

MasterGo MCP 教程

什么是 MCP

背景与概述

Model Context Protocol (MCP) 是由 Anthropic (Claude 的开发公司) 在 2023 年 11 月底推出的一个开放标准协议，旨在为 AI 模型与外部数据源和工具之间建立标准化的连接方式。MCP 的出现标志着 AI 应用开发进入了一个新的时代。

MCP 可以被理解为 AI 世界的"USB-C 接口"—它提供了一个通用的标准，让 AI 模型能够轻松连接到各种数据源和工具，就像我们可以通过 USB-C 接口连接各种外设一样。这种标准化的方式大大简化了 AI 与外部世界的交互过程。

在 MCP 出现之前，AI 应用开发主要集中在三个方面：Prompt（提示工程）、RAG（检索增强生成）和 Agent（智能代理）。而 MCP 的出现为这些技术提供了更加标准化和强大的实现方式，特别是在 AI 代理与外部世界交互方面。

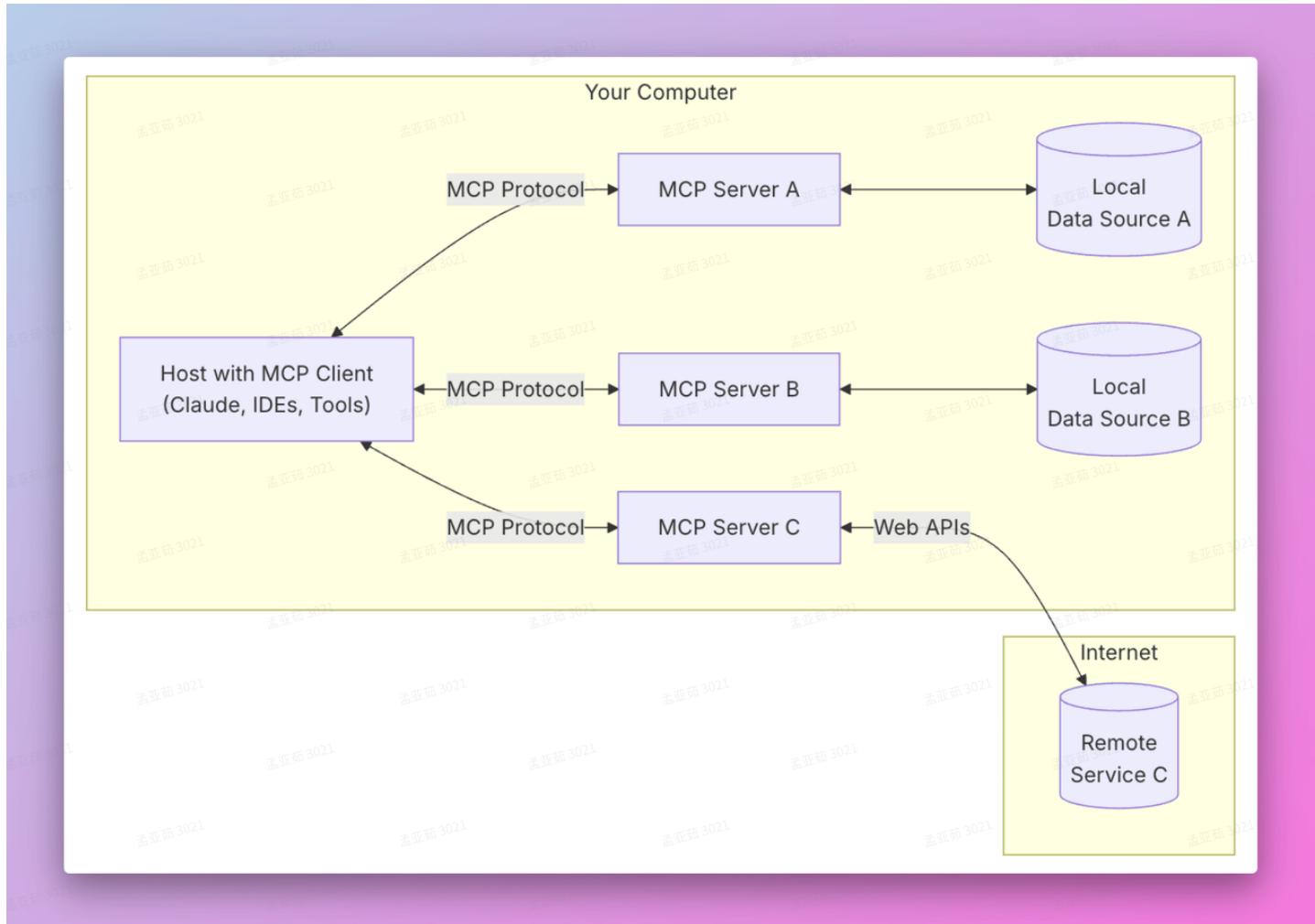
MCP 解决的问题

MCP 主要解决了以下几个关键问题：

- 数据访问的标准化：**在 MCP 之前，AI 模型访问外部数据通常需要复制粘贴、上传下载或使用特定的 API。MCP 提供了一个标准化的协议，使 AI 能够直接访问各种数据源，包括本地文件、Google Drive 文档、Slack 消息、数据库记录等。
- 工具使用的统一接口：**MCP 为 AI 提供了一个统一的接口来使用各种工具，无需为每个工具开发特定的集成方案。
- 跨平台兼容性：**作为一个开放标准，MCP 可以在不同的平台和应用程序之间工作，促进了生态系统的发展。
- 实时交互能力：**MCP 允许 AI 实时操作浏览器、文件系统和其他应用程序，大大增强了 AI 的交互能力。
- 安全性和隐私保护：**MCP 设计考虑了安全性和隐私保护，确保数据在 AI 和外部系统之间的安全传输。

通过解决这些问题，MCP 使 AI 能够更加无缝地融入我们的工作流程，提高了 AI 辅助工作的效率和可能性。

MCP 架构详解



MCP 采用客户端-服务器架构，主要包含以下几个核心组件：

1. MCP Host (MCP 主机)

MCP Host 是像 Claude 桌面端、集成开发环境（如 Cursor）或其他通过 MCP 协议访问资源的 AI 工具。它同时负责管理 MCP 服务器和提供 MCP 客户端功能。

2. MCP Client (MCP 客户端)

MCP Client 是用户交互的工具，它通过 MCP 协议与 MCP 服务器维持一对一的连接。客户端负责将用户的请求转发给服务器，并将服务器的响应呈现给用户。

3. MCP Server (MCP 服务器)

MCP Server 是提供特定功能的轻量级程序，它通过标准化的 Model Context Protocol 向客户端暴露各种能力。每个 MCP 服务器可以专注于提供特定类型的集成或功能，例如：

- 连接内部数据库的服务器
- 通过 API 与外部服务交互的服务器
- 访问本地文件的服务器

4. 数据源

MCP 架构中的数据源分为两类：

- **本地数据源**: MCP 服务器可以安全访问的计算机上的文件、数据库和服务
- **远程服务**: MCP 服务器可以通过互联网连接的外部系统（如通过 API）

通信流程

MCP 的通信流程如下：

1. MCP Host 通过 MCP 协议连接到一个或多个 MCP Server
2. 每个 MCP Server 可以访问本地或远程的资源
3. MCP Client 通过 MCP 协议与 MCP Server 进行双向通信
4. MCP Server 从本地或远程资源中获取数据，并将其返回给 MCP Client

这种架构形成了一个中心化的"枢纽"，每个服务器以标准化的方式暴露其功能。AI 代理或 LLM 可以轻松地在不同的数据源、工具和服务之间切换，而无需为每个新需求重写集成代码。

MasterGo MCP

MasterGo 本次提供的 MCP 就是作为 MCP server 的形式出现的，MasterGO MCP Server 目前只能获取图层的 DSL 数据，并不能通过 MCP 去修改画布数据。

希望使用 MCP 的用户，只需要使用支持 MCP 协议的 Host（比如 Cursor, Cline 插件），就可以在对应的 Host 上运行 MasterGo MCP，获取到图层的 DSL 了。

获取到图层的 DSL 之后，可以让大模型转成 HTML / Vue / React 等等形式，具体能用来做什么并没有限制。

搭建 MCP 环境

MCP 的运行需要基本条件

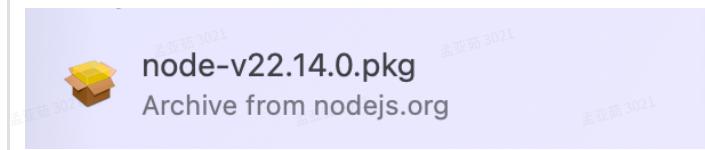
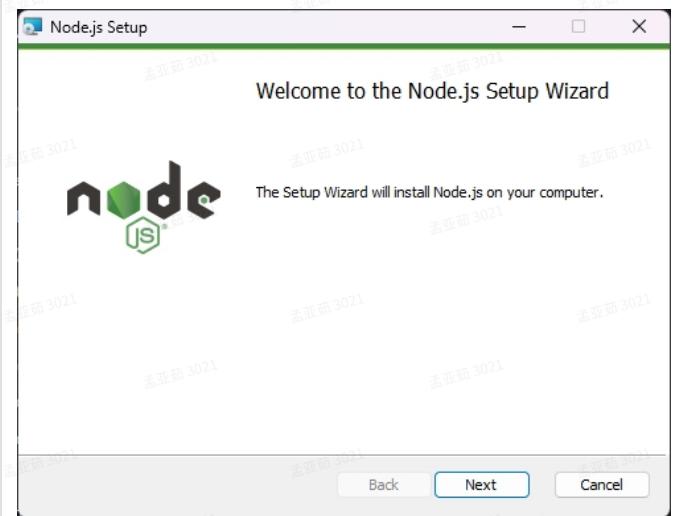
1. Node.js 运行环境（MCP server 需要这个环境才能启动）
2. MCP Host（需要有软件使用 MCP server）

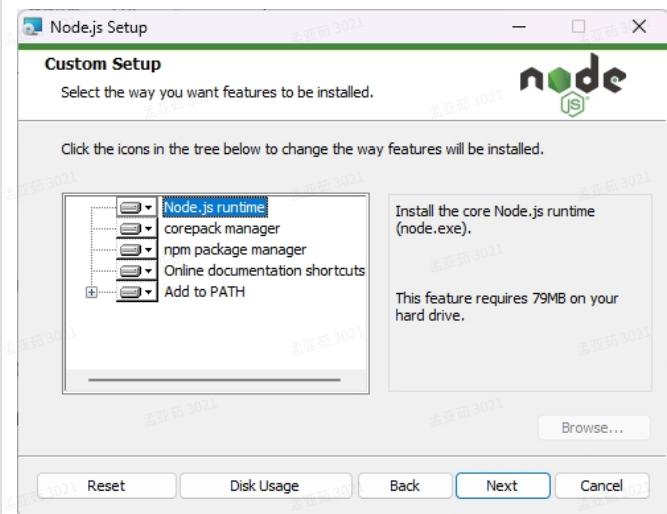
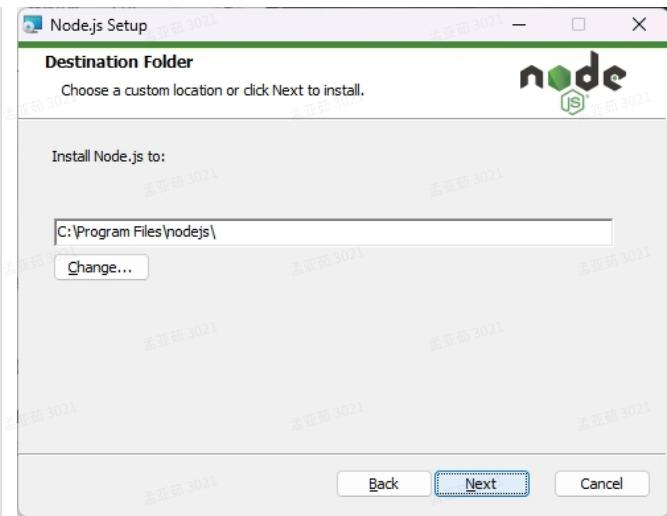
Hosts 我们默认使用 Cursor 和 Cline 做演示。

安装 Node.js

MasterGo MCP 的运行需要依赖 Node.js 环境，因此在使用 MCP 前需要在电脑上安装 Node.js 环境。

步骤

Mac 系统	Windows 系统
官网安装	官网安装
访问 https://nodejs.org/zh-cn	访问 https://nodejs.org/zh-cn
	
下载安装包	下载（基本上一路同意过去就行了，不需要关心具体的配置路径）
	
然后按照提示执行下去	



然后在终端中验证一下 node 是否下载成功，复制以下代码

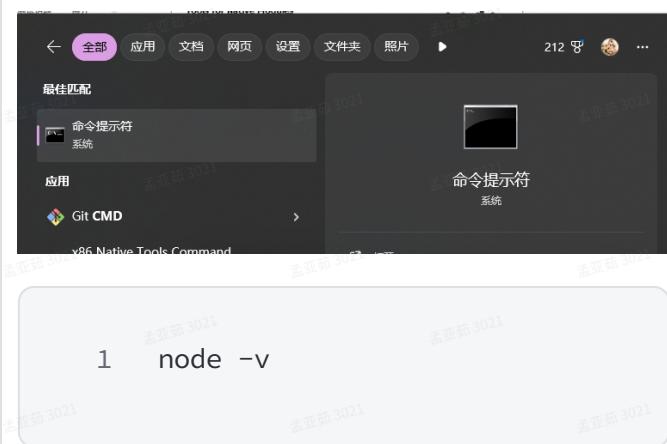
Terminal



```
1 node -v
```

执行之后有 v22.14.0 就行

然后在 CMD 中检验一下是否安装成功



执行之后的结果

```
Last login: Sun Mar 23 11:53:13 on console
~ % node -v
v18.20.4
~ %
```

其实只要版本号大于 18 即可。不一定需要 22 版本。



```
命令提示符
Microsoft Windows [版本 10.0.22631.5039]
(c) Microsoft Corporation, 保留所有权利。
C:\Users\77284%node -v
v22.14.0
C:\Users\77284%
```

其实只要版本号大于 18 即可。不一定需要 22 版本。

安装 Node 的常见问题

安装太慢

网络原因，解决方法有两种：

- 需要开代理去访问（推荐，因为很多环境问题都是因为网络访问不通导致的）
- 使用镜像源（部署在国内的资源）

使用镜像源安装

- 访问：<https://mirrors.aliyun.com/nodejs-release/v22.14.0/>
- 点击 node-v22.14.0 下载

node-v22.14.0-arm64.msi	26.1 MB	2025-02-11 09:23
node-v22.14.0-darwin-arm64.tar.gz	44.9 MB	2025-02-11 08:45
node-v22.14.0-darwin-arm64.tar.xz	23.3 MB	2025-02-11 08:46
node-v22.14.0-darwin-x64.tar.gz	46.4 MB	2025-02-11 11:13
node-v22.14.0-darwin-x64.tar.xz	25.1 MB	2025-02-11 11:14
node-v22.14.0-headers.tar.gz	8.4 MB	2025-02-11 13:56
node-v22.14.0-headers.tar.xz	525.7 KB	2025-02-11 13:57
node-v22.14.0-linux-arm64.tar.gz	51.1 MB	2025-02-11 08:37
node-v22.14.0-linux-arm64.tar.xz	27.3 MB	2025-02-11 08:39
node-v22.14.0-linux-armv7l.tar.gz	46.8 MB	2025-02-11 08:08
node-v22.14.0-linux-armv7l.tar.xz	24.2 MB	2025-02-11 08:11
node-v22.14.0-linux-ppc64le.tar.gz	54.2 MB	2025-02-11 08:09
node-v22.14.0-linux-ppc64le.tar.xz	28.9 MB	2025-02-11 08:12
node-v22.14.0-linux-s390x.tar.gz	51.8 MB	2025-02-11 08:18
node-v22.14.0-linux-s390x.tar.xz	27.1 MB	2025-02-11 08:21
node-v22.14.0-linux-x64.tar.gz	51.6 MB	2025-02-11 08:08
node-v22.14.0-linux-x64.tar.xz	28.5 MB	2025-02-11 08:12
node-v22.14.0-win-arm64.7z	18.8 MB	2025-02-11 09:23
node-v22.14.0-win-arm64.zip	29.5 MB	2025-02-11 09:23
node-v22.14.0-win-x64.7z	20.8 MB	2025-02-11 08:57
node-v22.14.0-win-x64.zip	33.3 MB	2025-02-11 08:57
node-v22.14.0-win-x86.7z	19.1 MB	2025-02-11 08:47
node-v22.14.0-win-x86.zip	30.8 MB	2025-02-11 08:47
node-v22.14.0-x64.msi	26.9 MB	2025-02-11 08:47
node-v22.14.0-x86.msi	80.2 MB	2025-02-11 12:34
node-v22.14.0.pkg	94.4 MB	2025-02-11 13:46
node-v22.14.0.tar.gz	45.5 MB	2025-02-11 13:52
win-arm64/	-	2025-02-11 09:23
win-x64/	-	2025-02-11 08:57
win-x86/	-	2025-02-11 08:47

安装 Node 的常见问题

安装太慢

网络原因，解决方法有两种：

- 需要开代理去访问（推荐，因为很多环境问题都是因为网络访问不通导致的）
- 使用镜像源（部署在国内的资源）

使用镜像源安装

- 访问：<https://mirrors.aliyun.com/nodejs-release/v22.14.0/>
- 点击 node-v22.14.0 下载

选择一个合适的版本下下，一般是 x86 的

node-v22.14.0-linux-armv7l.tar.gz	46.8 MB	2025-02-11 08:08
node-v22.14.0-linux-armv7l.tar.xz	24.2 MB	2025-02-11 08:11
node-v22.14.0-linux-ppc64le.tar.gz	54.2 MB	2025-02-11 08:09
node-v22.14.0-linux-ppc64le.tar.xz	28.9 MB	2025-02-11 08:12
node-v22.14.0-linux-s390x.tar.gz	51.8 MB	2025-02-11 08:18
node-v22.14.0-linux-s390x.tar.xz	27.1 MB	2025-02-11 08:21
node-v22.14.0-linux-x64.tar.gz	51.6 MB	2025-02-11 08:08
node-v22.14.0-linux-x64.tar.xz	28.5 MB	2025-02-11 08:12
node-v22.14.0-win-arm64.7z	18.8 MB	2025-02-11 09:23
node-v22.14.0-win-arm64.zip	29.5 MB	2025-02-11 09:23
node-v22.14.0-win-x64.7z	20.8 MB	2025-02-11 08:57
node-v22.14.0-win-x64.zip	33.3 MB	2025-02-11 08:47
node-v22.14.0-win-x86.7z	19.1 MB	2025-02-11 08:47
node-v22.14.0-win-x86.zip	30.8 MB	2025-02-11 08:47
node-v22.14.0-x64.msi	29.5 MB	2025-02-11 08:58
node-v22.14.0-x86.msi	26.9 MB	2025-02-11 08:47
node-v22.14.0.pkg	80.2 MB	2025-02-11 12:34
node-v22.14.0.tar.gz	94.4 MB	2025-02-11 13:46
node-v22.14.0.tar.xz	45.5 MB	2025-02-11 13:52
win-arm64/	-	2025-02-11 09:23
win-x64/	-	2025-02-11 08:57
win-x86/	-	2025-02-11 08:47

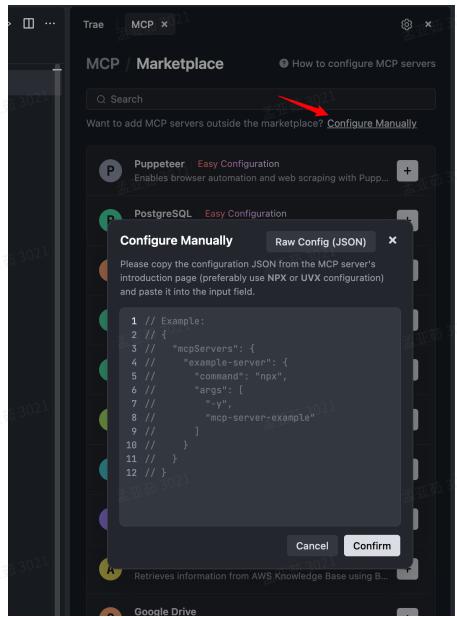
安装 MCP Hosts (Cursor / VSCode / Trae)

Cursor	Trae	Cline
https://www.cursor.com/cn	https://www.trae.ai/	Vscode / Trae 的插件商店中下载 Trae 和 (trae 已支持) VSCode 原生不支持 MCP，需要借助 Cline 去启动 MCP 的能力。
打开 Cursor settings	点击右上角头像，并选择 MCP 菜单	后续以 Trae 为例，VSCode 和 Trae 使用 Cline 插件本身没有区别，设置步骤是通用的。 在 cline 中配置 MCP server
切换到 MCP 页面，并点击 add new global MCP server	点击 Add MCP Servers	
复制以下代码到 mcp.json 文件中，然后去 MasterGo 中生成 Token	如果 windows 使用下面的配置不行，可以再试试下面的配置。MacOs 的配置只看这个就够了	复制以下代码到 cline_mcp_settings.json 文件中，然后去 MasterGo 中生成 Token
MacOs / Windows：		

```

2      "mcpServers": {
3        "mastergo-
4          magic-mcp": {
5            "command": "npx",
6            "args": [
7              "-y",
8              "@mastergo/mag
9              ic-mcp",
10             "--"
11             token=MG_MCP_T
12             OKEN",
13             "--"
14             url=https://ma
15             stergo.com"
16           ],
17           "env": {
18             "NPM_CONFIG_REGI
19             STRY": "https://regis
20             try.npmjs.org/
21             "
22           }
23         }
24       }

```



MacOs / Windows:

如果 windows 使用下面的配置不行，可以再试试下面的配置。

MacOs 的配置只看这个就够了

```

1   {
2     "mcpServers": {
3       "mastergo-
4         magic-mcp": {
5         "command": "npx",
6         "args": [
7           "-y",
8           "@mastergo/mag
9           ic-mcp",
10          "--"
11          token=MG_MCP_TOK
12          EN",
13          "--"
14          url=https://mast
15          ergo.com"
16        ],
17        "env": {
18          "NPM_CONFIG_REGI
19          STRY": "https://regis
20          try.npmjs.org/"
21        }
22      }
23    }

```

Windows:

```

1   {
2     "mcpServers": {
3       "mastergo-
4         magic-mcp": {
5         "command": "cmd",
6         "args": [
7           "-y",
8           "@mastergo/mag
9           ic-mcp",
10          "--"
11          token=MG_MCP_TOKE
12          N",
13          "--"
14          url=https://maste
15          rgo.com"
16        ],
17        "env": {
18          "NPM_CONFIG_REGIS
19          TRY": "https://regis
20          try.npmjs.org/"
21        }
22      }
23    }

```

Windows:

```

1   {
2     "mcpServers": {
3       "mastergo-
4         magic-mcp": {
5         "command": "cmd",
6         "args": [
7           "-y",
8           "@mastergo/mag
9           ic-mcp",
10          "--"
11          token=MG_MCP_TOKE
12          N",
13          "--"
14          url=https://maste
15          rgo.com"
16        ],
17        "env": {
18          "NPM_CONFIG_REGIS
19          TRY": "https://regis
20          try.npmjs.org/"
21        }
22      }
23    }

```

```

5      "args": [
6          [
7              "/c",
8              "npx",
9              "-y",
10             "@mastergo/magic-mcp",
11             "--",
12             "token=MG_TOKEN",
13             "--",
14             "url=https://mastergo.com"
15         ],
16     ],
17   },
18 }

```

```

"args": [
    [
        "/c",
        "npx",
        "-y",
        "@mastergo/magic-mcp",
        "--",
        "token=MG_TOKEN",
        "--",
        "url=https://mastergo.com"
    ]
]

```

```

5      "args": [
6          [
7              "/c",
8              "npx",
9              "-y",
10             "@mastergo/magic-mcp",
11             "--",
12             "token=MG_TOKEN",
13             "--",
14             "url=https://mastergo.com"
15         ],
16     ],
17   ],
18 }

```

Windows:

代码块

```

1  {
2      "mcpServers": {
3          "mastergo-magic-mcp": {
4              "command": "command",
5              "cmd",
6              "args": [
7                  "/c",
8                  "npx",
9                  "-y",
10                 "@mastergo/magic-mcp",
11                 "--",
12                 "token=MG_TOKEN",
13                 "--",
14                 "url=https://mastergo.com"
15             ]
16         }
17     }
18 }

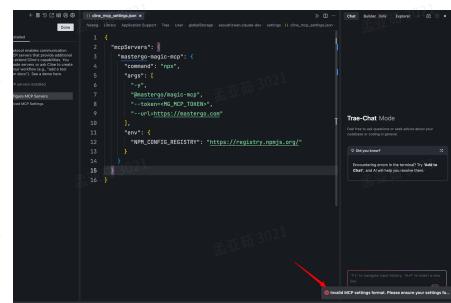
```

```

1  {
2      "mcpServers": {
3          "mastergo-magic-mcp": {
4              "command": "command",
5              "cmd",
6              "args": [
7                  "/c",
8                  "npx",
9                  "-y",
10                 "@mastergo/magic-mcp",
11                 "--",
12                 "token=MG_TOKEN",
13                 "--",
14                 "url=https://mastergo.com"
15             ]
16         }
17     }
18 }

```

如图所示

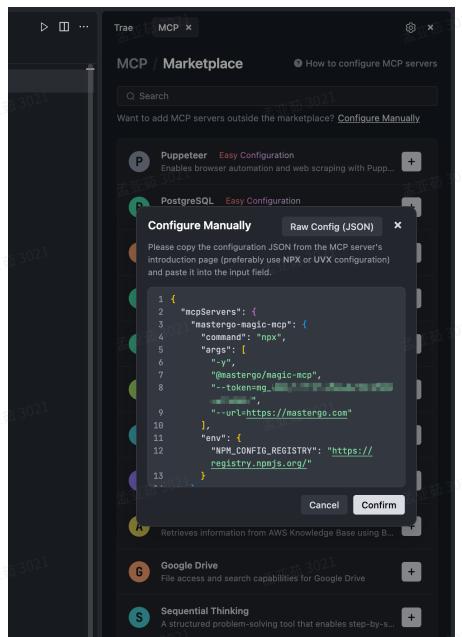


右下角可能会报错，这个问题是 Cline 插件的 bug，先不管，我们接着往下做

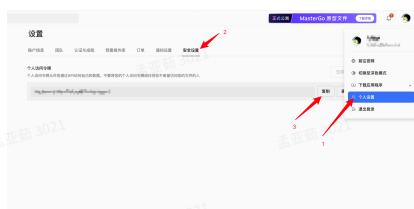
<https://github.com/cline/cline/issues/2459>

粘贴进去，

token 如何获取可以往下看



获取 Token



替换 Token，并保存文件

MG_MCP_TOKEN 替换成上一步中获取的 Token，如下图

```
curl -X POST https://mastergo.com/api/v1/mcp/getToken -H "Content-Type: application/json" -d '{"name": "mcpToken"}'
```

```
1
2   "mcpServers": [
3     "mastergo-magic-mcp": {
4       "command": "npx",
5       "args": [
6         "-y",
7         "-tokerne..._77",
8         "-tokerne..._77",
9         "-urlhttps://mastergo.com"
10      ],
11      "env": {
12        "NPM_CONFIG_REGISTRY": "https://registry.npmjs.org/"
13      }
14    }
15  ]
```

保存后，回到设置页面，如果显示绿色，并且出现了 Tools 有 mcp__getDSL 就是成功了。



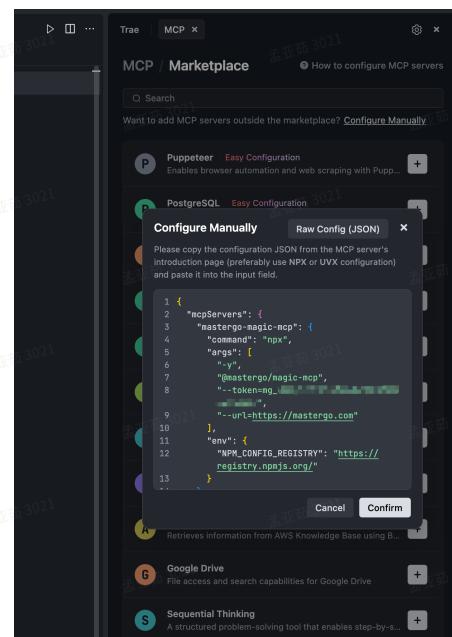
如果 client close，考虑重启一下 Cursor

获取 Token



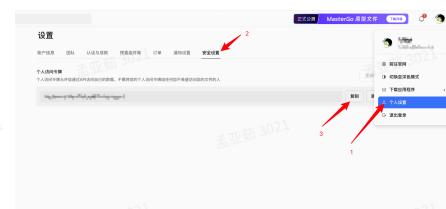
替换 Token，并保存文件，点击 Confirm

MG_MCP_TOKEN 替换成上一步中获取的 Token，如下图



点击同意

获取 Token



替换 Token，并保存文件

MG_MCP_TOKEN 替换成上一步中获取的 Token，如下图

```
curl -X POST https://mastergo.com/api/v1/mcp/getToken -H "Content-Type: application/json" -d '{"name": "mcpToken"}'
```

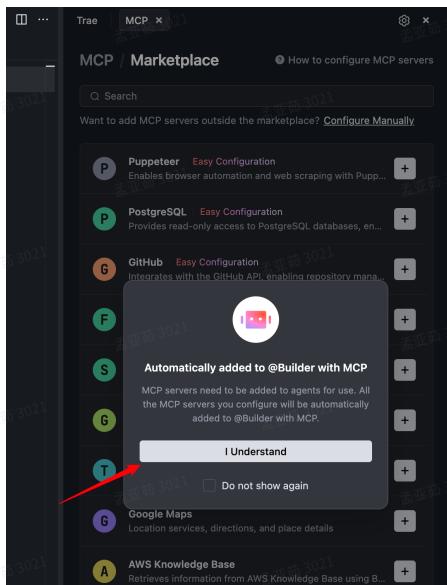
```
1
2   "mcpServers": [
3     "mastergo-magic-mcp": {
4       "command": "npx",
5       "args": [
6         "-y",
7         "-tokerne..._77",
8         "-tokerne..._77",
9         "-urlhttps://mastergo.com"
10      ],
11      "env": {
12        "NPM_CONFIG_REGISTRY": "https://registry.npmjs.org/"
13      }
14    }
15  ]
```

你在保存设置的时候，应该会遇到报错，我们不管他，保存 json 之后，直接重启 traef / VSCode 就可以了

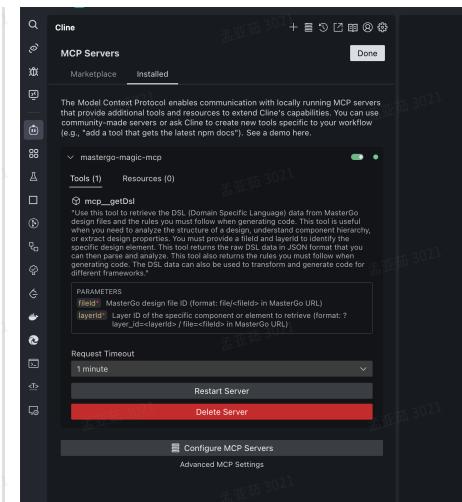
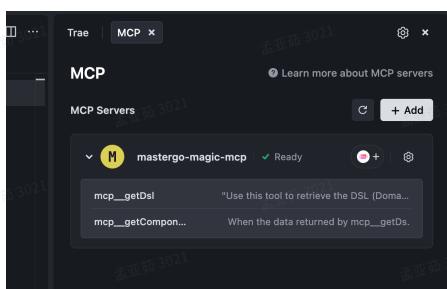
```
"disabled": false,
"autoApprove": []
```

Model MCP settings format: Please ensure your settings follow the correct JSON format.

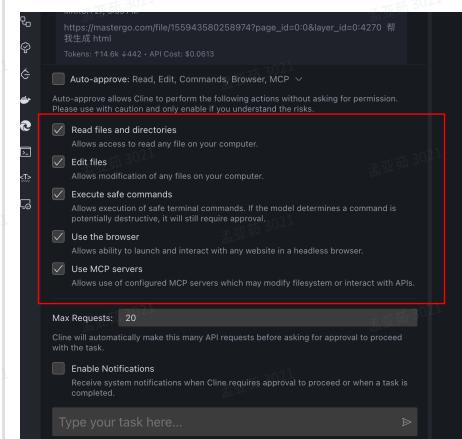
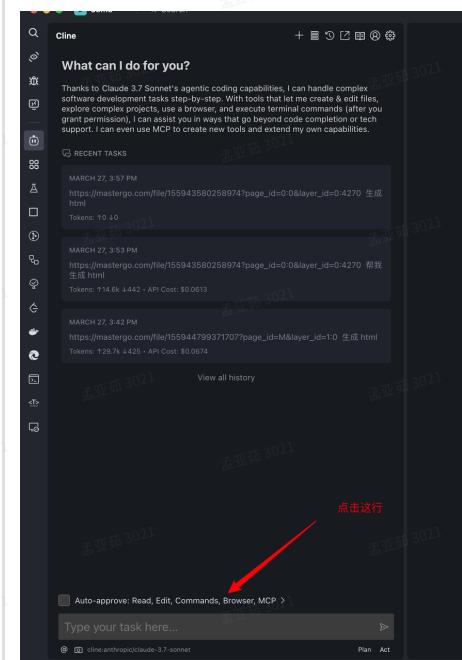
重启完之后，你能看到如下配置，就是成功了



安装成功



接着配置一下 Cline 本身，让其可以调用 MCP



来到 MasterGo 中，选中一个图层，并复制链接

来到 MasterGo 中，选中一个图层，并复制链接

来到 MasterGo 中，选中一个图层，并复制链接

将连接粘贴到 Cursor 中，让其生成页面

需要保证复制的 URL 有

- layerId**

在 Trae 中点击 Agent，切换 Agent 到 Builder With MCP

将连接粘贴到 Cursor 中，告诉 Trae。

将连接粘贴到 Cline 中，让其生成页面

需要保证复制的 URL 有

- layerId**

DEMO 效果

