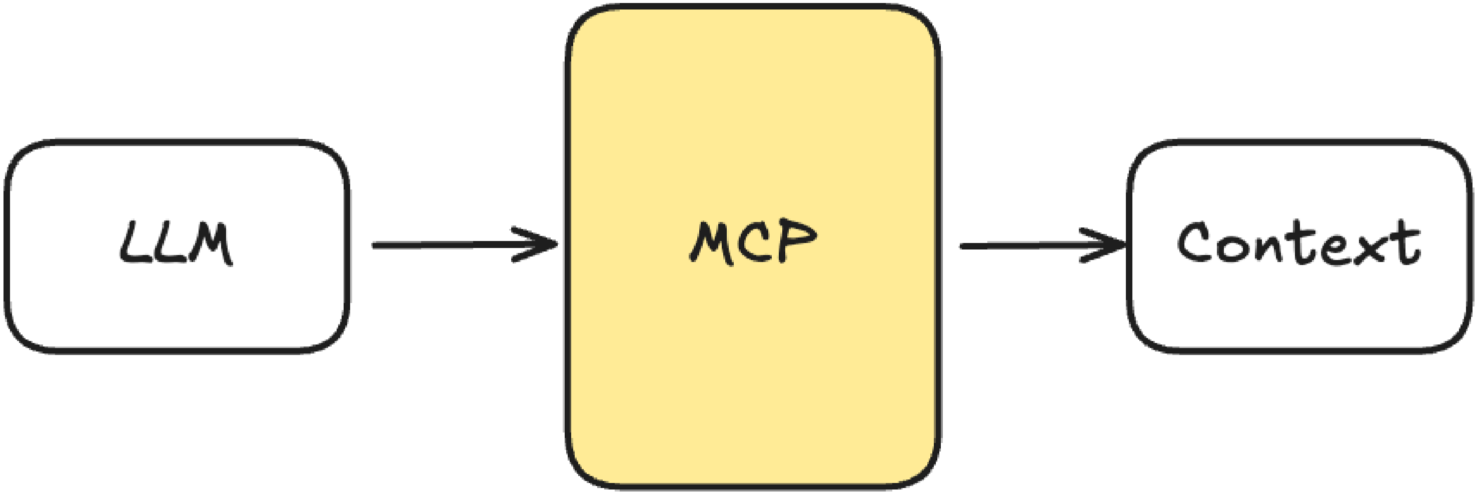
Building Agents with MCP

The HTTP Moment of AI?

Introduction to MCP

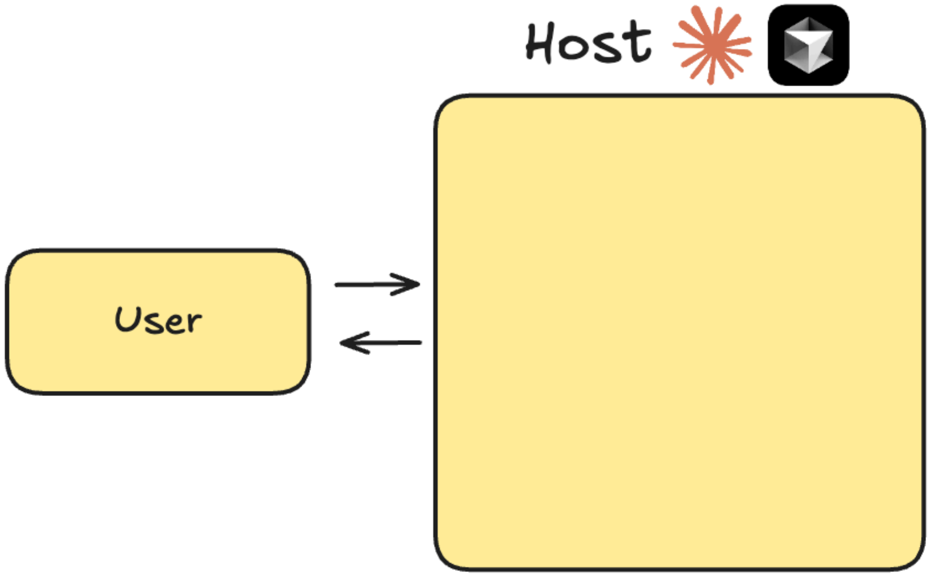
What is MCP?

Open Protocol to standardize connections between LLMs and Context

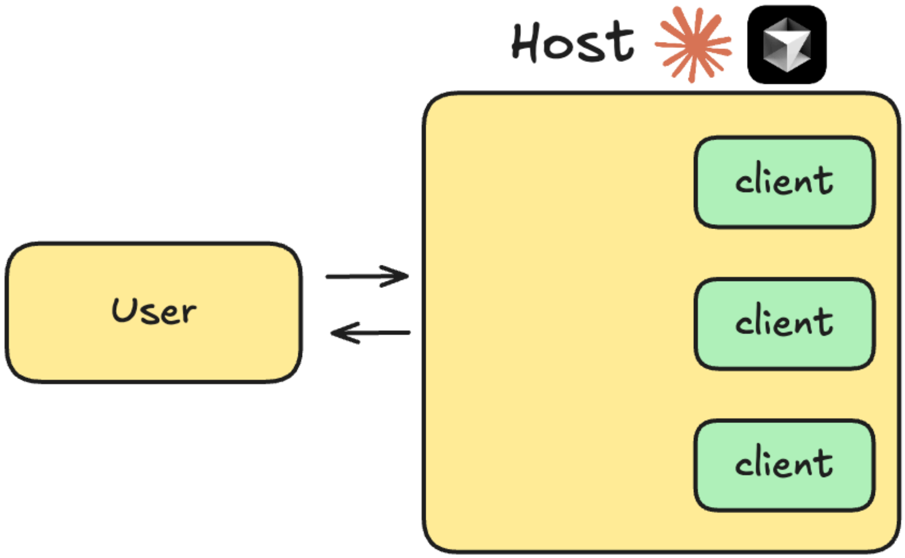


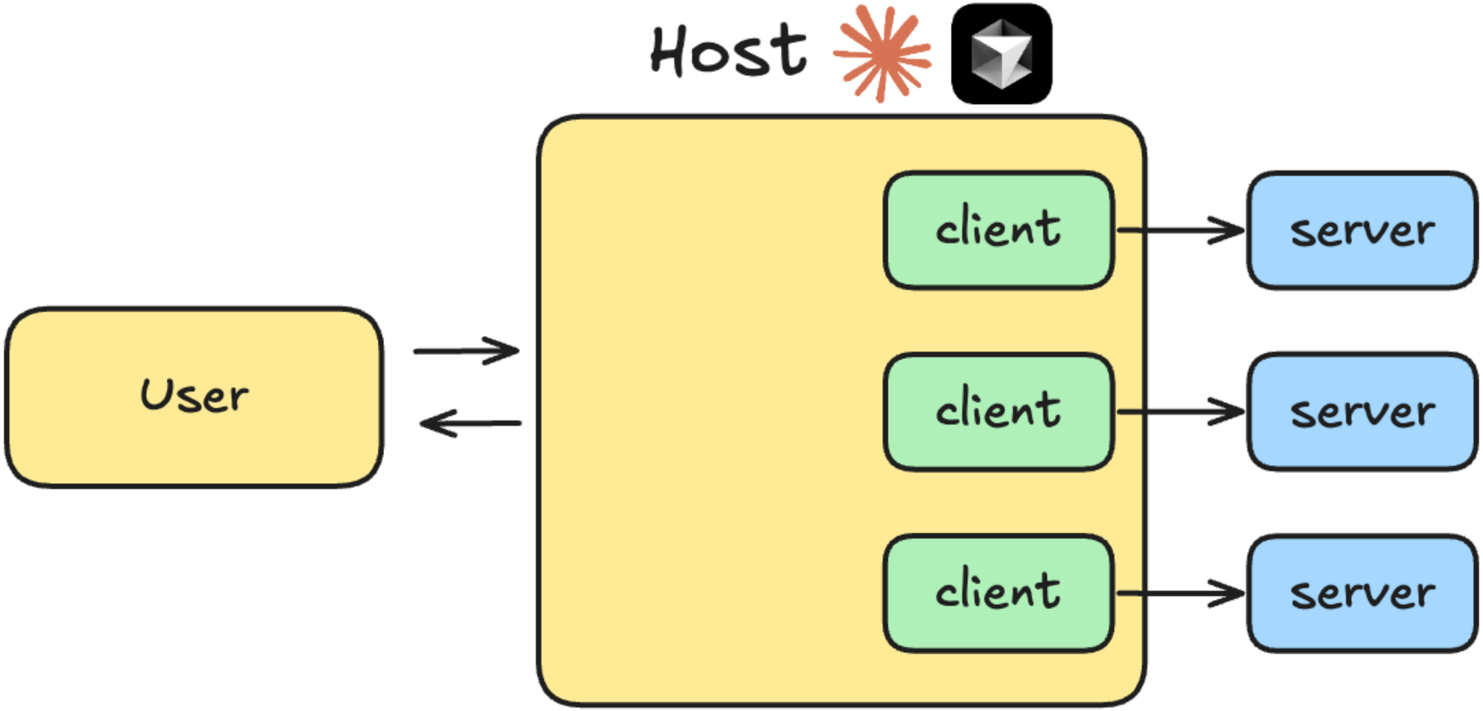
MCP is what makes AI actually useful for real apps

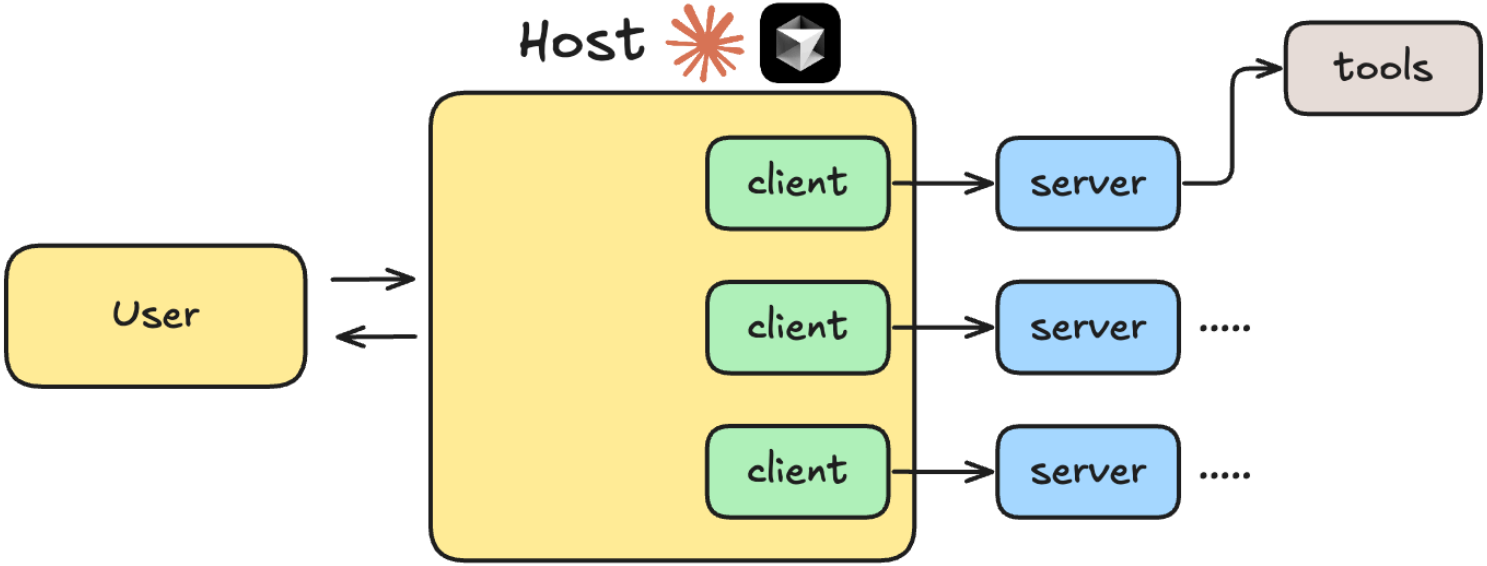
MCP Core Components



Host
User


Host
User
client
client
client


Host
User
client
client
client
server
server
server



Host

client

client

client

server

server

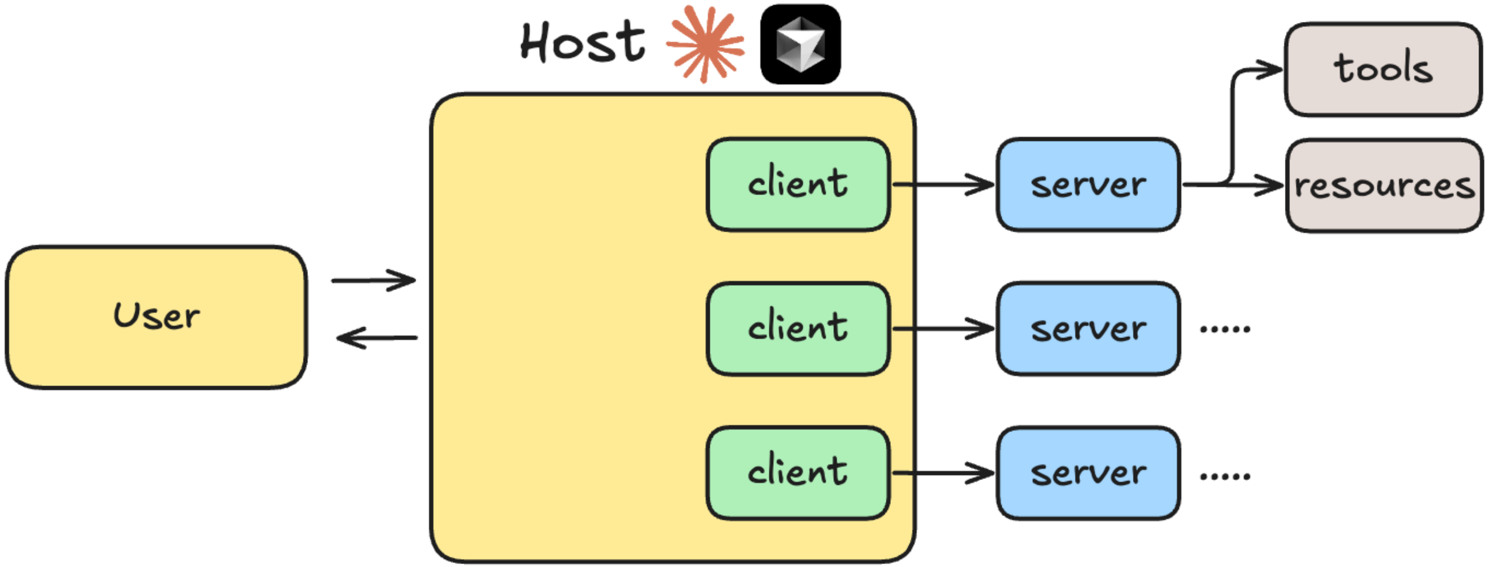
server

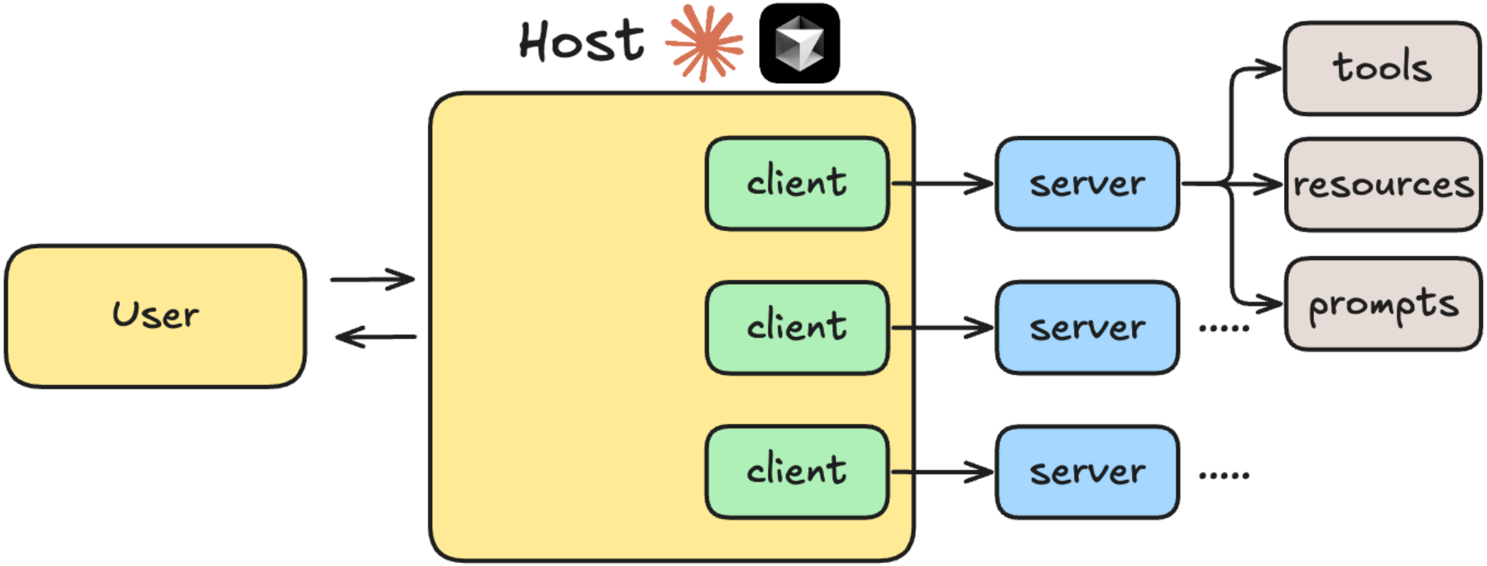
tools

User

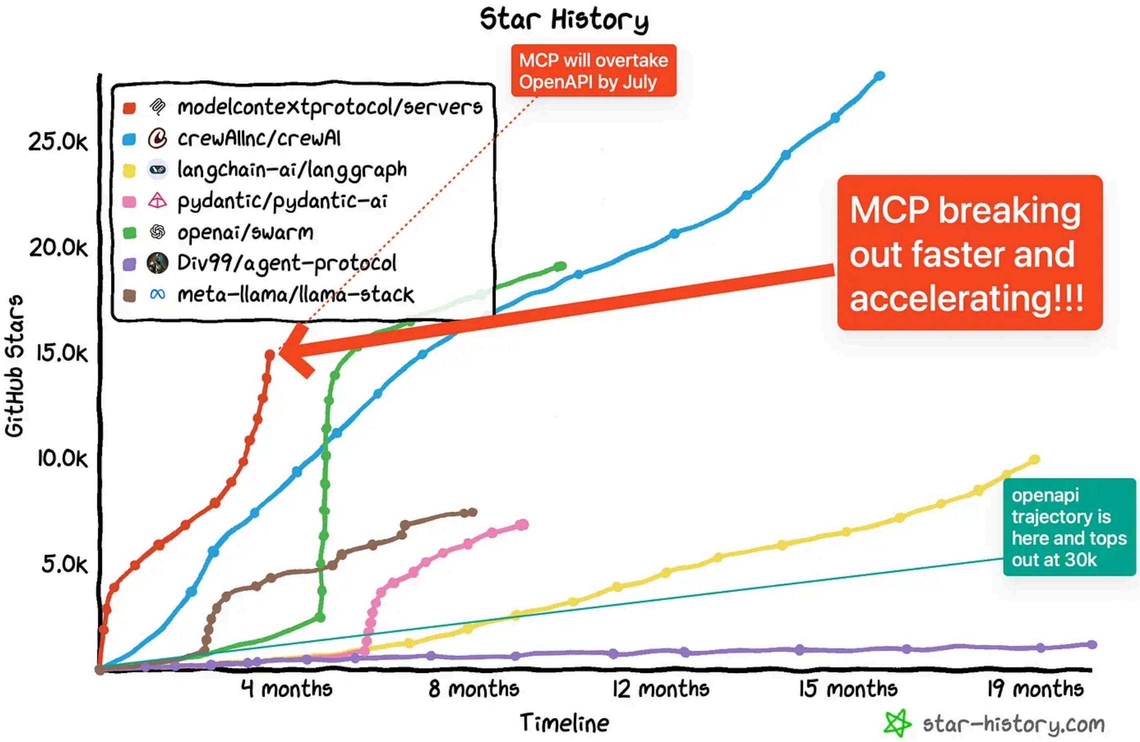
.....

.....


Host
User
client
client
client
server
server
server
.....
.....
tools
resources


Host
User
client
client
client
server
server
server
.....
.....
tools
resources
prompts

MCP is Growing Super Fast



MCP will overtake OpenAPI by July

MCP breaking out faster and accelerating!!!

openapi trajectory is here and tops out at 30k

Host

- User-facing AI application (ChatGPT, Claude Desktop, Cursor)

Host

- User-facing AI application (ChatGPT, Claude Desktop, Cursor)
- Manages user interactions and permissions

Host

- User-facing AI application (ChatGPT, Claude Desktop, Cursor)
- Manages user interactions and permissions
- Orchestrates flow between user requests, LLM, and tools

Host

- User-facing AI application (ChatGPT, Claude Desktop, Cursor)
- Manages user interactions and permissions
- Orchestrates flow between user requests, LLM, and tools
- Renders results back to users

Client

- 1:1 connection with a single Server

Host

- User-facing AI application (ChatGPT, Claude Desktop, Cursor)
- Manages user interactions and permissions
- Orchestrates flow between user requests, LLM, and tools
- Renders results back to users

Client

- 1:1 connection with a single Server
- Handles protocol-level MCP communication

Host

- User-facing AI application (ChatGPT, Claude Desktop, Cursor)
- Manages user interactions and permissions
- Orchestrates flow between user requests, LLM, and tools
- Renders results back to users

Client

- 1:1 connection with a single Server
- Handles protocol-level MCP communication
- Acts as intermediary between Host and Server

Host

- User-facing AI application (ChatGPT, Claude Desktop, Cursor)
- Manages user interactions and permissions
- Orchestrates flow between user requests, LLM, and tools
- Renders results back to users

Client

- 1:1 connection with a single Server
- Handles protocol-level MCP communication
- Acts as intermediary between Host and Server
- Manages capability discovery and invocatio

Server

- External program/service exposing capabilities

Server

- External program/service exposing capabilities
- Lightweight wrapper around existing functionality

Server

- External program/service exposing capabilities
- Lightweight wrapper around existing functionality
- Can run locally or remotely

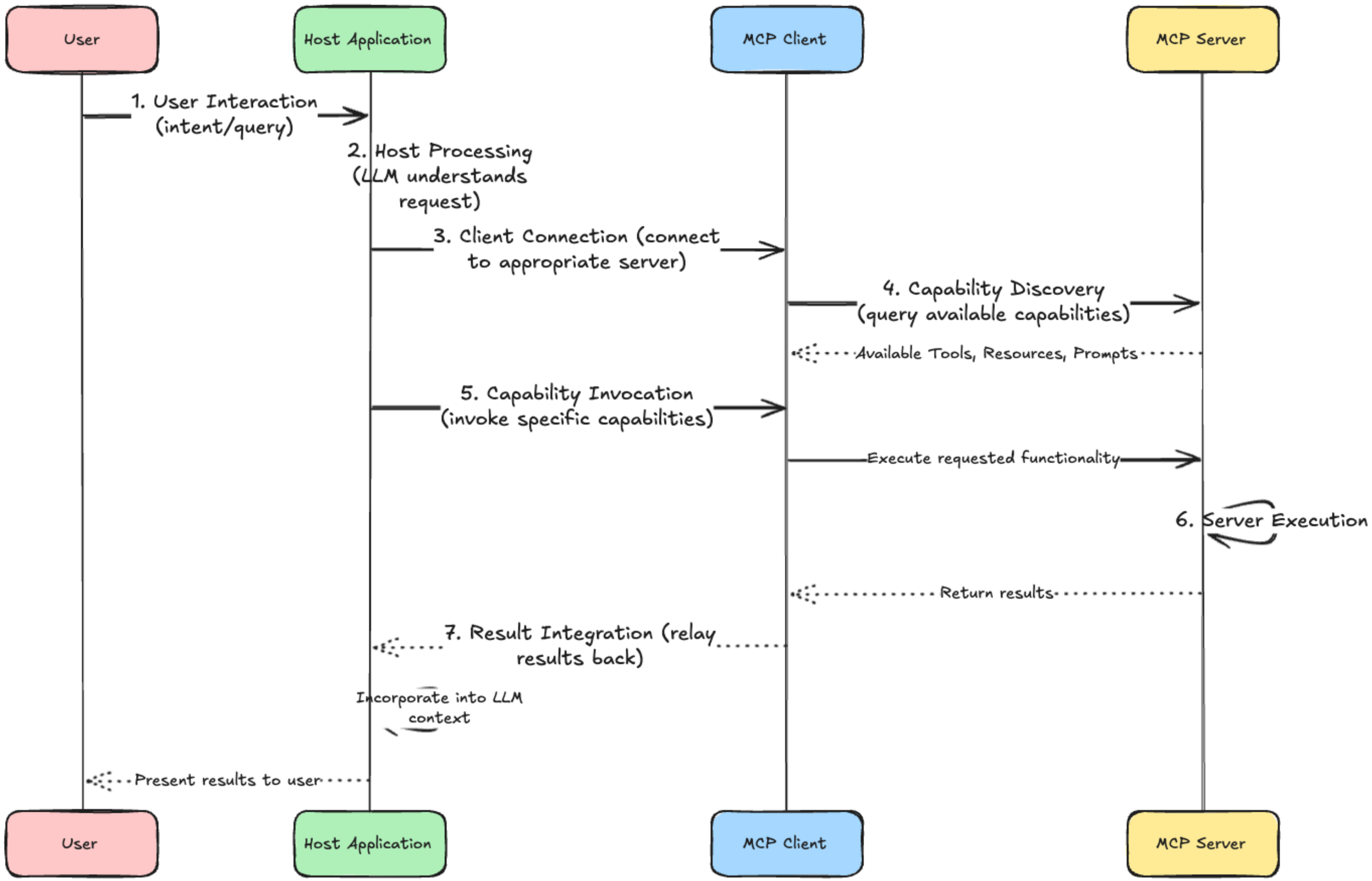
Server

- External program/service exposing capabilities
- Lightweight wrapper around existing functionality
- Can run locally or remotely
- Exposes capabilities in standardized format

Server

- External program/service exposing capabilities
- Lightweight wrapper around existing functionality
- Can run locally or remotely
- Exposes capabilities in standardized format
- Provides access to tools, data sources, or services

Communication Flow



User

Host Application

MCP Client

MCP Server

1. User Interaction (intent/query)

2. Host Processing (LLM understands request)

3. Client Connection (connect to appropriate server)

4. Capability Discovery (query available capabilities)

Available Tools, Resources, Prompts

5. Capability Invocation (invoke specific capabilities)

Execute requested functionality

6. Server Execution

Return results

7. Result Integration (relay results back)

Incorporate into LLM context

Present results to user

User

Host Application

MCP Client

MCP Server

Communication Flow

1. **User Interaction**
2. **Host Processing**
3. **Client Connection**
4. **Capability Discovery**
5. **Capability Invocation**
6. **Server Execution**
7. **Result Integration**

Demo - Practical Introduction to MCP SDK

Demo - Creating our First MCP Server

(and using it with Claude Desktop!)

MCP Capabilities: Tools, Resources, Prompts & Sampling

MCP Capabilities

Core Primitives

- **Tools**

Core Primitives

- **Tools**
- **Resources**

Core Primitives

- **Tools**
- **Resources**
- **Prompts**

Core Primitives

- **Tools**
- **Resources**
- **Prompts**
- **Sampling**

Tools

Tools

- Model-controlled executable functions

Tools

- Model-controlled executable functions
- Require user approval

Tools

- Model-controlled executable functions
- Require user approval
- Can have side effects

Tools

- Model-controlled executable functions
- Require user approval
- Can have side effects
- Example: Fetching GitHub repository data, sending emails, or updating a database

```python
def send_email(to: str, subject: str, body: str) -> str:
    """
    Send an email to the given address
    """
    ... # Implementation logic
    return {
        "status": "success"
    }
```

Tools

- Model-controlled executable functions
- Require user approval
- Can have side effects
- Example: Fetching GitHub repository data, sending emails, or updating a database

```python
def send_email(to: str, subject: str, body: str) -> str:
    """
    Send an email to the given address
    """
    ... # Implementation logic
    return {
        "status": "success"
    }
```

- **Tools** are the most powerful MCP capabilities

Resources

Resources

- Application-controlled data access

Resources

- Application-controlled data access
- Read-only operations

Resources

- Application-controlled data access
- Read-only operations
- Example: File contents, database records

```python
def get_file_contents(file_path: str) -> str:
    """
    Get the contents of a file
    """
    ... # Implementation logic
    return {
        "contents": "File contents"
    }
```

Prompts

- User-controlled templates

Prompts

- User-controlled templates
- Structure interactions

Prompts

- User-controlled templates
- Structure interactions
- Guide workflows

Prompts

- User-controlled templates
- Structure interactions
- Guide workflows
- Example: Code review templates

```python
def plan_project(project_name: str) -> str:
    """
    Plan a project
    """
    ... # Implementation logic
    return {
        "plan": "Project plan"
    }
```

Sampling

- Server-initiated LLM interactions

Sampling

- Server-initiated LLM interactions
- Requires client facilitation

Sampling

- Server-initiated LLM interactions
- Requires client facilitation
- Enables agentic behaviors

Sampling

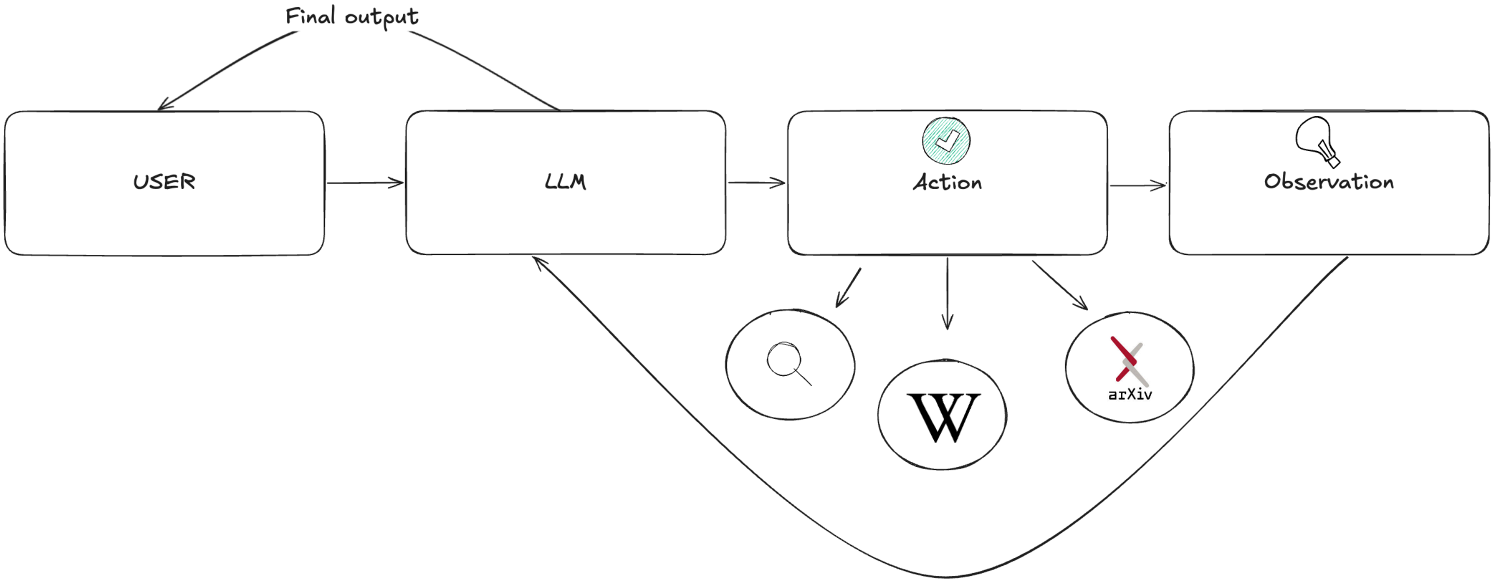
- Server-initiated LLM interactions
- Requires client facilitation
- Enables agentic behaviors
- Example: Multi-step analysis

```python
def request_sampling(messages, system_prompt=None, include_context="none"):
    """Request LLM sampling from the client."""
    ... # Implementation logic
    return {
        "role": "assistant",
        "content": "Analysis of the provided data..."
    }
```

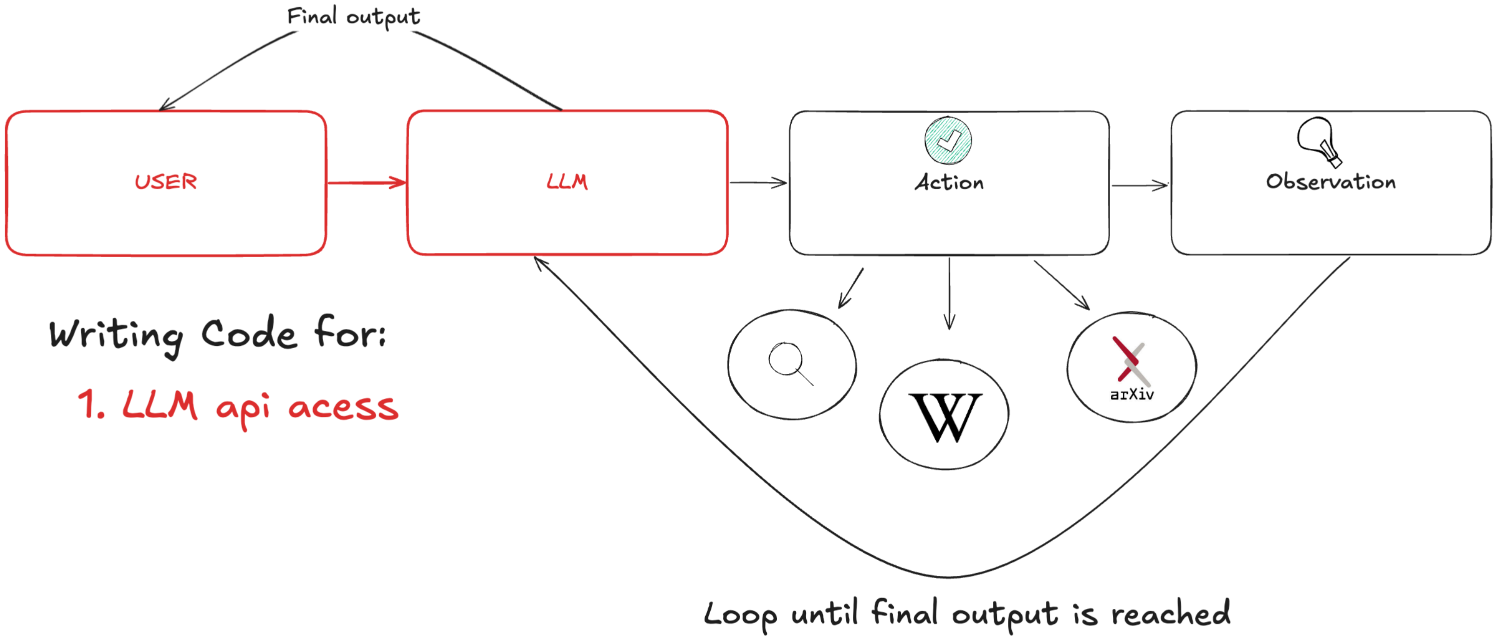
Whiteboard - How MCP Capabilities Work Together

Demo - Implementing MCP Tools, Resources, Prompts & Sampling

Building Agents with MCP

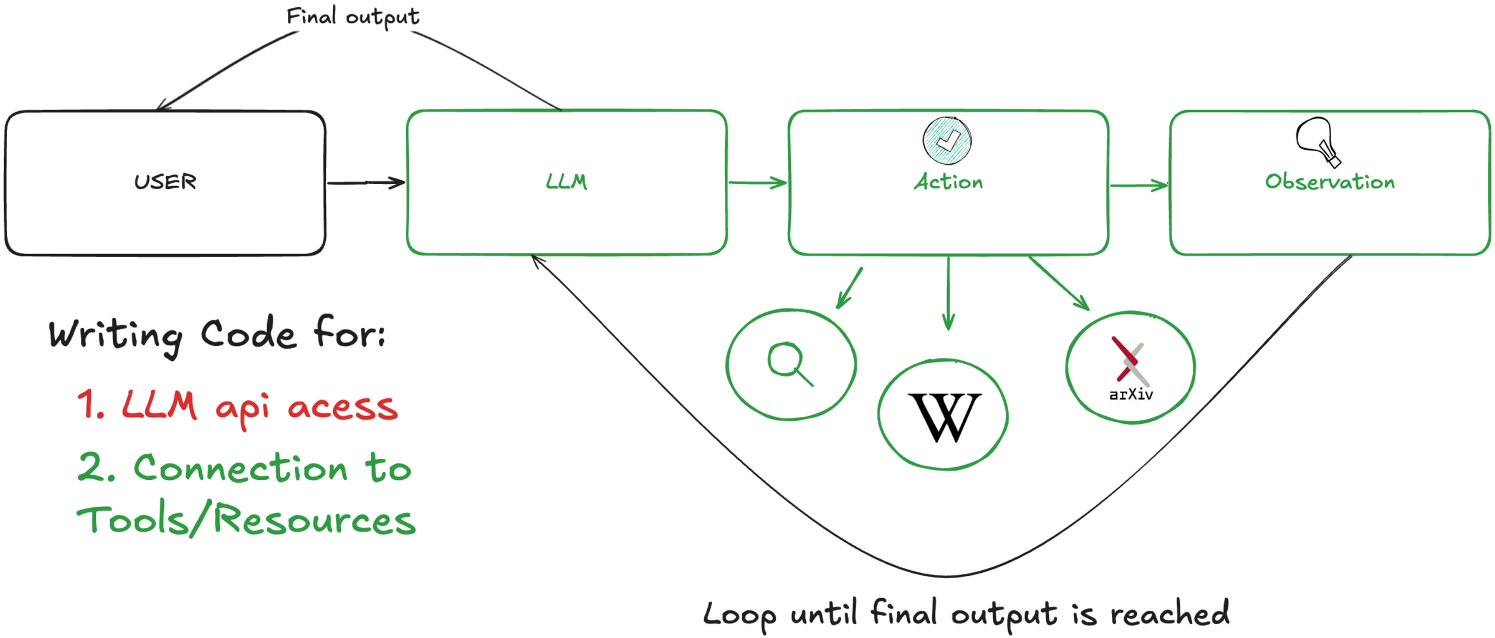


Loop until final output is reached



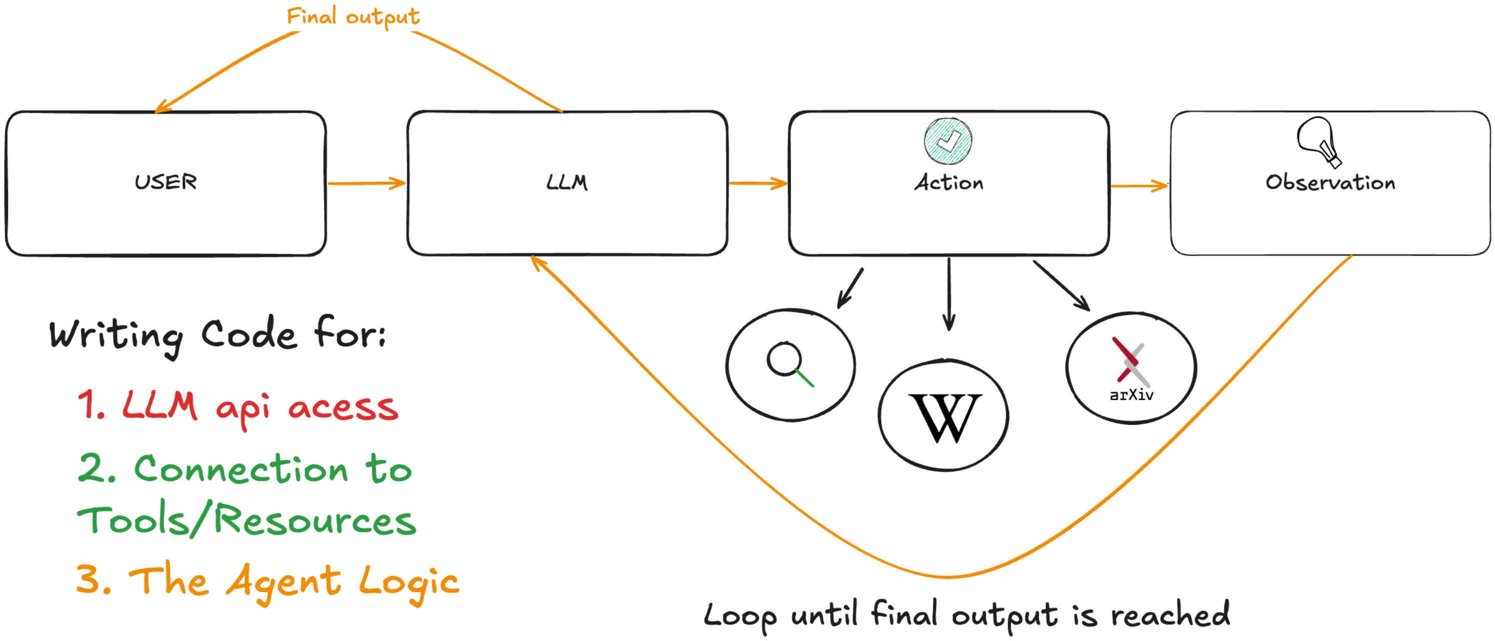
Writing Code for:

1. LLM api acess



Writing Code for:

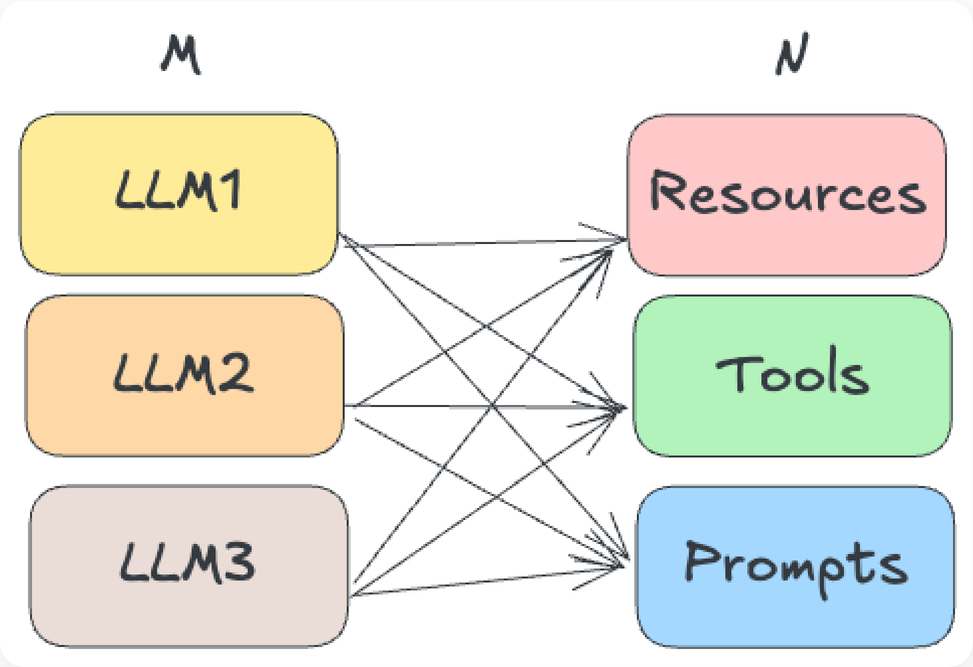
1. LLM api acess
2. Connection to Tools/Resources



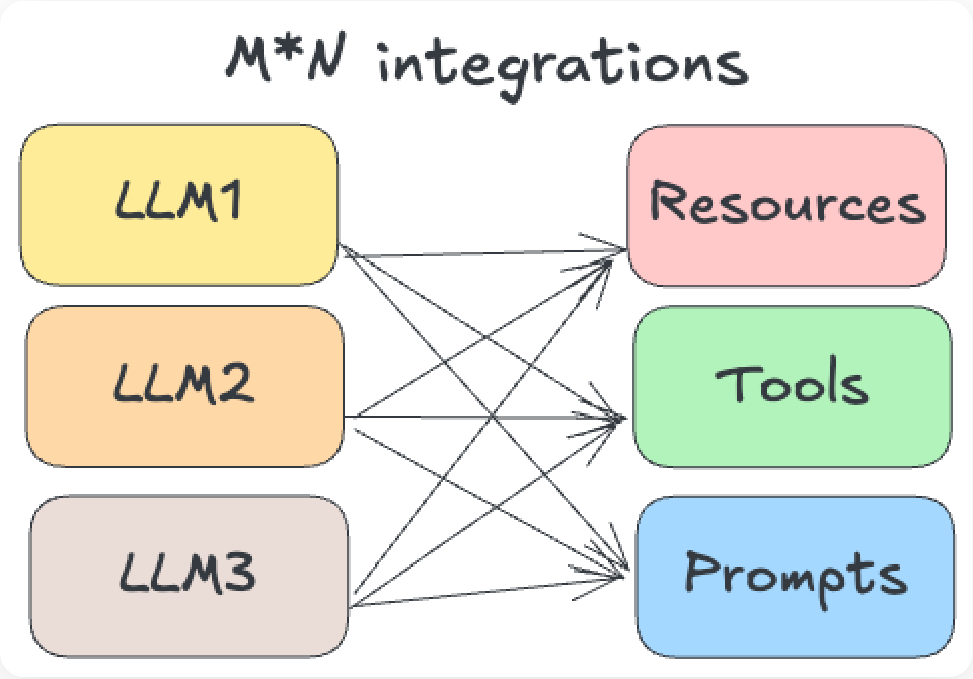
Writing Code for:

1. LLM api acess
2. Connection to Tools/Resources
3. The Agent Logic

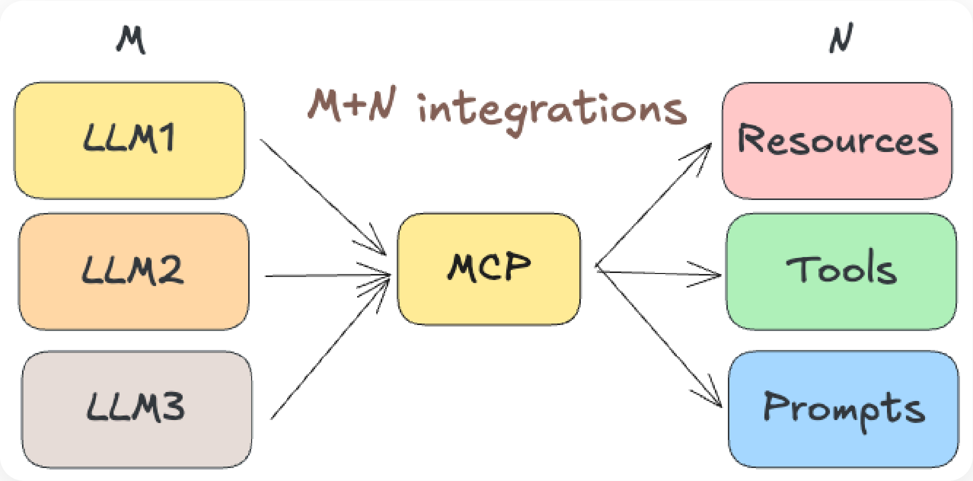
The MN Integration Problem: Multiple LLMs



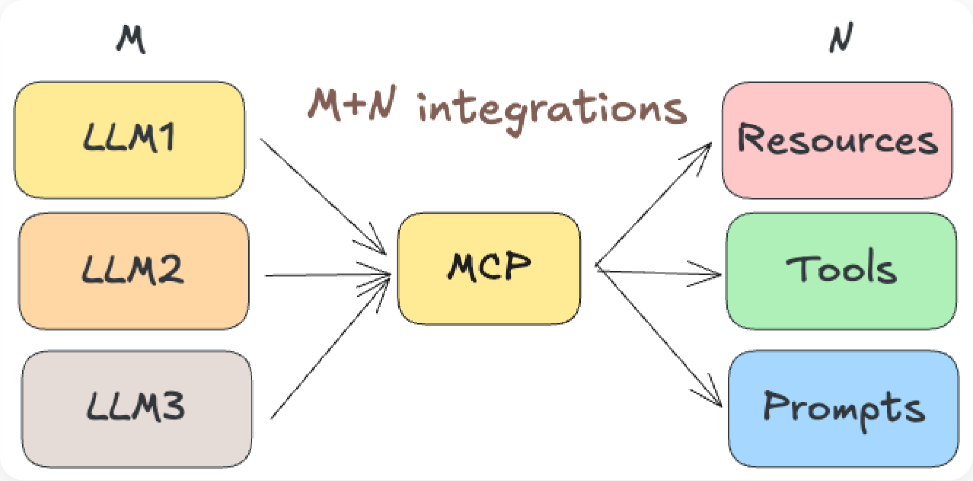
The MN Integration Problem: Multiple LLMs



The MN Integration Problem: Multiple LLMs



MCP Simplifies the Integration of Problem for Agent Development



Whiteboard - Agent Development in the Era of MCP

Demo - Building Agents with MCP Using Google's ADK

Demo - Building Agents with MCP Using LangGraph

Demo - Building Agents with MCP Using OpenAI's Agent SDK

Workflow Automation Revolution

Workflow Automation Revolution

- Pro tip: use MCP in an app like Claude or Cursor to feel the power of MCP

Workflow Automation Revolution

- Pro tip: use MCP in an app like Claude or Cursor to feel the power of MCP
- Data analysis across multiple systems without custom code

Workflow Automation Revolution

- Pro tip: use MCP in an app like Claude or Cursor to feel the power of MCP
- Data analysis across multiple systems without custom code
- Automated reporting and insights

Workflow Automation Revolution

- Pro tip: use MCP in an app like Claude or Cursor to feel the power of MCP
- Data analysis across multiple systems without custom code
- Automated reporting and insights
- Context-Aware Applications that can communicate with context and other apps easily

Workflow Automation Revolution

- Pro tip: use MCP in an app like Claude or Cursor to feel the power of MCP
- Data analysis across multiple systems without custom code
- Automated reporting and insights
- Context-Aware Applications that can communicate with context and other apps easily
- Personal assistants with deep system access (Claude Desktop)

Workflow Automation Revolution

- Pro tip: use MCP in an app like Claude or Cursor to feel the power of MCP
- Data analysis across multiple systems without custom code
- Automated reporting and insights
- Context-Aware Applications that can communicate with context and other apps easily
- Personal assistants with deep system access (Claude Desktop)
- Development environments with intelligent tooling (Claude-Code, Cursor)

Workflow Automation Revolution

- Pro tip: use MCP in an app like Claude or Cursor to feel the power of MCP
- Data analysis across multiple systems without custom code
- Automated reporting and insights
- Context-Aware Applications that can communicate with context and other apps easily
- Personal assistants with deep system access (Claude Desktop)
- Development environments with intelligent tooling (Claude-Code, Cursor)
- Multi-Language Support (Python, TypeScript, Swift, Kotlin, Java, Go)

Fun Demo Time! - Using MCP from Claude Desktop and Cursor! Hacks, Tips, and Tricks!

MCP Security Considerations

⚠️ Security Risks: MCP Vulnerabilities

- **Critical "Tool Poisoning Attacks" Discovered**

⚠️ Security Risks: MCP Vulnerabilities

- **Critical "Tool Poisoning Attacks" Discovered**
- Malicious instructions embedded in MCP tool descriptions

⚠️ Security Risks: MCP Vulnerabilities

- **Critical "Tool Poisoning Attacks" Discovered**
- Malicious instructions embedded in MCP tool descriptions
- Instructions invisible to users but visible to LLMs

⚠️ Security Risks: MCP Vulnerabilities

- **Critical "Tool Poisoning Attacks" Discovered**
- Malicious instructions embedded in MCP tool descriptions
- Instructions invisible to users but visible to LLMs
- **Potential Damage:** Data exfiltration, hijacked agent behavior

⚠️ Security Risks: MCP Vulnerabilities

- **Critical "Tool Poisoning Attacks" Discovered**
- Malicious instructions embedded in MCP tool descriptions
- Instructions invisible to users but visible to LLMs
- **Potential Damage:** Data exfiltration, hijacked agent behavior
- **Mitigation Strategies:** Tool pinning, clear UI patterns, cross-server protection

⚠️ Security Risks: MCP Vulnerabilities

- **Critical "Tool Poisoning Attacks" Discovered**
- Malicious instructions embedded in MCP tool descriptions
- Instructions invisible to users but visible to LLMs
- **Potential Damage:** Data exfiltration, hijacked agent behavior
- **Mitigation Strategies:** Tool pinning, clear UI patterns, cross-server protection
- **Reference:** Invariant Security Research

⚠️ Security Risks: MCP Vulnerabilities

- **Critical "Tool Poisoning Attacks" Discovered**
- Malicious instructions embedded in MCP tool descriptions
- Instructions invisible to users but visible to LLMs
- **Potential Damage:** Data exfiltration, hijacked agent behavior
- **Mitigation Strategies:** Tool pinning, clear UI patterns, cross-server protection
- **Reference:** Invariant Security Research

Key Takeaway: Extensive guardrailing needed for production deployments

The Protocol "Wars"

The Protocol "Wars"

MCP vs A2A vs ACP

Three Major Players Competing/Complimenting? for Standardization:

The Protocol "Wars"

MCP vs A2A vs ACP

Three Major Players Competing/Complimenting? for Standardization:

MCP (Anthropic): AI-to-tool communication and data access

The Protocol "Wars"

MCP vs A2A vs ACP

Three Major Players Competing/Complimenting? for Standardization:

MCP (Anthropic): AI-to-tool communication and data access

A2A (Google): Agent-to-agent system communication, secure collaboration

The Protocol "Wars"

MCP vs A2A vs ACP

Three Major Players Competing/Complimenting? for Standardization:

MCP (Anthropic): AI-to-tool communication and data access

A2A (Google): Agent-to-agent system communication, secure collaboration

ACP (IBM Research): Agent Communication Protocol, focuses on practical adoption first

The Protocol "Wars"

MCP vs A2A vs ACP

Three Major Players Competing/Complimenting? for Standardization:

MCP (Anthropic): AI-to-tool communication and data access

A2A (Google): Agent-to-agent system communication, secure collaboration

ACP (IBM Research): Agent Communication Protocol, focuses on practical adoption first

The Stakes: Who becomes the "HTTP" of AI agent communication?

The Protocol "Wars"

MCP vs A2A vs ACP

Three Major Players Competing/Complimenting? for Standardization:

MCP (Anthropic): AI-to-tool communication and data access

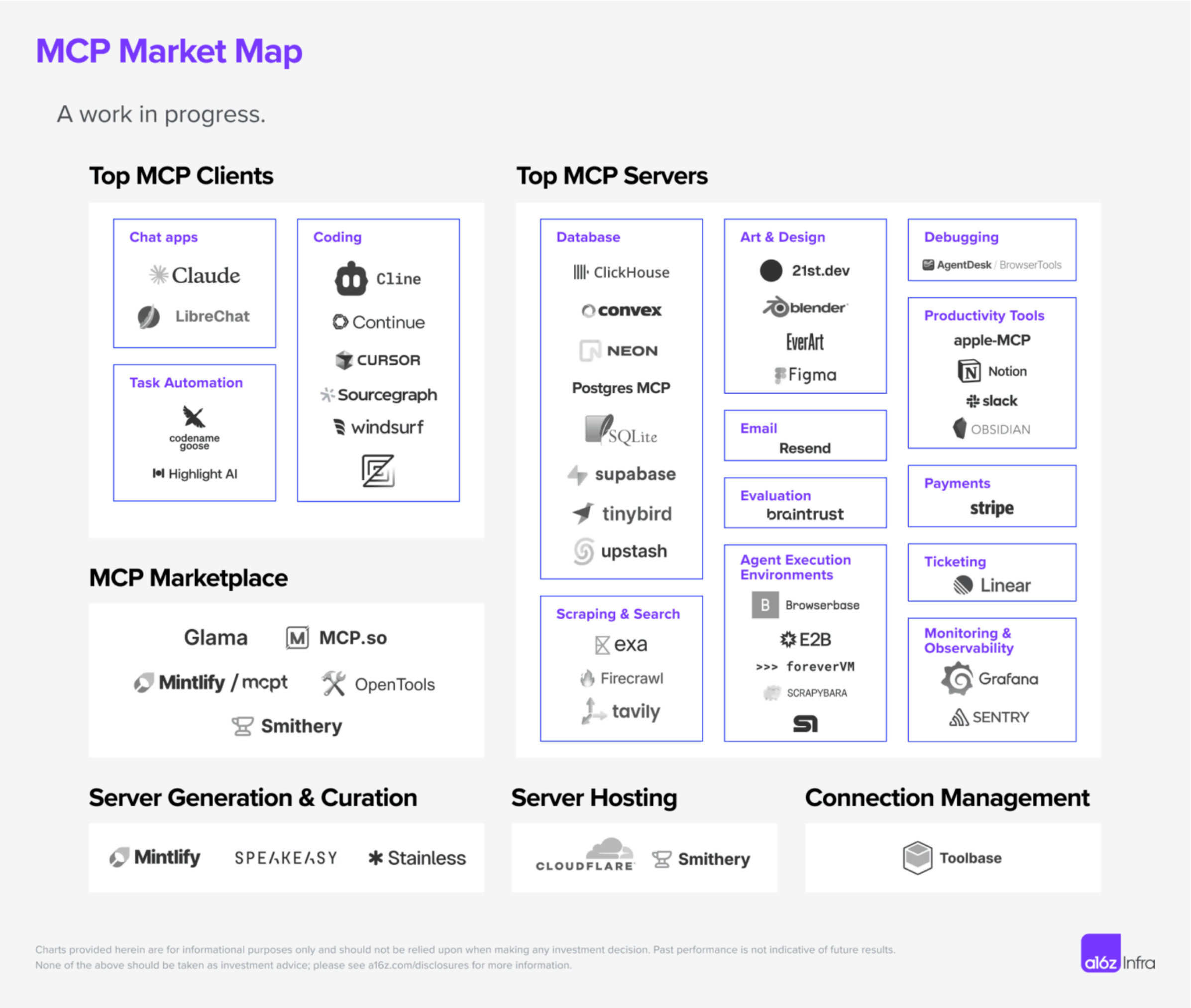
A2A (Google): Agent-to-agent system communication, secure collaboration

ACP (IBM Research): Agent Communication Protocol, focuses on practical adoption first

The Stakes: Who becomes the "HTTP" of AI agent communication?

Current Reality: Fragmentation risk vs innovation through competition

The Growing MCP Ecosystem



MCP Market Map

A work in progress.

Top MCP Clients

Chat apps
- Claude
- LibreChat

Coding
- Cline
- Continue
- CURSOR
- Sourcegraph
- windsurf

Task Automation
- codename goose
- Highlight AI

Top MCP Servers

Database
- ClickHouse
- convex
- NEON
- Postgres MCP
- SQLite
- supabase
- tinybird
- upstash

Scraping & Search
- exa
- Firecrawl
- tavily

Art & Design
- 21st.dev
- blender
- EverArt
- Figma

Email
- Resend

Evaluation
- braintrust

Agent Execution Environments
- Browserbase
- E2B
- >>> foreverVM
- SCRAPYBARA

Debugging
- AgentDesk / BrowserTools

Productivity Tools
- apple-MCP
- Notion
- slack
- OBSIDIAN

Payments
- stripe

Ticketing
- Linear

Monitoring & Observability
- Grafana
- SENTRY

MCP Marketplace

- Glama
- MCP.so
- Mintlify / mcpt
- OpenTools
- Smithery

Server Generation & Curation

- Mintlify
- SPEAKEASY
- Stainless

Server Hosting

- CLOUDFLARE
- Smithery

Connection Management

- Toolbase

Charts provided herein are for informational purposes only and should not be relied upon when making any investment decision. Past performance is not indicative of future results. None of the above should be taken as investment advice; please see a16z.com/disclosures for more information.

a16z Infra

Whiteboard + Demo - Practical MCP Security Tips

Connect With Me

📚 Blog

🔗 LinkedIn

🐦 Twitter/X

📺 YouTube

📧 Email: lucasenkrateia@gmail.com