Building an Ekahau MCP Server as a First Layer of Design Validation



WLPC Prague 2026

Christos Hassis
Senior Consulting Engineer, @cchassis
chassis@cisco.com

2026-10-14

Agenda

00 What is MPC anyways?

01 Managing Ekahau at scale

02 AP placement as context

03 Advanced usecases

04 Ekahau for the non-technical

05 Limitations & future steps

This presentation reflects the personal views of the presenter and is provided for informational purposes only.
It does not constitute the official position of Cisco Systems Inc. or any of its affiliates, and should not be relied upon as legal, financial, or professional advice.
Cisco Systems Inc. makes no representations as to the accuracy or completeness of the content.

© 2026 Cisco and/or its affiliates. All rights reserved.

Model Context Protocol (MCP)

MCP (Model Context Protocol) is an open-source standard for connecting AI applications to python scripts.

MCP follows a client-server architecture where an MCP host – an AI application like VS Code or OpenCode – establishes connections to one or more MCP servers.

Local MCP servers that use the STDIO transport typically serve a single MCP client, whereas remote MCP servers that use the Streamable HTTP transport will typically serve many MCP clients.

© 2026 Cisco and/or its affiliates. All rights reserved.

How to turn existing scripts in to MCP tools

```python
from mcp.server import MCPServer

mcp = MCPServer("CalculatorServer")


@mcp.tool()
def add(a: int, b: int) -> int:
    """Add two integers and return their sum."""
    return a + b
```

By importing the MCPServer python library, we're able to create a new MCP server that will accept connections from AI applications and expose tools.

Those tools will in turn act as an interface for existing functions that always act deterministically.

© 2026 Cisco and/or its affiliates. All rights reserved.

Offloading user intent translation to LLMs

```python
from mcp.server.mcpserver import MCPServer

mcp = MCPServer("esxevo_mcp", version="0.1.0")

@mcp.tool()
def esx_project_summary(source_path: str):
    """Return metadata from one or more Ekahau projects."""
    projects = resolve_esx_sources(source_path)
    summaries = []

    for project_path in projects:
        project_data = read_json(project_path, "project.json")
        summaries.append({
            "source": project_path.name,
            "summary": extract_project_metadata(project_data),
        })

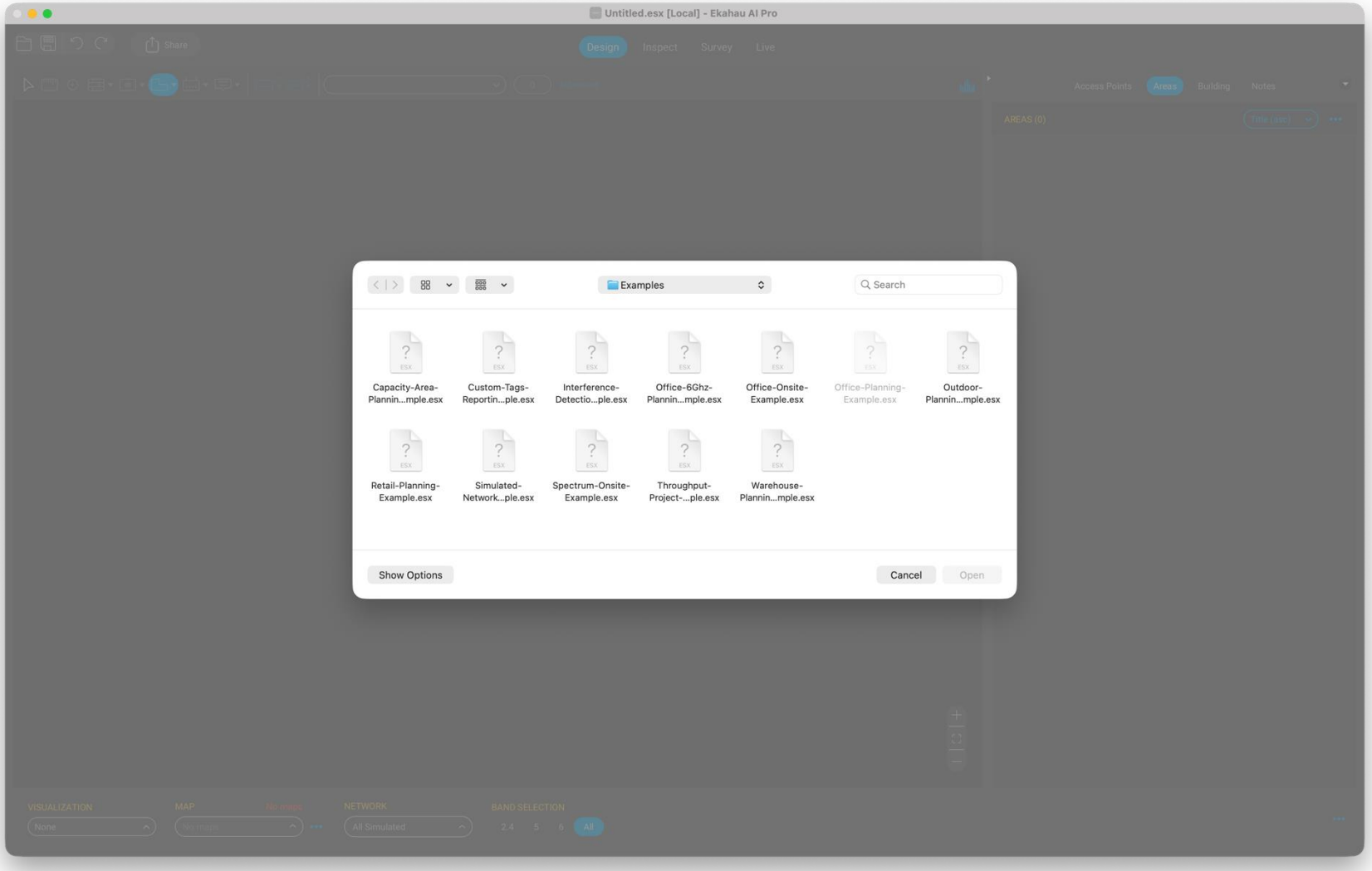
    return summaries[0] if len(summaries) == 1 else summaries
```

Memorizing correct CLI syntax poses a high barrier of entry for most network engineers, risking tool adoption.

By deploying LLMs we offload translation of intent into the correct tool call (AI agents).

© 2026 Cisco and/or its affiliates. All rights reserved.

Working on multiple .esx files in parallel



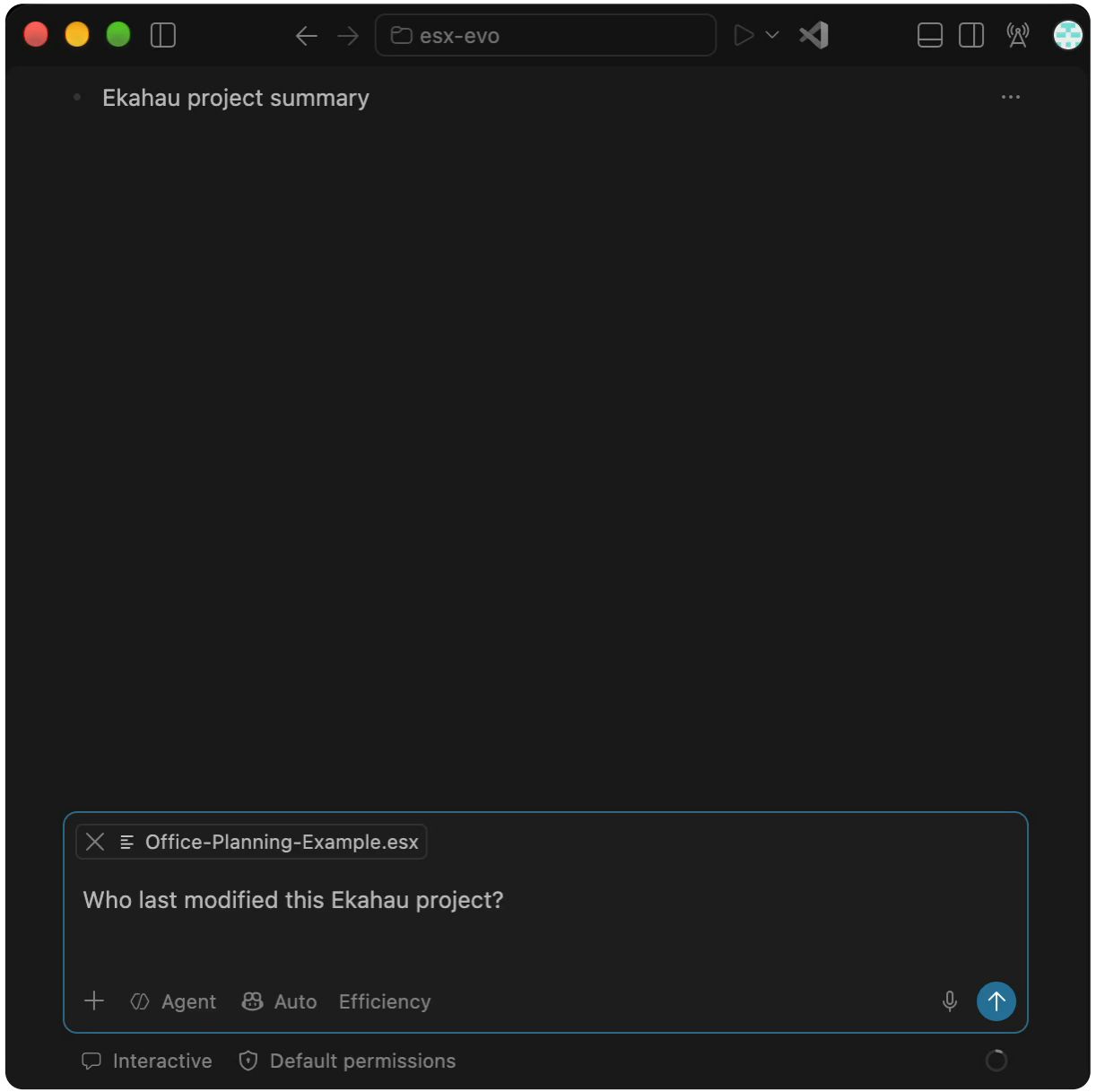
Humans fundamentally work in a single-threaded fashion.

By leveraging the ability to spawn multiple sub-agents, routine tasks can be offloaded to LLMs.

© 2026 Cisco and/or its affiliates. All rights reserved. Total credit count: 0.0

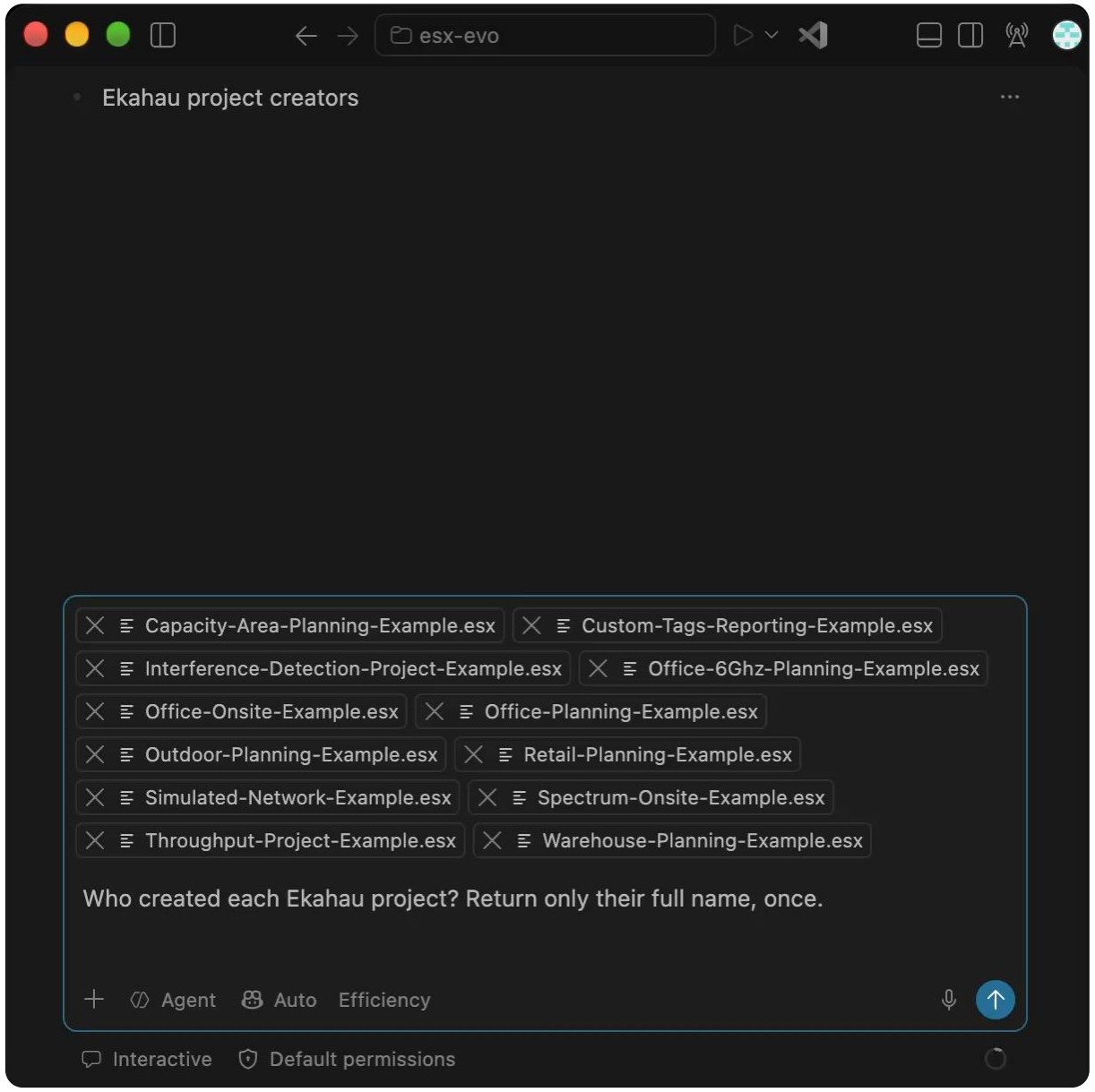
esx_project_summary


```
{
"source": "Office-Planning-Example.esx",
"summary": {
"schemaVersion": "1.7.0",
"name": "Office-Planning-Example",
"title": "Office-Planning-Example",
"customer": "",
"location": "",
"responsiblePerson": "",
"noteIds": [],
"history": {
"modifiedAt": "2022-11-08T08:19:12.087Z",
"modifiedBy": "Mikko Lauronen",
"createdAt": "2022-11-08T07:55:55.651Z",
"createdBy": "Mikko Lauronen"
},
"tags": []
}
}
```

© 2026 Cisco and/or its affiliates. All rights reserved. Total credit count: 0.4

But what about scale?



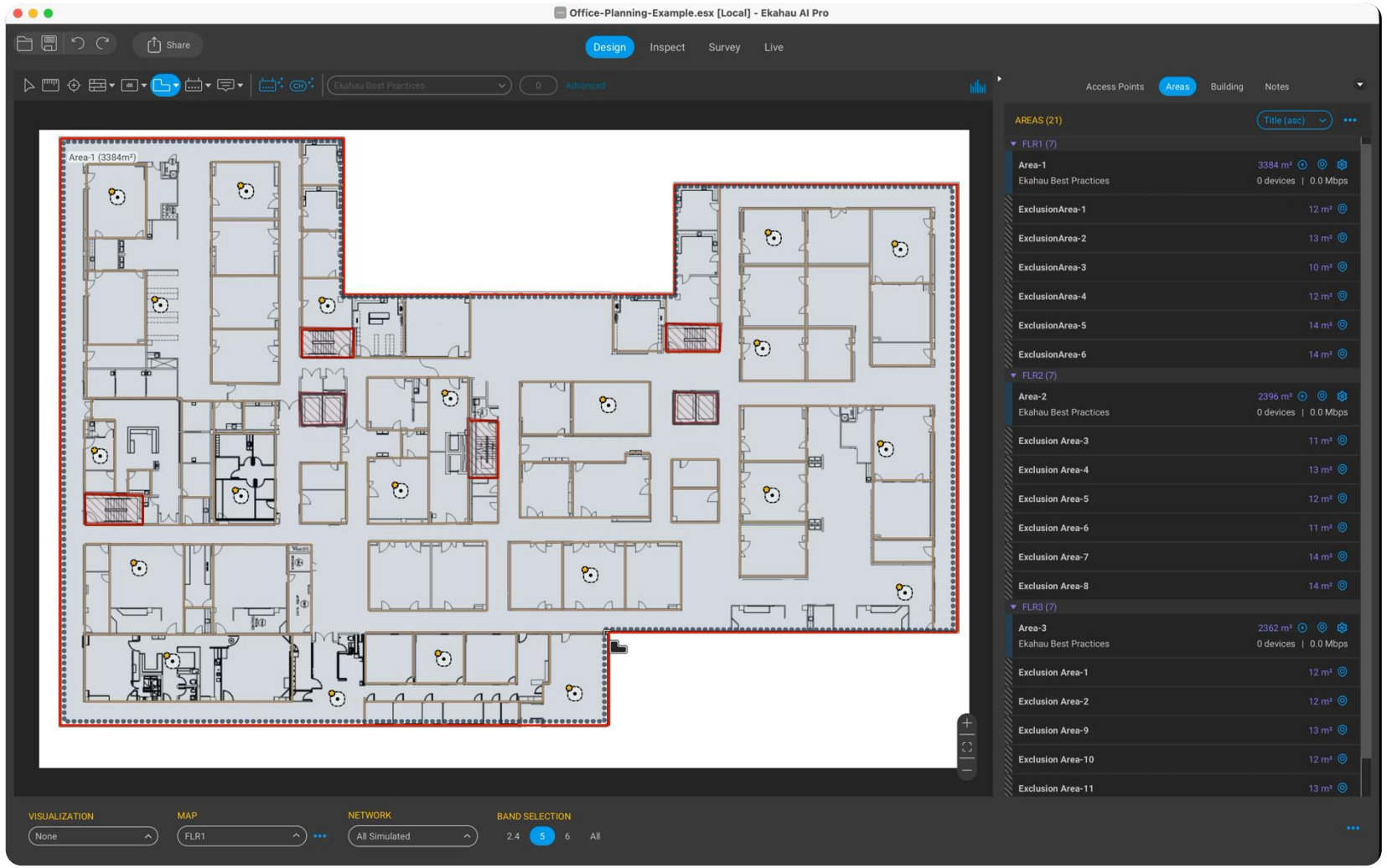
Similar to a logical loop, an agent will recursively use tools until the task is accomplished.

As many files are attached to the chat window as context, will result to an equal amount of tool calls.

No extra logic is used to eliminate duplicates in this case, but this is easily handled by the LLM itself.

© 2026 Cisco and/or its affiliates. All rights reserved. Total credit count: 0.5

Metadata is cool, but what about more context?

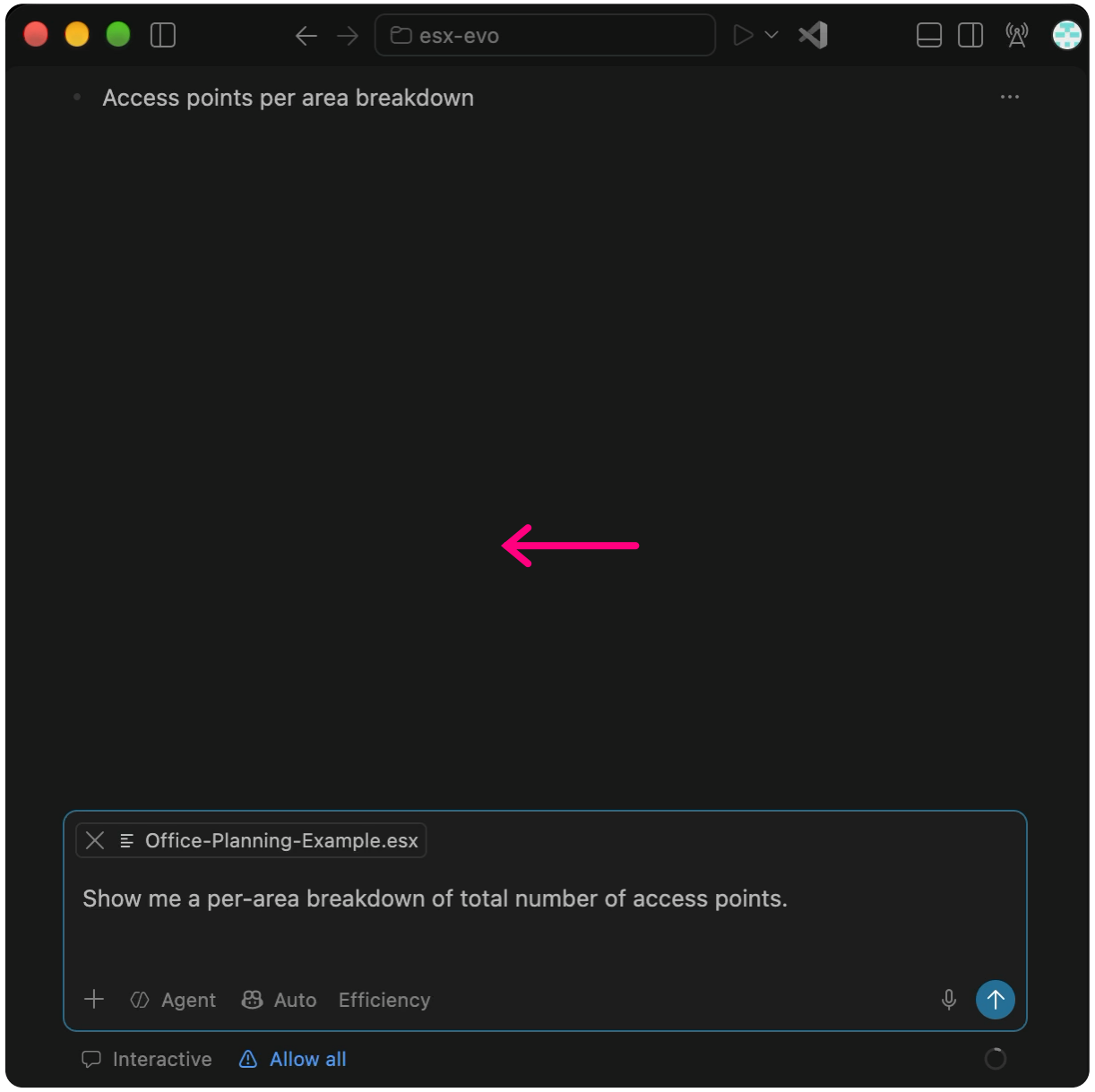


An important enabler for the next examples is the ability for a script to locate access points relevant to requirement areas.

From this we can derive useful metrics such as average distance between nodes, density per square unit, adherence to deployment best practices, etc.

© 2026 Cisco and/or its affiliates. All rights reserved.

Letting the agent fill in the gaps

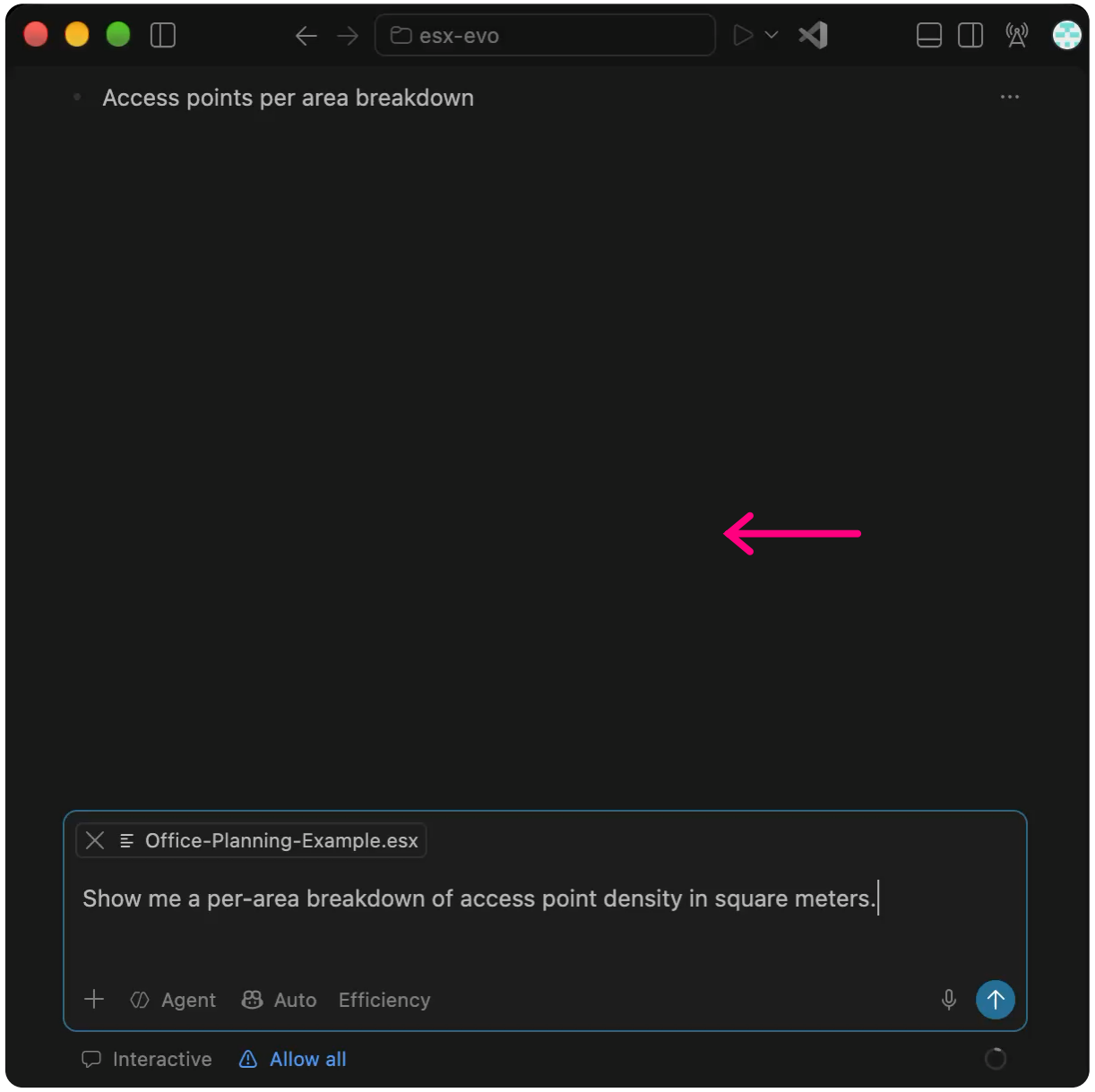


The MCP server does not provide an explicit “access_point_per_area” tool to avoid overlap.

The agent instead makes the decision to call “calculate_ap_density” since it already includes this metric and further filters its output.

© 2026 Cisco and/or its affiliates. All rights reserved. Total credit count: 0.6

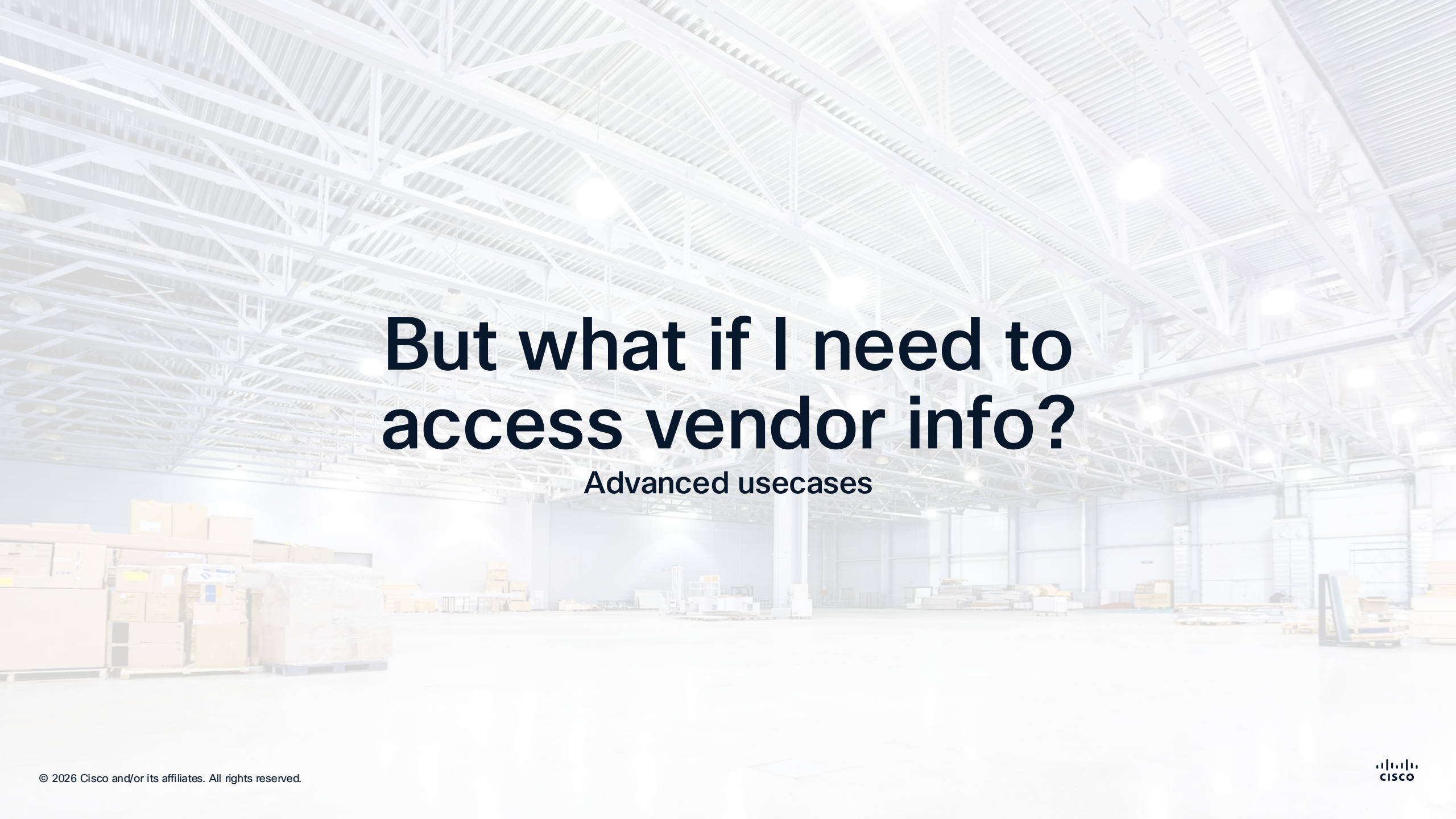
Using this toolkit to spot anomalies in our design



By treating interactions with AI agents as another level of abstraction, we're able to uncover potential outliers that warrant a closer look.

This functionality aims to augment data points provided by wireless design tools rather than replace them and speed up design decisions.

© 2026 Cisco and/or its affiliates. All rights reserved. Total credit count: 0.8



But what if I need to access vendor info?

Advanced usecases

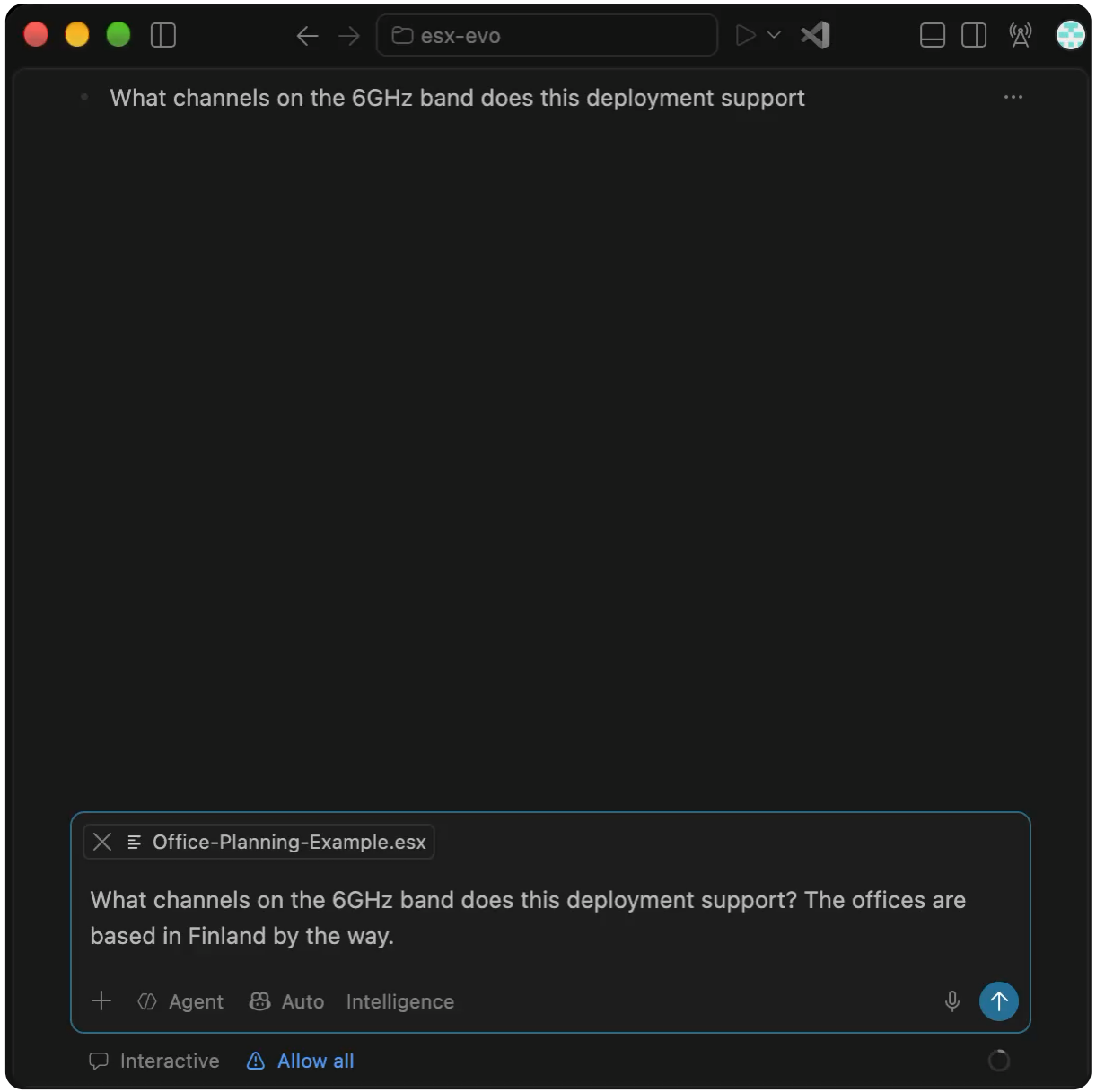
© 2026 Cisco and/or its affiliates. All rights reserved.

Integrating with apchannels.cisco.com

CW9166I Finland, On Prem, Indoor

2.4 GHz	5 GHz	6 GHz Low Power Indoor	6 GHz Standard Power

	Channel	Frequency (MHz)	Supported Since Software Version	Supported Until Software Version
UNII-5	1	5955	17.15.1	-
	5	5975	17.15.1	-
	9	5995	17.15.1	-
	13	6015	17.15.1	-
	17	6035	17.15.1	-
	21	6055	17.15.1	-
	25	6075	17.15.1	-
	29	6095	17.15.1	-
	33	6115	17.15.1	-
	37	6135	17.15.1	-
	41	6155	17.15.1	-
	45	6175	17.15.1	-
	49	6195	17.15.1	-
	53	6215	17.15.1	-
	57	6235	17.15.1	-
	61	6255	17.15.1	-
	65	6275	17.15.1	-
	69	6295	17.15.1	-
	73	6315	17.15.1	-
	77	6335	17.15.1	-
	81	6355	17.15.1	-
	85	6375	17.15.1	-
	89	6395	17.15.1	-
	93	6415	17.15.1	-



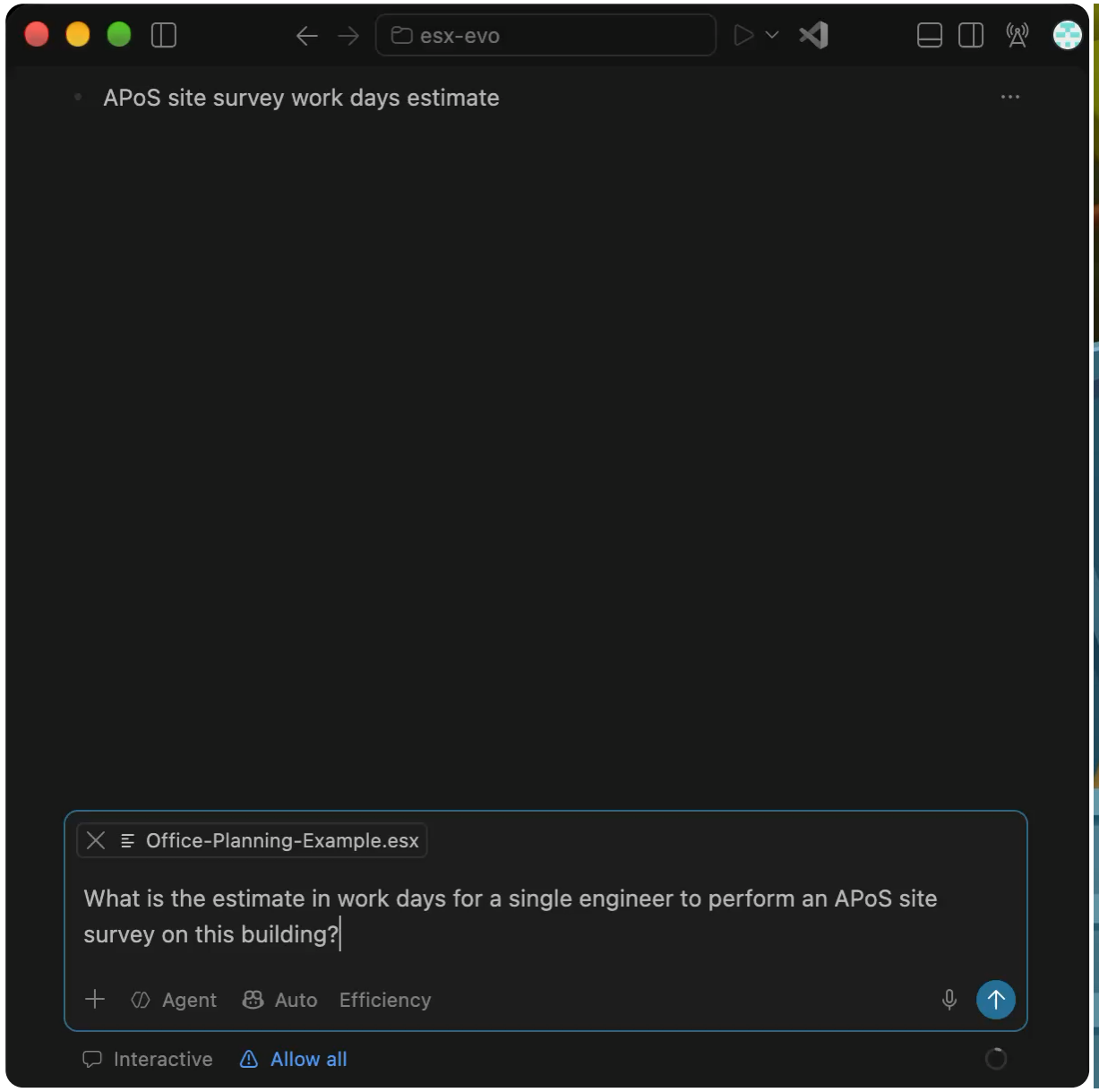
© 2026 Cisco and/or its affiliates. All rights reserved. Total credit count: 1.4



But what if I need to talk to my PM?

Ekahau MCP for the non-technical

© 2026 Cisco and/or its affiliates. All rights reserved.





© 2026 Cisco and/or its affiliates. All rights reserved. Total credit count: 1.6

What is important to remember

Human-in-the-loop

Despite the impressive capabilities of LLMs especially when augmented by purpose-build tools, human expertise is still very much necessary. Wireless design remains extremely context-heavy.

Access to data is critical

The move to cloud services without corresponding API access severely limits cool usecases like this. Hamina recently introduced OpenIntent project export* support which can be retrofitted to this MCP project.

Local models can easily compete

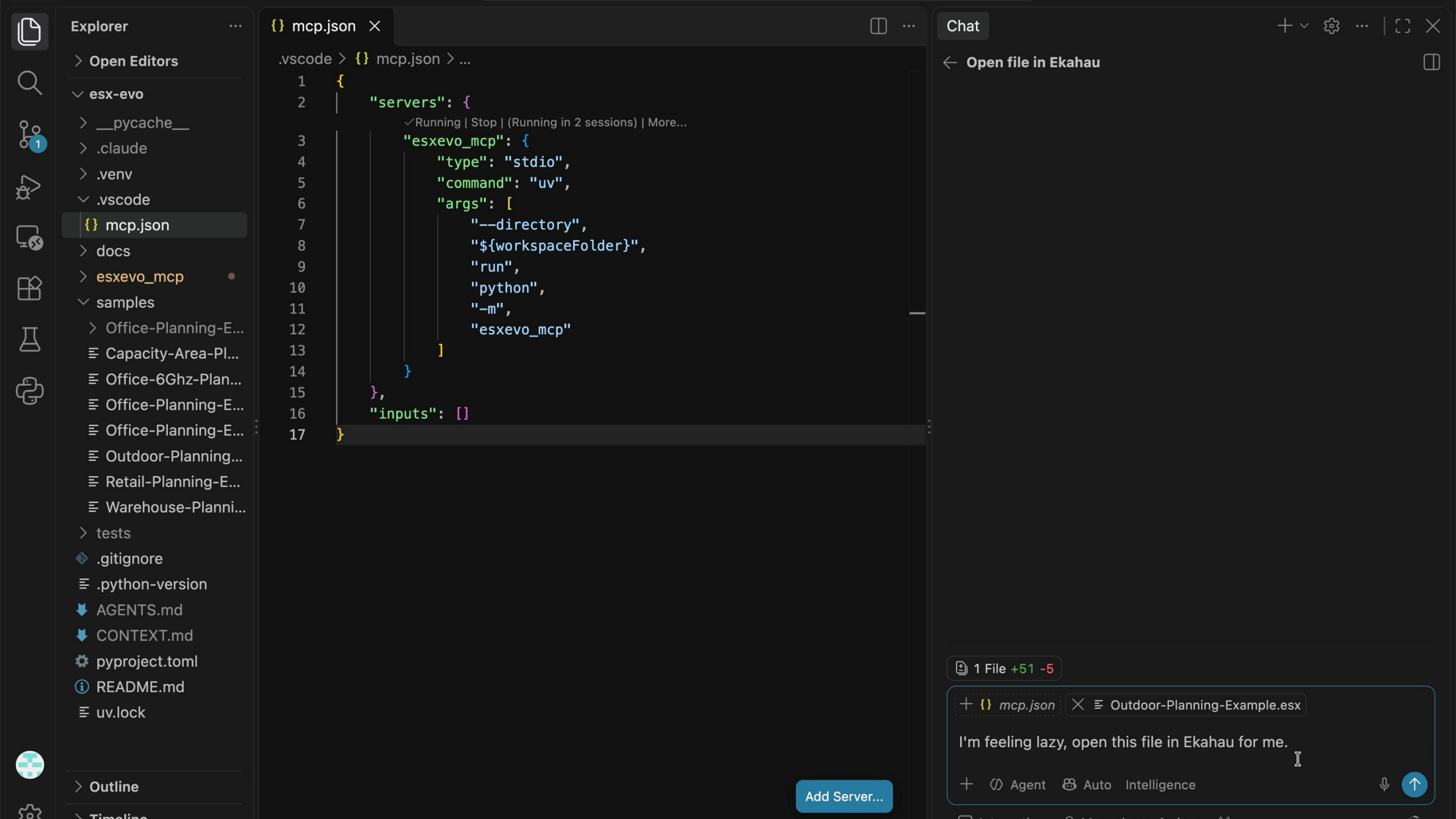
Credit usage when in Copilot “Efficiency” mode totaled just $0.02. Local LLMs provide equivalent performance without the need for internet access.

© 2026 Cisco and/or its affiliates. All rights reserved.

Future steps

- Further refinement of the underlying scripts supporting more accurate tool calculations.
- Evangelizing responsible LLM usage across the industry stands to benefit everyone.
- The move towards OpenIntent will further abstract such usecases regardless of tool vendor.
- v3.0 aims to enable file edits by the MCP tools, but further work needs to be done to ensure readiness.
- Open-sourcing this tool can help integrate more wireless vendors' best practices and making WiFi design a little less complicated.

© 2026 Cisco and/or its affiliates. All rights reserved.


Explorer
Open Editors
esx-evo
__pycache__
.claude
.venv
.vscode
mcp.json
docs
esxevo_mcp
samples
Office-Planning-E...
Capacity-Area-Pl...
Office-6Ghz-Plan...
Office-Planning-E...
Office-Planning-E...
Outdoor-Planning...
Retail-Planning-E...
Warehouse-Planni...
tests
.gitignore
.python-version
AGENTS.md
CONTEXT.md
pyproject.toml
README.md
uv.lock
Outline
Timeline
mcp.json
.vscode > mcp.json > ...
{
"servers": {
Running | Stop | (Running in 2 sessions) | More...
"esxevo_mcp": {
"type": "stdio",
"command": "uv",
"args": [
"--directory",
"${workspaceFolder}",
"run",
"python",
"-m",
"esxevo_mcp"
]
}
},
"inputs": []
}
Add Server...
Chat
Open file in Ekahau
1 File +51 -5
mcp.json
Outdoor-Planning-Example.esx
I'm feeling lazy, open this file in Ekahau for me.
Agent
Auto
Intelligence

Thank you



Christos Hassis

CCNP, CMSS | Automating big problems away