

The Handoff Protocol

“ **One-liner:** Split a problem across two separate AI sessions with different roles and contexts, then synthesize their outputs yourself — like managing a real team.

? Jump in (Tinkerers start here)

You'll need two AI chat windows open at the same time (two browser tabs, or two different AI tools — either works).

Pick a project or decision that has at least two distinct dimensions. For example: "Create a content strategy for launching our new product."

Chat A — The Strategist. Open your first chat and send:

“ You are a **brand strategist** with 15 years of experience. Your focus is positioning, audience targeting, and messaging clarity. You do NOT think about implementation details — that's someone else's job.

I'm working on: **[your project]**

Give me your strategic recommendations. Focus on: who the audience is, what the core message should be, and how to position this differently from competitors. Be specific and opinionated.

Chat B — The Executor. Open your second chat and send:

“ You are an **operations-focused content producer**. Your focus is practical execution: channels, formats, timelines, and resource requirements. You do NOT set strategy — you receive it and figure out how to make it real.

I'm working on: **[your project]**

Give me an execution plan. Focus on: which channels to prioritize, what content formats work best, a realistic timeline, and what resources I'll need. Be specific and practical.

Now you're the manager. Read both outputs. Notice what Chat A assumed that Chat B would question, and vice versa. Then write your own synthesis:

- Where do these perspectives align?
- Where do they conflict?
- What did each one miss that the other caught?
- What's your actual plan, informed by both?

Optional bonus round: Take your synthesis and paste it back into one of the chats:

“Here's the combined strategy and execution plan I've built from two different advisors. Poke holes in it. What's still weak?”

? Plan first (Planners start here)

Here's what you're about to do:

1. **Choose a project** — Something real with both a strategic and practical dimension. Content launches, product decisions, event planning, and hiring processes all work well.
2. **Set up Chat A (Strategist)** — Give it a clear strategic role with explicit boundaries. Tell it *not* to worry about implementation.
3. **Set up Chat B (Executor)** — Give it a clear operational role with explicit boundaries. Tell it *not* to set strategy.
4. **Run both chats** — Send the same project description to each, but with their respective role prompts.
5. **Synthesize manually** — You are the integration point. Compare outputs, find gaps, resolve conflicts, and produce a combined plan.

"Done" looks like: You have a plan that neither AI session could have produced alone, and you can articulate what each perspective contributed.

? Why this matters (Strategists start here)

In [AC-Basic-01](#), you simulated multiple perspectives in a single chat. Here, you're practicing a fundamentally different skill: **managing separate agents with isolated contexts**. This mirrors how real multi-agent systems work — each agent has a specific role, limited scope, and doesn't see the other's work. The human (you) acts as the orchestrator. This is the skill that scales: from two chats to entire AI-assisted workflows with specialized roles, handoff points, and quality gates.

Reflection

- How did the outputs differ when each AI had a constrained role vs. a single AI doing both? Was the split worth the extra effort?
- What context got lost in the handoff between sessions? How would you design a better transfer summary?
- Did the synthesis step feel harder or easier than you expected? What made it difficult?
- ☐ *Run this exercise with two colleagues, each managing one AI session. Compare the experience of synthesizing someone else's AI output vs. your own — it highlights how much implicit context lives in your head. (Social Learners)*

?? Level up

Ready for more? Try [AC-Advanced-01](#) — where you'll design a complete multi-agent workflow with defined roles, handoffs, and feedback loops.

Back to [Agent Collaboration](#)

Revision #1

Created 2026-03-16 15:48:25 UTC by Admin

Updated 2026-03-16 15:48:25 UTC by Admin