

★ Member-only story

Claude Skills Might Quietly Replace MCP

3 min read · 16 hours ago



Algo Insights

Follow



Listen



Share



More

I've been experimenting with Claude's new *Skill* feature lately — and honestly, it feels quite impactful. Larger than MCP, even.

If you've ever attempted to extend Claude with MCP (Model Context Protocol), you know it's powerful but also... cumbersome—too many tokens, too many tool descriptions, too much mental overhead.

Skills fix that — in a surprisingly elegant way.



So what's a “Skill,” exactly?

In simple terms, **skill** is simply a combination of a prompt and some tools or files Claude can access when needed.

Like this:

```
Skill ≈ prompt + tools/assets
```

And the setup? Dead simple.

```
.claude/  
└─ skills/  
    └─ [skill-name]/  
        └─ SKILL.md  
        └─ (maybe a few helper scripts or prompts)
```

That `SKILL.md` file is where the magic lives.

It usually begins with a small YAML header indicating when Claude *should* use the skill, followed by some markdown instructions explaining *how* to apply it — but here's the clever part:

Claude only loads the full instructions **when it decides to use that skill**. No wasted tokens. No overstuffed context window.

. . .

Why it's smarter than MCP

MCP was like giving Claude a whole toolbox — with every hammer, screwdriver, and wrench — even if he only needed a single nail.

Each MCP bundle adds thousands of tokens because it needs to describe every tool. And unless you guide Claude on which one to use and when, it often gets stuck.

Skills turn that upside down.

Only relevant information is loaded, only when needed.

It's sleek and quick.

. . .

Example

There's a "shadcn MCP" circulating that includes items like:

```
- get_project_registries
- list_items_in_registries
- search_items_in_registries
- view_items_in_registries
- get_item_examples_from_registries
- get_add_command_for_items
- get_audit_checklist
```

That's around 4,200 **tokens** to describe the tools.

Now picture transforming all that into a single skill:

```
.claude/
├── skills/
│   ├── shadcn-helper/
│   │   ├── SKILL.md
│   │   └── helper_functions.py
```

Your `SKILL.md` could just say:

```
name: shadcn-helper
description: Helps with project registry searches and audits.
use_when: "Working on shadcn UI or registry management."
```

Then a few markdown lines explaining the workflow to Claude (such as “search items first, then get examples, then add if needed”).

Done. Fewer than 70 tokens. Much cleaner.

. . .

The fun one — “slack-gif-creator”

One of Claude’s demo skills is a small script called **slack-gif-creator**. It simply creates a GIF for any topic — but it’s a perfect example of how Skills function.

The entire description is very brief. But once it’s triggered, Claude executes a workflow — write some Python, call a helper function, and generate a GIF.

Something like this:

```
from PIL import Image, ImageDraw

def create_gif(text, frames=10):
    imgs = []
    for i in range(frames):
        img = Image.new("RGB", (200, 200), (255, 255, 255))
        draw = ImageDraw.Draw(img)
        draw.text((50, 100), text, fill=(0, 0, 0))
        imgs.append(img)
    imgs[0].save("out.gif", save_all=True, append_images=imgs[1:], duration=100
```

Nothing fancy — but it works. And since it’s all contained within a Skill, you can replace or improve that function without affecting the rest of your setup.

. . .

Build your own

This is where it becomes interesting.

You can create Skills for your **own codebase** — essentially building your own “developer knowledge base.”

```
.claude/  
└─ skills/  
    ├── frontend/  
    ├── backend/  
    └─ database/
```

Each folder can contain prompts, helper scripts, or notes for Claude to follow.

Like:

- How to name components so they fit your design system,
- how to migrate a database safely
- or how to add a new route without breaking anything

You're essentially explaining to Claude *how your team operates*.

. . .

Where this is going

My gut? MCP might fade out over time.

Because a Skill can load MCP tools programmatically if it wants to. So why bother with the whole bulky structure?

Skills are lightweight, modular, and efficient.

And they actually make Claude feel like it “knows” your project.

. . .

If MCP was a Swiss Army knife with every blade open, a Skill is the same knife — but it only flips open what you need, when you need it.

Simple. Smart.

Claude

Anthropic Claude

Mcp Server

AI

Ai Agent



Follow

Written by Algo Insights

3.4K followers · 6 following

Algo Insights: Unlocking the Future of Trading with Code, Data, and Intelligence.
https://linktr.ee/Algo_Insights


No responses yet



Bgerby

What are your thoughts?

More from Algo Insights


 In Coding Nexus by Algo Insights

4 Open-Source Tools That Made Me Rethink My Dev Setup

I've been coding for a while now. Most of the tools we use every day... they've been the same for years. Editors, browsers, frameworks. Just...

★ Sep 3 🤝 450 💬 9

🔖⁺ ⋮

 In Coding Nexus by Algo Insights

oLLM: How I Ran an 80B Model on My 8GB GPU

I didn't expect this. Running a 160GB model on a card with 8GB VRAM feels impossible.



Sep 30



510



7



In Coding Nexus by Algo Insights

How to Build Your Own AI Rig for Running Local LLMs (Gemma, Mistral, Qwen, GPT-OSS and Llama)

About three months ago, I realised I was utterly dependent on companies that didn't care about anything except power, Money, and control.



Oct 8



350



12



In Coding Nexus by Algo Insights

DSPy: Stop Prompting, Start Programming Your AI


If you've ever spent hours tweaking prompts, you know how frustrating it can be.

✦ Oct 1 🖱 247 💬 6



See all from Algo Insights

Recommended from Medium


 Riccardo Tartaglia

5 Essential MCP Servers Every Developer Should Know

I've been experimenting with Model Context Protocol servers for a few months now, and I have to say, they've changed the way I work.

Oct 11 🖱 22 💬 3



 aakash

Perplexity Just Unleashed 10 FREE AI Agents That Do Your Entire Job (The “Comet” Shortcut)

Stop what you are doing. The era of the simple AI search engine is officially over.

★ Oct 7 🖱️ 693 💬 24



 In Dare To Be Better by Max Petrusenko

Claude Skills: The \$3 Automation Secret That’s Making Enterprise Teams Look Like Wizards

How a simple folder is replacing \$50K consultants and saving companies literal days of work



6d ago



191



3



In nginity by Reza Rezvani

Master Claude Code “Skills” Tool to transform repetitive AI prompts into permanent, executable...

Discover how the Anthropic’s new tool for Claude Code called “Skills” transform AI Coding assistant from a generic assistant into your...



6d ago



94



2





Manojkumar Vadivel

The .claude Folder: A 10-Minute Setup That Makes AI Code Smarter

If you're new to Claude Code, it's a powerful AI coding agent that helps you write, refactor, and understand code faster. This article...

Sep 15

👏 57

💬 2



In The Context Layer by Jannis

Claude's New "Skills" Show How Anthropic Is Layering Intelligence on Top of MCP



5d ago

👏 242

💬 1



See more recommendations