# Agent Marketplace

**What is MCP**?

* MCP is an open protocol that standardizes how applications provide context to LLMs.
* MCP provides a standardized way to connect AI models to different data sources and tools.

**Why MCP**?

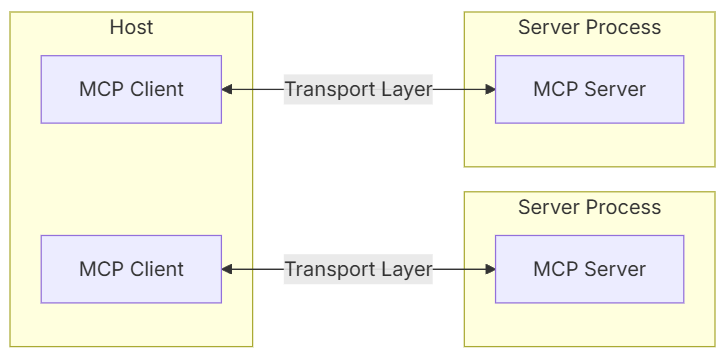
MCP helps you build agents and complex workflows on top of LLMs. LLMs frequently need to integrate with data and tools, and MCP provides:

* A growing list of pre-built integrations that your LLM can directly plug into
* The flexibility to switch between LLM providers and vendors
* Best practices for securing your data within your infrastructure

**Core Architecture**:

MCP follows a client-server architecture where:

* **Hosts** are LLM applications (like Claude Desktop or IDEs) that initiate connections
* **Clients** maintain 1:1 connection with servers, inside the host application
* **Servers** provide context, tools to clients



**Transport layer**:

The transport layer handles the actual communication between clients and servers.

MCP supports following transport mechanisms:

1. **Stdio transport**
   * Uses standard input/output for communication
   * Ideal for local processes
2. **HTTP with SSE transport**
   * Uses Server-Sent Events for server-to-client messages
   * HTTP POST for client-to-server messages

All transports use [**JSON-RPC**](https://www.jsonrpc.org/) 2.0 to exchange messages. See the [**specification**](https://modelcontextprotocol.io/specification) for detailed information about the Model Context Protocol message format.

**MCP Tools**:

* Enable LLMs to perform actions through your server
* Tools are a powerful primitive in the Model Context Protocol (MCP) that enable servers to expose executable functionality to clients.
* Through tools, LLMs can interact with external systems, perform computations, and take actions in the real world.

**Marketplace**:

**Agentic architectures** are redefining how systems are built—modular, autonomous, and dynamically orchestrated.

At the heart of this evolution is the **Agent Marketplace**—a centralized registry where **MCP servers** and **LLM-powered agents** are onboarded and made discoverable. Each agent exposes a specific skill—like document parsing, classification, or RAG operations—wrapped in a standardized interface.

This standardization enables true composability. Developers can seamlessly plug agents into pipelines, reuse them across projects, and scale orchestration dynamically.

The marketplace acts as a bridge between standalone agent logic and production workflows. Agents onboarded here can be versioned, governed, and seamlessly injected into pipelines via the **Workflow Designer**—our low-code canvas to visually model multi-agent orchestration, trigger conditions, and data flows.

The marketplace integrates directly with the **Workflow Designer**—a low-code canvas where teams can visually model multi-agent workflows, control data routing, define triggers, and deploy solutions—all without boilerplate glue code.