Why We Build on Claude Code, Not Cowork

It comes down to one thing: who controls the context.

If you're a knowledge worker — a professional who wants AI to *create* your work and multiply your impact, not just answer the occasional question — the tool you build on is not a detail. It quietly sets a ceiling on how good your outputs can ever get. That's why we build on Claude Code. Cowork — the friendlier-looking option — is quicker to get started with, but easier to begin isn't the same as better.

To see why, we have to start with what's actually changed.

WHAT'S ACTUALLY CHANGED

It's not that AI can do knowledge work — ChatGPT showed us that years ago. It's that AI can now do **all** of it. Every output you'd otherwise create yourself — the email, the brief, the analysis, the document, the proposal — AI can produce, under your direction, supervision and quality control. Not autocomplete. Not the occasional assist. All of it. You direct, you decide, you approve.

The effect is hard to overstate. People who adopt this way of working produce multiples more — measured from their real work, not a survey — while, unusually, feeling *less* drained, because they've shifted from making everything to directing and refining it. And it compounds: every week is better than the last.

But here's the part almost everyone misses. Hardly any of this is about the AI model. It's about one thing.

THE ONE THING THAT MATTERS: CONTEXT

Imagine a painter who mixed every colour together on their palette. Red, blue, yellow, green — all swirled into a muddy brown. That's raw AI before you give it context. It has been trained on the whole internet — everything, all at once — so out of the box it produces the average of everything: generic, bland, faintly off. The telltale AI output your colleagues can spot in a second.

Now give that painter a clean palette with just the colours the picture needs, and precision becomes possible. It's the same with AI. Provide focused, relevant context — your clients, your projects, your standards, your voice — and the output transforms from generic to genuinely *yours*.

> This is the whole game: **the quality of what AI produces is determined almost entirely by the quality of the context you give it — not by the model.** Master context and your outputs are excellent. Neglect it and they're mediocre, whatever tool you use.

THE SHARED BRAIN

There's a catch. Tools like ChatGPT forget everything between conversations — like a brilliant colleague with total amnesia each morning, so you re-explain your whole world every time. That makes giving AI rich context practically impossible.

The answer is a **shared brain**: one place on your computer — a folder of plain files — holding everything the AI needs to know about your work. It reads from it every session, so your context is always there. Two things make it powerful:

- **It builds itself, under your supervision.** You don't maintain it by hand. As you work, the AI captures and files what matters — so the brain gets richer with every task, and tomorrow's work begins from everything you've done before.
- **It has a map.** One file tells the AI where everything lives — so you don't fetch context yourself. You say what you want, and it draws on whatever the task needs from across all your work.

This is the new way of working: **AI creates all your outputs under your supervision.** You work by conversation — giving direction, asking questions, refining — and you approve everything before it goes out. Your expertise doesn't disappear; it shifts from making the work to judging it. Faster, and far less draining.

WHY THE TOOL IS NOT A DETAIL

If context is everything, the thing you cannot compromise on is your **command of that context** — it is the source of every output's quality. So the real test of a tool isn't "is it easy to start?" It's three harder questions:

1. Can you control how your work gets done?

Above all your saved ways of working — the judgement you've encoded. Can you shape it directly and instantly, or is it locked behind someone else's process?

2. Can the AI work across your whole shared brain?

Everything in one place, with a map — so it draws on whatever the task needs, not just the slice in front of you.

3. Can you see your work in the context of everything else?

So you always know where things live, and can point the AI straight to the right context.

This is where two tools that look alike diverge completely.

CLAUDE CODE AND COWORK

Both are made by Anthropic, and both are genuinely capable — you can do real, substantial work in Cowork, and for many people it's a big step up from a chatbot. The difference is what each is *designed* for, and it shows up in three places.

> **Controlling how your work is done.** The highest-leverage context of all is your **skills** — saved instructions for how *you* want a recurring job done. In Claude Code they're plain files you (or the AI) can open, read and change in seconds — as is the map itself. You keep direct control of the two things that matter most. Cowork keeps skills in a separate, managed place — packaged tools you install, hard to see and awkward to change — built on the assumption that someone *else* crafts the workflow and you just run it. We believe the opposite: you know your work best, so you must own how it's done.

Working across your whole shared brain. Claude Code simply runs inside whatever folder you open it in — so you point it at the one folder holding all your work, and the map orients it to everything. Cowork has no such map. It's built to work project by project: you open a folder for the job in front of you, work inside it, then move on. So the AI only ever sees the slice you've scoped — never the whole picture. And that quietly wrecks quality: real work draws on context scattered across everything you do — the way your own mind brings your whole experience to bear, rather than opening a single drawer. Confine the AI to one folder and it works with most of its memory switched off.

Seeing your work in context. Subtler, but it matters. As you work, Cowork shows you the files it has changed — on their own, not set within everything else you have. Over time you lose your feel for what context lives where — and that feel is what lets you direct the AI well ("read the notes from that meeting first", "pull everything on this client") and keep your shared brain in order. With your shared brain open beside Claude Code, every change lands in full view of the whole.

THE SIMPLICITY MOST PEOPLE MISS

Here's the irony. Claude Code *looks* like the harder tool — it grew up in a terminal, which can seem technical. But the terminal is now a choice, not a requirement: the same Claude desktop app that hosts Cowork also carries a **Code tab** — Claude Code itself, with very similar functionality in a friendlier window. The two ways of working sit side by side in one app, which makes the real difference plain: it was never the interface. It's who controls the context.

And whichever window you choose, once it's set up (a few guided installs), Claude Code is far simpler to work in. No projects to switch between, no skills to load, nothing to click. You just **speak**, and the work happens — the tool fades into the background and you're simply talking to your computer. Cowork's easier-looking surface actually asks more of you, not less.

THE BOTTOM LINE

This way of working only delivers when you own and control your context — completely, with your whole shared brain in view. Claude Code gives you precisely that: it's the choice for the professional who wants to go all the way — to have AI create *all* their work, every output, with their whole shared brain behind it. Cowork, by design, keeps you a step removed from that.

But the tool is only half of it. The seismic difference isn't Claude Code versus Cowork — it's between someone skilled at working this way and someone who isn't, above all at managing context. And no tool teaches you that. Claude Code won't show you how to build a shared brain, encode your work into skills, or manage context well; left to work it out alone, most people never do.

That's what Multiply Academy teaches. We use AI to personalise the whole programme to your role and your real work, and we guide you through the hardest part — rewiring how you've worked for years. The tool is the easy part. Learning to wield it is where we come in.

Reflects our hands-on evaluation as of June 2026; desktop-app note updated July 2026. We re-test as these tools evolve.