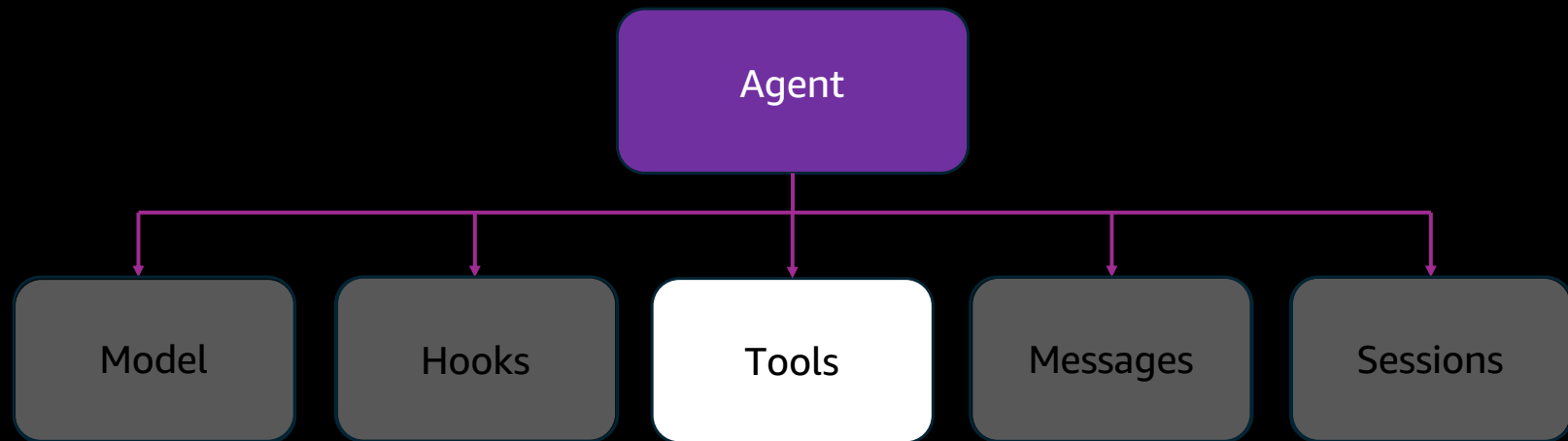


04 Tools and MCP Integration

Agenda

- Use pre-built tools from the Strands toolkit
- Create custom tools with the `@tool` decorator
- Integrate MCP servers for thousands of additional tools
- Build agents that can create their own tools

Strands Agent() Architecture

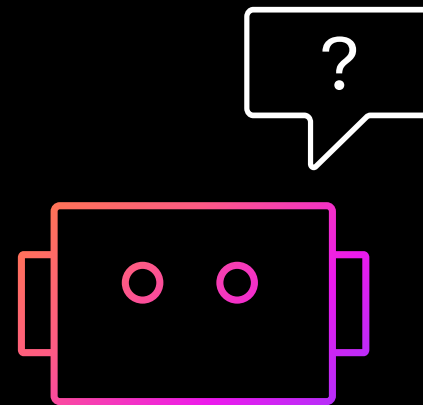


"Think of an Agent as an orchestrator that coordinates between your model, tools, and custom logic through hooks."

Connecting Tools to Agents

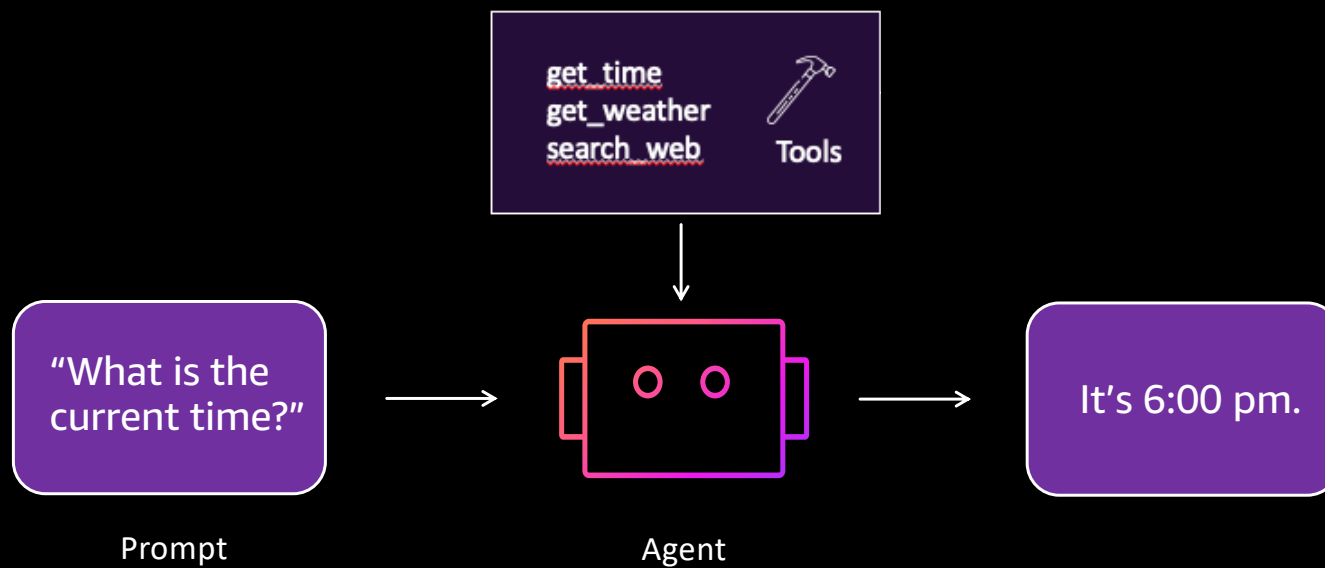
"What is the
current time?"

Prompt



Foundation Model

Connecting Tools to Agents



Pre-built Tools

```
from strands import Agent
from strands_tools import current_time, http_request

# Create an agent with default settings
agent = Agent(
    system_prompt="""You are an assistant""",
    tools=[current_time, http_request]
)

# Test the agent with a query that might benefit from tools
query = """
Answer the following questions:
1. What is the current time in UTC?
2. Based on Wikipedia, which CS concept can be traced back to Paul Bachmann?
"""

response = agent(query)
```

Custom Tools

```
from strands import Agent, tool

@tool
def hello_world() -> str:
    """Function to say the magic word"""
    return "Hello, world!"

def main():
    agent = Agent(tools=[hello_world])
    agent("What is the magic word ?")

if __name__ == "__main__":
    main()
```

Model Context Protocol (MCP)

```
from strands import Agent
from strands.tools.mcp import MCPClient

from strands_tools import current_time
from mcp import stdio_client, StdioServerParameters

# MCP Server
mcp_client = MCPClient(
    lambda: stdio_client(
        StdioServerParameters(
            command="uvx",
            args=["aws-labs.aws-documentation-mcp-server@latest"]
        )
    )
)

with mcp_client:
    # Create an agent with default settings
    agent = Agent(
        tools=[current_time] + mcp_client.list_tools_sync()
    )

    agent("List your tools.")
```


Can agents build
own tools?

Self-extending agents

Strands allows agents to watch tools directory and immediately load tools to be used.

```
from strands import Agent
from strands_tools import editor

agent = Agent(
    system_prompt="""
    Goal:
    - Create a python tool under cwd()/tools/*.py using given python tool decorator.
    - I have hot-reloading abilities, after writing the file, I can use it immediately.

    Building tools:

    from strands import tool

    @tool
    def name(name: str, description: str) -> str:
        """
        Create a tool under cwd()/tools/*.py.
        """
        return ""

    """,
    tools=[editor],
    load_tools_from_directory=True
)

while True:
    query = input("\n> ").strip()

    if query == "":
        continue

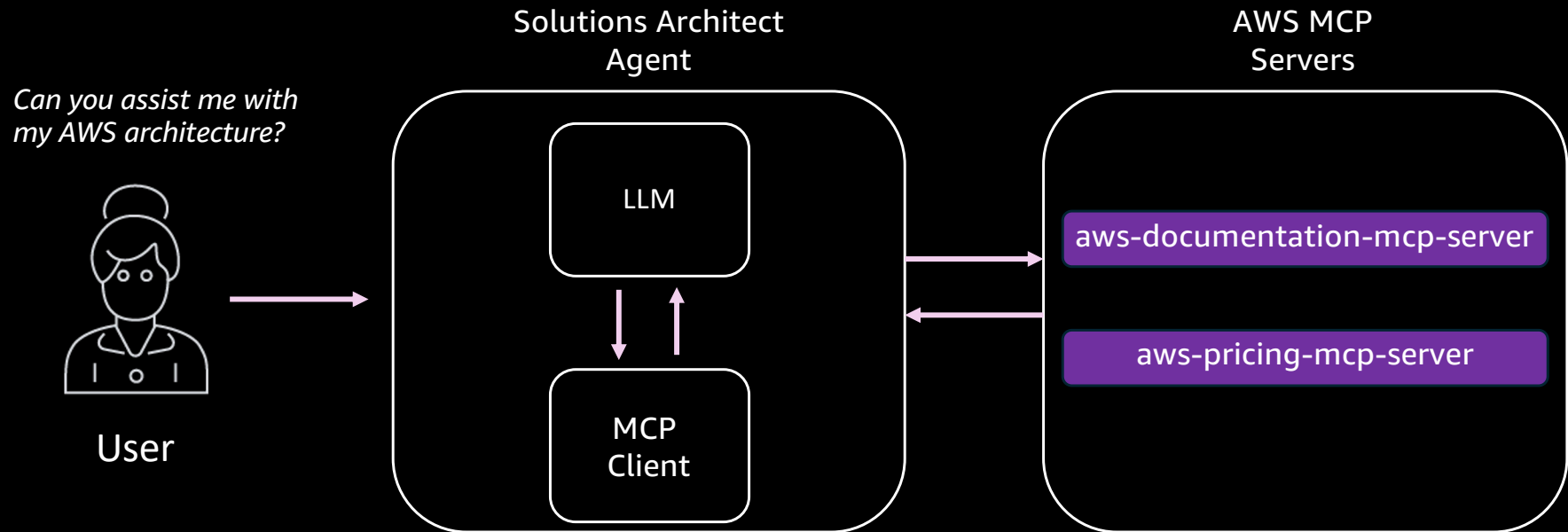
    if query.lower() in ["exit", "quit"]:
        print("\nGoodbye!")
        break

    agent(query)
```



Demo

Example 1: MCP



Example 2: Meta-Tooling

*Use a tool to calculate
compound interest.*



User



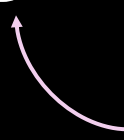
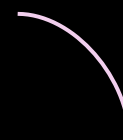
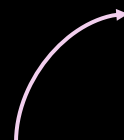
Agent

LLM

Write Tool

Load Tool

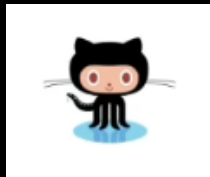
Use Tool



Key Takeaways

- Tool use in Strands
- Creating custom tools
- Integrating MCP
- Agents building their own tools

Where's the code?



<https://s12d.com/advanced-strands-agents>