



# Let's Learn MCP

Model Context Protocol

Katie Savage

James Montemagno

Technical PMs for Developer Community, Microsoft

# Agenda

1. What is MCP? How does it work? How do you use it?
2. In-depth overview of MCP features
3. Building an MCP server in C#

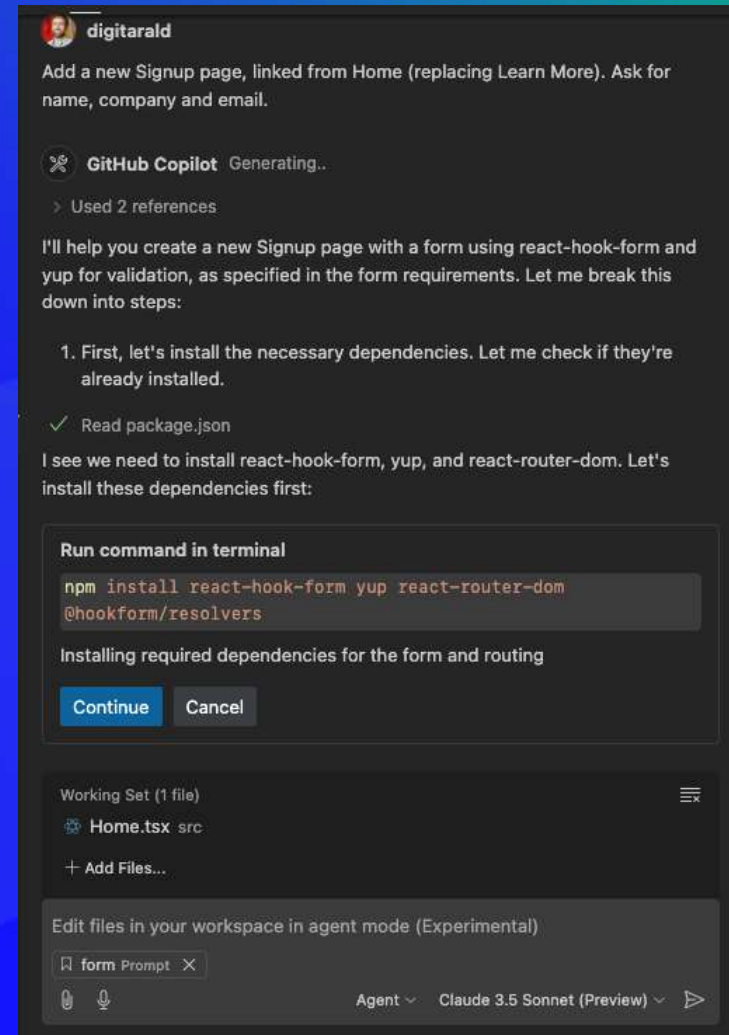
Tutorial: <https://aka.ms/letslearnmcp-csharp>

 Agent mode

# Agentic coding

Accelerate delivery by shifting from code to outcomes

Autonomous AI pair programmer that **plans, executes, and iterates** on multi-step coding tasks – from editor to terminal, tests to debugger.



But how do we provide real time context?

How do we connect to our database? Query GitHub Issues? Run tests with natural language?

How can we bridge the gaps between our code, the client and LLM, and everything else?

# The need for MCP arises – a Timeline

**Tools are integral to Agent performance**



**Integration chaos!!!**

**The need for a standard (MCP)**



**MCP is an open protocol that standardizes how applications provide context to LLMs.**

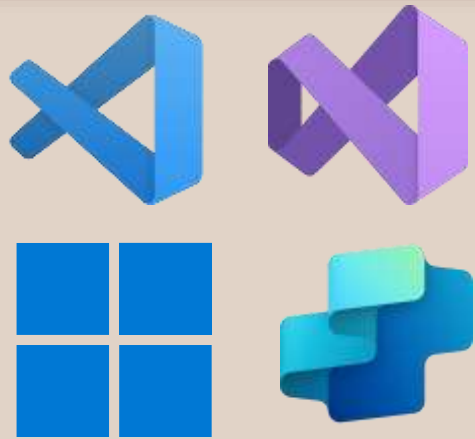
***It's like USB-C but for AI***

# Model Context Protocol

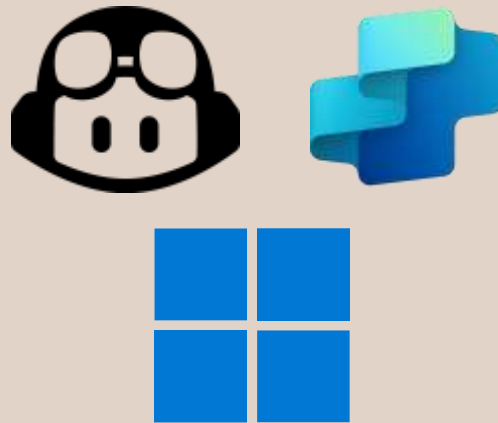
Created by Anthropic  
Adopted by the AI community  
Evolving Together

# Parts of MCP

## Hosts



## Clients



## Servers



# Demo

Setting up & using MCP servers  
[aka.ms/letslearnmcp-csharp](https://aka.ms/letslearnmcp-csharp)

# MCP Deep-Dive

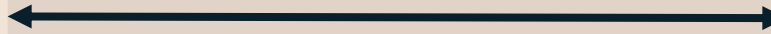
Adapted from Mahesh Murag (Anthropic) AI Engineer session

## MCP Client

- Invokes **Tools**
- Queries for **Resources**
- Interpolates **Prompts**

## MCP Server

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



## Tools

*Model-controlled*

Functions invoked by  
the model

Retrieve / search

Send a message

Update DB records

## Resources

*Application-controlled*

Data exposed to the  
application

Files

Database records

API Responses

## Prompts

*User-controlled*

Pre-defined templates  
for AI interactions

Document Q&A

Transcript Summary


Output as JSON

The popular basic primitive

# Tools

- Functions that LLMs can call to perform specific **actions** - basically function calling.
- Send email
- Update todos
- Run tests
- File issue bug

Run the website and take a screenshot

 **GitHub Copilot**

> Used 1 reference

I'll help you run the website and take a screenshot. Let me start the development server and then take a screenshot of the website.

✓ Started [npm: Next: Start Development Server](#)

Now I'll wait a moment for the server to start up, then navigate to the website and take a screenshot.

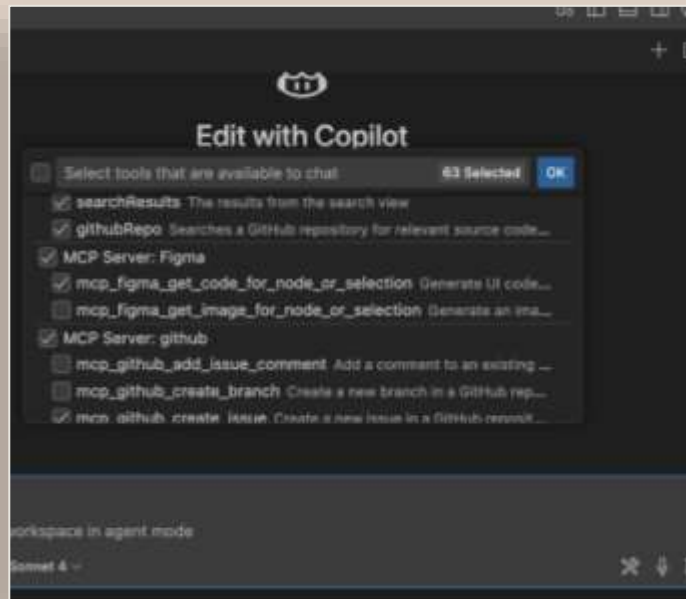
> Ran Navigate to a URL – Playwright (MCP Server) ✓

Perfect! The website is running. Now let me take a screenshot of the page.

> Ran Take a screenshot – Playwright (MCP Server) ✓

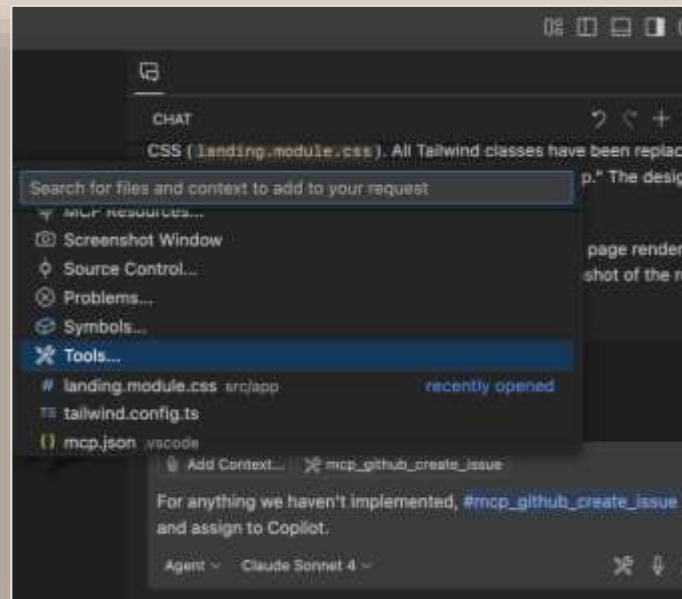
# Tools: User controls

Provide user controls to fine-tune and improve tool calls



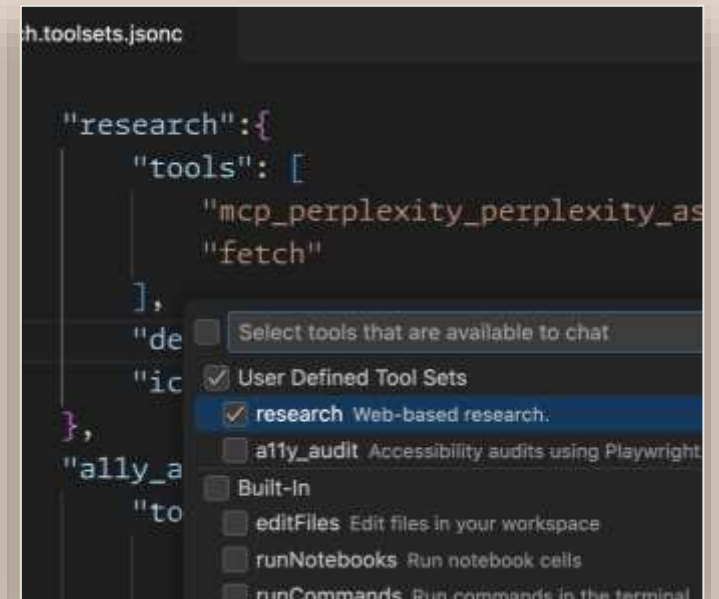
Per-chat tool selection

Manually reduce tools required to complete a specific task



Mention tools in prompts

Reduce the guessing and reference tools as needed.



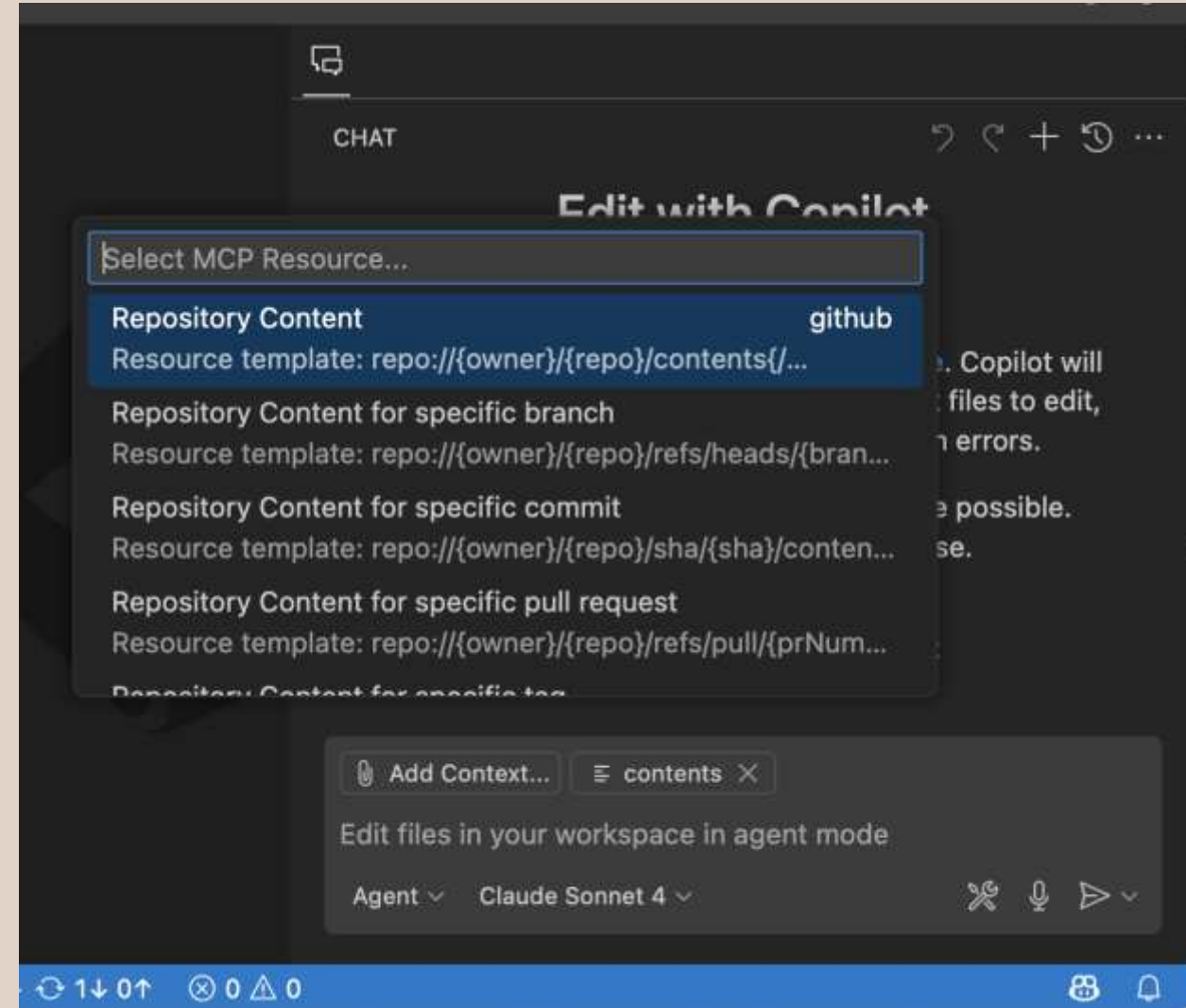
User-defined tool sets

Re-use task-specific tools across tasks

Ease of adding and returning data

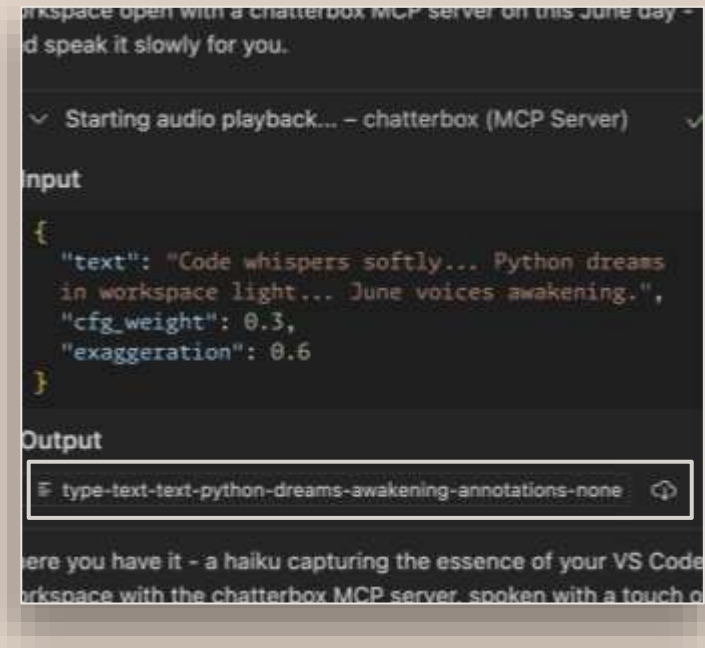
# Resources

- Expose data in a read-only fashion as URIs client/agent/user.
- Files
- Documents
- Database entries/schemas
- Images



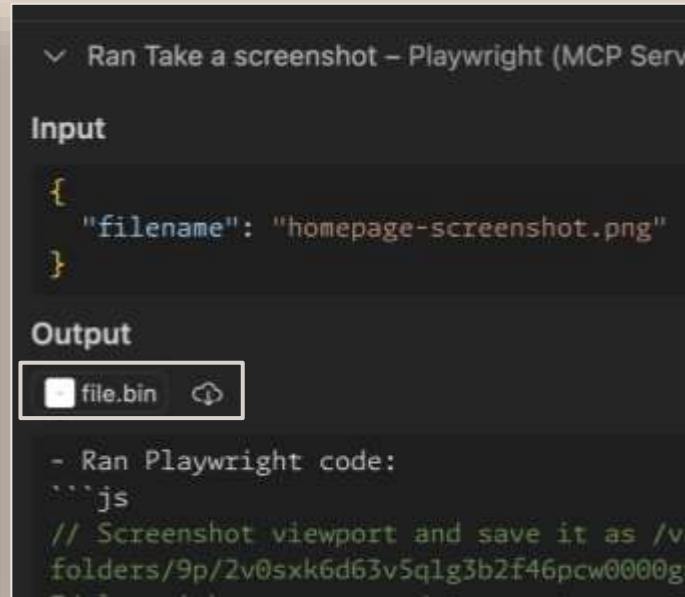
# Resources: Use cases

Make data accessible to clients and users



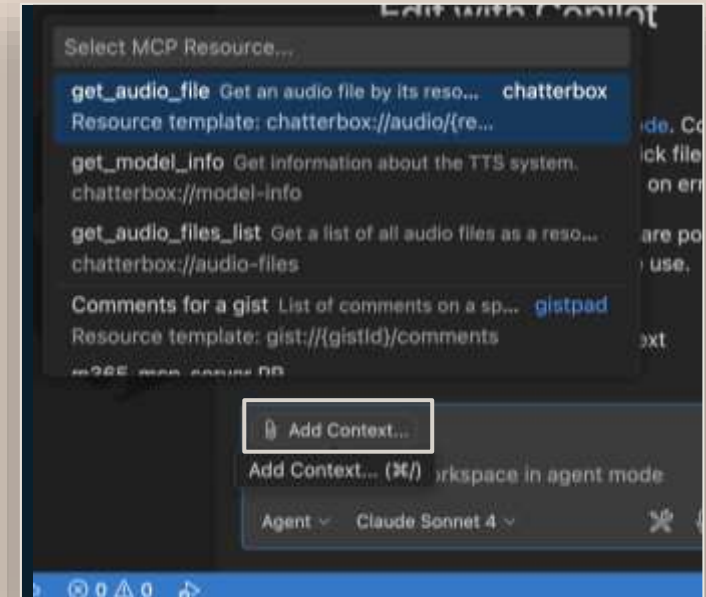
Reduce response tokens

Returning embedded resources lets the agent pull the data.



Expose data/files to user

Provide to assets for the user to act on, not LLM context.



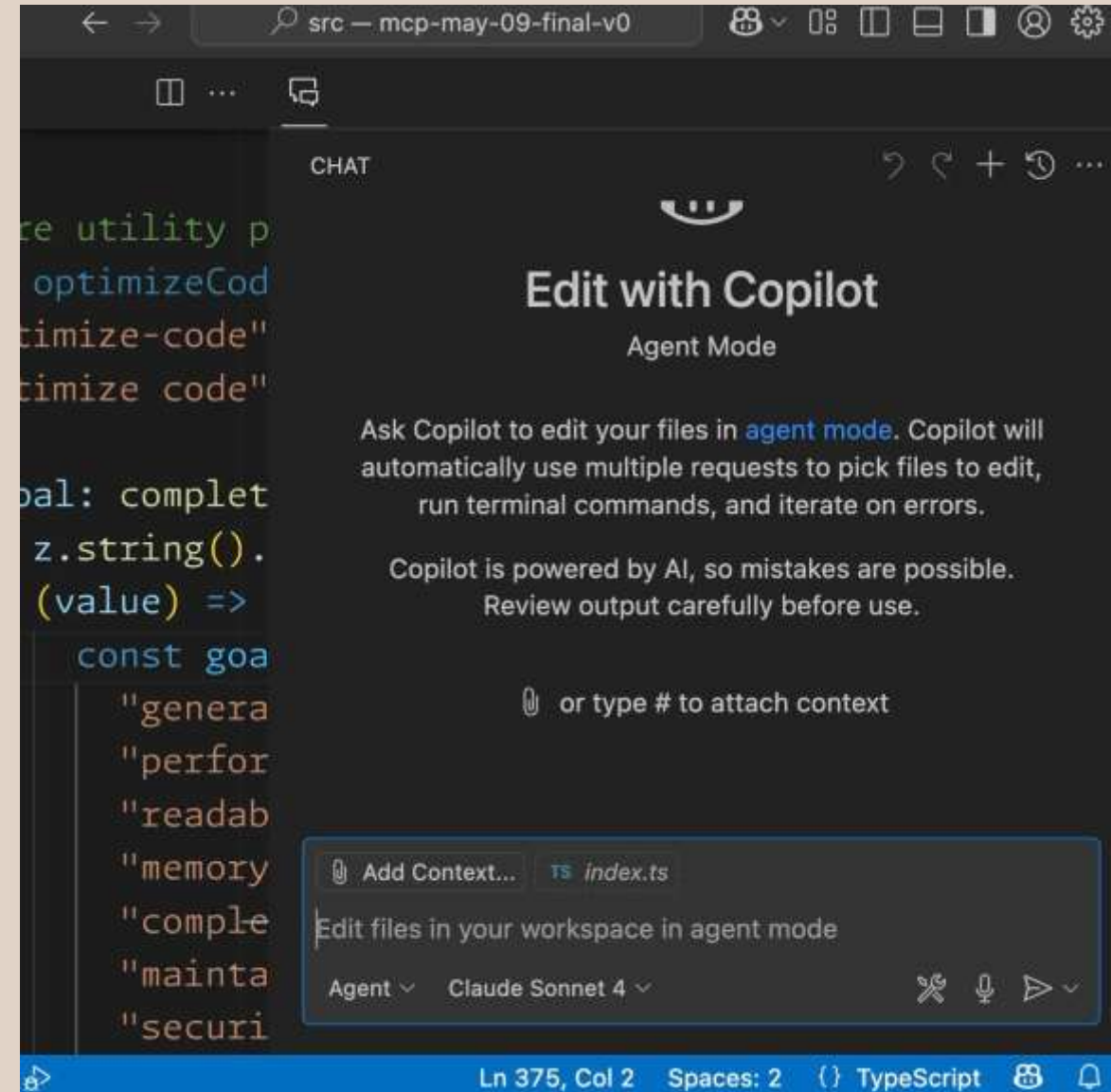
Attach as context

Reducing tool lookups with specific resources attached.

Quick

# Prompts

- Pre-defined templates for AI tasks, explicitly invoked by the user.
- Static presets
- Reusable with placeholders
- Dynamically generated



# Prompts: Use cases

From static templates to dynamically generates workflows

- `/welcome`

- `/setup`

## Onboarding prompts

Welcome prompts for users to verify setup and tour the tools

- `/need-info {issue}`

- `/impl-metric {name}`

## Common workflows

One-shot tasks, parameterized for reusability

- `/migrate-db ...`

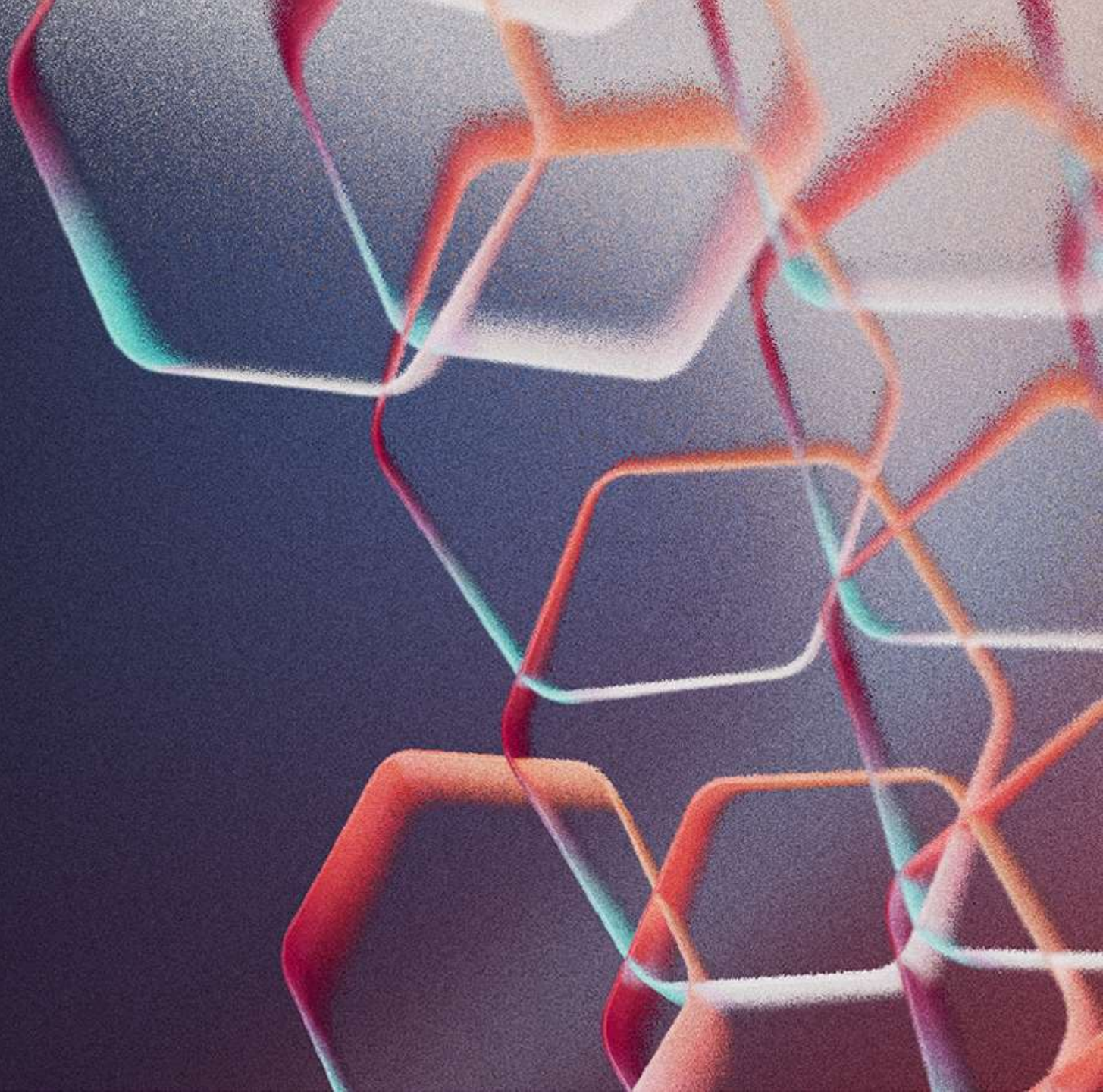
- `/root-cause-wizard`

## Context-aware workflows

Dynamically customized for the current user/codebase

# Demo

Prompts, resources, & more!



Your Computer

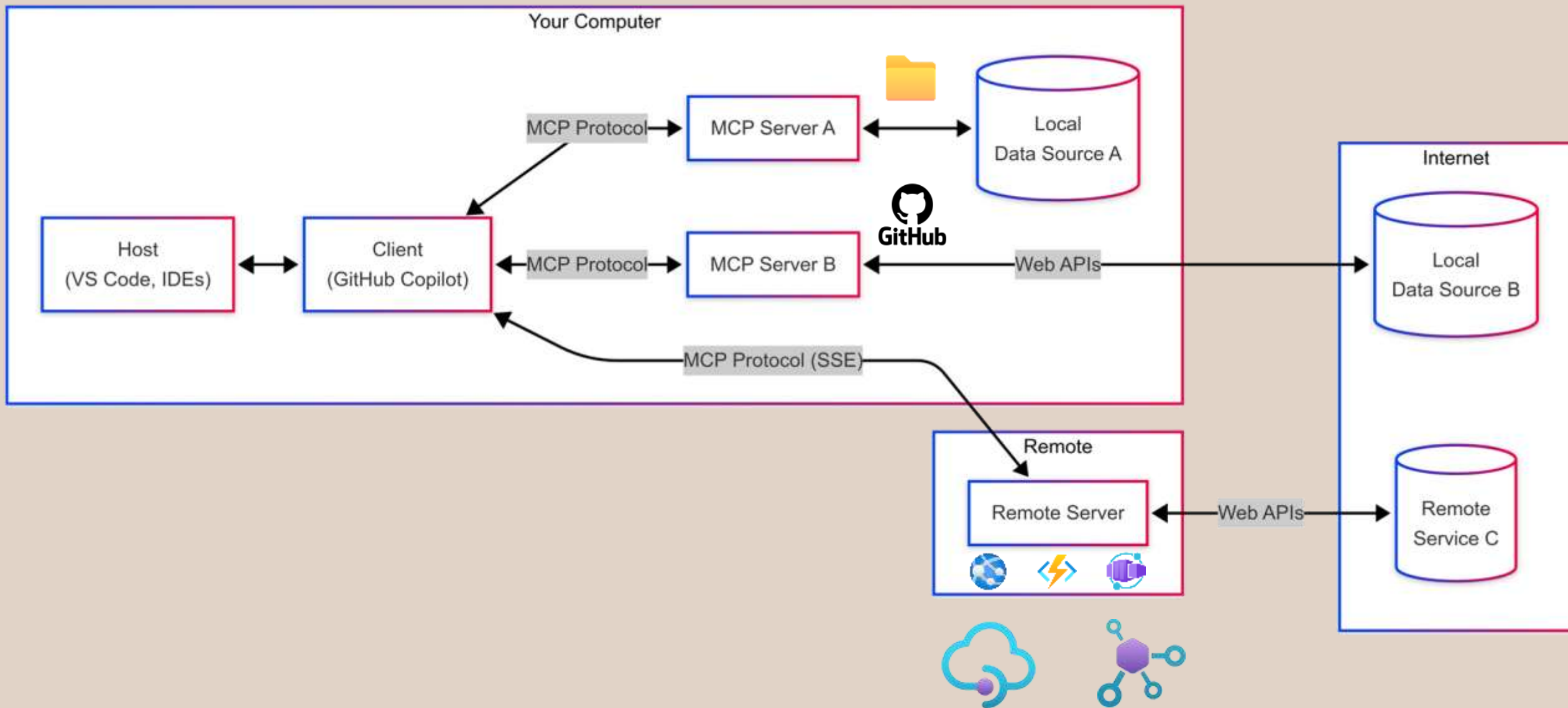
Host  
(VS Code, IDEs)



Client  
(GitHub Copilot)



Internet



# Building an MCP

[aka.ms/letslearnmcp-csharp](https://aka.ms/letslearnmcp-csharp)

# Official SDKs



# Official C# SDK for MCP

**Clients**

**Servers**

**Local – Stdio Transport**

**Remote – Server Sent Events**

**Remote – Streamable HTTP**

# NuGet Packages

**ModelContextProtocol.Core**

**ModelContextProtocol**

**ModelContextProtocol.AspNetCore**

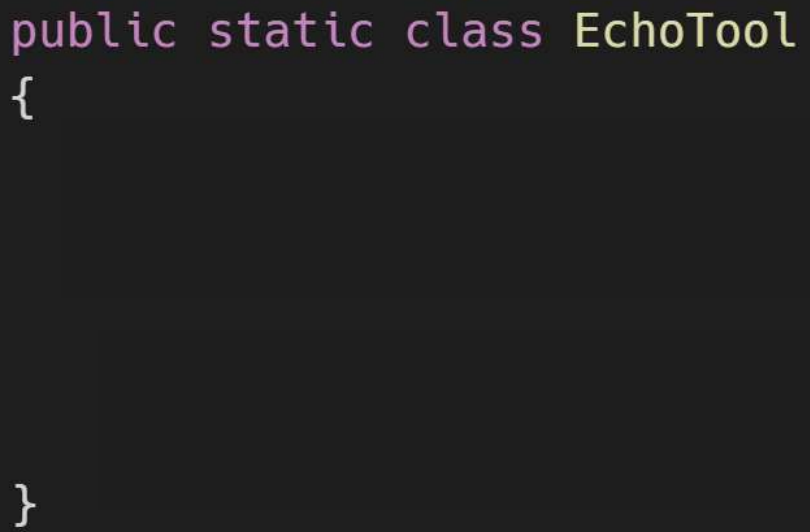
# Get Started



```
dotnet new console -n MyMCP
```

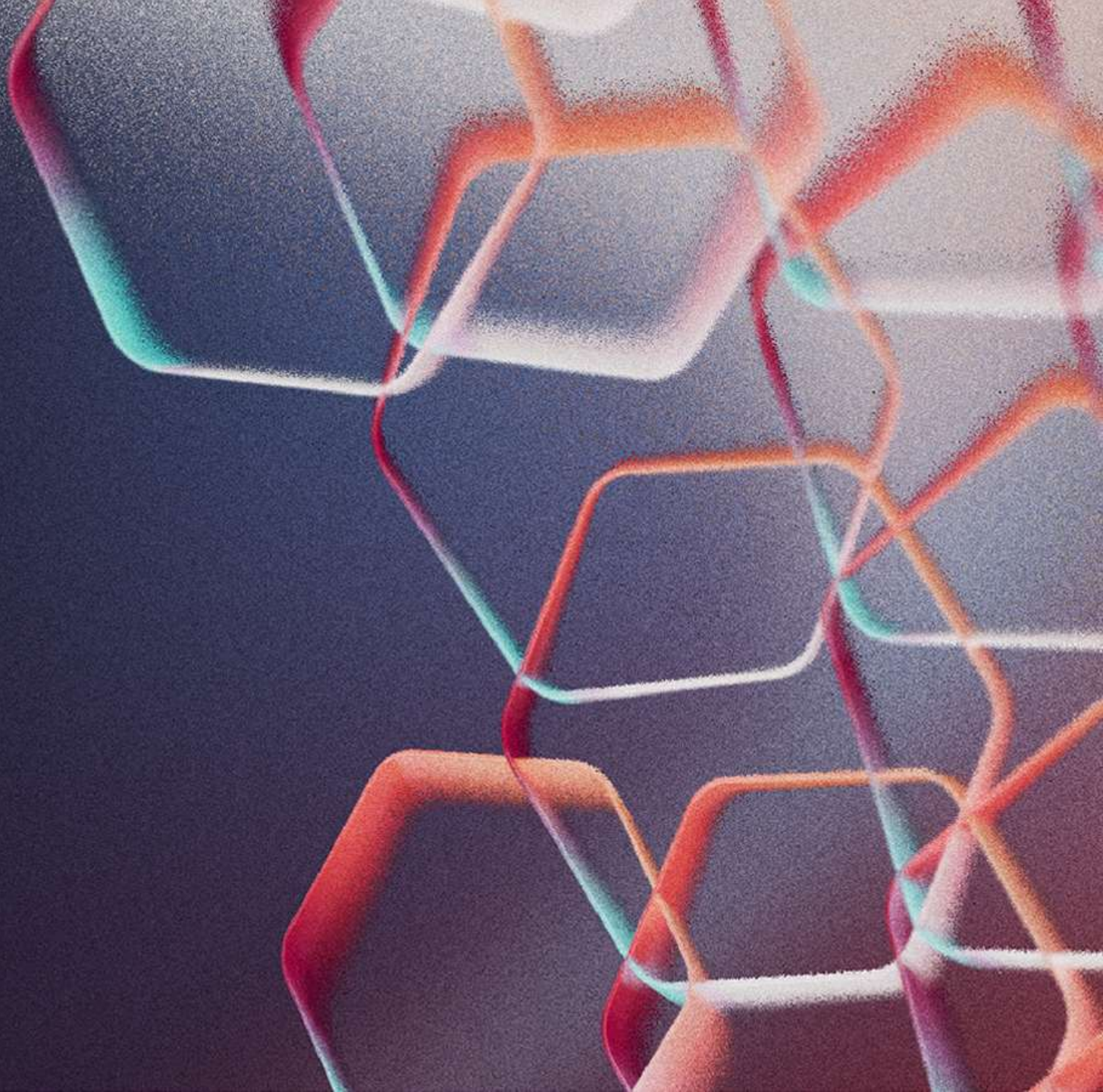


```
using Microsoft.Extensions.DependencyInjection;  
using Microsoft.Extensions.Hosting;  
using ModelContextProtocol.Server;  
using System.ComponentModel;
```

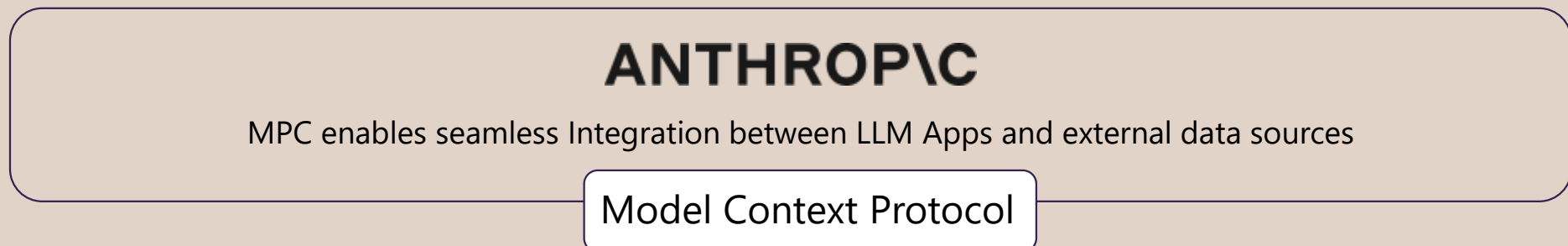
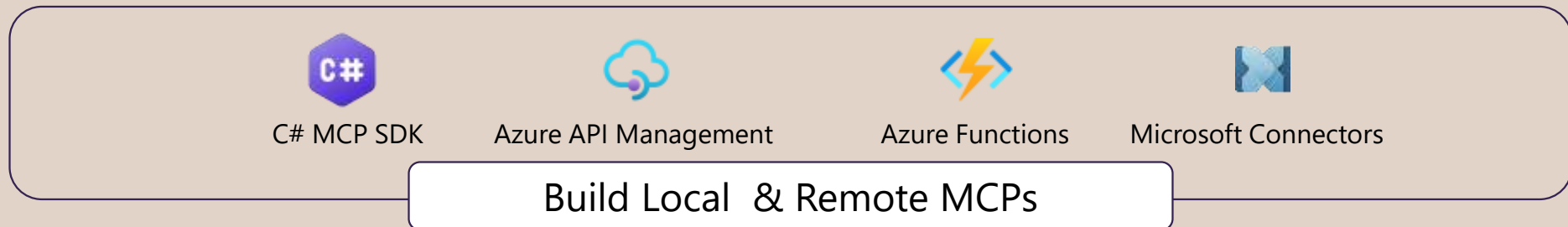
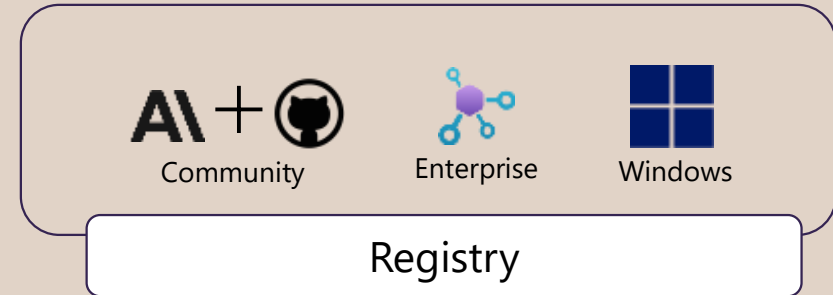
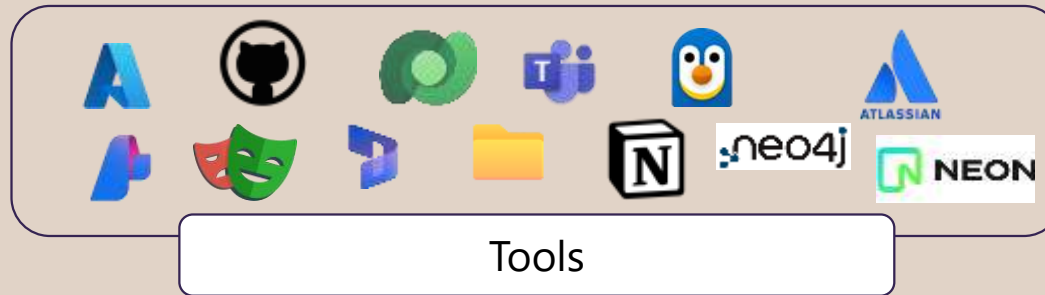


# Demo

Building an MCP Server



# Model Context Protocol At Microsoft





# Join us for MCP Dev Days

 [aka.ms/mcpdevdays](https://aka.ms/mcpdevdays)

July 29<sup>th</sup> & 30<sup>th</sup>

MCP Dev Days



Learn more

## Today's tutorial

[aka.ms/letslearnmcp-csharp](https://aka.ms/letslearnmcp-csharp)

## More training

[aka.ms/mcp-for-beginners](https://aka.ms/mcp-for-beginners)

## C# SDK for MCP

[aka.ms/csharp-sdk-mcp](https://aka.ms/csharp-sdk-mcp)

## VS Code MCP Support

[aka.ms/vscode-mcp](https://aka.ms/vscode-mcp)

## Model Context Protocol

[modelcontextprotocol.io](https://modelcontextprotocol.io)

## MCP Dev Days – July 29 & 30

Join us for MCP Dev Days, a two-day virtual event exploring the growing ecosystem around the Model Context Protocol (MCP), a standard that bridges AI models and the tools they rely on.

Register: <https://aka.ms/mcpdevdays>