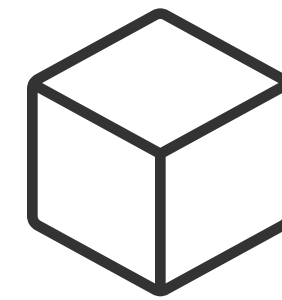


MCP Registry: The Path To (De-)Centralizing Discovery

How we're empowering a rich
ecosystem of discovery
solutions



Who are we?

2 of the 4 maintainers of the MCP Registry,
members of the MCP Steering Committee



Tadas Antanavicius

 PulseMCP.com

Everything MCP. Server
directory and weekly MCP
newsletter.



Toby Padilla

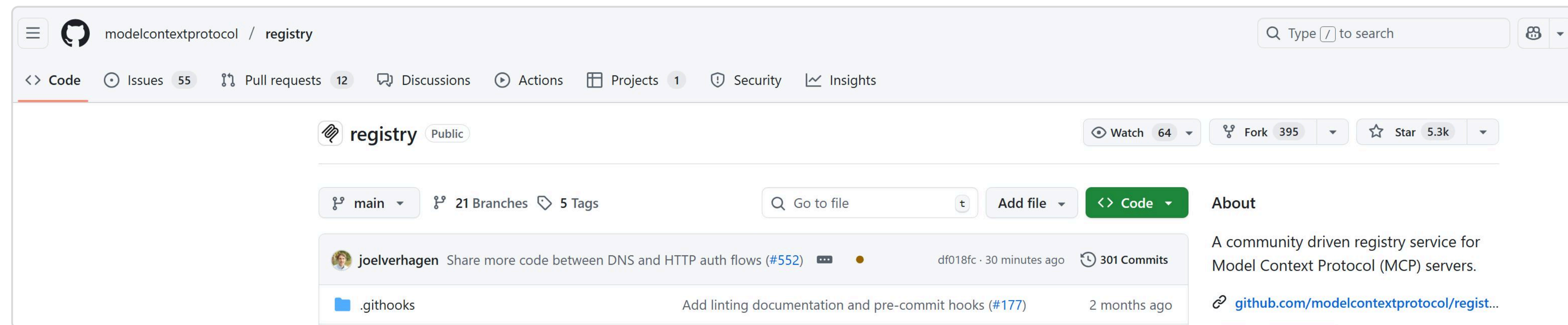
 GitHub

Head of MCP. GitHub
server sub-registry and
GitHub MCP server.

What we are talking about?

REST API for MCP Servers

registry.modelcontextprotocol.io



⚠ There is no UI for the Registry

- Community members have built UIs
- It's not obvious whether a UI on raw registry data is particularly valuable

MCP Registry Documentation

The MCP registry provides MCP clients with a list of MCP servers, like an app store for MCP servers.

I want to...

- Publish my MCP server → [Publishing Guide](#)
- Consume registry data → [API Usage Guide](#)
- Understand the registry's purpose → [Ecosystem vision](#)
- Look up specific information → [server.json spec](#) | [API spec](#) | [CLI reference](#)
- Add reference to my community project → [Community Projects](#)



Updates and upcoming

 September 8: Live Preview Launch

⋮

 October 10: Breaking Changes / Freeze

- Excellent community feedback; breaking changes
- Close to locking in an initial version of the public interfaces

⋮

 Next few months: General Availability

- Final list of TODO's still TBD

Community shout-outs

Maintainers



Adam Jones
Anthropic



Radoslav Dimitrov
Stacklok

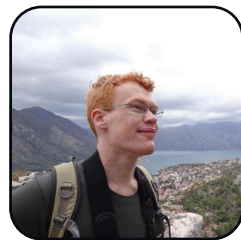
Regular Contributors



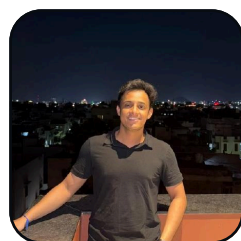
Preeti Dewani
Last9.io



Joel Verhagen
NuGet



Connor Peet
VS Code



Avish Porwal
Microsoft

Thank you all!

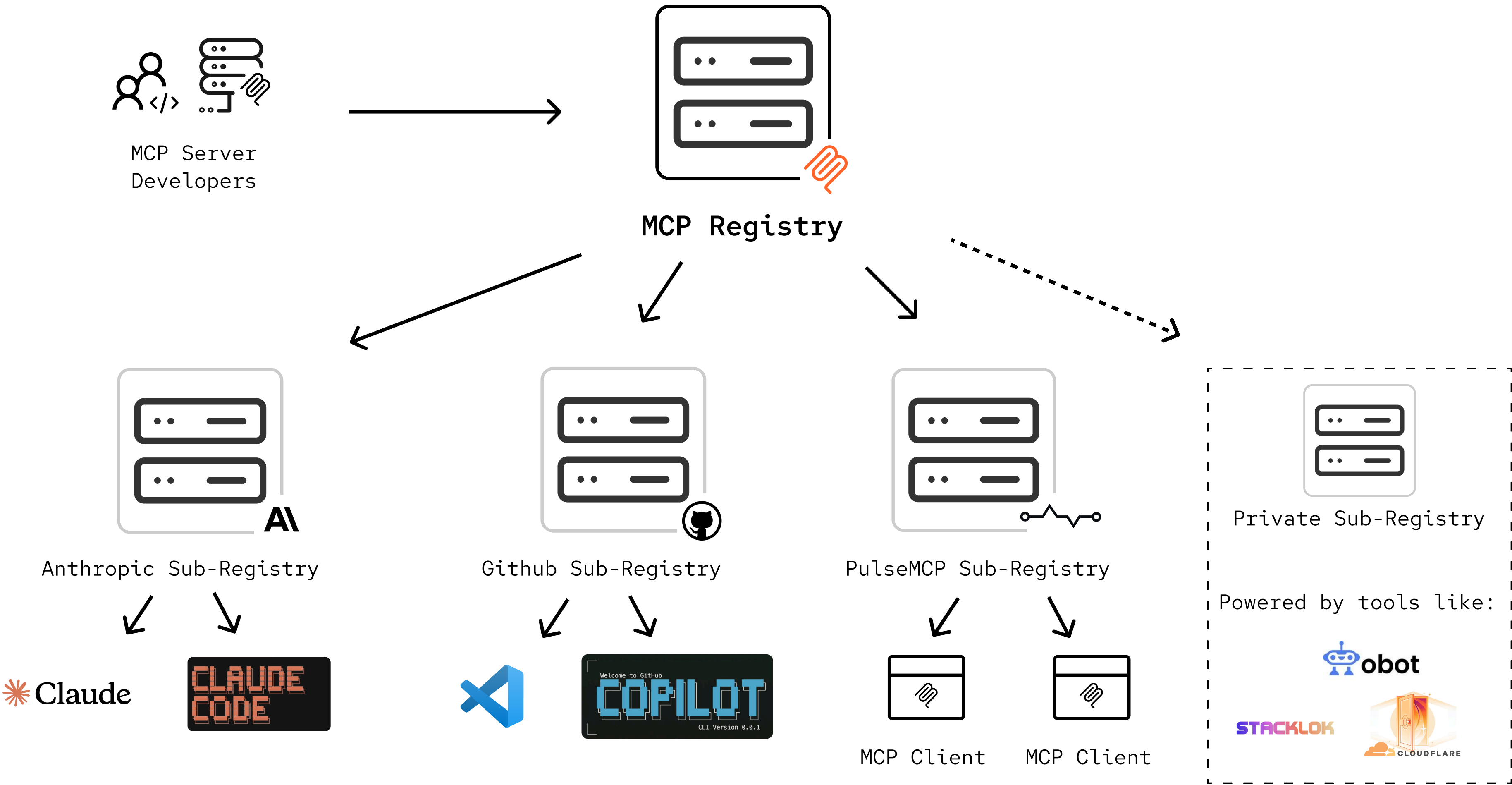
Contributors 46



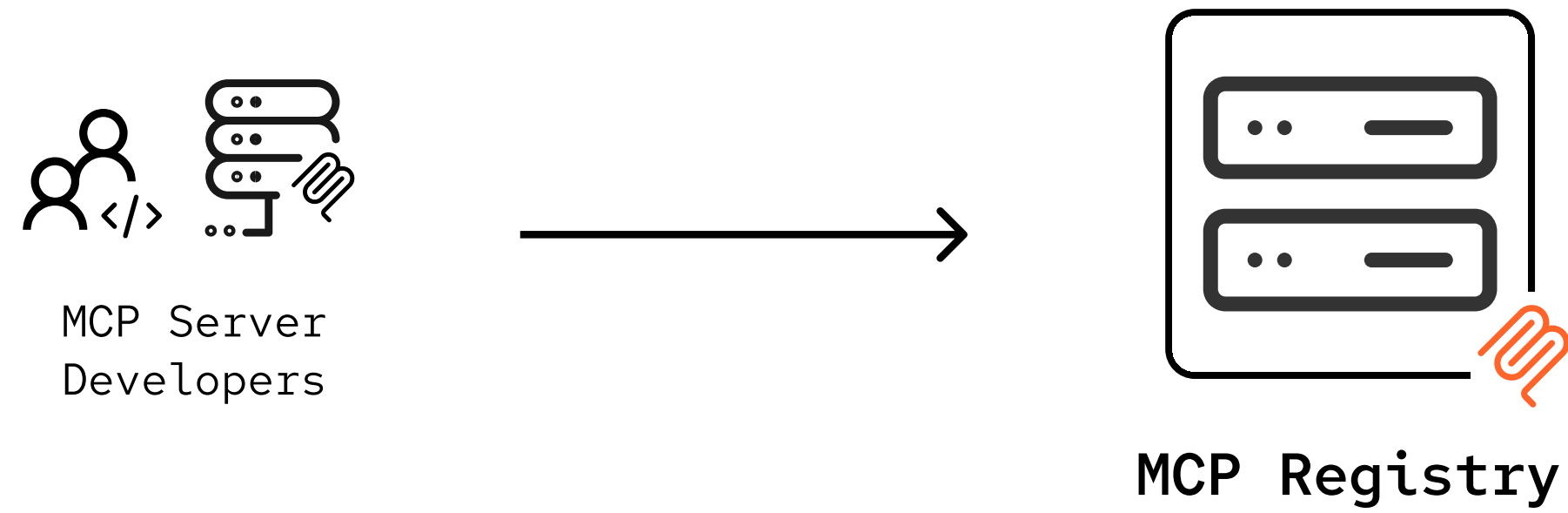
[+ 32 contributors](#)



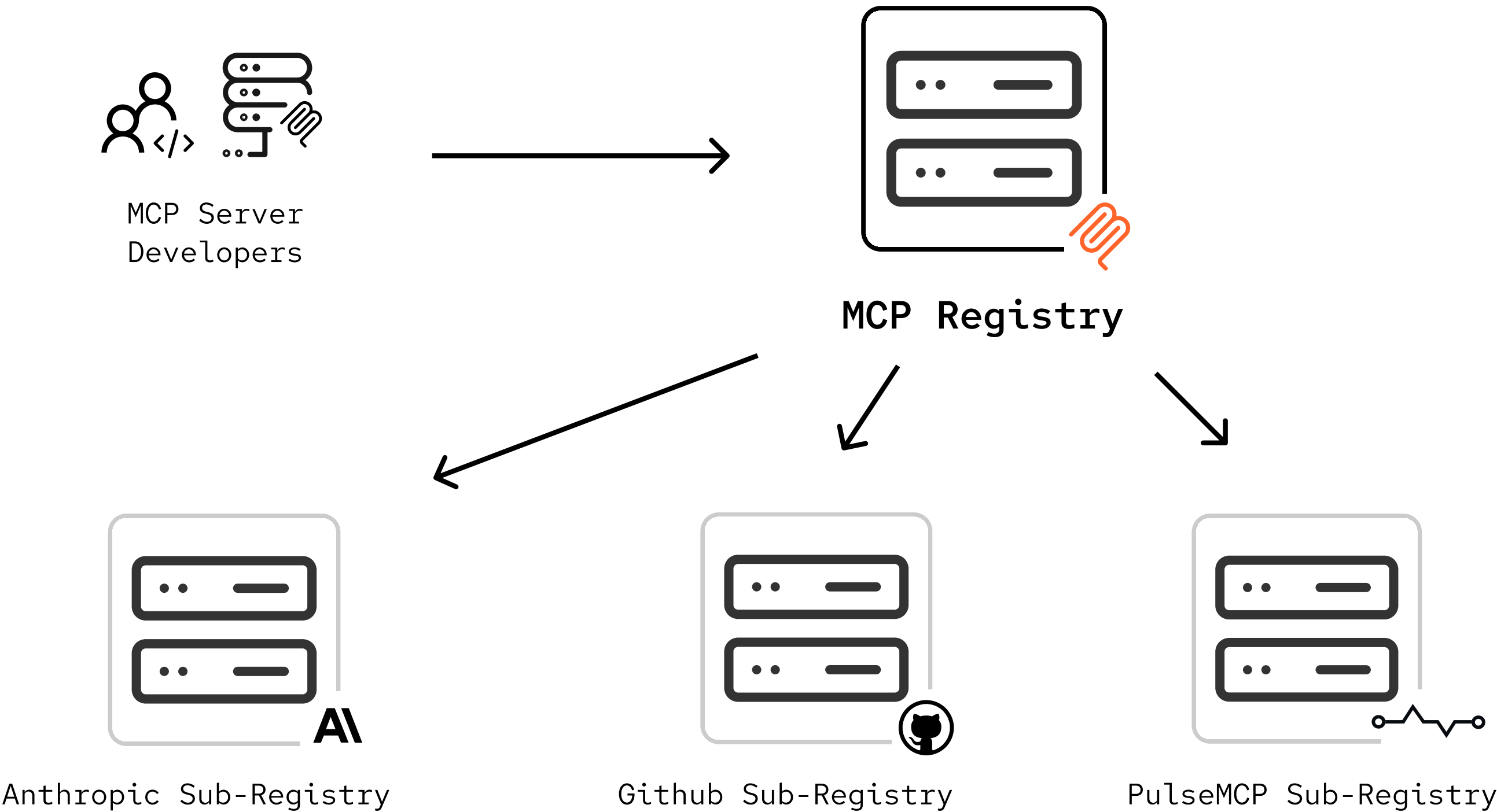
Registry Ecosystem Architecture



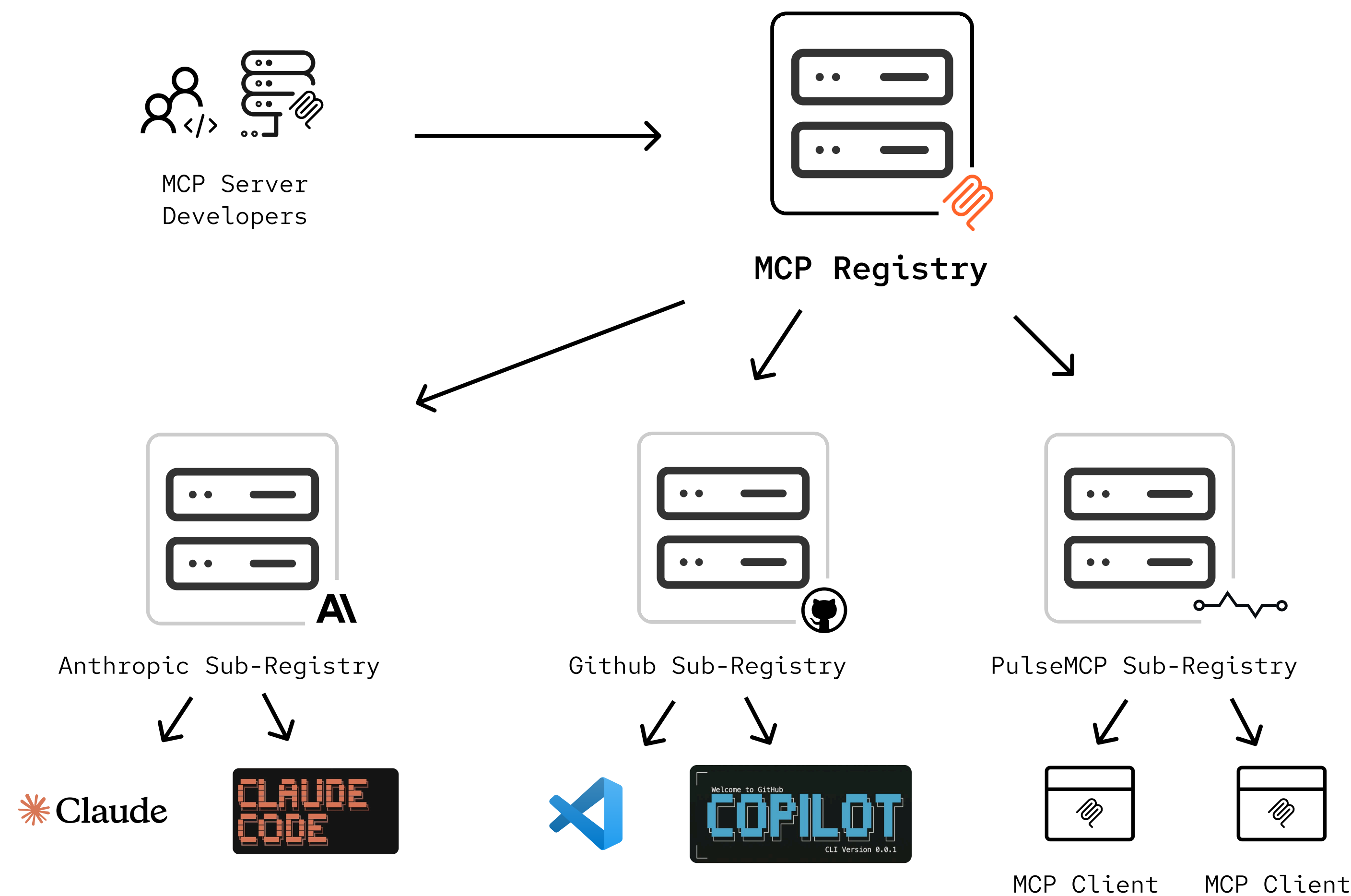
Server devs publish to Registry



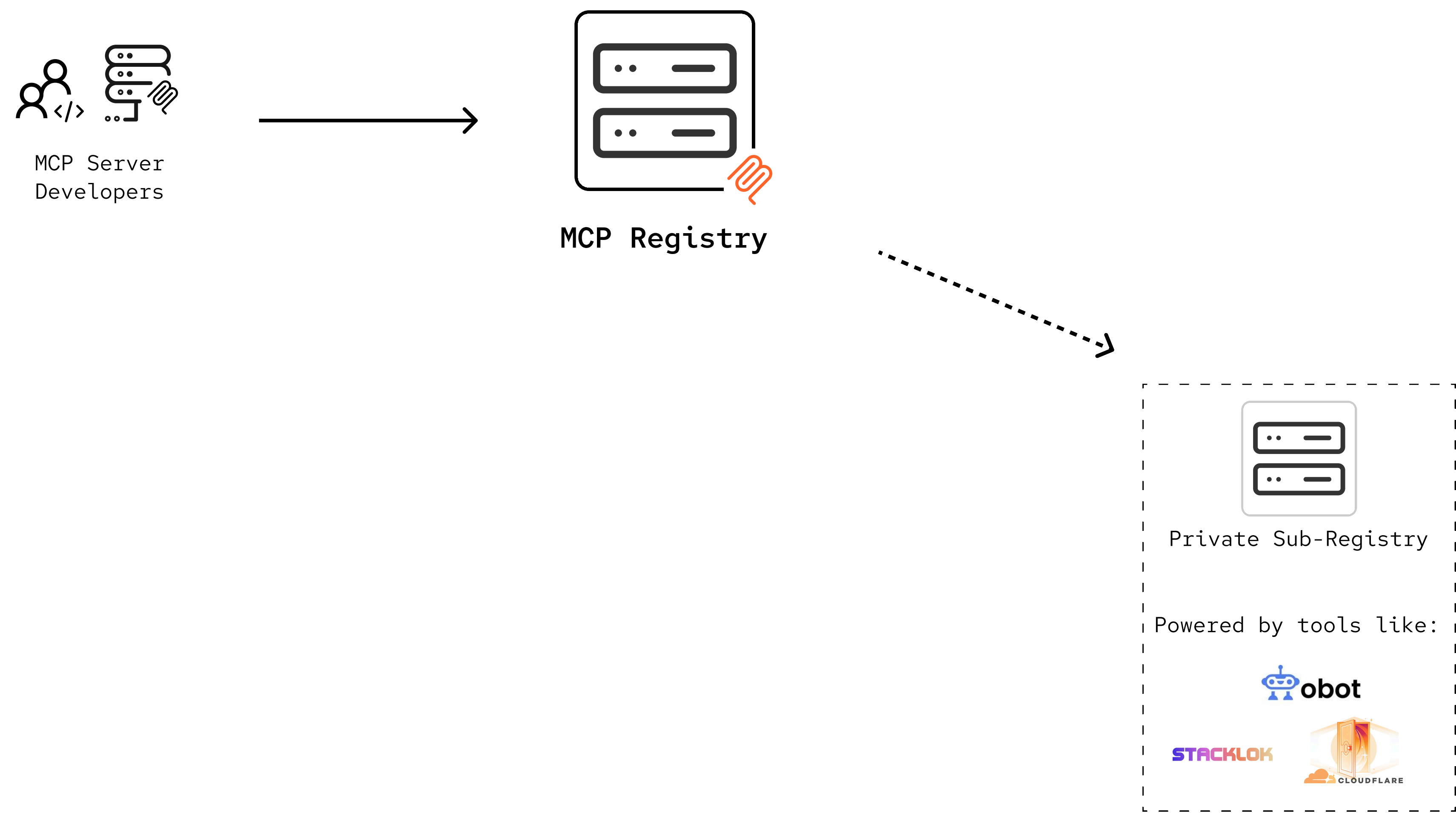
Sub-Registries ETL MCP Registry



MCP Clients Query Sub-Registries



Private Enterprises Self-Host



Introducing server.json

A static definition
of an MCP server

- 1 Identity
- 2 Capabilities
- 3 Location
- 4 Configuration

server.json example

```
1  {
2    ...
3    "name": "com.microsoft/playwright", -- identity
4    "version": "1.0.2",
5    "description": "Provides browser automation capabilities for interaction with web pages
through structured accessibility snapshots.", -- capabilities
6    "repository": {
7      "url": "https://github.com/microsoft/playwright-mcp",
8      "source": "github"
9    },
10   "packages": [
11     {
12       "registryType": "npm",
13       "registryBaseUrl": "https://registry.npmjs.org",
14       "identifier": "@playwright/mcp", -- location
15       "version": "1.0.2",
16       "transport": {
17         "type": "stdio"
18       },
19       "packageArguments": [ -- configuration
20         {
21           "type": "named",
22           "name": "--allowed-hosts",
23           "description": "Comma-separated list of hosts this server is allowed to serve from.
Defaults to the host the server is bound to.",
24           "valueHint": "host1.com,host2.com",
25           "isRequired": false
26         }
27       ]
28     }
29   ]
30 }
31
```

server.json Identity

```
1  {
2    ...
3    "name": "com.microsoft/playwright",
4    "version": "1.0.2"
5    ...
6  }
7
8    "url": "https://github.com/microsoft/playwright-mcp",
9    "source": "github"
10 },
11 "packages": [
12   {
13     "registryType": "npm",
14     "registryBaseUrl": "https://registry.npmjs.org",
15     "identifier": "@playwright/mcp", -- location
16     "version": "1.0.2",
17     "transport": {
18       "type": "stdio"
19     },
20     "packageArguments": [ -- configuration
21       {
22         "type": "named",
23         "name": "--allowed-hosts",
24         "description": "Comma-separated list of hosts this server is allowed to serve from.
25 Defaults to the host the server is bound to.",
26         "valueHint": "host1.com,host2.com",
27         "isRequired": false
28       }
29     ]
30   }
31 }
```

server.json

Capabilities

```
1  {
2    ...
3    "description": "Provides browser automation capabilities for interaction with web pages through
structured accessibility snapshots.", -- capabilities
4    ...
5    "packages": [
6      {
7        "registryType": "npm",
8        ...
9        "transport": {
10          "type": "stdio"
11        },
12        ...
13      }
14    ]
15  }
16  ...
17  "packageManagers": [
18    {
19      "type": "configuration",
20      {
21        "type": "named",
22        "name": "--allowed-hosts",
23        "description": "Comma-separated list of hosts this server is allowed to serve from.
Defaults to the host the server is bound to.",
24        "valueHint": "host1.com,host2.com",
25        "isRequired": false
26      }
27    }
28  ]
29  ]
30  }
31
```

server.json Location

```
1  {
2    ...
3    "name": "com.microsoft/playwright", -- identity
4    "version": "1.0.2",
5    "description": "Provides browser automation capabilities for interaction with web pages
```

```
1  {
2    ...
3    "packages": [
4      {
5        "registryType": "npm",
6        "registryBaseUrl": "https://registry.npmjs.org",
7        "identifier": "@playwright/mcp",
8        "version": "1.0.2",
9        ...
10     }
11   ]
12 }
```

```
1  {
2    ...
3    "remotes": [
4      {
5        "type": "streamable-http",
6        "url": "https://mcp.notion.com/mcp"
7      },
8      {
9        "type": "sse",
10       "url": "https://mcp.notion.com/sse"
11     }
12   ]
13   ...
14 }
```

```
19     "packageArguments": [ -- configuration
20       {
21         "type": "named",
22         "name": "--allowed-hosts",
23         "description": "Comma-separated list of hosts this server is allowed to serve from.
24 Defaults to the host the server is bound to.",
25         "valueHint": "host1.com,host2.com",
26         "isRequired": false
27       }
28     ]
29   ]
30 }
31
```

server.json

Configuration

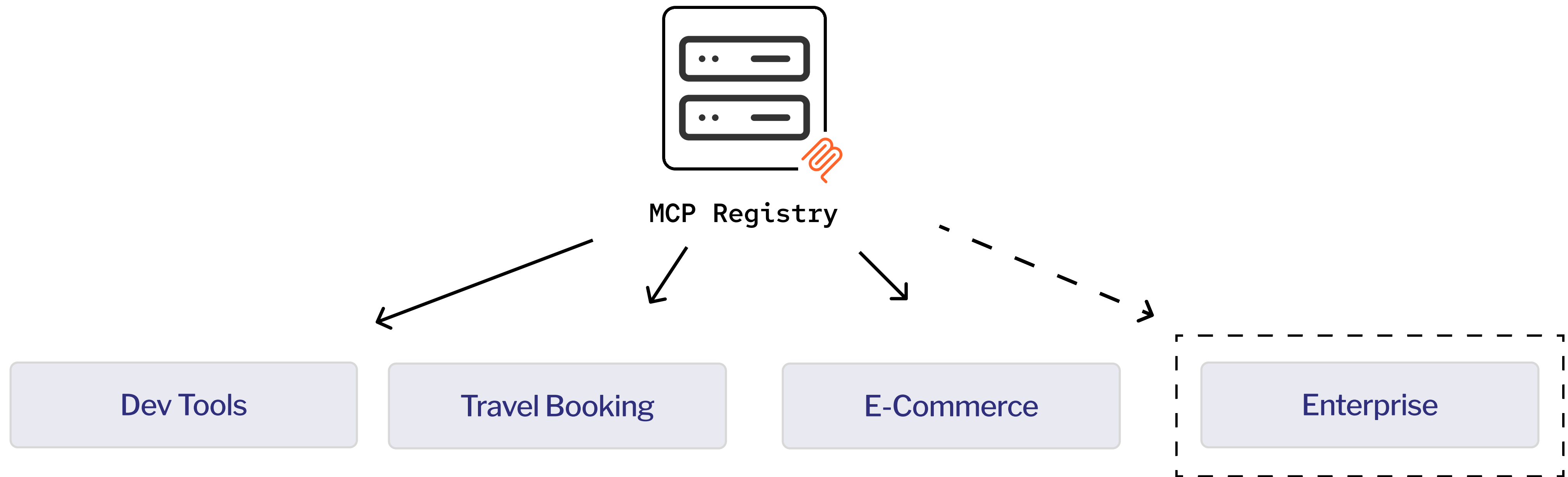
```
1  {
2    ...
3    "name": "com.microsoft/playwright", -- identity
4    "version": "1.0.2",
5    "description": "Provides browser automation capabilities for interaction with web pages
through structured accessibility snapshots.", -- capabilities
6    "repository": {
7      "url": "https://github.com/microsoft/playwright-mcp",
8      "source": "github"
9    },
10   "packages": [
11     {
12       "registryType": "npm",
13       "registryBaseUrl": "https://registry.npmjs.org",
```

```
1  ...
2    "packages": [
3      {
4        ...
5        "packageArguments": [
6          {
7            "type": "named",
8            "name": "--allowed-hosts",
9            "description": "Comma-separated
list of hosts this server is allowed to serve
from. Defaults to the host the server is
bound to.",
10           "valueHint": "host1.com,host2.com",
11           "isRequired": false
12         }
13       ]
14     }
15   ]
```

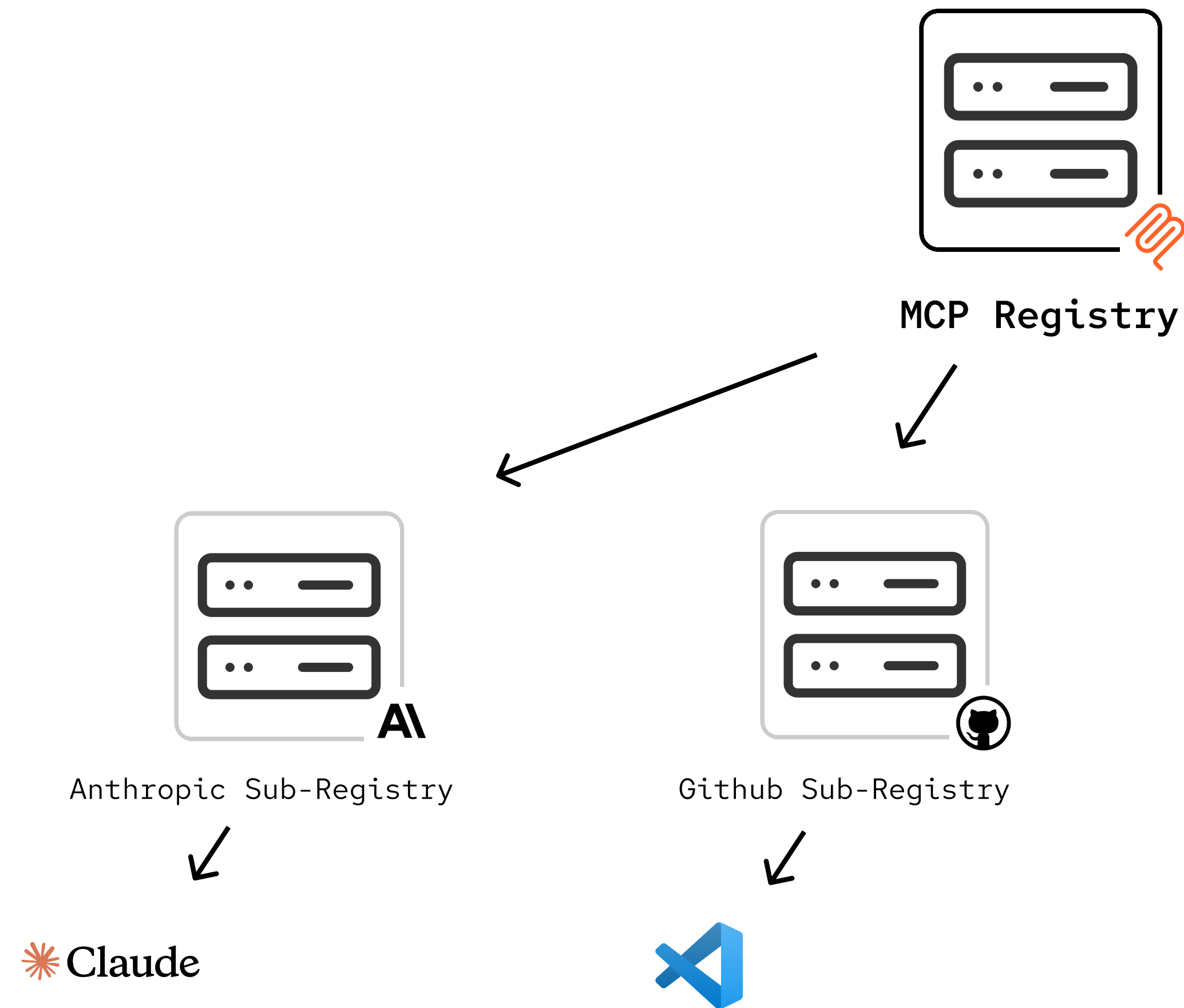
```
1  ...
2    "remotes": [
3      {
4        "type": "streamable-http",
5        "url": "https://server.smithery.ai/@smithery/
notion/mcp",
6        "headers": [
7          {
8            "description": "Bearer token for Smithery
authentication",
9            "isRequired": true,
10           "value": "Bearer {smithery_api_key}",
11           "isSecret": true,
12           "name": "Authorization"
13         }
14       ]
15     }
16   ]
```



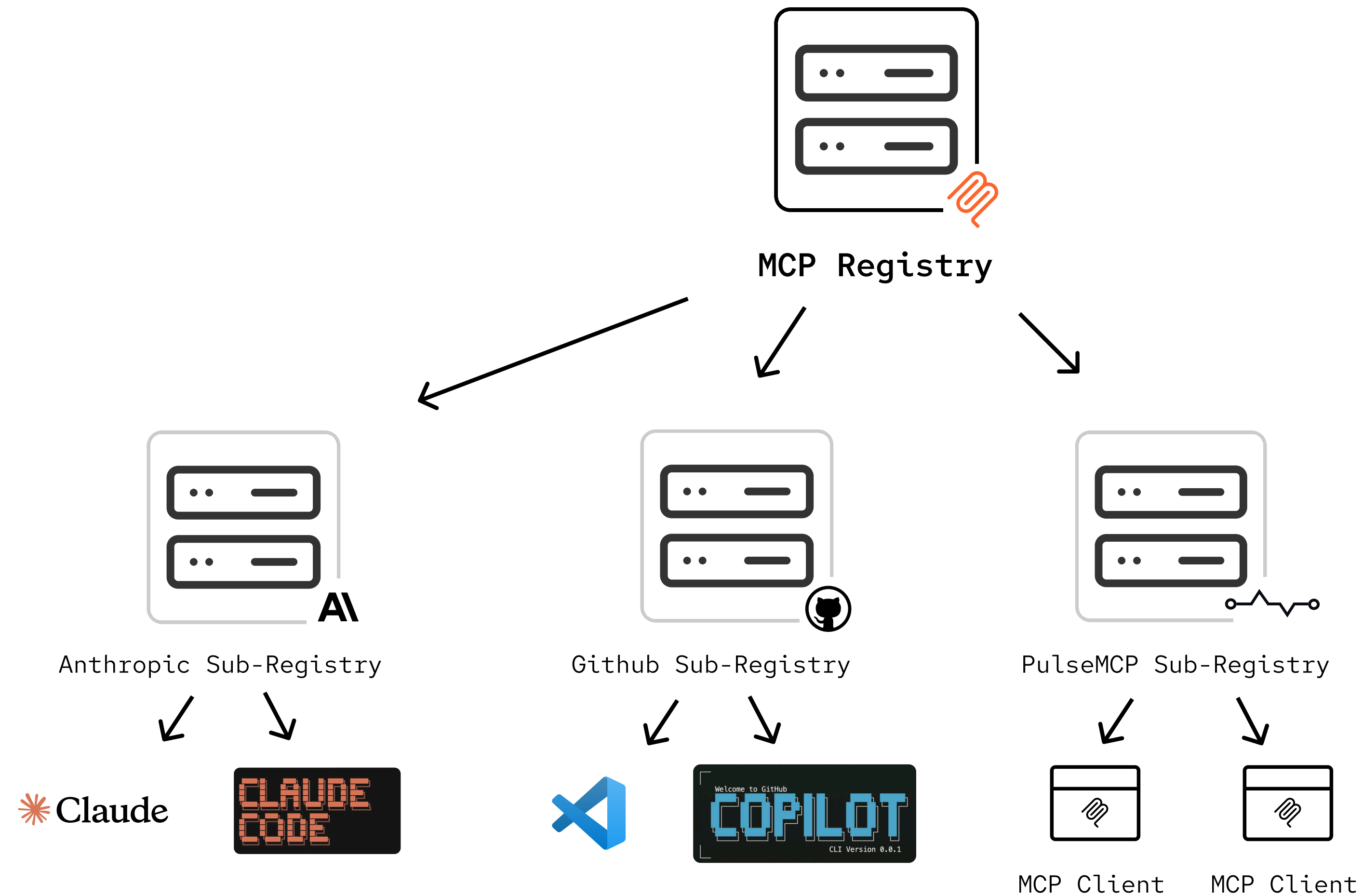
Sub-Registries: Why We Need Them



Every MCP Client needs a sub-registry



Many will have cross-cutting concerns

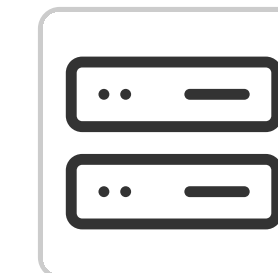
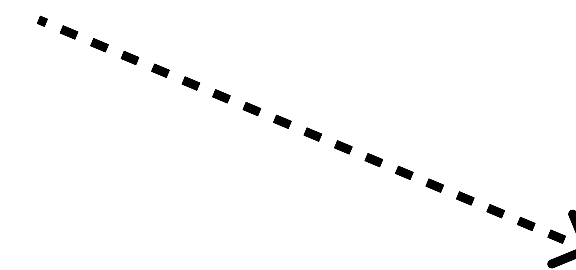


Private sub-registries

Can use enterprise solutions



MCP Registry



Private Sub-Registry

Powered by tools like:



STACKLOK



Server Devs/Maintainers

What does this mean for you?

One place to publish (via CI/CD)

No more: sub-registry formats, publication processes

No more: complex READMEs laden with JSON



MCP Client Owners

What does this mean for you?

You get a structured source of truth

Reliably install any MCP server (or recognize you can't)

No more scraping the web to find them all



DEMO



What's next?

Push for stability: Preview → General Availability

server.json formally into MCP specification via SEP

Further de-centralization: .well-known discovery

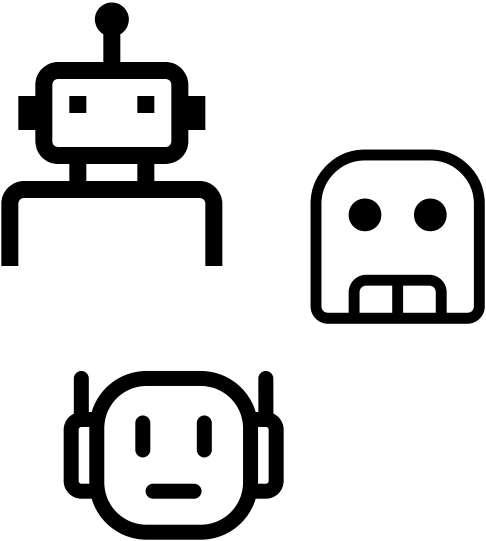


.well-known and server.json

Expecting: {domain}/.well-known/mcp/servers.json

```
1  [  
2    { ...server.json },  
3    { ...server.json },  
4    { ...server.json }  
5  ]
```

.well-known and server.json



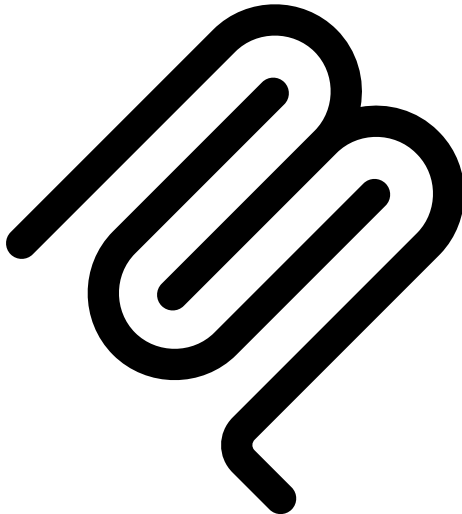
Agent

→
Deep
Research

local-plumber.com/.well-known/mcp/servers.json

```
1  [  
2    { ...server.json }  
3  ]
```

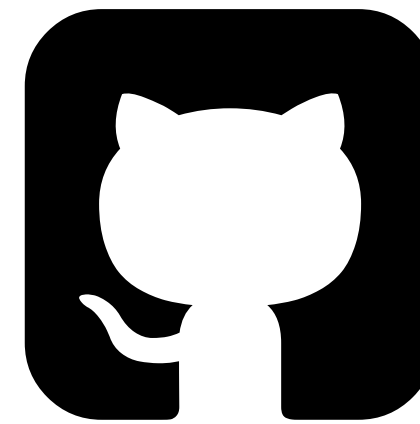
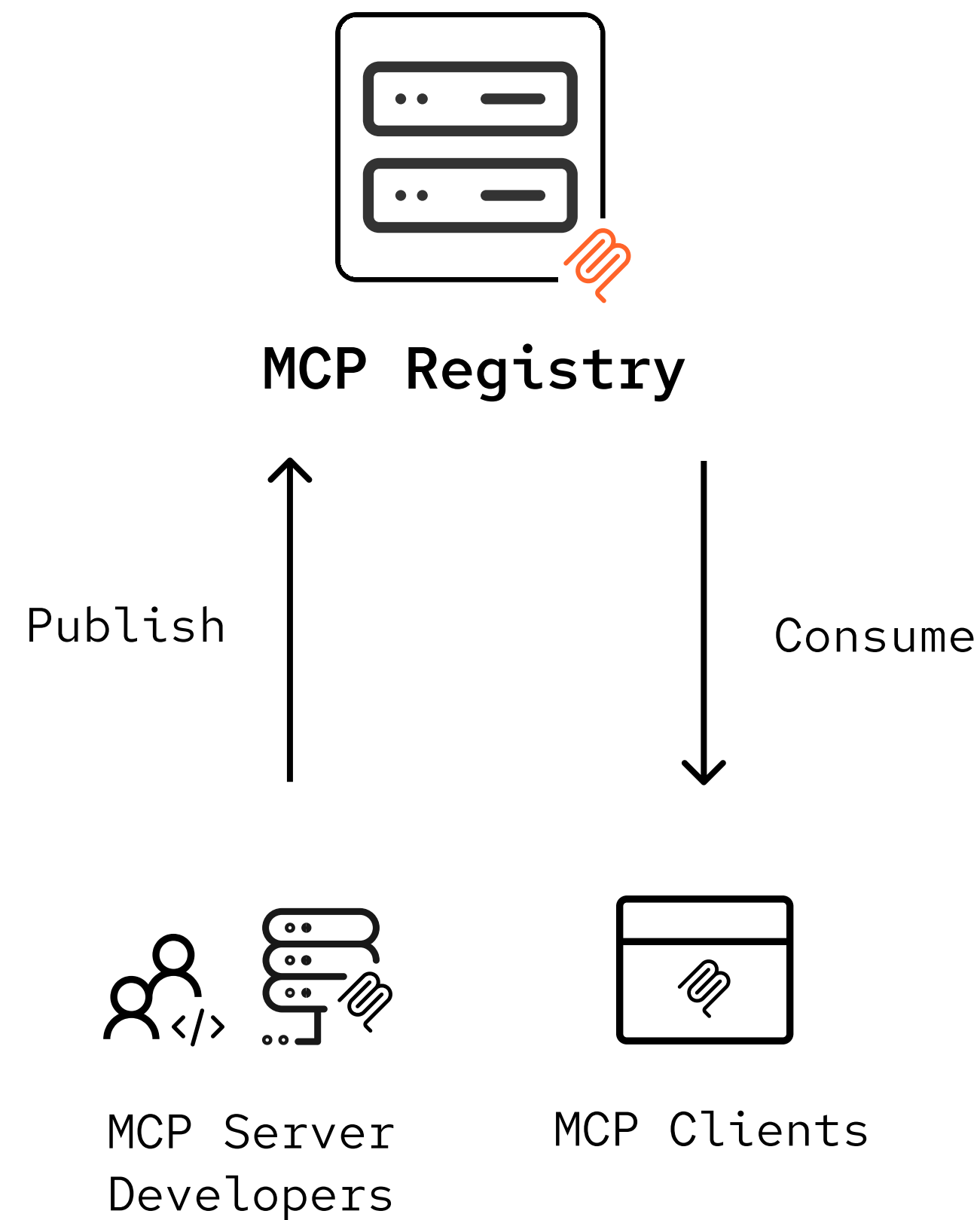
→
Install



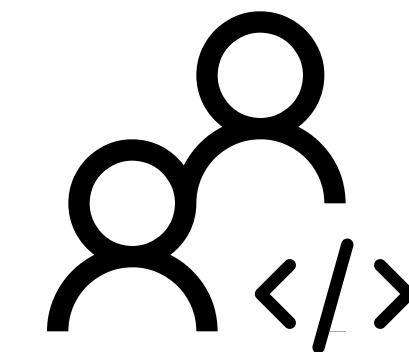
MCP Tools:
check_
availability,
book_
appointment

Want to lend a hand?

Contribute to the community effort



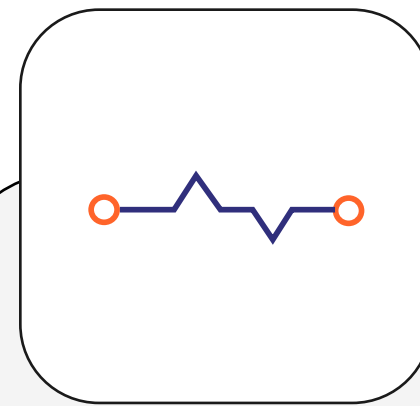
Tackle
issues,
review PRs



Self-directed
but aligned
contributions



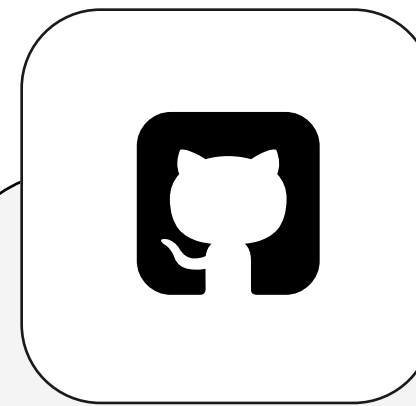
Thank You!



PulseMCP

[pulsemcp.com/
partner-with-us](https://pulsemcp.com/partner-with-us)

Sponsorships and
marketing collabs



GitHub

github.com/mcp

Explore the [GitHub MCP
Registry](#) (self-
publishing coming soon)