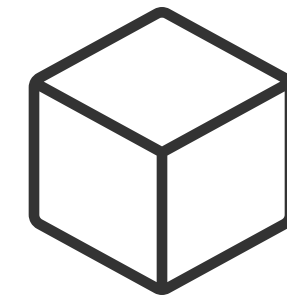


# MCP Registry: Designing For Server Discovery

The building block to unlock rich  
and comprehensive server discovery



# Leading the Registry project



**Tadas Antanavicius**

PulseMCP

One of the earliest  
MCP server directories



**Alex Hancock**

Block / Goose

One of the earliest  
MCP clients



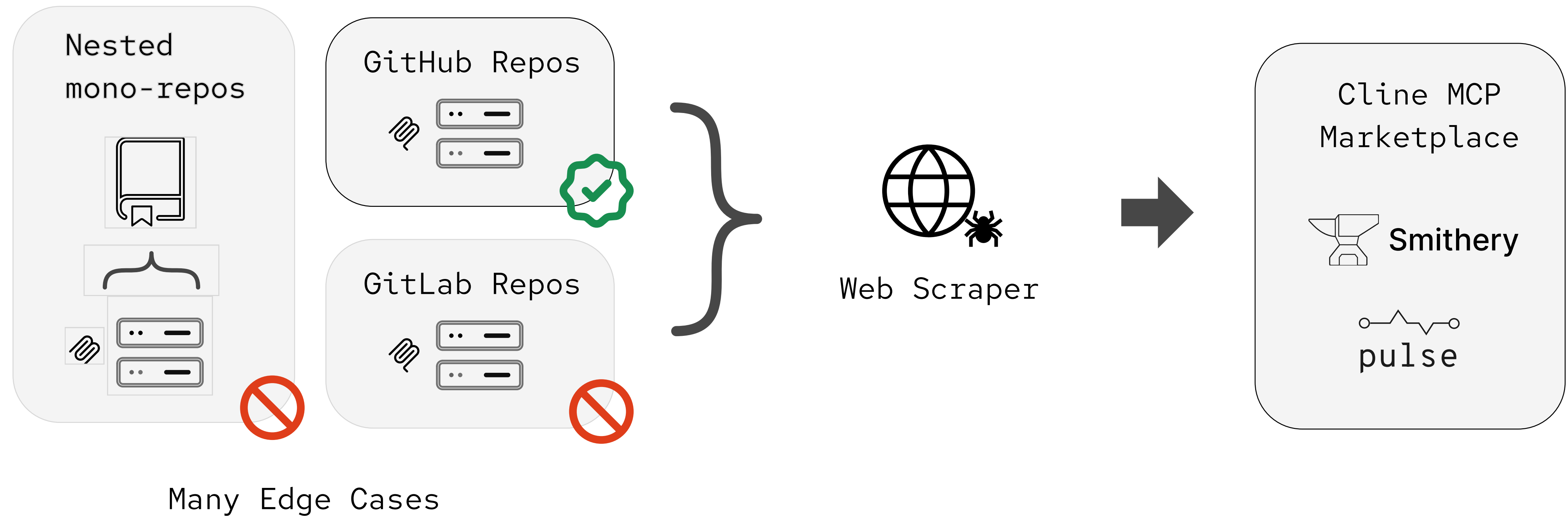
**Toby Padilla**

GitHub

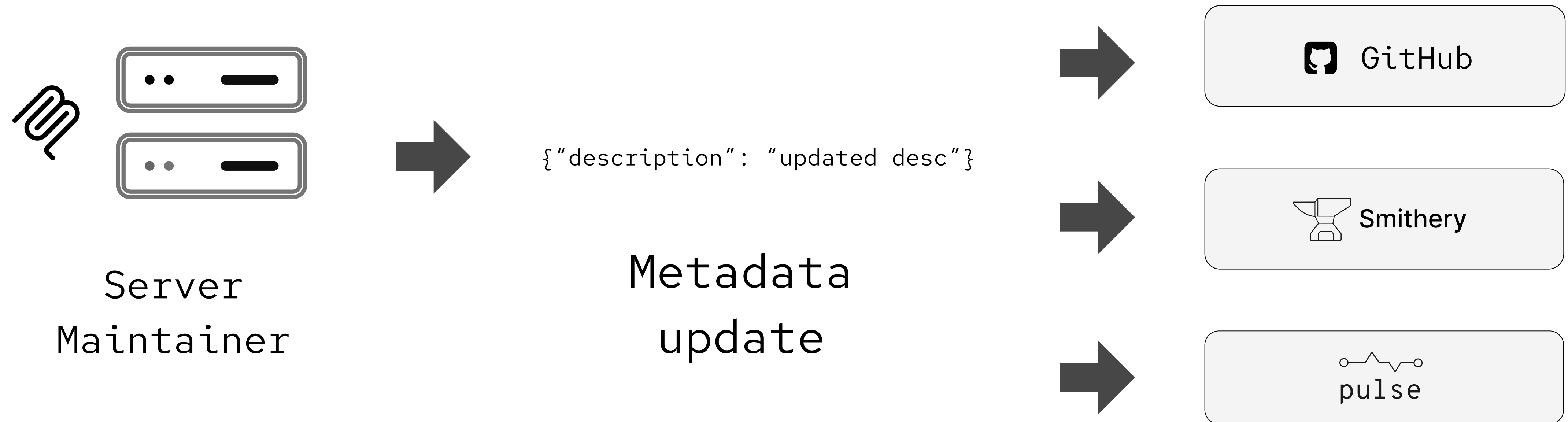
The most popular MCP  
server to-date

Problems we are  
solving for

# Third-party registries can't get comprehensive server lists



# Server maintainers have to manage metadata everywhere



# Client apps need a source-of-truth for installation instructions

### Authentication

To authenticate and save credentials:

1. Run the server with the `auth` argument: `node ./dist auth`
2. This will open an authentication flow in your system browser
3. Complete the authentication process
4. Credentials will be saved in the root of this repo (i.e. `servers/.gdrive-server-credentials.json`)

### Usage with Desktop App

To integrate this server with the desktop app, add the following to your app's server configuration:

#### Docker

Authentication:

Assuming you have completed setting up the OAuth application on Google Cloud, you can now auth the server with the following command, replacing `/path/to/gcp-oauth.keys.json` with the path to your OAuth keys file:

```
docker run -i --rm --mount type=bind,source=/path/to/gcp-oauth.keys.json,target=/gcp-oauth.keys.json -v mcp-gdrive:/gdrive-server
```

The command will print the URL to open in your browser. Open this URL in your browser and complete the authentication process. The credentials will be saved in the `mcp-gdrive` volume.

Once authenticated, you can use the server in your app's server configuration:

```
{  "mcpServers": {    "gdrive": {
```



VS Code

Goose

Cline

Today, we unreliaibly parse READMEs

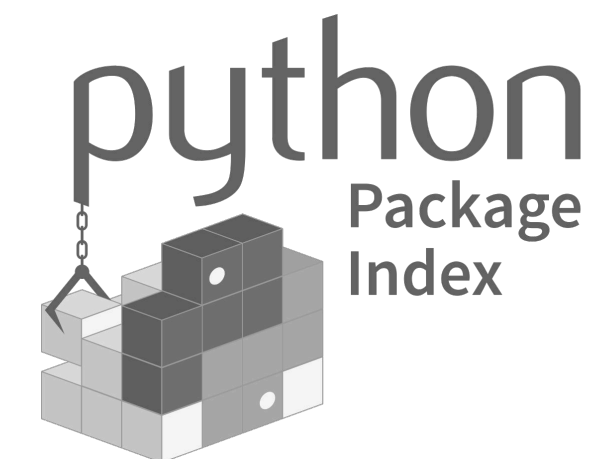


Problems we are not  
solving for

# Source code storage

Will be delegated to other registries





```
1  "packages": [  
2    {  
3      "registry_name": "npm",  
4      "name": "@modelcontextprotocol/server-filesystem",  
5      "version": "1.0.2",  
6      ...  
7    },  
8    {  
9      "registry_name": "docker",  
10     "name": "mcp/filesystem",  
11     "version": "1.0.2",  
12     ...  
13   }  
14 ]
```





# Advanced search, filter, ranking, etc.

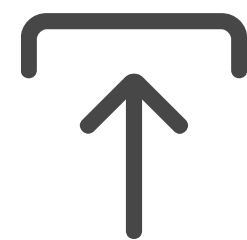
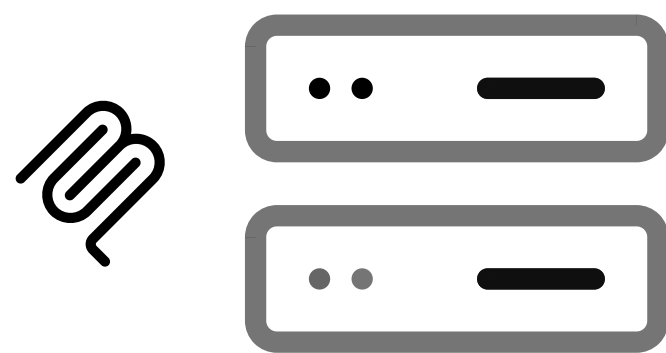
Will be delegated to MCP client marketplaces

Search by name	
Vector search	
Filter by category	
Rank by quality	



**What the solution  
looks like**

# MCP server maintainers publish a `server.json` via CLI



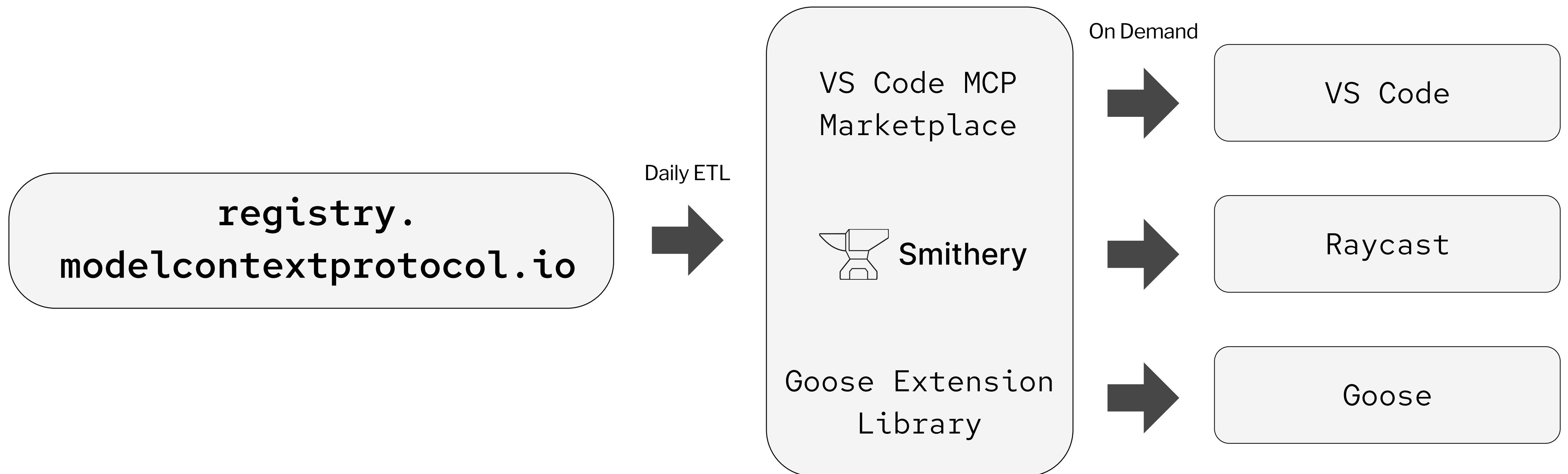
Server  
Maintainer

```
$ mcpr publish ./server.json
```

```
1 {  
2   "name": "io.modelcontextprotocol/filesystem",  
3   "description": "Node.js server implementing Model  
Context Protocol (MCP) for filesystem operations.",  
4   "repository": { ... },  
5   "version_detail": { ... },  
6   "packages": [ ... ]  
7 }
```



# MCP clients have an ETL layer between the official registry REST API



Where we'd love  
more input

# Will MCP client apps have the data they need?

```
1  {
2    "name": "io.modelcontextprotocol/filesystem",
3    "description": "Node.js server implementing Model Context Protocol (MCP) for filesystem operations.",
4    "repository": { ... },
5    "version_detail": { ... },
6    "packages": [
7      {
8        "registry_name": "npm",
9        "name": "@modelcontextprotocol/server-filesystem",
10       "version": "1.0.2",
11       ...
12       "environment_variables": [ ... ]
13     }
14     ...
15   ],
16   "remotes": [
17     {
18       "transport_type": "sse",
19       "url": "https://mcp-fs.example.com/sse"
20     }
21   ]
22 }
```

# Will third party MCP marketplaces be able to use our API shapes?

`registry.modelcontextprotocol.io`

`{"a", "b", "c"}`



Acme Corp Internal Registry

`{"a", "b", "c", "f"}`

Microsoft API Center

`{"a", "b", "c", "d"}`

Cline MCP Marketplace

`{"a", "b", "c", "g"}`



# Demo

Server publication and client installation





# Thank You!



**PulseMCP**

[pulsemcp.com](https://pulsemcp.com)

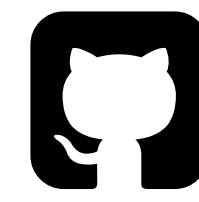
Sign up for our weekly  
MCP newsletter



**Goose**

[block.github.io/goose](https://block.github.io/goose)

Try out Goose, the local  
agentic MCP client app



**GitHub**

[github.com/github/github-  
mcp-server](https://github.com/github/github-mcp-server)

The GitHub MCP server