

Make your AI useful with MCP

Tobias Nyholm

Why?

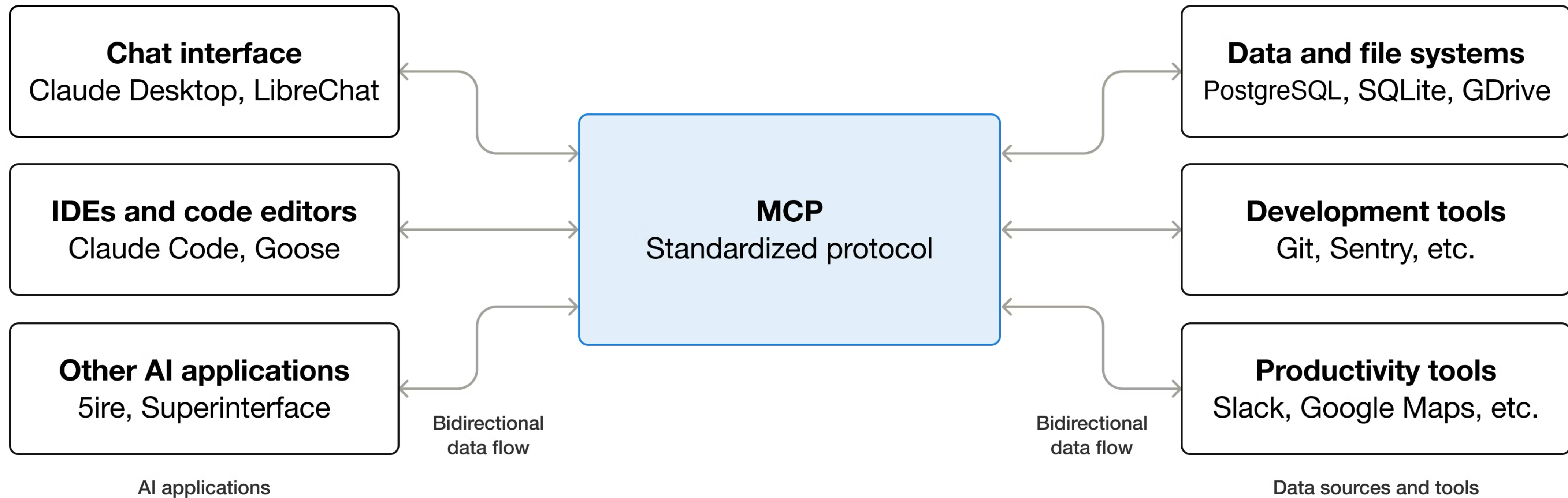
Tobias Nyholm

- Creating good stuff at Eneba.com
- Certified Symfony developer
- Symfony core member
- Symfony CARE
- PHP-Stockholm
- Open source

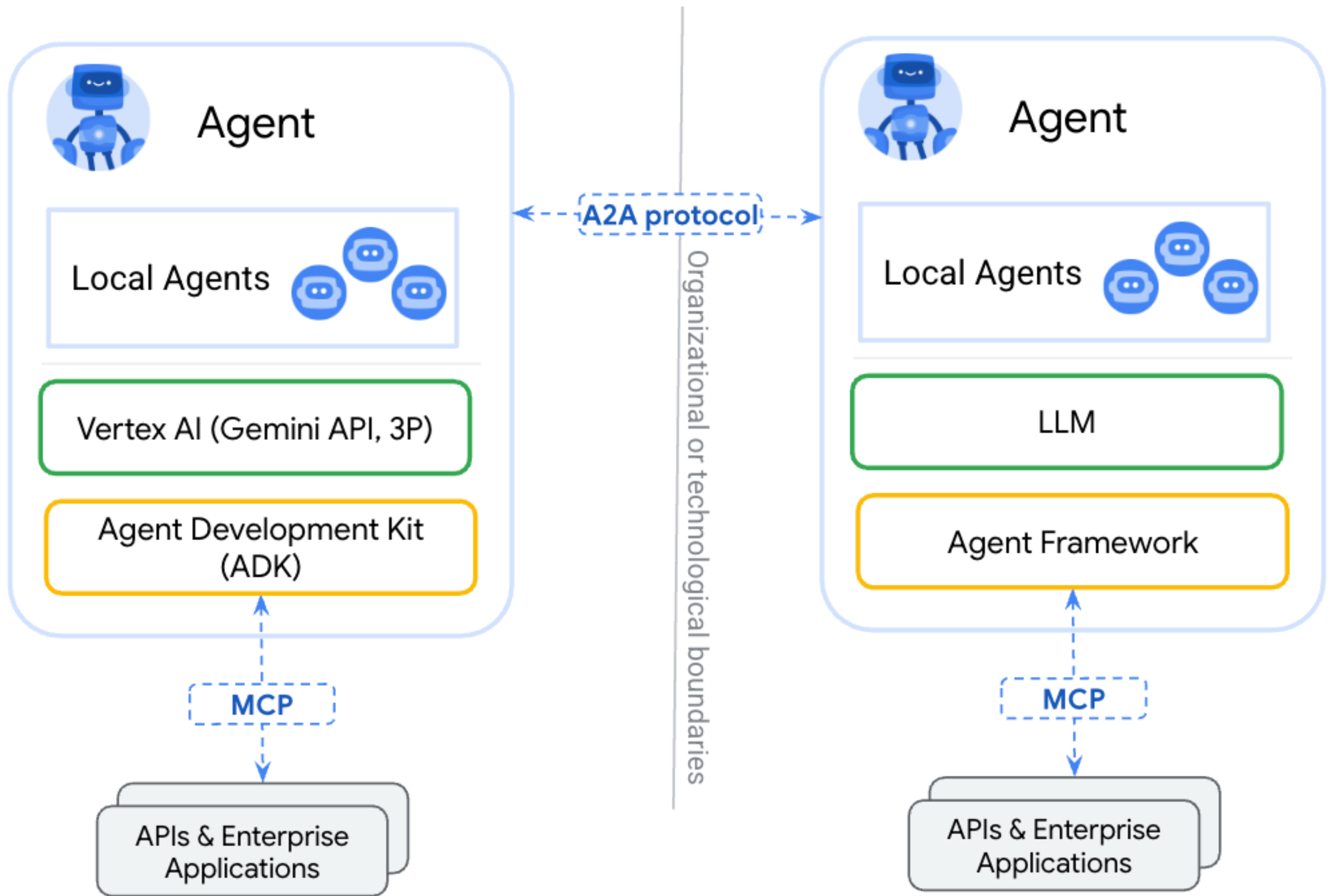
Open source

Assert
Neo4j FriendsOfApi/boilerplate
Backup-manager/symfony Guzzle Buzz nyholm/effective-interest-rate
Swap Puli LinkedIn API client
php-http/discovery NSA
PSR7 PHP-cache PHP-Translation
CacheBundle
league/geotools Mailgun
KNP Github API happyr/normal-distribution-bundle
Stampie HTTPPlug SymfonyBundleTest
MailgunBundle php-http/httpplug-bundle php-http/multipart-stream
SimpleBus integrations PHP-Geocoder
PSR HTTP clients BazingaGeocoderBundle

Model Context Protocol



Not the same as
Agent2Agent (A2A) Protocol



Your API can be ~~RESTful~~ GraphQL

MCP is evolving, but stable

Version 2025-11-25

Version 2025-06-18

Version 2025-03-26

Version 2024-11-05

Let's look at the protocol

AI Chat

MCP server

AI Chat
(Jetbrains AI)

MCP server
(Your application)

AI Chat
(Jetbrains AI)

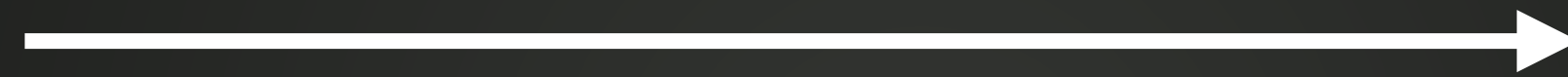
What can you do?



Version and features



Okey, good



Use tool X



Terminate



MCP server
(Your application)

JSON-RPC

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "initialize",
  "params": {
    "protocolVersion": "2024-11-05",
    "capabilities": {
      "roots": {
        "listChanged": true
      },
      "sampling": {},
      "elicitation": {}
    },
    "clientInfo": {
      "name": "ExampleClient",
      "title": "Example Client Display Name",
      "version": "1.0.0"
    }
  }
}
```

Initialize Request

“What can you do?”



```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "protocolVersion": "2024-11-05",
    "capabilities": {
      "logging": {},
      "prompts": {
        "listChanged": true
      },
      "resources": {
        "subscribe": true,
        "listChanged": true
      },
      "tools": {
        "listChanged": true
      }
    },
    "serverInfo": {
      "name": "ExampleServer",
      "title": "Example Server Display Name",
      "version": "1.0.0"
    },
    "instructions": "Optional instructions for the client"
  }
}
```

Initialize Response

“Version and features”



```
{  
  "jsonrpc": "2.0",  
  "method": "notifications/initialized"  
}
```

Initialized Notification

"Okey, good"



```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "error": {  
    "code": -32602,  
    "message": "Unsupported protocol version",  
    "data": {  
      "supported": ["2024-11-05"],  
      "requested": "1.0.0"  
    }  
  }  
}
```

Error Response



AI Chat
(Jetbrains AI)

What can you do?



Version and features



Okey, good

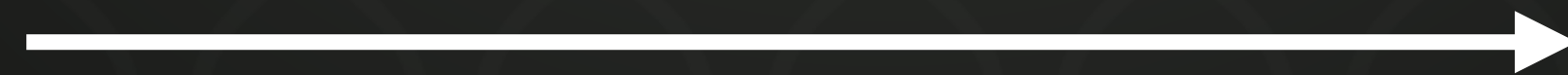


MCP server
(Your application)

Use tool X



Terminate



More of Model Context Protocol

Standard I/O

HTTP

But now...
it is getting interesting

MCP Elasmobranchiidae

MCP Features

Prompts

Pre-defined templates or instructions that guide language model interactions

Resources

Structured data or content that provides additional context to the model

Tools

Executable functions that allow models to perform actions or retrieve information

modelcontextprotocol.io

mcp/sdk

`composer require mcp/sdk`

```
#!/usr/bin/env php
<?php

require_once __DIR__ . '/vendor/autoload.php';

use Mcp\Server;
use Mcp\Server\Session\FileSessionStore;
use Mcp\Server\Transport\StdioTransport;

$server = Server::builder()
    ->setServerInfo('My demo', '1.0')
    ->setSession(new FileSessionStore(__DIR__ . '/sessions'))
    ->setDiscovery(__DIR__)
    ->build();

$result = $server->run(new StdioTransport());

exit($result);
```

```
use Mcp\Capability\Attribute\McpTool;

final class MyTool
{
    #[McpTool(name: 'addition')]
    public function addition(float $a, float $b): float
    {
        return $a + $b;
    }
}
```

```
use Mcp\Capability\Attribute\McpTool;

class KingOfSwedenTool
{
    #[McpTool(
        name: 'king-of-sweden',
        description: 'Get the name of the King of Sweden'
    )]
    public function getKing(): string
    {
        return 'Tobias';
    }
}
```

With Claude desktop

```
{
  "preferences": {
    "quickEntryShortcut": "off",
    "menuBarEnabled": false
  },
  "mcpServers": {
    "local-test": {
      "command": "php",
      "args": [
        "/full/path/to/my/server.php"
      ]
    }
  }
}
```



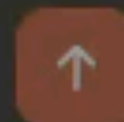
Free plan · [Upgrade](#)

🌟 Good evening, Tobias

How can I help you today?



Sonnet 4.5



Upgrade to connect your tools to Claude



How about without magic?

```
#!/usr/bin/env php
<?php

require_once __DIR__ . '/vendor/autoload.php';

use Mcp\Server;
use Mcp\Server\Session\FileSessionStore;
use Mcp\Server\Transport\StdioTransport;

$server = Server::builder()
    ->setServerInfo('My demo', '1.0')
    ->setSession(new FileSessionStore(__DIR__ . '/sessions'))
    ->setDiscovery(__DIR__)
    ->build();

$result = $server->run(new StdioTransport());

exit($result);
```

```
#!/usr/bin/env php
<?php

require_once __DIR__ . '/vendor/autoload.php';

use Mcp\Server;
use Mcp\Server\Session\FileSessionStore;
use Mcp\Server\Transport\StdioTransport;

$server = Server::builder()
    ->setServerInfo('My demo', '1.0')
    ->setSession(new FileSessionStore(__DIR__ . '/sessions'))
    ->addTool(KingOfSwedenTool::class, 'king-of-sweden')
    ->build();

$result = $server->run(new StdioTransport());

exit($result);
```


And now...
the advanced parts

AI Chat
(Jetbrains AI)

What can you do?



Version and features



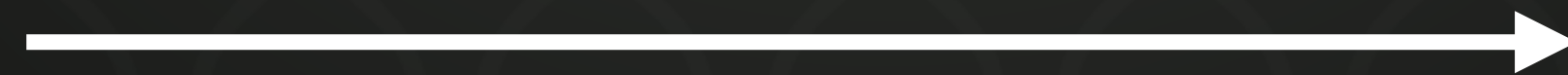
Okey, good



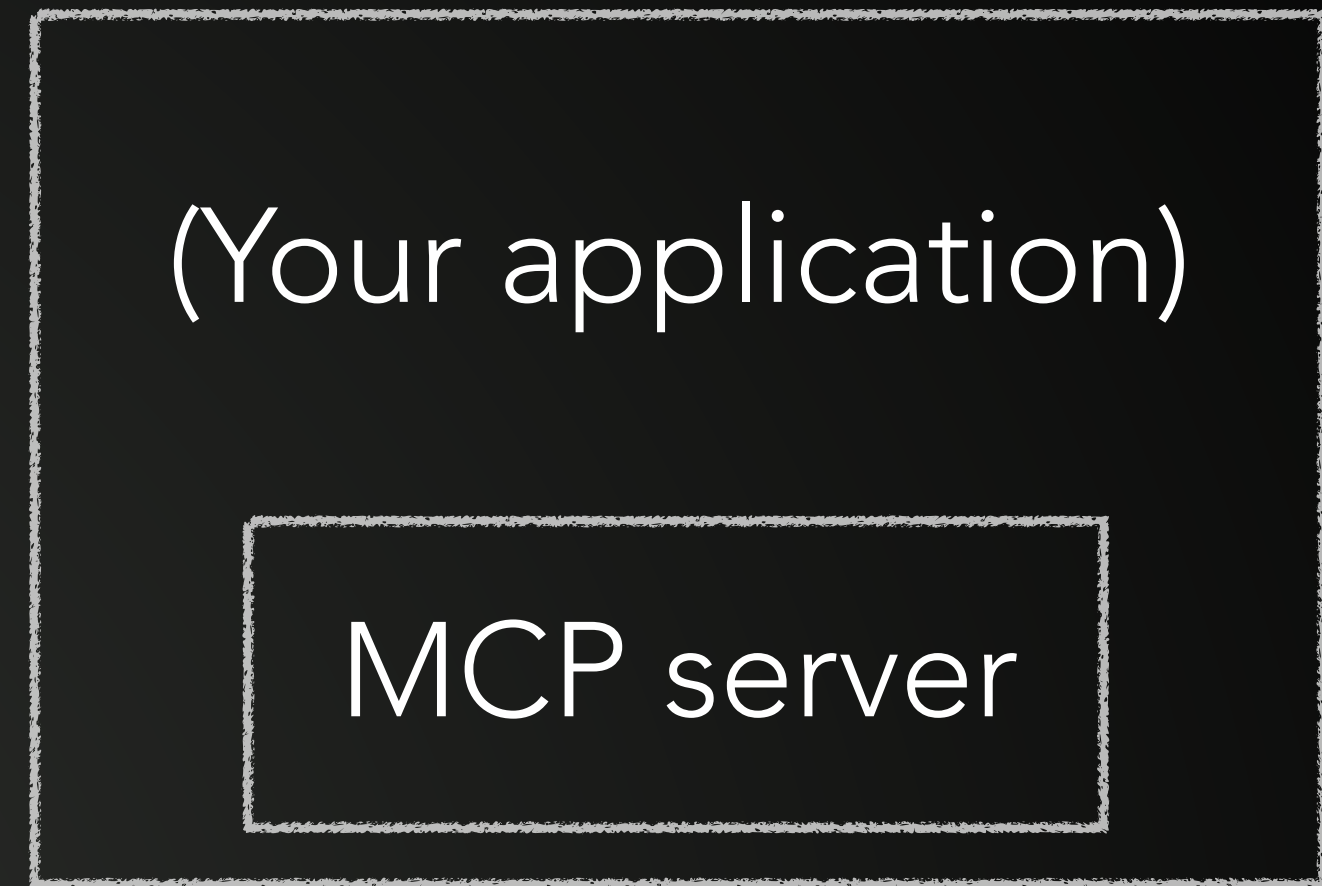
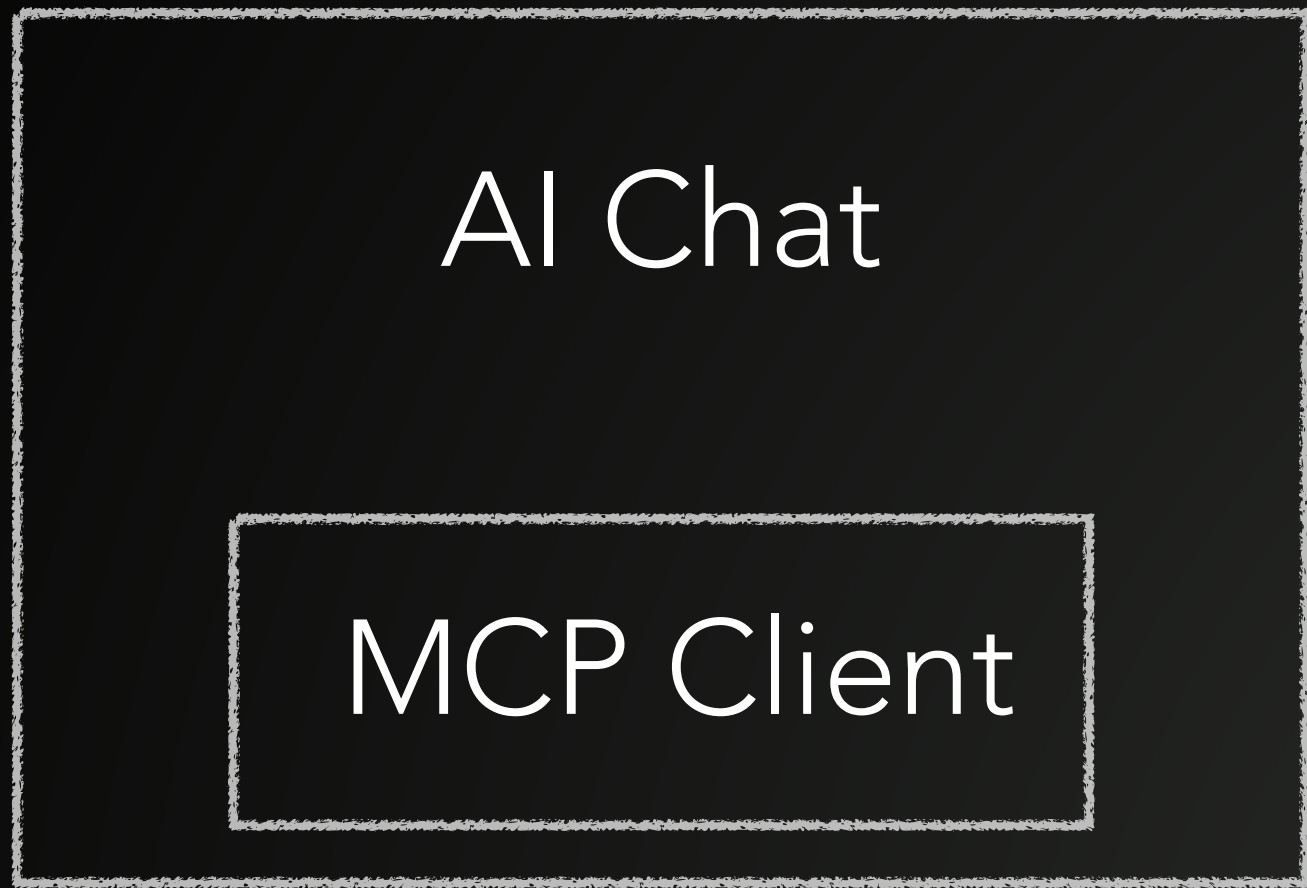
Use tool X

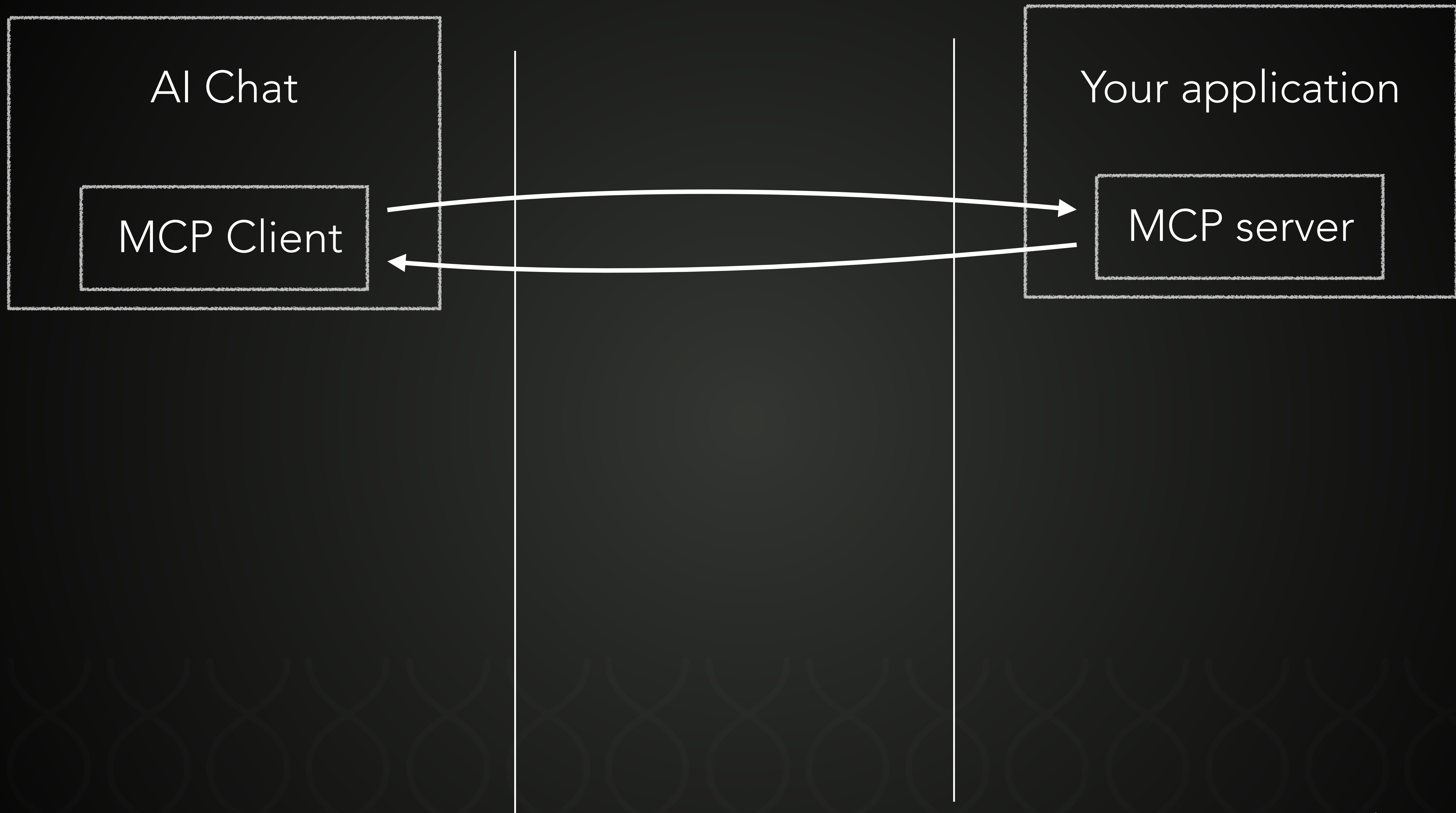


Terminate



MCP server
(Your application)





MCP Specification: OAuth 2.1

MCP PHP SDK: Authentication not supported

If we own the client: _(ツ)_/

```
composer require symfony/mcp-bundle nyholm/psr7
```

```
# config/packages/mcp.yaml
```

```
mcp:
```

```
  app: 'My Secure Server'
```

```
  version: '1.0.0'
```

```
  http:
```

```
    path: /_mcp
```

```
    session:
```

```
      store: file
```

```
      directory: '%kernel.cache_dir%/mcp-sessions'
```

```
      ttl: 3600
```

```
# config/packages/security.yaml

security:
    # ...

    firewalls:
        main:
            # normal firewall
            # ...

    access_control:
        - { path: ^/_mcp, roles: ROLE_USER }
```

```
use Mcp\Capability\Attribute\McpTool;

class KingOfSwedenTool
{

    #[McpTool(
        name: 'king-of-sweden',
        description: 'Get the name of the King of Sweden'
    )]
    public function getKing(): string
    {

        return 'Tobias';
    }
}
```

```
use Mcp\Capability\Attribute\McpTool;
use Symfony\Bundle\SecurityBundle\Security;

class KingOfSwedenTool
{
    public function __construct(
        private Security $security,
    ){}

    #[McpTool(
        name: 'king-of-sweden',
        description: 'Get the name of the King of Sweden'
    )]
    public function getKing(): string
    {
        $user = $this->security->getUser();

        return $user->getName();
    }
}
```

```
use Mcp\Capability\Attribute\McpTool;
use Symfony\Bundle\SecurityBundle\Security;
use App\Service\OrderManager;

class LastOrderStatusTool
{
    public function __construct(
        private Security $security,
        private OrderManager $orderManager,
    ){

#[McpTool(
    name: 'last-order-status',
    description: 'Get the status on latest order'
)]
    public function getLastOrder(): string
    {
        $user = $this->security->getUser();
        $order = $this->orderManager->getLastOrder($user);

        return $order->getHumanReadableStatus();
    }
}
```

How do we know if it works?

```
npx @modelcontextprotocol/inspector \  
-e DEBUG=1 -e FILE_LOG=1 \  
php examples/complex-tool-schema/server.php
```


MCP is complex

**MCP SDK makes
things simple**

**Want to do more fun
with me?**

<https://ene.ba/tobias>

