



CHAPTER THREE

Your business needs one place to be true.

Everyone who quits AI quits at the same point. Not because it could not do the work, but because the work felt like it was scattered across forty conversations and none of them remembered the others. That is not an AI problem. It is a filing problem, and it has six drawers.

7 DRAWERS 2 LOAD EVERY TIME 5 LOAD ON DEMAND BUILD IT IN AN AFTERNOON

Why it feels chaotic

The discouragement is real and it is predictable. It goes like this.

Week one is magic. You ask for a programme description and get something better than you would have written. Week two you ask again and it has forgotten your tone. Week three you paste your brand colours in for the fourth time. Week four you are re-explaining a client's shoulder injury to something that helped you with that exact shoulder a fortnight ago.

So you conclude the tool is unreliable, and you go back to doing it yourself.

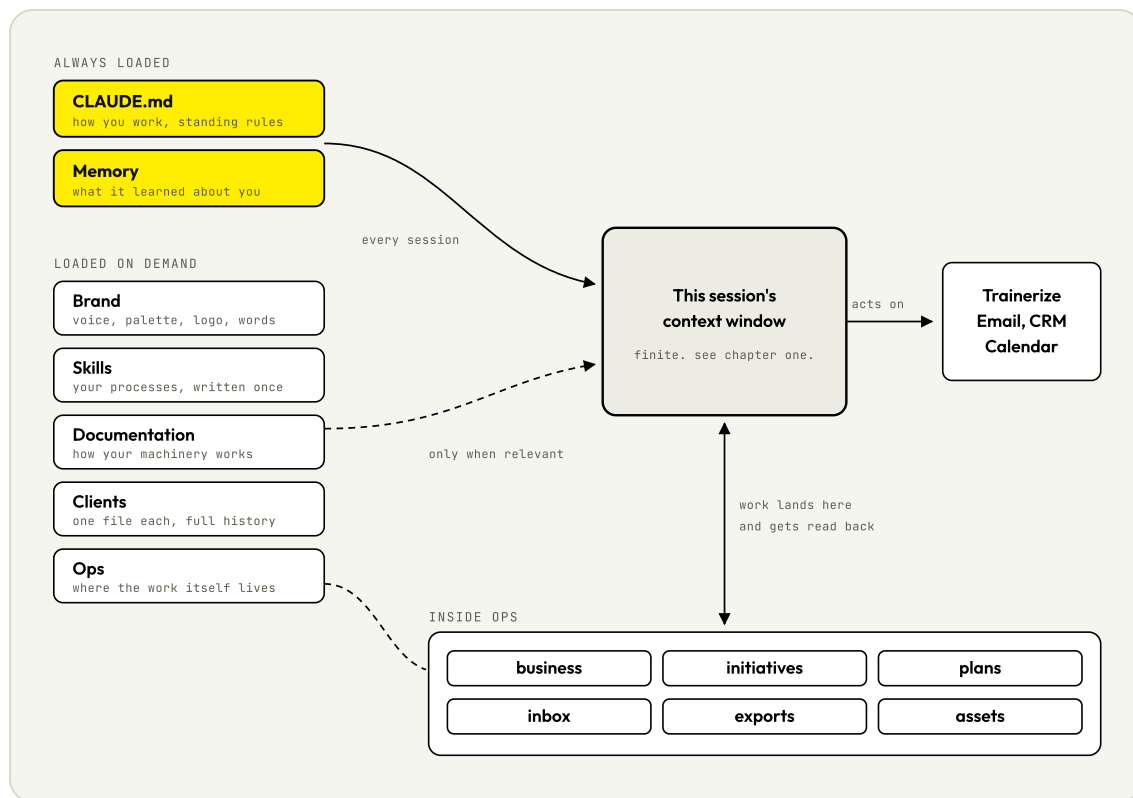
Nothing was unreliable. You were starting from zero every time, because nothing you told it was written anywhere it could read. Chapter One called this the difference between the *context window*, which is working memory and empties every session, and *memory*, which has to be written down somewhere and deliberately loaded back.

THE REFRAME

You are not "using an AI". You are onboarding a very capable new hire who has total recall of anything written down and zero recall of anything that was not. Everything in this chapter is the staff handbook you would have written for a human, except this one gets read in full, every single time, forever.

The shape of it

Seven drawers. Two open automatically, five open only when the job calls for them. Six hold what Claude needs to know about you, and the seventh holds the work itself. That is the whole structure.



SEVEN DRAWERS. TWO LOAD THEMSELVES, FIVE WAIT TO BE CALLED. SIX HOLD WHAT CLAUDE NEEDS TO KNOW ABOUT YOU. OPS IS THE SEVENTH, AND IT HOLDS THE WORK ITSELF.

WHY ON-DEMAND IS A FEATURE, NOT A LIMITATION

A beginner's instinct is to load everything so Claude "knows it all". That is exactly how you cause context rot. If all forty client files are open, the one client you are actually discussing is competing for attention with thirty-nine who are irrelevant. **A good source of truth is not the one that loads the most. It is the one that loads the right thing.**

The seven drawers

WHAT EACH DRAWER IS, AND THE QUESTION IT ANSWERS		
DRAWER	THE QUESTION IT ANSWERS	LOADS
CLAUDE.md	How does this person work, and what must never happen?	Every session
Memory	What have I learned about them that is not written anywhere else?	Every session
Brand	What do they sound like and look like?	Anything client-facing
Skills	How do they do this particular job?	When the task matches
Documentation	How does my machinery actually work, and where are the keys?	When touching it
Clients	Who is this person and what is their history?	When that client is named
Ops	Where is the work itself, and everything we are producing?	When working on it

1• CLAUDE.md, the standing instructions

One file, read at the start of every session. Who you are, what business you run, how you like things done, what is off limits, and where everything else lives.

This is the highest-return file you will ever write, and the rule for filling it is simple: **every correction you find yourself repeating belongs in it, the same day you notice**. If you have told it twice not to use exclamation marks, that is not a preference any more, it is a rule, and rules go in the file.

2• Memory, the running record

Distinct from CLAUDE.md. That file is what you deliberately wrote. Memory is what got learned along the way, one fact per note, indexed so the index loads cheaply and the detail loads only when it matters. Covered properly further down, because this is the drawer people manage worst.

3• Brand, so it sounds like you

Voice and tone with real examples of both. Words you use and words you refuse. Colours with their hex codes and where each is allowed. Fonts. Logo files and the rules for using them. Your promise, your positioning, who you are for and who you are not for.

Skip this drawer and every piece of copy will be competent, generic and unmistakably not yours. Fill it once and everything downstream inherits it.

Where it lives. Brand is the one drawer that sits inside ops, at `ops/assets/brand/`, because most of it is files: logos, a social kit, a website theme. Start with `guidelines/brand-guidelines.md`. If you already have brand guidelines (a page on your site, or a PDF), you don't need to rewrite them. Keep one text copy there so Claude can read it quickly, put the link to the original at the top, and say the original wins if they ever disagree.

4 • Skills, your processes made executable

Every repeatable job you do, written down as instructions. Onboarding a new client. Writing a check-in. Building a twelve-week block. Handling a cancellation. Publishing a video.

Written as a skill, the process runs identically at 9am Monday and 11pm Friday, whether you run it, Claude runs it, or the person you hire next year runs it. This is the drawer that turns a coach into a business.

THE GOTCHA THAT SILENTLY BREAKS THIS

Claude Code only discovers skills in a hidden folder called `.claude/skills/`. Put them anywhere else and they simply never load, with no error and no warning. But a hidden folder is a miserable place to keep something you edit weekly.

The fix is one line, and it gets you both: keep the folder visible at `skills/` and point Claude Code at it with a shortcut.

```
# run once, from your business folder
$ mkdir -p .claude && ln -s ../skills .claude/skills
```

Now the drawer is where you can see it and your assistant still finds it. If you ever move the folder, recreate that link or your skills will quietly stop working.

5 • Software documentation, how your stack works

One note per platform you depend on. What it does for you, how it authenticates, where the credentials live, which parts of it have an API, and the specific gotchas you have hit and solved.

The gotchas are the valuable part. Every hour you have lost to a platform quirk is an hour you will lose again unless it is written down. Write the quirk and the fix, and it becomes an hour you never lose again.

6 • Clients, one file each

Goals, history, injuries and limitations, preferences, what has been tried and what happened, decisions you made and the reasoning behind them.

This is the drawer that changes how coaching feels. "Draft Sarah's check-in" stops being a prompt you have to load with context and becomes an instruction that already has it. The file

makes the difference between an assistant who knows your clients and one who has to be told every time.

7•Ops, where the work itself lives

The first six drawers are things Claude reads to understand you. Ops is different: it is where the actual work sits, and **everything Claude produces for you lands here**.

Claude writes into other drawers too, and should. It records what it learns into memory, and it writes up how a platform behaves into documentation. The difference is not read versus write, it is knowledge versus work.

It is one drawer with subfolders, not seven more drawers, and that matters. You are not memorising a filing system. You are opening one drawer and finding the obvious compartment inside it. Covered in full below.

The client one-sheet, and where it comes from

The Clients drawer is the one that sounds like admin until you see what fills it. You do not write these. Your sales calls write them.

Zoom is the input you already have

If you run consults on Zoom, you are already generating the single richest document about every client you have, and almost certainly throwing it away. The recording holds their exact words: the real reason they came, the thing they are embarrassed about, the last three things that failed and why they think it failed.

Connect Zoom and that transcript becomes readable. Point Claude at it with a template and you get a filled-in client profile instead of a two-hour recording nobody will ever rewatch.

WHY THIS BEATS NOTES YOU TOOK YOURSELF

Your consult notes capture what you thought was important while you were also selling. The transcript captures what *they* said, in their words. Those words are the ones that should show up later in their check-ins and their renewal conversation, and you will not remember them in March.

The one-sheet template

One file per client. Filled from the consult transcript and the intake form, then updated as things change. This is the shape worth standardising on:

```

# Sarah Chen · client profile
## Identity
Started      2026-09-01      Package  12-week 1:1
Timezone     PST                Check-in  Sundays
Contact      sarah@...           Source   Instagram, saw the knee video

## Why she came (her words, from the consult)
"I've lost the same 20 pounds four times. I don't want another diet,
I want to stop doing this every two years."
Emotional driver: her daughter's wedding, but she led with the pattern.

## Goal
Primary      Body recomposition, hold it past 12 months
Measurable   Down 15 lb, deadlift bodyweight, 3x/week consistency
Deadline     Wedding, 2027-06-14

## History and limits
Training age  ~3 years, on and off
Injuries      L knee, meniscus 2019, no surgery. Pain on deep flexion.
Medical       Hypothyroid, medicated and stable
Equipment     Commercial gym, full access
Schedule      Mon/Wed/Fri early, 45 min hard ceiling

## Nutrition baseline
Start weight  168 lb
Initial macros  P 140 / C 180 / F 60  (~1900 kcal)
Restrictions   No dairy, dislikes fish
Notes         Skips breakfast most days. Not negotiable, worked around.

## Training baseline
Squat 135x5 · Bench 85x5 · DL 165x5 · no overhead pressing yet

## What has failed before, and why
Two coached programs, both abandoned around week 7.
Her read: "the check-ins stopped feeling personal."
⇒ Flag: week 6-8 is the risk window. Do something human there.

## Decisions
2026-09-01    Leg press over back squat until knee tested. Revisit wk 4.
2026-09-01    Macros set conservative on purpose. She has dieted hard
               before and needs room to move down later.

```

Every section on that sheet answers a question you would otherwise ask again, or worse, guess at. The two most valuable blocks are the ones most coaches never write down: **what failed before and why**, and **the decisions with their reasoning**. Six months from now you will not remember why her macros started conservative, and without the note you will read it as a mistake instead of a plan.

THE COMPOUNDING BIT

Once these exist, "draft Sarah's week 6 check-in" is a complete instruction. It already knows week 6 is her risk window, that the knee is being tested, that fish is out and that her real fear is the fourth relapse. That is the moment coaching with AI stops feeling generic.

The memory layer, and how to keep it honest

This is the drawer people manage worst, because it is the only one that gets actively worse if you ignore it.

The other five drawers are things you write. Memory is what accumulates. Left alone it becomes a landfill: contradictory facts, notes about a client who churned last year, a rule you changed your mind about in March that is still quietly steering things in September.

A wrong memory is worse than no memory. No memory makes Claude ask. A wrong one makes it confidently proceed.

Six rules that keep it useful

1. **One fact per note.** Not a diary entry, not a session log. One thing that is true, in a file named after it. Notes that hold five facts get half-read and half-updated.
2. **Write the why, not just the what.** "Use 1900 calories for Sarah" rots the moment it changes. "Started conservative because she has dieted hard before and needs room to move down" survives, because it explains itself.
3. **Keep a cheap index.** One line per memory, in one file that loads every session. The detail sits behind it and loads only when relevant. The index is what makes the whole layer affordable.
4. **Date anything that can expire.** Prices, packages, platform quirks, team members, tools. A fact true in March is a liability in September if nothing says when it was written.
5. **Do not store what is already written somewhere.** If it is in the client file, the brand doc or the skill, it does not belong in memory too. Two copies means one of them is wrong and you will not know which.
6. **Delete confidently.** Pruning is maintenance, not loss. A memory that is no longer true should be gone, not annotated.

THE TRIGGER TO WATCH FOR

If you have corrected the same thing twice, stop and write it down. Not the third time, the second. That single habit is most of what separates a setup that gets sharper every week from one that plateaus in a fortnight and gets abandoned in a month.

Inside the ops drawer

Six drawers hold what Claude knows. This one holds what you are doing. Six compartments inside it, and you create each one the first time you actually have something to put in it.

THE QUESTION THAT SORTS EVERYTHING

"Does Claude need to know this, or are we working on it?" Knowledge goes in one of the first six drawers. Work goes in ops. That single question resolves almost every "where does this go" moment, and it is why the structure still holds up once there are a few hundred files in it.

business

The business as a business. Contracts and templates, pricing, policies, insurance, registration, anything you would hand an accountant or a lawyer. Named **business** rather than **documents** on purpose, so it never gets confused with the documentation drawer.

initiatives

One folder per thing you are actively building. The website rebuild. The group programme launch. The certification you are studying for. Each holds its own notes, drafts and decisions, and gets archived whole when it finishes.

This is the folder that stops long projects leaking into everything else. Without it, a three-month build scatters across forty conversations and you can never answer "where did we get to on that?"

plans

Thinking before doing. Rough, dated, disposable. Every plan either graduates into an initiative or gets deleted, and both outcomes are fine. Keeping plans separate from initiatives is what stops half-formed ideas being mistaken for commitments.

inbox

Capture. Voice notes from the car, links, half-thoughts, the thing you must not forget. It has exactly one rule: **it gets emptied**. An inbox you never process is just a pile with a nicer name.

This is the smallest folder on the list and the one that changes your week most, because it is what lets you stop holding things in your head between clients.

exports

Your Claude conversation history, saved as files so it is searchable later. Sessions are not permanent by default, and the good ones contain reasoning you will want again. Exporting them turns your own past work into something you can search, quote and reuse instead of half-remember.

assets

The files. `brand/` holds your guidelines, logos and social kit (the brand drawer lives here). `images/` holds headshots and photos, and `copy/` holds finished text like bios and profile copy. Kept together so your text folders stay light and quick to search. Videos are too big for this folder, so keep them on a drive or in your video host.

DO NOT BUILD ALL SIX ON DAY ONE

Create a compartment the first time you have something to put in it. An ops drawer with three used folders beats one with six empty ones, and empty folders are how a structure starts feeling like homework instead of a tool.

What it looks like on disk

Seven drawers, one folder. This is genuinely the whole thing.

```

my-coaching-business/
├── CLAUDE.md           ← read every session. start here.
├── memory/
│   ├── MEMORY.md      ← the cheap index
│   └── ...             one fact per file
├── skills/
│   ├── onboard-client/
│   ├── write-checkin/
│   └── build-program/
├── documentation/     ← how your machinery works
│   ├── software/
│   │   ├── trainerize/  how you use it, where the keys are
│   │   ├── stripe/
│   │   └── zoom/
│   ├── blueprints/     what each automation does
│   └── workflows/
├── clients/
│   ├── sarah-chen.md   ← the one-sheet
│   ├── marcus-webb.md
│   └── _template.md    so every new one matches
├── ops/               ← the work itself. the only drawer
│   ├── business/       contracts, pricing, policies, legal
│   ├── initiatives/
│   │   ├── client-journey/ ← first touch to renewal
│   │   └── website-rebuild/
│   ├── plans/          dated, rough, disposable
│   ├── inbox/          ← capture. gets emptied.
│   ├── exports/        saved conversations, searchable
│   ├── assets/         the files
│   │   ├── brand/      ← the brand drawer lives here
│   │   │   ├── guidelines/ start with brand-guidelines.md
│   │   │   ├── logos/
│   │   │   └── social/
│   ├── images/         headshots, photos
│   └── copy/           bios, profile copy, finished text

```

That is a filing cabinet, not a software project. If you can keep a client folder in Dropbox, you can keep this. The only genuinely new idea is that something now reads it.

ONE FOLDER, BACKED UP, VERSIONED

Keep the whole thing in a single folder that syncs and has version history. That way the entire operating system of your business is one directory you could hand to a new hire, restore onto a new laptop, or roll back if something goes wrong. Chapter One called this a *repo*. It is the safety net that makes it reasonable to let Claude edit real files.

How to actually build it

In order, smallest first. Do not try to fill six drawers in one sitting. Every coach who tries that finishes none of them.

1. **CLAUDE.md, twenty minutes.** Who you are, what you do, three things you always want and three you never want. It will be bad. Ship it anyway; it improves every time you correct something.
2. **One client file.** Your next consult. Record it, transcribe it, fill the one-sheet. One file proves the whole idea faster than any amount of planning.
3. **Brand voice.** Paste in five things you have written that sound like you and ask it to describe the voice. Edit what comes back. That is your voice doc, and it took ten minutes.
4. **Your first skill.** The job you do most and enjoy least. Write down how you do it. That is a skill.
5. **Documentation, as you go.** Do not sit down to document your stack. Write the note the next time a platform wastes an hour of your life. Write the client journey the first quiet afternoon you get; it is the one worth scheduling.
6. **Memory, continuously.** Never a task. Just the twice-corrected rule, forever.
7. **Ops compartments, on first use.** Make `ops/inbox/` today because you will use it today. Make the rest the first time you have something to put in one.

The whole chapter in four lines

1. **Seven drawers.** Standing instructions, memory, brand, skills, documentation, clients, ops.
2. **Two load every session. Five load when called.** That is context management, not filing.
3. **Ops is the seventh and the only one Claude writes into.** Ask "does Claude need to know this, or are we working on it?"
4. **Your consults already contain the client files.** You are just not keeping them yet.
5. **Corrected twice means write it down.** The habit matters more than the folder.

It stops feeling chaotic the moment you can name where a thing belongs. That is the whole trick, and you now have every name on the list.

A source of truth is not a project you finish. It is a place you put things,
starting with the next thing you would otherwise have to explain twice.

FITNESS MARKETING MACHINE CONCORD, NC FITNESSMARKETINGMACHINE.NET
SCALE WITHOUT LIMITS