

Lab 012 : Qu'est-ce que MCP ? Anatomie du protocole

Niveau : L100 Parcours : ? MCP Durée : ~20 min ? Coût : Gratuit ? Aucun compte nécessaire

Ce que vous apprendrez

Ce qu'est MCP et pourquoi il a été créé Les trois primitives MCP : Tools, Resources et

Prompts Les options de transport MCP : stdio vs HTTP/SSE Comment explorer les serveurs MCP avec le MCP Inspector Comment MCP s'intègre avec GitHub Copilot, Foundry, Claude et d'autres agents

Introduction

Le problème que MCP résout

Avant MCP, si vous vouliez connecter un agent IA à une base de données, vous deviez écrire du code d'intégration personnalisé pour chaque combinaison agent + source de données. $N \text{ agents} \times M \text{ sources de données} = N \times M \text{ intégrations personnalisées}$.

MCP résout ce problème avec un protocole standard :

Architecture MCP

MCP a été créé par Anthropic en 2024 et rapidement adopté par Microsoft, OpenAI, Google et d'autres comme standard de l'industrie.

Les trois primitives MCP

1. ? Tools

Fonctions que le LLM peut appeler ? la primitive la plus courante.

[diagram or code sample]

Le LLM lit la description et décide quand appeler cet outil.

2. ? Resources

Données que le LLM peut lire ? comme des fichiers, des URL ou des enregistrements de base de données.

[diagram or code sample]

3. ? Prompts

Modèles de prompts réutilisables que le LLM peut utiliser.

[diagram or code sample]

Options de transport

MCP communique via deux transports :

Transport | Cas d'utilisation | Fonctionnement | stdio | Outils locaux, agents CLI | Le processus parent lance le serveur MCP en tant que processus enfant ; communique via stdin/stdout | HTTP/SSE | Serveurs distants, agents cloud | Le serveur MCP expose des points de terminaison HTTP ; l'agent se connecte via Server-Sent Events |

Dans les labs de ce hub, nous utilisons HTTP/SSE afin que les agents hébergés dans le cloud (comme Foundry Agent Service) puissent atteindre votre serveur MCP.

Flux de messages MCP

Lorsqu'un agent appelle un outil, voici ce qui se passe :

[diagram or code sample]

Utiliser le MCP Inspector

Le MCP Inspector est un outil basé sur le navigateur permettant d'explorer n'importe quel serveur MCP sans écrire de code.

tip "Essayez-le maintenant ? aucune installation nécessaire" Exécutez cette commande (nécessite Node.js) : [diagram or code sample] Ou visitez inspector.modelcontextprotocol.io dans votre navigateur.

Avec l'inspecteur, vous pouvez :

Vous connecter à n'importe quel serveur MCP (local ou distant) Parcourir les outils, ressources et prompts disponibles Appeler des outils manuellement et voir les réponses JSON

Déboguer vos propres serveurs MCP

MCP dans l'écosystème Microsoft

Produit | Support MCP | GitHub Copilot (VS Code) | ? Connecter des serveurs MCP dans .vscode/mcp.json | Microsoft Foundry Agent Service | ? Connecter des serveurs MCP comme outils d'agent | Semantic Kernel | ? Adaptateur de plugin MCP disponible | Claude Desktop | ? Support MCP natif | Azure MCP Server | ? Serveur MCP hébergé par Microsoft pour les services Azure |

Configuration MCP dans VS Code (.vscode/mcp.json)

[diagram or code sample]

Une fois configuré, GitHub Copilot dans VS Code peut appeler les outils de votre serveur MCP depuis le chat.

Résumé

MCP est un standard ouvert qui permet à n'importe quel agent IA de se connecter à n'importe quel outil ou source de données via un protocole cohérent. Il possède trois primitives (Tools, Resources, Prompts) et deux transports (stdio et HTTP/SSE). Le MCP Inspector vous permet d'explorer les serveurs sans écrire de code.

Étapes suivantes

Prêt à construire votre propre serveur MCP ?

Python : ? Lab 020 ? Construire un serveur MCP en Python C# : ? Lab 021 ? Construire un serveur MCP en C#

Modernisation de Phase 3

warning "Verified 2026-10-27" Ciblez MCP 2026-07-28. Préférez serveur stateless, discovery, SDK Tier 1 actuel et état multi round-trip explicite. Ajoutez Registry et autorisation selon le modèle de confiance.

[diagram or code sample]

Model Context Protocol specification MCP authorization MCP Registry

Kit d'accompagnement

Le build du site génère des téléchargements PDF et ZIP stables. Le ZIP contient le guide du lab et tous les fichiers d'accompagnement déclarés.

Télécharger le PDF généré Télécharger le ZIP généré

Supporting file: CHEATSHEET.md

Fichiers d'accompagnement: Lab 012 : Qu'est-ce que MCP ? Anatomie du protocole

Commencez par le parcours local ou gratuit documenté. Consignez les hypothèses et les preuves de validation. Gardez les services hébergés facultatifs sauf exigence explicite du lab.

Supporting file: README.md

Lab 012 : Qu'est-ce que MCP ? Anatomie du protocole

Utilisez ces fichiers avec le guide du lab et le kit généré.