Member-only story

Basics of MCPs. Why and what !

Understanding basics of MCPs and why they came in picture when we have such powerful LLMs.

5 min read · Oct 3, 2025

Chinmay Bhalerao Follow

Listen Share More

LLMs have limitations.
They **can** answer your questions, write procedures, provide step-by-step instructions, perform text understanding and generation, reasoning and problem-solving, and knowledge retrieval.

But LLMs **cannot** perform actions like executing code, sending emails, carrying out ordered ste

We often think LLMs can do a lot of things, but they are essentially advanced prediction models that are very good at generating text.

If you are familiar with Cursor, Augment, or GitHub Copilot, then you know they use LLMs to execute changes like reading, writing, or updating files. It’s interesting to see how these tools combine the prediction power of LLMs with actual execution of actions.

The idea is, I can make different tools for different functionalities. For example, like Cursor or Copilot, I can have a **read tool**, **write tool**, **update tool**, **database interaction tool**, **console executor**, or **Google search tool**.

But I also need to ensure that these tools follow some kind of protocol when being created.

You can think of it like REST API protocols. We use REST APIs in a standardized way to connect front end and back end. For LLMs, we currently don’t have a standardized protocol to connect inputs and outputs.

For example, APIs use standard status codes. Everyone knows and accepts worldwide what each status code means, so it’s easy to adapt and use. Similarly, for connecting or authentication, we have the right set of protocols that everyone follows.

In the same way, while connecting with LLMs or designing tools for LLMs, we also need to consider some kind of protocol.

For this purpose, Anthropic introduced an interesting idea called **Model Context Protocols (MCP).**

A **protocol** that defines how AI models can **communicate with external contexts** (like APIs, databases, code repos, or plugins). It helps AI systems fetch the **right information at the right time** instead of relying only on their pre-training.

External data sources → e.g., a company's database, APIs, documents, logs.

Tools and services → e.g., Jira, GitHub, Slack, SQL, cloud APIs.

Local environment → e.g., files on your machine, configs, code repos.

This **context** is *not* part of the model's training weights — it's something the model can access **dynamically at runtime** via MCP.

> *Why context is important ?*

There are thousands of repositories, but the LLM needs to know **which repo to take and work with**. We don't want generalized solutions — we want solutions related to our specific context and problems.

So **context** can be tools, environments, prompts, or resources.

> *What is a Protocol?*

A **protocol** in computing is a set of rules and standards that define how two systems communicate with each other.

It’s similar to the requests we make to connect with APIs.

MCP client a

> *MCP Server*

The component that **provides context** (e.g., a data source, tool, or service). It *serves* the AI with information.

Examples:

> *GitHub MCP server → provides repository info.*
>
> *SQL MCP server → provides query results.*
>
> *Jira MCP server → provides ticket data.*

Welcome back. You are signed into your member account **bg••••@jaxondigital.com**.

MCP Client

The component that **consumes context** (usually the AI model or the app hosting it).It requests in wledge or actions.

Examples:

> *Claude (or GPT) acting as an MCP client → asks the GitHub MCP server:* “Give me open PRs.”

Your AI-powered app using MCP → connects to a SQL MCP server for data.

> Client = AI asking for context
> Server = Tool or data source providing context

What an MCP Server Contains

An MCP server is essentially a **wrapper around your data or tools,** so the AI client can use them safely.

It usually contains:

> ***APIs / Data Sources*** *→ The actual source of info (databases, APIs, files, logs).*
>
> *Example: Jira API, SQL database, file system.*
>
> ***Tools (Actions)*** *→ Small functions or commands the LLM can call.*
>
> *Example:* `list_issues` , `get_pull_requests` , `query_database` .
>
> ***Schema / Definitions*** *→ A structured description of what the server can do.*
>
> ***Security Rules*** *→ Define what the AI is allowed to do safely.*

Analogy

[This analogy I have take from ChatGPT to explain things in better way, Thank you ChatGPT]

Think of an MCP server like a **vending machine**:

> *Inside it has snacks (APIs/data).*
>
> *The buttons are the tools (actions).*

> *The labels are the schema (how to use them)*
>
> *The coin sl*
>
> *The client (AI) is just pressing buttons to get the right snack.*

Example: GitHub MCP Server

Contains: GitHub API connection.

Tools: `list_repos`, `get_pull_requests`, `create_issue`.

Schema: JSON definition of inputs needed.

Security: Allows reading repos, but not deleting.

So when the LLM (client) asks:

> *"Find all open PRs in repo XYZ."*

The MCP server:

1. Runs the GitHub API call.
2. Returns results in structured format.

> MCP Server = APIs + tools + schema + security rules
> Client = AI app that requests data
> Protocol = the rules for how they talk

If you find this article useful

It is a proven fact that "**Generosity makes you a happier person**"; therefore, give claps to the article if you liked it. If you found this article insightful, follow me on **LinkedIn** and **Medium**. You can also **subscribe** to get notified when I publish articles. Let's create a community! Thanks for your support!

If you have found this article insightful and want to join the conversation on building reliable AI, then follow me. I share pragmatic insights for builders in the AI space. You can also subscribe to get my articles delivered directly to your inbox.

Here are a few other posts you might find useful:

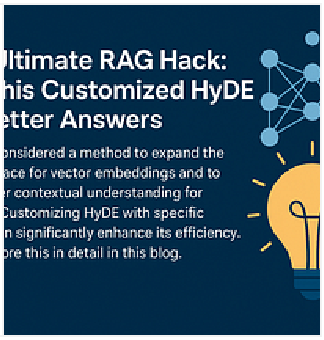
Why Grap

Limitations
Graph stru

medium.com

The Ultimate RAG hack: Use this customized HyDE for better answers

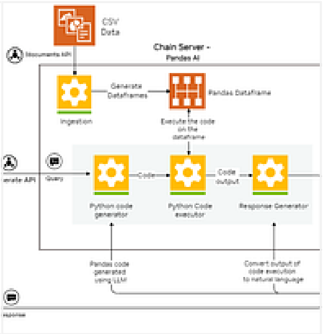
medium.com



RAG is Not Enough: Why Your Next AI Project Demands Structured Data RAG

The FAST-RAG system without complex embedding models or vector databases

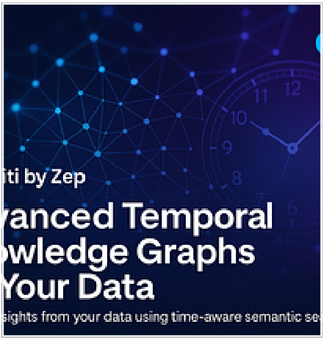
medium.com



Graphiti by Zep: Advanced Temporal Knowledge Graphs for Your Data

A Framework to Build, Query, Time-Aware Knowledge Graphs that work on episodic inputs

medium.com



Signing off,

Chinmay !

Large Language Models

Model Context Protocol

Artificial Intelligence

Welcome back. You are signed into your member account **bg••••@jaxondigital.com**.

llow

Written by Chinmay Bhalerao

1.8K followers · 124 following

Senior AI Engineer | 3X Top Writer in AI, Computer Vision & Object Detection | Generative AI, RAG & Fine-Tuning | Making AI work for everyone, not just experts.

Responses (3)

Bgerby

What are your thoughts?

Rpmk
4 days ago

informational and simply explained.

5 1 reply Reply

Prakash Gupta
Oct 10

Nice to know this topic's philosophy !

5 1 reply Reply

mohamad shahkhajeh
Oct 5

It's useful that MCPs define a standard protocol for LLMs to interact with external APIs and databases reliably.

5 1 reply Reply

Welcome back. You are signed into your member account **bg••••@jaxondigital.com**.

More from Chinmay Bhalerao

In Data And Beyond by Chinmay Bhalerao

RAG is Not Enough: Why Your Next AI Project Demands Structured Data RAG

The FAST-RAG system without complex embedding models or vector databases

✦ Jul 9 787 13 •••

In Data And Beyond by Chinmay Bhalerao

Why Graph RAG Matters? All about Graph RAG

Limitations of production level vanilla RAG systems, emergence of Graph structures and linking RAG system with graph databases

✦ Sep 23 👏 258 💬 4

Welcome back. You are signed into your member account **bg••••@jaxondigital.com**.

In Data And Beyond by Chinmay Bhalerao

“Agentic AI”: The Buzzword You Actually Need to Understand

✦ Oct 10 👏 123

In Data And Beyond by Chinmay Bhalerao

Graphiti by Zep: Advanced Temporal Knowledge Graphs for Your Data

A Framework to Build, Query, Time-Aware Knowledge Graphs that work on episodic inputs

May 14 361 6

See all from Chinmay Bhalerao

Recommended from Medium

Welcome back. You are signed into your member account **bg••••@jaxondigital.com**.

aakash

Perplexity Just Unleashed 10 FREE AI Agents That Do Your Entire Job (The “Comet” Shortcut)

Stop what you are doing. The era of the simple AI search engine is officially over.

✦ Oct 7 662 19

Riccardo Tartaglia

5 Essential MCP Servers Every Developer Should Know

I’ve been experimenting with Model Context Protocol servers for a few months now, and I have to say, they’ve changed the way I work.

Oct 11 22

Welcome back. You are signed into your member account **bg••••@jaxondigital.com**.

Alex Suzuki

Cretaceous Software Engineering — Why I don’t use AI to write code.

👋 Hi! I’m a Cretaceous Software Engineer.

5d ago 641 28

In Netflix TechBlog by Netflix Technology Blog

How and Why Netflix Built a Real-Time Distributed Graph: Part 1— Ingesting

Authors: Ad

Welcome back. You are signed into your member account **bg••••@jaxondigital.com**.

4d ago 👏 528 💬 12

In Coding Nexus by Civil Learning

MarkItDown: Convert Anything into Markdown — the Smart Way to Feed LLMs

You know that feeling when you're trying to feed a PDF or a Word document into an LLM, and it just doesn't understand what's going on...

✦ 6d ago 👏 125 💬 1

Welcome back. You are signed into your member account **bg••••@jaxondigital.com**.

Neal Davis

Cloud Skills that will soon be obsolete — and what to learn instead

If you’re learning cloud right now — pay attention. Some of the most common skills people still spend hours practicing are already...

Oct 13 👏 17 💬 2

See more recommendations