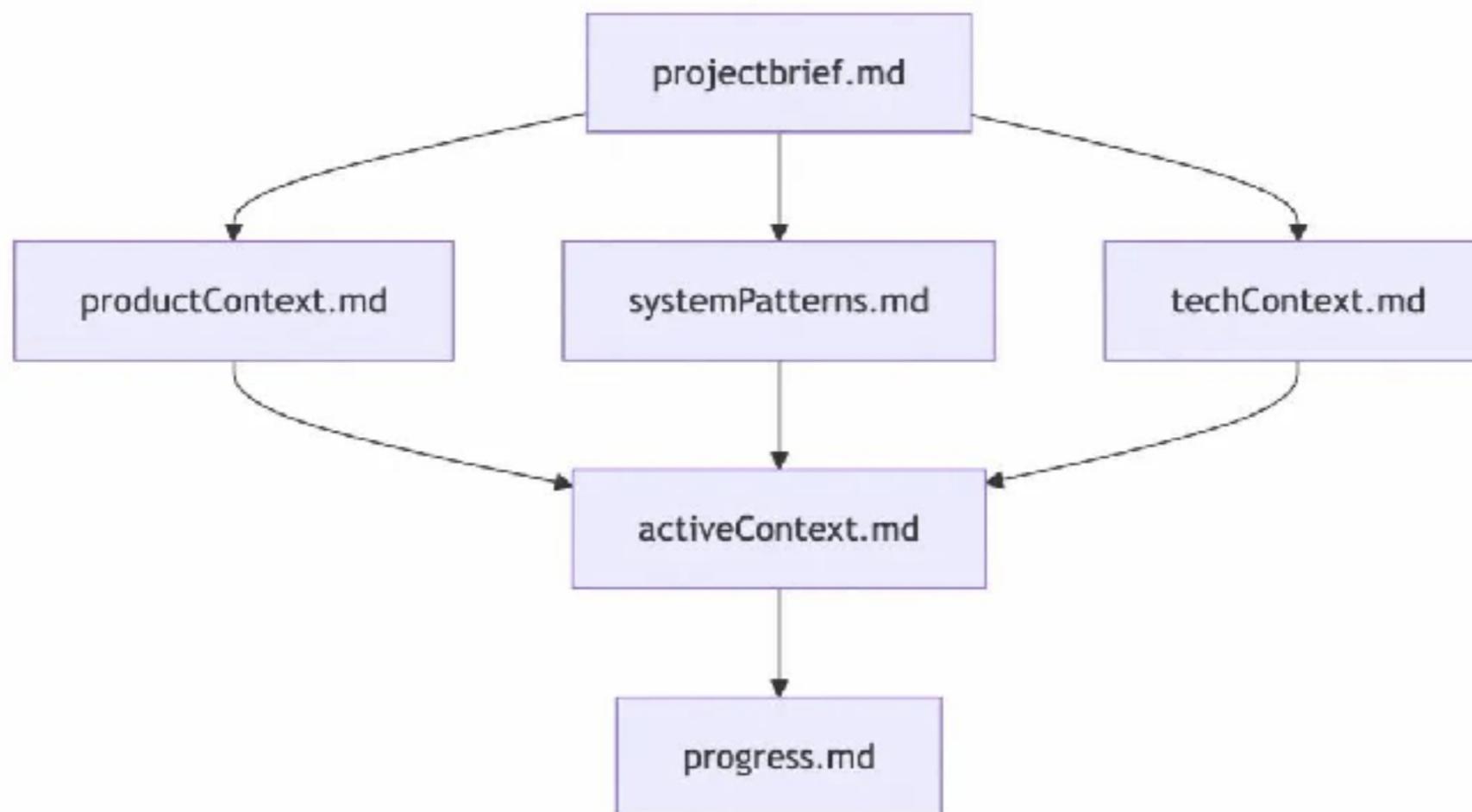
- PlatformIO** by jl-codes (7 stars, 334 installs). Description: Board-agnostic embedded development server supporting 1000+ development boards across 30+ platforms, enabling firmware building, uploading, and library management through PlatformIO's ecosystem. Tags: developer-tools, embedded-systems, firmware, iot, microcontrollers, hardware-development.
- Workflowy** by daniel137 (24 stars, 271 installs). Description: Enables AI assistants to interact with Workflowy lists, providing tools for searching, creating, updating, and managing nodes with completion status tracking. Tags: note-taking, workflow, task-management, note-organization, project-planning, productivity.

Click to MCP server



# Cline's Memory Bank

Manage context for AI across sessions  
by using structured document



<https://docs.cline.bot/features/plan-and-act>



# Agent Skills in Cline

- Enable Background Edit

**Experimental:** Allows editing files in background without opening the diff view in editor.
- Enable Multi-Root Workspace

**Experimental:** Allows cline to work across multiple workspaces.
- Enable Skills

**Experimental:** Enables Skills for reusable, on-demand agent instructions from .cline/skills/ directories.
- Enable YOLO Mode

**EXPERIMENTAL & DANGEROUS:** This mode disables safety checks and user confirmations. Cline will automatically approve all actions without asking. Use with extreme caution.

*Experimental feature !!*



# Q/A

