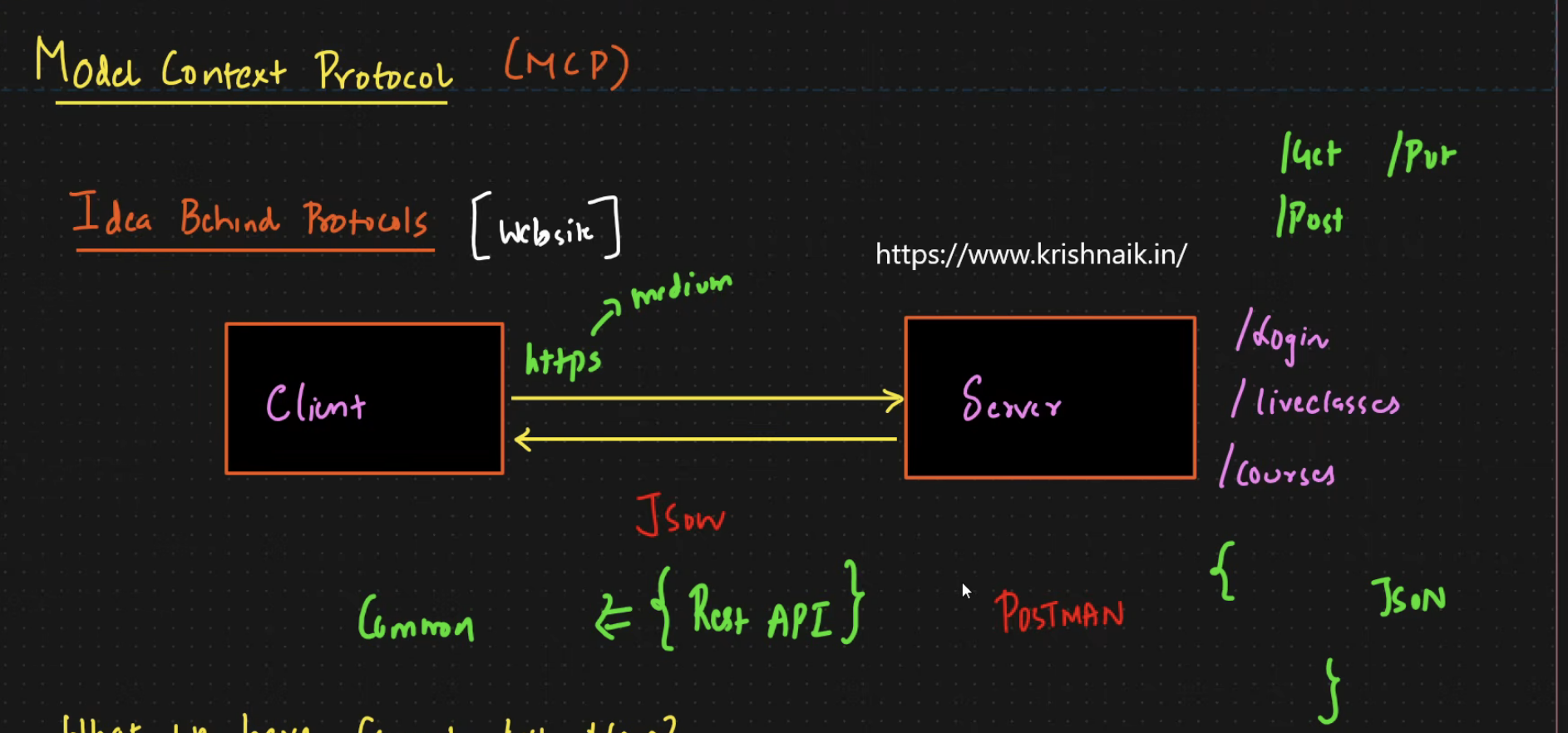
MCP:

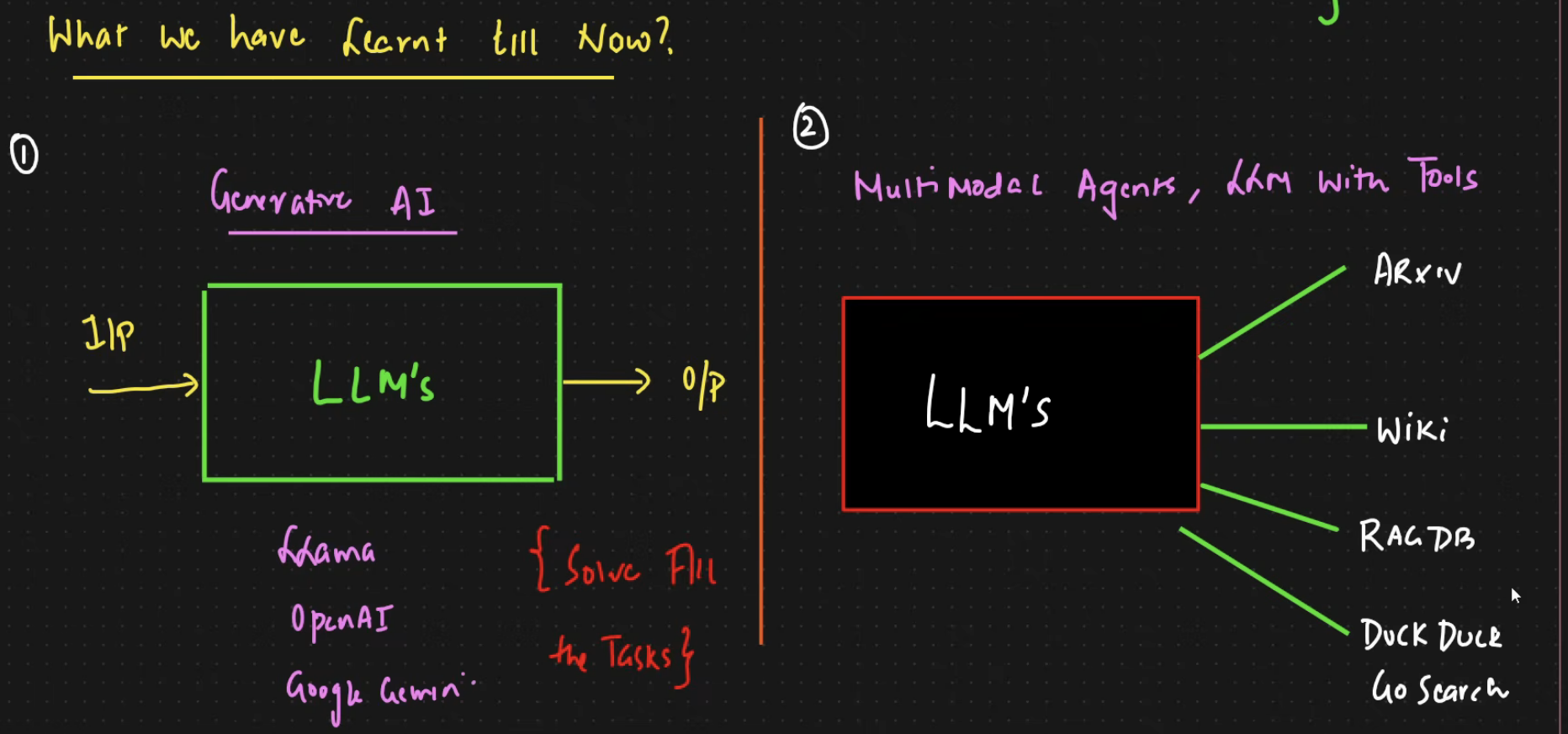
A close up of a text

AI-generated content may be incorrect.

In traditional applications we use REST API to communicate with application

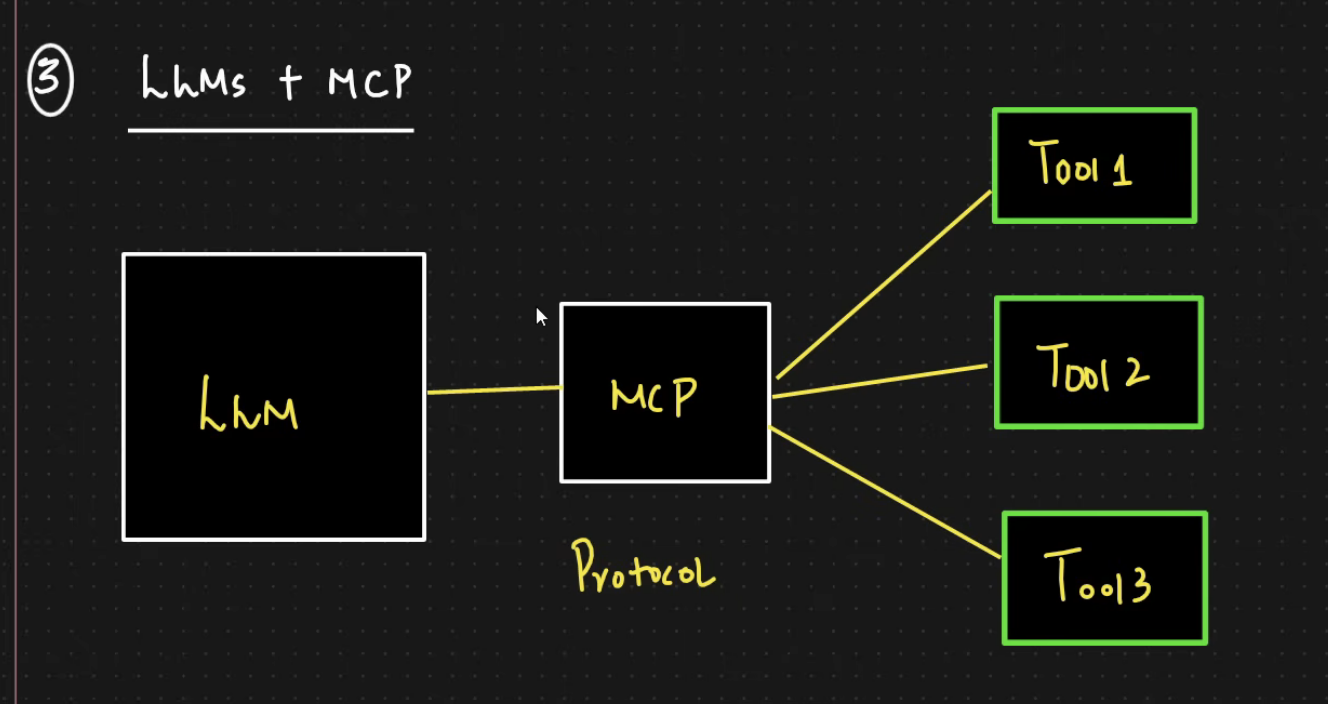


IN LLM we use multiple tools to do some specific task, suppose If I want to read some research paper then I will use ARXIN tool



Main challenge is if I want to integrate 100 different tools then It becomes difficult to manage it and suppose if underlying tool changes then it becomes difficult to manage

We can’t scale with multiple tools

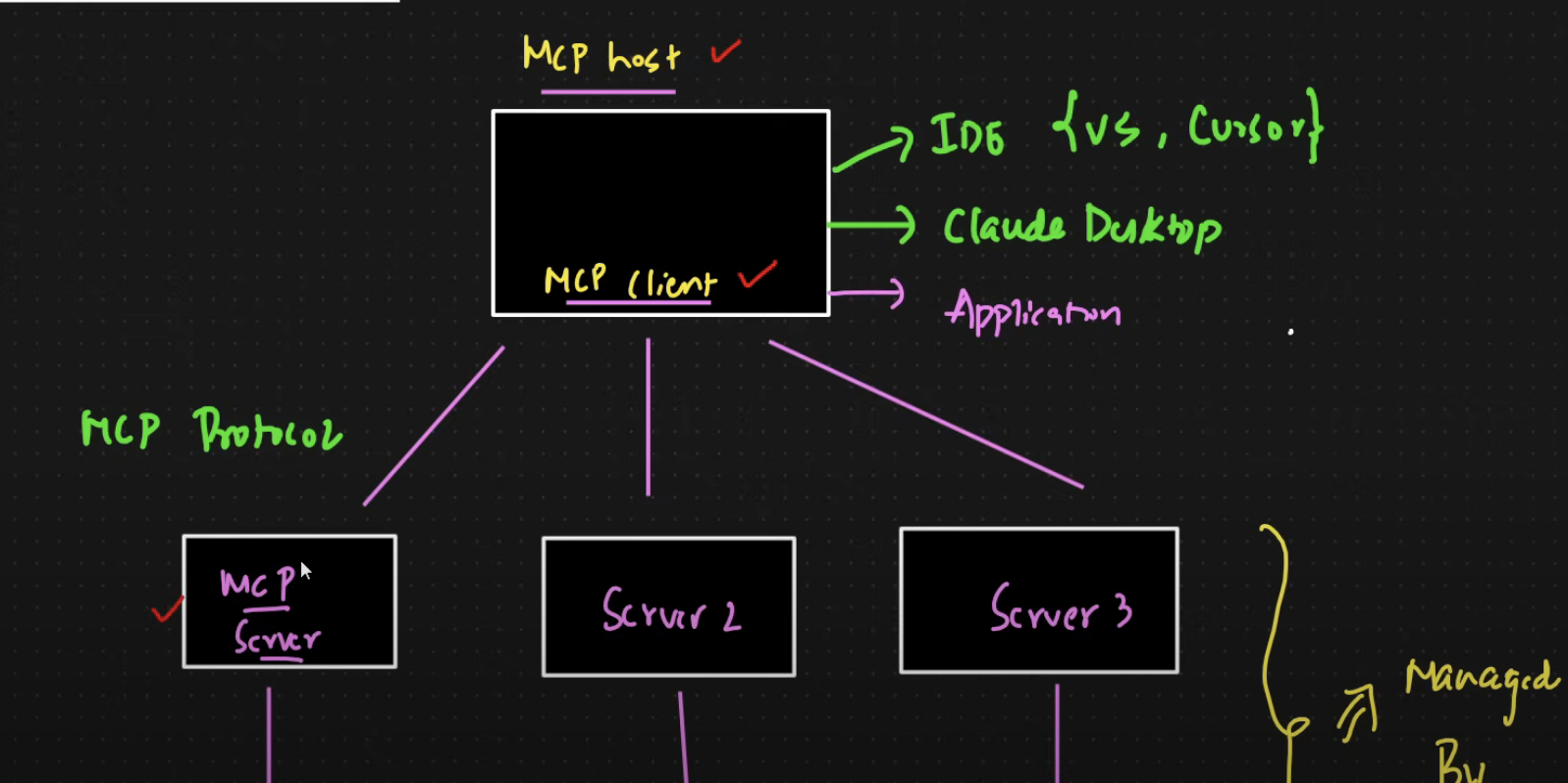


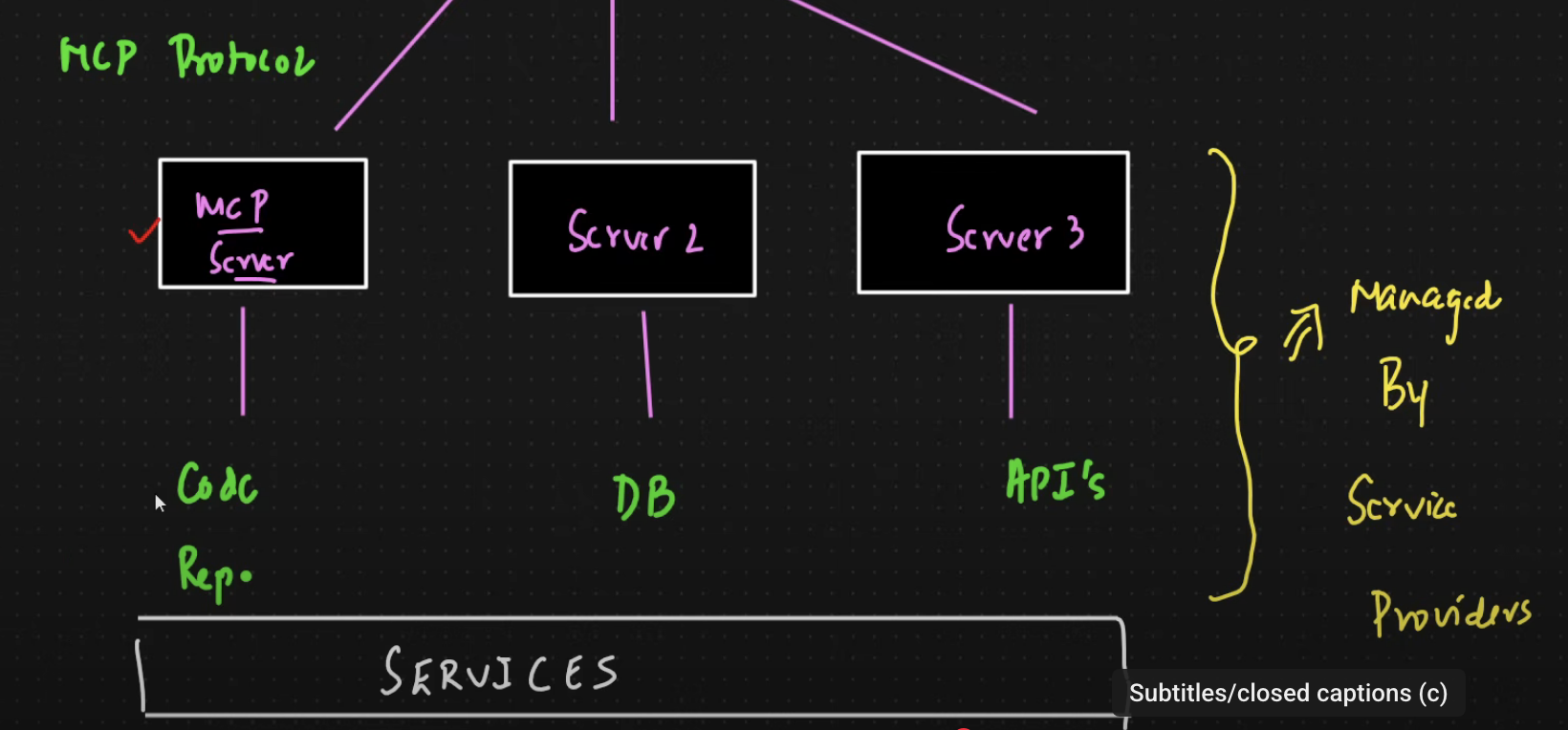
To handle that we will use MCP where tools needs to adhere to MCP standards with that we can add any number of tools

Components in MCP:

MCP Host- It can be IDE like vs code / claude desktop

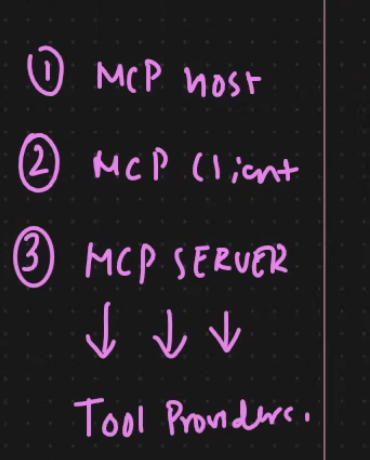
This MCP host can create a client to interact with MCP server



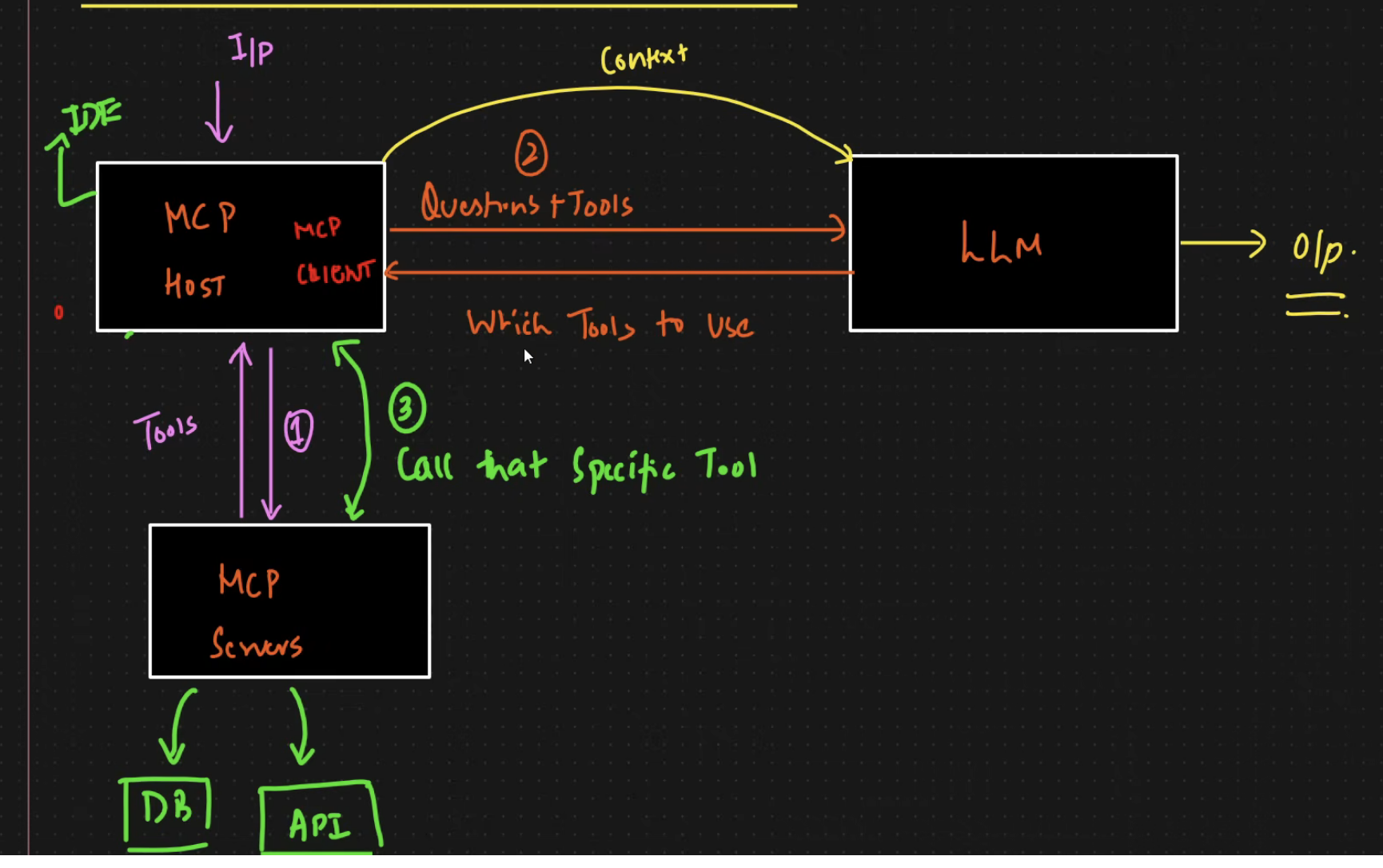


MCP client is communicating with MCP server using MCP protocol

MCP Server and underlying tools will be completely managed by tool provider



Communication Between Components:

* When I give input in IDE, it will hive MCP server which returns what all tools are available
* Now MCP host will hit LLM with tool info & input prompt
* 
* LLM will tell which all tools to use then it will go back to MCP server with specific tool and then entire context will be given back to LLM and LLM will give output