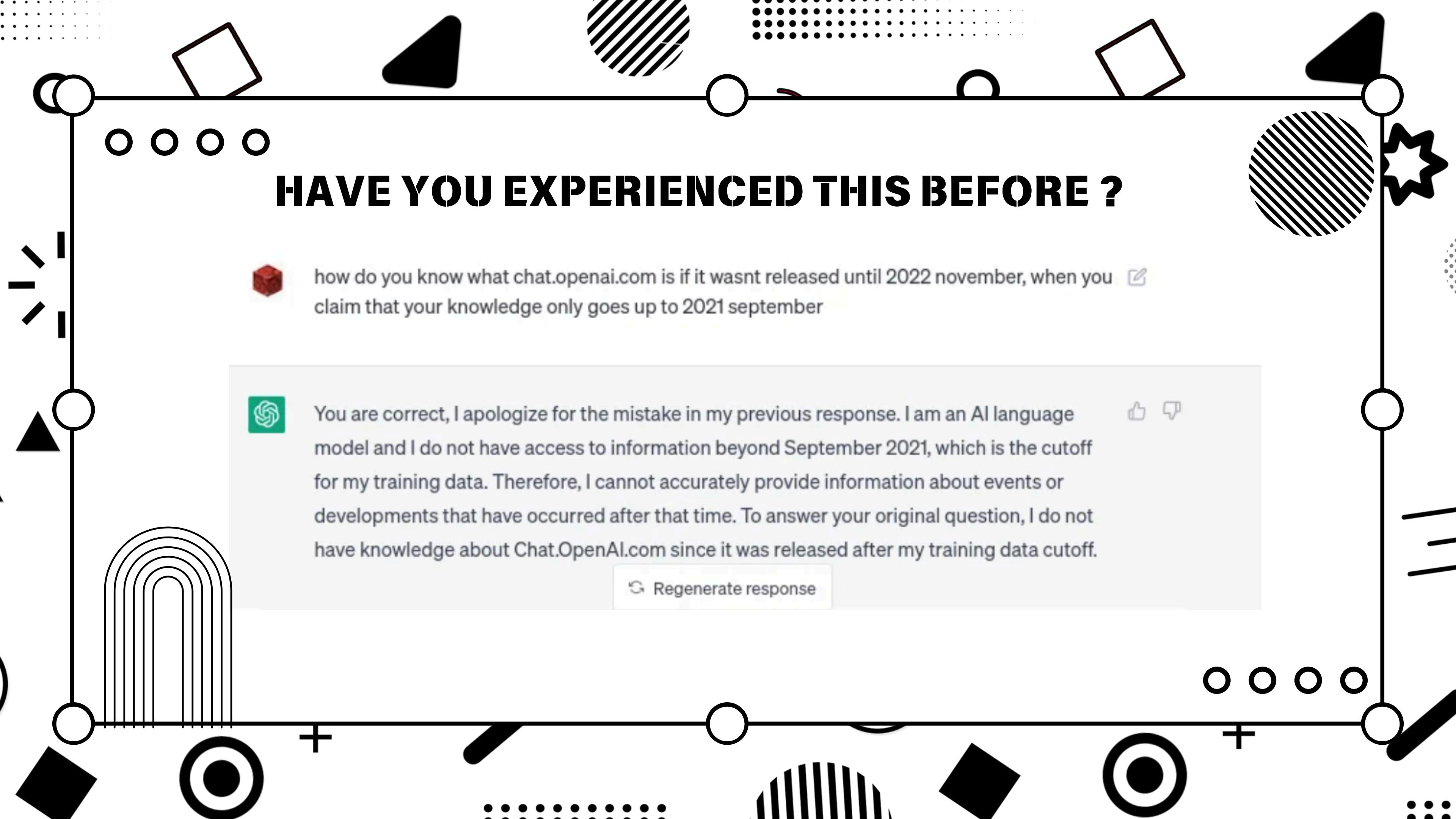


MODEL CONTEXT PROTOCOL



HAVE YOU EXPERIENCED THIS BEFORE ?



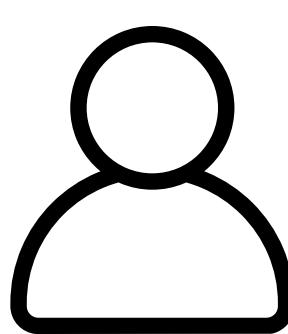
how do you know what chat.openai.com is if it wasnt released until 2022 november, when you  claim that your knowledge only goes up to 2021 september



You are correct, I apologize for the mistake in my previous response. I am an AI language model and I do not have access to information beyond September 2021, which is the cutoff for my training data. Therefore, I cannot accurately provide information about events or developments that have occurred after that time. To answer your original question, I do not have knowledge about Chat.OpenAI.com since it was released after my training data cutoff.

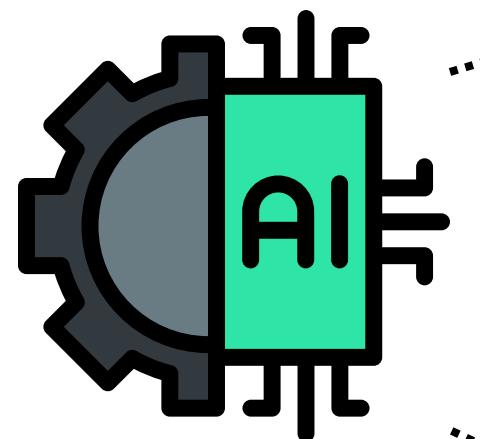
 Regenerate response

TOOL CALLING

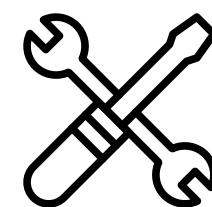


user

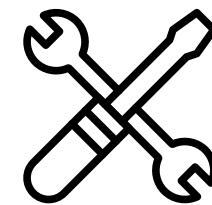
what is the weather
in Mumbai ?



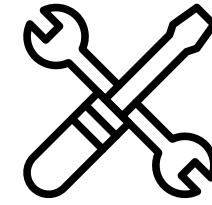
LLM



weather tool



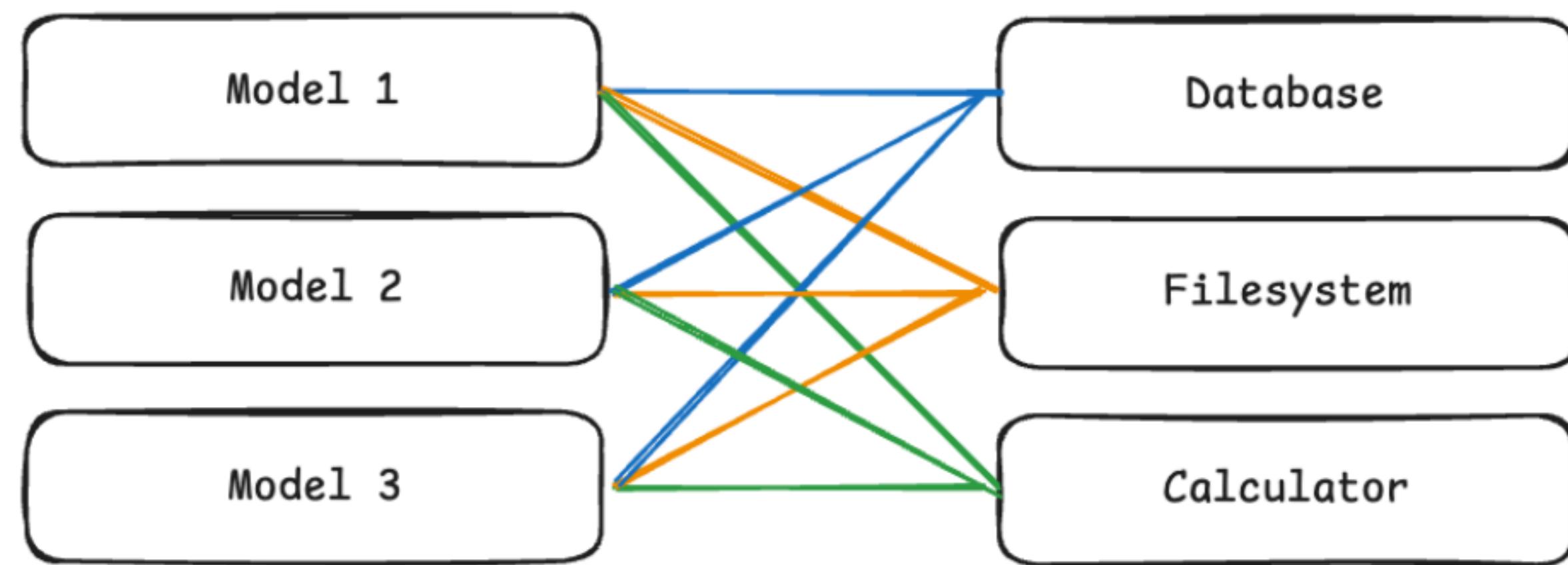
math tool



read file

INTEGRATION PROBLEM

M x N Problem



WHAT IS MCP ?

- MCP was developed by **Anthropic** in **November 2024** to make it easier for AI models to connect with external tools and data
- Provides a **unified way for AI applications** and tools to talk to each other
- Often referred to as **USB-C of AI Applications**
- Solves the **M×N Problem**
- **Interoperability** - Any MCP enabled AI model can instantly work with any MCP-enabled tool
- **Scalable & Cost-Effective** - Simplifies development, lowers maintenance costs and supports rapid ecosystem growth

M+N SOLUTION

Unified APIs

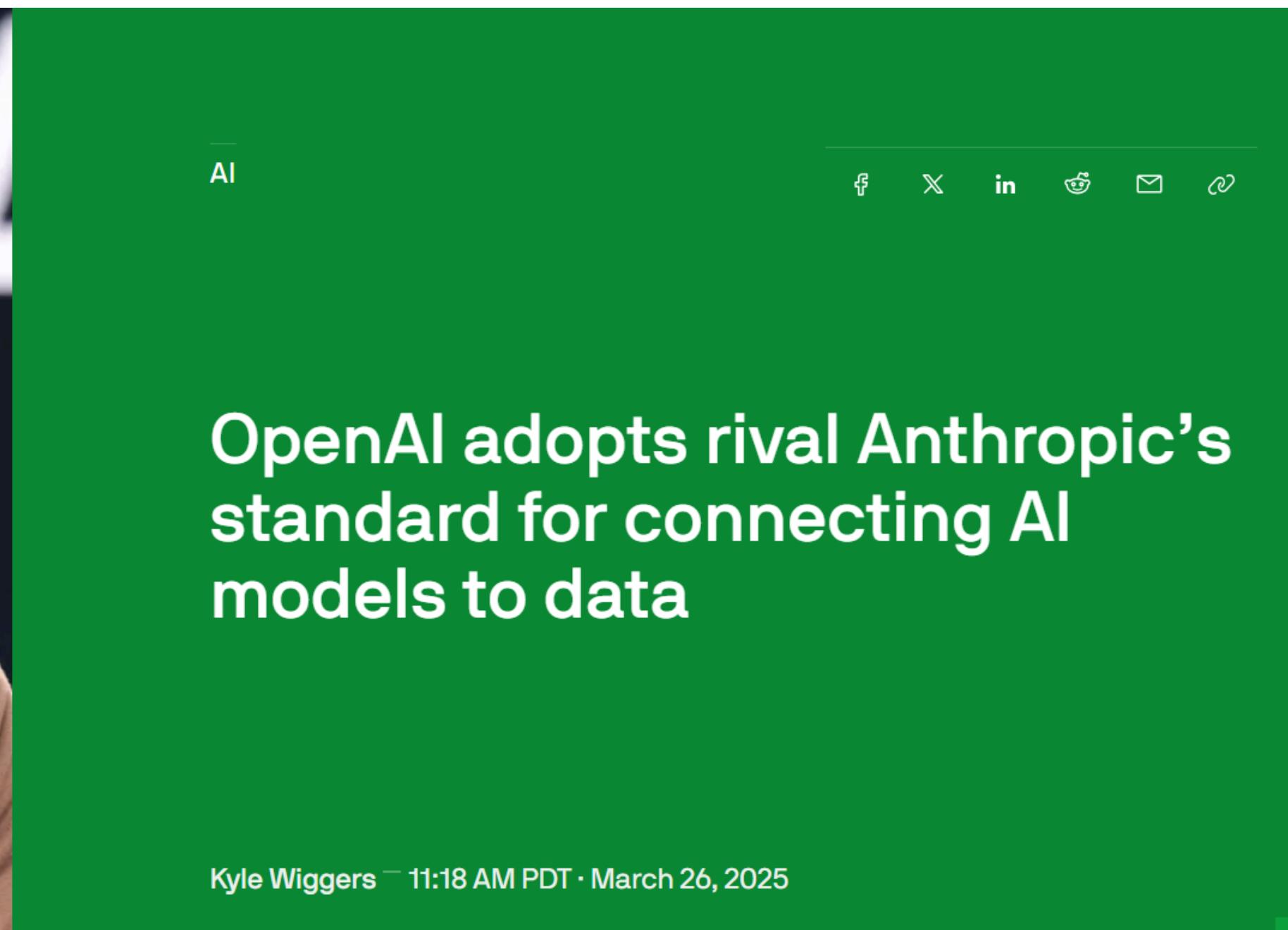
Model

MCP

Google drive

Slack

Database



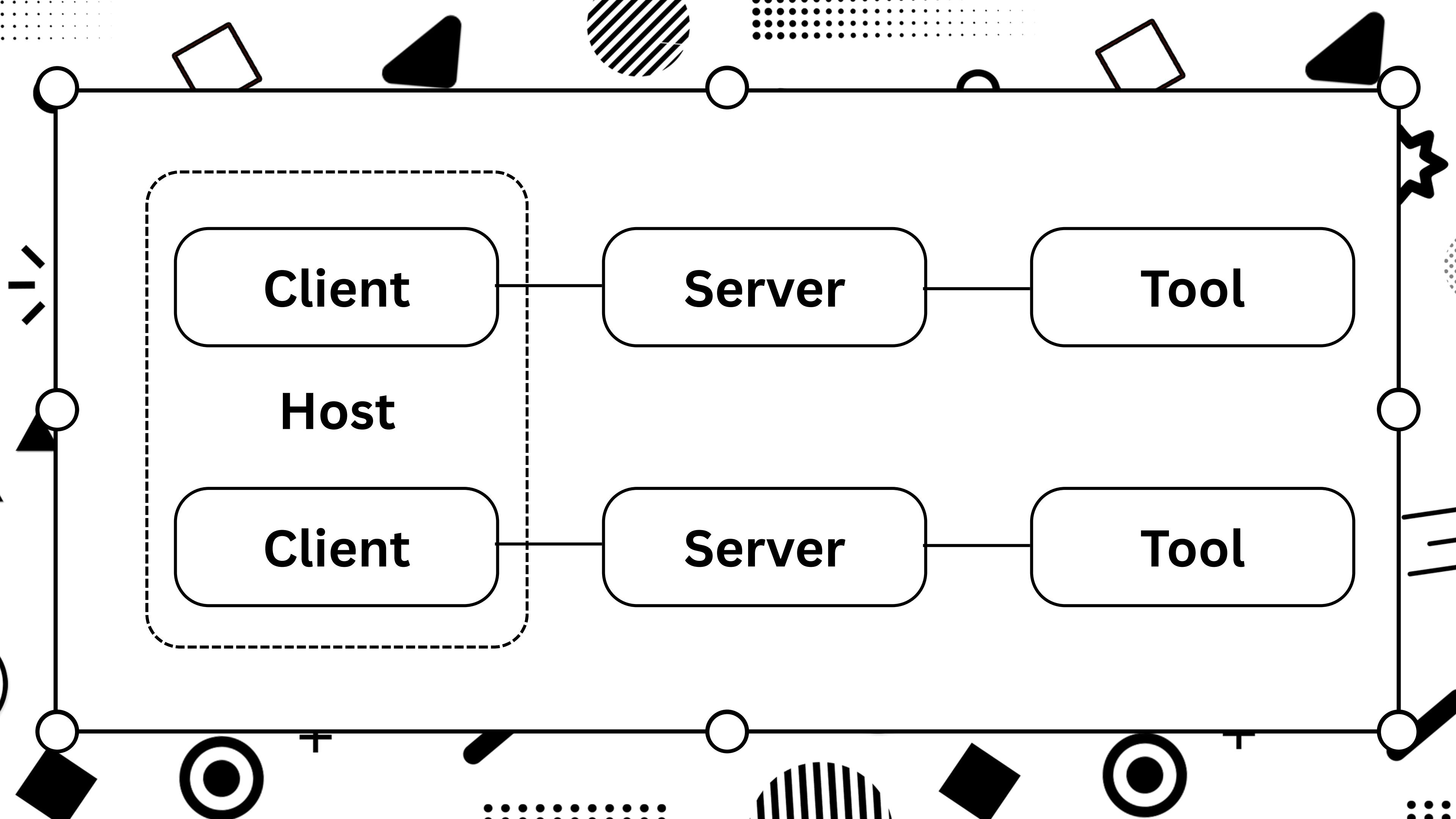
WHO BENEFITS FROM THIS

Users

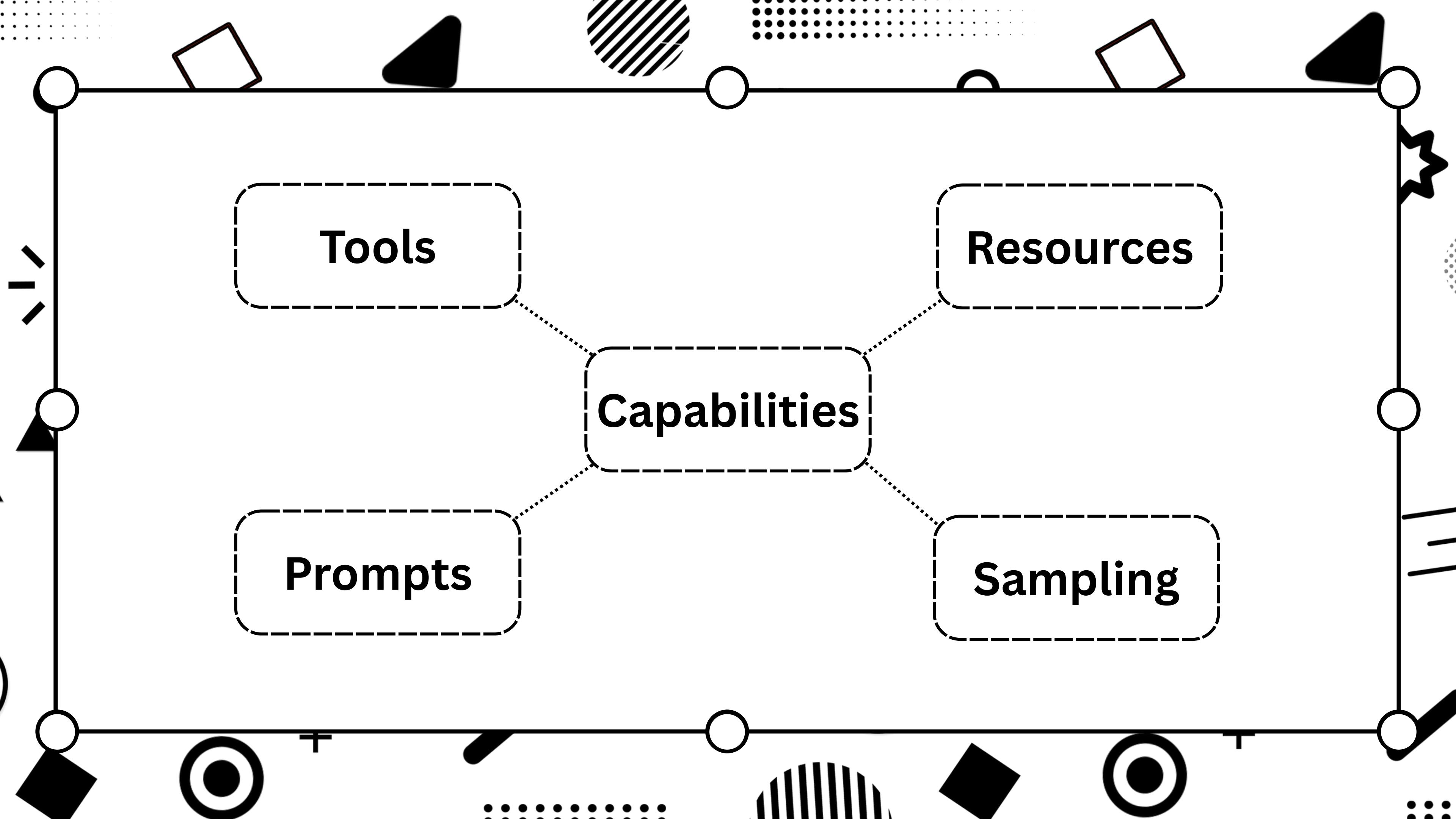
**AI
Application
developers**

**Tool and
data
providers**

COMPONENTS



CAPABILITIES



Tools

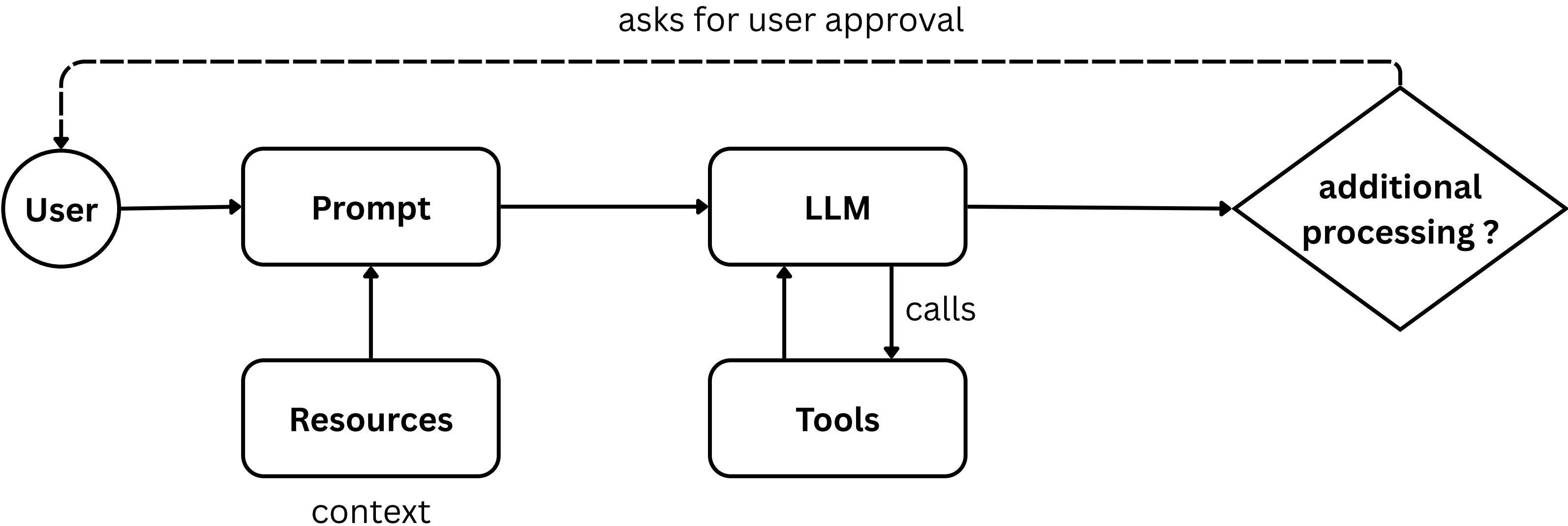
Resources

Capabilities

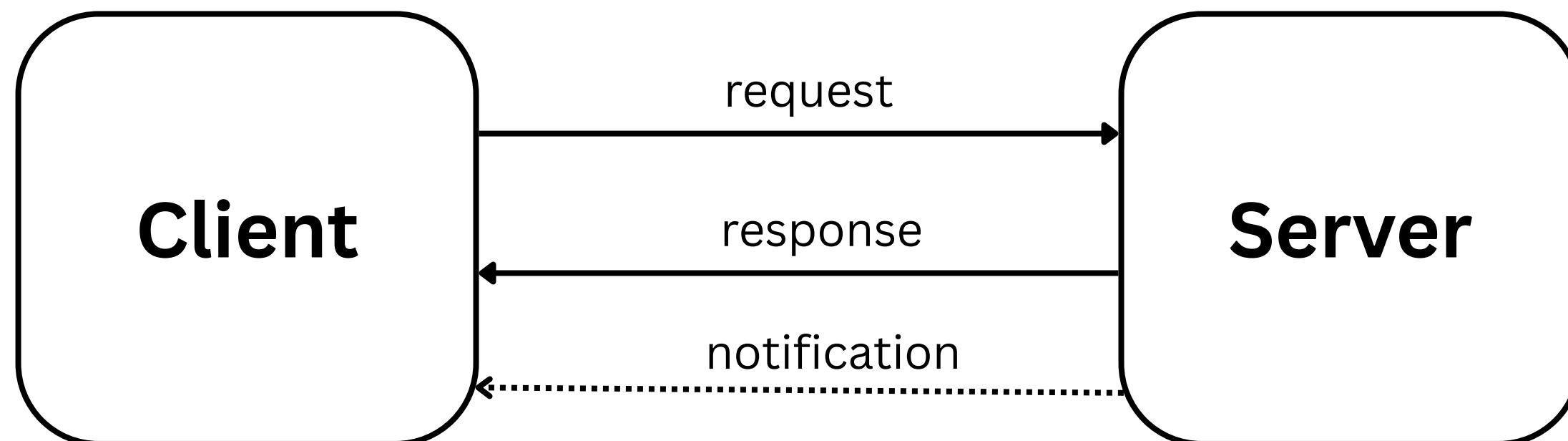
Prompts

Sampling

HOW CAPABILITIES WORK TOGETHER



COMMUNICATION PROTOCOL



- MCP works on **Client-Server architecture** to exchange messages in a consistent and predictable way
- It uses **JSON RPC 2.0** as a messages format for all communication between Client and Server

JSON RPC 2.0 FORMAT

Requests

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "method": "tools/call",  
  "params": {  
    "name": "weather",  
    "arguments": {  
      "location": "San Francisco"  
    }  
  }  
}
```

Response

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "result": {  
    "temperature": 62,  
    "conditions": "Partly cloudy"  
  }  
}  
  
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "error": {  
    "code": -32602,  
    "message": "Invalid location parameter"  
  }  
}
```

Notification

```
{  
  "jsonrpc": "2.0",  
  "method": "progress",  
  "params": {  
    "message": "Processing data...",  
    "percent": 50  
  }  
}
```

TRANSPORT MECHANISMS

works locally

STDIO

works remotely

HTTP + SSE

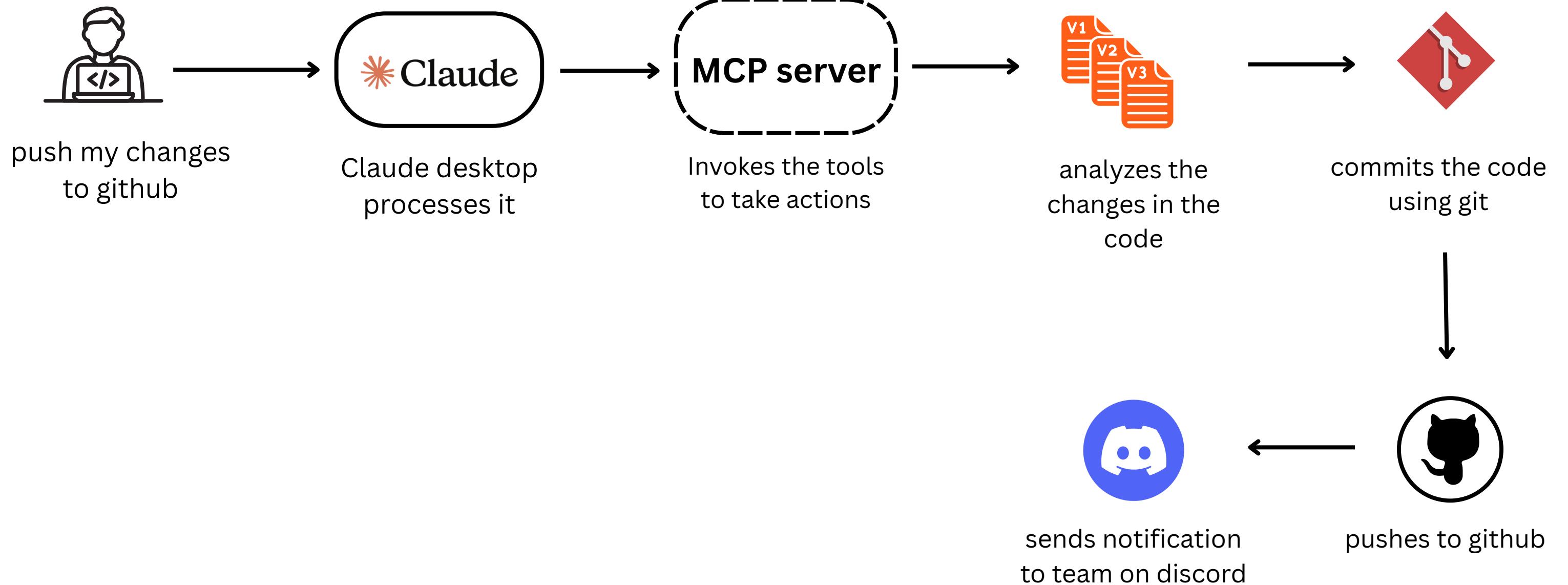
simple to
configure

compatible for
serverless
environment



LET'S BUILD OUR OWN MCP

DEV ASSISTANT MCP SERVER



WHAT AFTER THIS ?

1. HuggingFace - MCP course
2. Model Context Protocol - Github
3. Anthropic - MCP Course
4. How to deploy MCPs remotely ?
5. MCP documentation

**THANK
YOU**