



FROM "WHERE DID I SAVE THAT" TO "CLAUDE ALREADY FILED IT"

Claude Cowork

From Zero to Hero.

One desktop. One agent. The status meeting just got shorter.

THE ARGUMENT, IN THREE BEATS

- 01** • The loop flipped for office work too.
- 02** • The deliverable is the unit, not the chat.
- 03** • Scheduled tasks compound while you sleep.

Anthropic shipped Cowork in ten days. Claude wrote the whole thing.

If the agent can build the agent, your spreadsheet doesn't stand a chance.

Thesis — Cowork moves the unit of work from "the chat" to *"the deliverable"* — and the new skill is choosing what to delegate, not what to type.

THREE THINGS THIS CLAIM COMMITS ME TO DEFEND

- 01** • The unit changed — chat turns out, deliverables in.
- 02** • The desktop is the surface — not the browser, not the chat box.
- 03** • The skill is delegation — picking the right boundary, not the right prompt.

WHY THIS IS A THESIS, NOT A TOPIC

A reasonable PM could disagree. The "but I want to see every step" counter-position is named on slide 16.

SOURCE ANCHOR

“The human role becomes validation, refinement, and decision-making. Not repetitive execution.” — Anthropic, 2026.

Cowork is the same agentic engine as Claude Code — *but pointed at the desktop instead of the repo.*

THE TWO PRODUCTS, SIDE BY SIDE — AND WHERE COWORK ACTUALLY SITS

TAB IN THE DESKTOP APP

Chat

one prompt · one reply

You ask a question, Claude answers. The unit is the message. You copy the result somewhere else.

TAB IN THE DESKTOP APP

Code

scope: a code repo

Edits files, runs tests, opens PRs. The unit is the patch. Built for engineers.

TAB IN THE DESKTOP APP

Cowork

scope: your whole desktop

Reads your files, opens Excel, sends emails, runs scheduled tasks. The unit is the deliverable.

Describe → Plan → Approve → Execute → Deliver.



REPEATS ON A SCHEDULE — DAILY, WEEKLY, WHATEVER YOU SET

WHERE THE HUMAN STAYS IN THE LOOP

You set the goal at **step 01** and approve the plan at **step 03**. Steps 04 and 05 happen whether you watch or not.

Anthropic calls this "validation, refinement, and decision-making" — the actual job of the human.

Evidence I — Anthropic shipped Cowork itself in ten days.

TIME FROM KICKOFF TO LAUNCH

10 days

from "let's build this" to a public preview — one and a half weeks.

Claude Code wrote 100% of the production code.

WHAT THE DOGFOODING PROVES

The team that builds the agent uses the agent to build the agent.

If a 10-day launch cycle is now the floor at Anthropic, the deck you're scheduled to deliver next Friday is the new bar for everyone else.

So this shows: the bottleneck on knowledge work is no longer typing speed or research depth. It's whether you trust the agent enough to ship its output.

Evidence II — the output is a file, not a message.

“

“Claude delivers finished work instead of step-by-step updates: a formatted spreadsheet, a memo, a briefing doc. You review, refine, and decide what's next.”

— Anthropic, claude.com/product/cowork · product page, 2026

WHAT SHOWS UP IN YOUR FOLDER

.xlsx

receipts → budget tracker

.docx

notes → first-draft report

.pptx

template + data → deck

.pdf

scattered docs → briefing

So this shows: the conversation is the input, not the output. The unit of value is what shows up on disk — not what's in the scrollback.

Evidence III — the work runs while you sleep.

BEFORE — YOUR MONDAY MORNING

"Let me pull together last week's numbers..."

25-40 minutes · every Monday · forever.



AFTER — YOUR SCHEDULED TASK

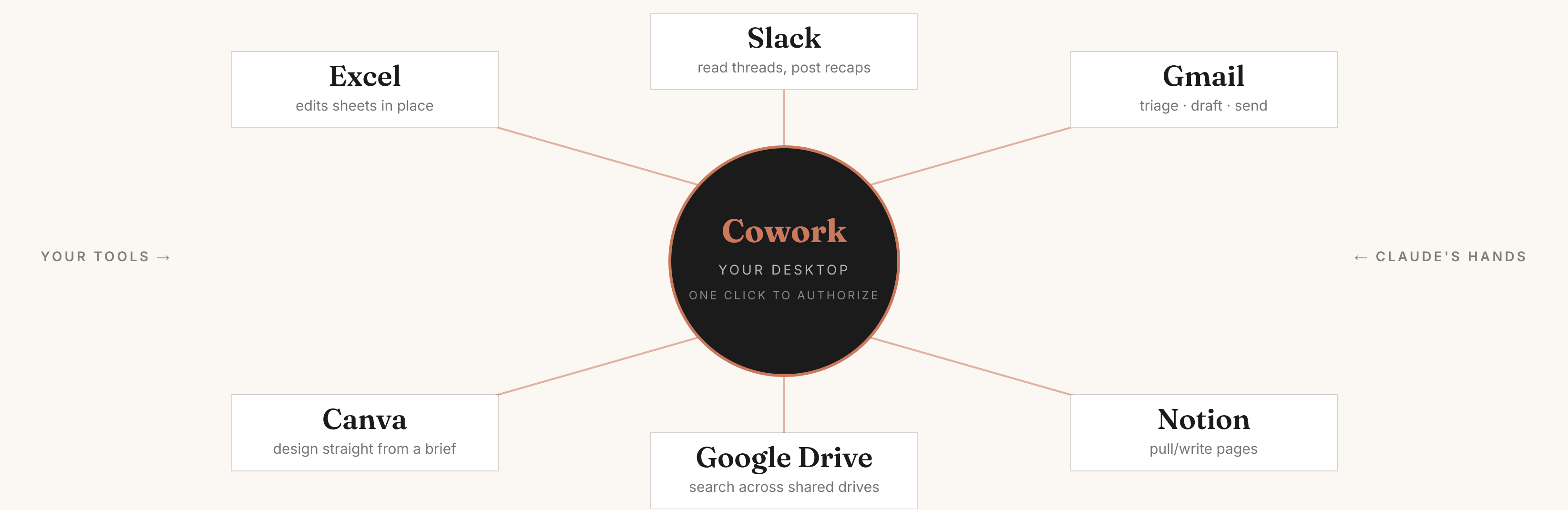
`/schedule weekly digest`
`every Mon · 8:00 AM`

Set once. The .xlsx is in your inbox at 8:05.

THE MATH THE CALENDAR RUNS ON YOU

Thirty minutes a week is *26 hours a year*. A scheduled task that costs you 90 seconds to set up pays back a full work-week, every year, until you turn it off.

Connectors — one-click access to the apps your work actually lives in.



The point: Claude doesn't ask you to copy-paste. It works inside the apps where the work already lives — with the permissions you grant, one click each.

Projects — one local folder, scoped permissions, persistent memory.

~/Projects/q1-board-prep/

SCOPED TO
COWORK

- ▶ **CONTEXT.md** # tone, audience, "ask first" rules
- ▶ **templates/**
 - ├─ deck-template.pptx
 - └─ memo-template.docx
- ▶ **source/**
 - ├─ q1-financials.xlsx
 - ├─ board-feedback-q4.pdf
 - └─ product-roadmap.md
- ▶ **drafts/**
 - └─ (Claude writes here)

From any session in this project:

> draft the Q1 board deck using **templates/** and **source/**

WHAT A PROJECT GIVES YOU

A scoped sandbox. Claude can read and write in this folder only. Permissions don't leak across projects.

Persistent context. Templates, brand voice, "ask first" rules — loaded into every session automatically.

The CLAUDE.md analogue. Same idea as Claude Code's memory file — one place to teach Claude your conventions.

A Project is to Cowork what a repo is to Claude Code: the named container of the work.

Skills — the third time you do something, you teach Cowork to do it.

```
.claude/skills/expense-reconcile/SKILL.md
```

```
ONE-TIME  
SETUP
```

```
---  
name: expense-reconcile  
description: Match credit-card statement to receipts;  
             flag missing or duplicate charges.  
---
```

Inputs

- The latest statement.csv from Downloads/.
- All .pdf receipts from Downloads/receipts/.

Output

- One reconciliation.xlsx with:
 - matched rows (green)
 - missing receipt (orange)
 - suspected duplicate (red)
- A two-line summary at the top.

WHY SKILLS, NOT JUST PROMPTS

Reusable. Write it once. Invoke by name. The five-paragraph prompt becomes one line.

Shareable. Email the folder to your teammate. They get the same skill, instantly.

Composable. A skill can call other skills. The workflow compounds.

Every Skill is a tax cut. The boring task you used to do every Tuesday now takes 30 seconds, forever.

Scheduled tasks — the recurring work that used to anchor your calendar.

Cowork · Scheduled · three real examples /SCHEDULE

▶ EVERY WEEKDAY · 08:00

/schedule morning brief
pull yesterday's metrics from Looker,
summarise the 5 most-replied Slack threads,
email it to me at 08:00.

▶ FRIDAY · 16:00 · WEEKLY

/schedule team-recap
read all #eng-* Slack channels for the week,
write a 200-word recap into the team Notion page.

▶ 1ST OF EVERY MONTH · 09:00

/schedule budget-refresh
reconcile invoices/ against last month's budget.xlsx,
flag variance > 10%, file under reports/2026/.

THE MATH

90s

to set up — once.

26 hrs

saved per year, per task.

Your status meeting just turned into a Slack message that writes itself.

Dispatch — text the desktop from your phone. It picks up where you left off.

```
Claude iOS · in the airport 14:32

You · 14:32
draft the Q1 board memo
using the source folder
+ feedback from the
board call yesterday

Cowork · 14:33
▶ Routing to your Mac.
▶ Reading source/* (8 files)
▶ Reading meeting-notes-
  2026-q1-board.md
▶ Drafting...

Cowork · 14:47
✓ Draft saved to
  drafts/q1-board-memo.docx
Open to review →
```

WHAT JUST HAPPENED

- 01 **You're not at your desk.** The message goes to Anthropic, which routes it to your laptop — which is logged in to Cowork and idle.
- 02 **The laptop does the work.** Local files, local apps, full permissions you already granted — the agent has hands where the data lives.
- 03 **You land, you have a draft.** The file is waiting in the Project folder. Time-in-the-air paid for itself.

The deliverable doesn't wait for you to be at your desk. The desk is the agent's, now — you just review.

Computer Use — when there's no API, Claude clicks the buttons itself.

WHAT IT DOES, IN ONE DIAGRAM

Cowork takes a *screenshot*, reasons about the pixels, and moves the *cursor*.



USE IT — AND DON'T

✓ USE IT WHEN

A legacy desktop app has no API.

Internal tool, vendor portal, accounting software your company won't replace — Cowork drives the GUI like a junior would.

✗ DON'T REACH FOR IT WHEN

A connector exists.

A connector is 10× faster and never breaks on a UI redesign. Computer Use is the escape hatch, not the default.

Computer Use is what makes Cowork “anything you can do, Claude can do.” Use it sparingly — it earns its name.

So what —

The PM's edge moves from *drafting fast* to *deciding fast*.

THE PATTERN, RESTATED

Pay tracks the ability to reject a wrong deck in five minutes — not write a passable one in two hours.

THREE CONCRETE CONSEQUENCES FOR THE NEXT 12 MONTHS

01 · SKILL THAT GETS PAID

Editing > writing.

Catching that wrong number in the third row of an autogenerated deck is the new load-bearing skill.

02 · PRACTICE THAT RETURNS

Templates are policy.

A good company template is now an artefact the agent reads. Brand voice is enforced by file, not by review.

03 · WHERE SENIORS WIN

Picking what to build.

The agent won't propose the right strategy. Choosing which deliverable matters — that's the work that doesn't get automated.

01 THE STRONGEST OPPOSING VIEW

“Knowledge work isn't deliverables. It's thinking.”

Critics argue that the deliverable was always a fossil — a memo, a deck, a sheet — left behind by the real work, which was reasoning together. Optimise the fossil and you optimise the wrong thing. **Worse:** when a 2-hour draft turns into a 20-minute review, the cuts go to the hours where ideas formed, not the hours where slides were polished.

Source: Cal Newport, "Deep Work" (2016) — the position, restated for the agentic era.

02 BUT THE THESIS STILL HOLDS —

The criticism is right about thinking — and wrong about which hours get cut.

The 2-hour draft is mostly typing, formatting, and finding files — *not* thinking. A senior PM can describe their reasoning in five minutes; the remaining 1h55 is converting it to a deliverable. Cowork eats the conversion tax. The thinking still has to happen — and you finally have time for it.

Source: this deck's argument, defended.

Five failure modes — if you do these, the agent costs more than it saves.

01 **No Project, just chat**
Treating Cowork like ChatGPT — one-off prompts, no folder, no templates. You re-explain your brand voice forty times.

02 **Computer Use when a connector exists**
Watching Cowork click through Gmail's UI when the Gmail connector ships in two clicks. 10× slower; breaks on every redesign.

03 **Approving plans you didn't read**
Hitting "approve" on a plan that touches 40 files. The agent will do all 40 — including the wrong 7. Read the plan or revise it.

04 **Scheduled tasks with no review step**
A weekly digest that gets emailed to leadership without you ever opening it. The first wrong number reaches your VP before you do.

05 **Delegating the thinking, not just the typing**
Asking Cowork to "decide our Q1 strategy from the board notes." It will produce confident-looking slop. The decision is still yours.

Zero to hero — the title of this deck, owed in receipts.

WEEK 1 · ZERO

Tame one folder.

Point Cowork at your Downloads folder. Have it sort and rename every file. Read each rename before approving.

Outcome:
a clean Downloads/ & calibrated trust

WEEK 2 · PROJECT

Set up one Project.

Pick the recurring deliverable that drains your week. Build the Project: `CONTEXT.md`, templates, source.

Outcome:
one deliverable, halved time

WEEK 3 · SCHEDULE

Automate the cadence.

Wrap that deliverable in a scheduled task. Always email it to *you* first — not the team. Review for 4 weeks before opening the firehose.

Outcome:
Mondays free

WEEK 4 · HERO

Ship a Skill.

Package what you built into a Skill folder. Share it with one teammate. Now two of you have Mondays back.

Outcome:
leverage that compounds

How a PM's Monday morning gets spent — before and after Cowork

Each bar = the first 3 hours of a Monday. Same PM, same role, before and after a Project + scheduled tasks setup.

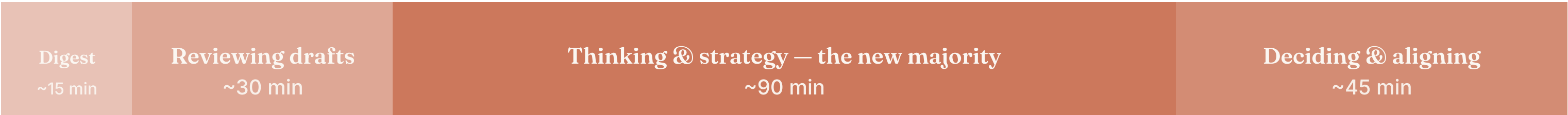
PRE-COWORK · A TYPICAL MONDAY

→ THINKING WAS 14% OF THE MORNING



POST-COWORK · PROJECT + SCHEDULED TASKS

→ THINKING IS NOW 50% OF THE MORNING



A 180-minute Monday morning, broken down by activity

So this shows: the thinking didn't expand — the typing collapsed. What used to fight for a quarter of the morning now owns it.

Source: composite, illustrative — derived from self-reported time logs in aiblewmymind.substack.com "Cowork: 10 use cases I tested" + Aakash Gupta "You Should Be Using Claude Cowork" (2026). Not a controlled study.

SLIDE 20 · PICK A SIDE

Which Monday morning task do you delegate to Cowork *first* — and which one will you *never* hand over?

DEFEND YOUR PICK · 30 SECONDS

YOU HAVE

30s

NAME BOTH, DEFEND ONE

THREE COMMON PICKS · PICK ONE FOR DELEGATION · BE SPECIFIC

(a)

The status report.

Pull data, format, send. Highest ROI — same shape every week.

(b)

Inbox triage.

Sort, draft replies, surface urgent. Daily reward, fast feedback loop.

(c)

Board deck.

Highest visibility, lowest tolerance for error. Start with the draft, not the send.

▲ MY PICK TO DELEGATE

I argue (b). *And the one I'll never delegate: deciding which deliverable matters. That's the human's job, forever.*