



Build it **once**

This is where most people quietly lose the time they just saved. They write a great brief, get a great result, and next week open a blank chat and do it again from memory — slightly worse, because they forgot two things.



README

What this is. How to split a brief into the part that never changes and the part that does, save the stable half, and turn a forty-minute job into a ninety-second one that gets better every week.

How to use it. Bring a brief you've already written — or one good result you'd like to repeat. Page 3 splits it. Page 5 shows the finished version.

The running example. Throughout, you're an operations manager at a mid-size company. Every Monday you roll up what the team shipped for your manager, Priya. It takes forty minutes and it shouldn't.

Two halves, one `saved`

Every recurring brief is really two briefs stuck together. Separate them once and you only ever type the small half again.

STABLE — SAVE THIS

Never changes

Who reads it and what they already know. The format. What to cut. The tone. Where your judgment ends and its begins. Written once, improved over time.

VARIABLE — PASTE THIS

Changes every run

This week's raw material. The notes, the thread, the export, the numbers. That's genuinely all that should be different between Monday and next Monday.

- a Look at the ratio.** When people actually split a brief, the stable half is usually eighty percent of the words. That ratio is the entire argument for doing this.
- b The rule that makes it compound.** When the output is wrong, fix the brief, not the output. Fixing the output solves today. Fixing the brief solves every week after.
- c Consistency is the underrated win.** A saved brief doesn't drift. Doing it from memory means a slightly different report every week, which is quietly worse for whoever reads it.

What goes where

PART OF YOUR BRIEF	HALF	WHY
Who reads it, what they know	Stable	Your audience rarely changes.
Format and length	Stable	The shape is the point of reusing it.
Tone and what to cut	Stable	Encodes taste you'd otherwise retype.
Your call vs. its call	Stable	The judgment line holds across runs.
This week's notes and numbers	Variable	The only genuinely new input.

Split it, then `reuse` it

The first prompt does the separation for you. The second is what you actually run every week from then on.

1 · Split an existing brief

`COPY-PASTE · THE SPLIT`

Here's a brief I wrote for a task I do every week: [paste your brief]

Split it into two blocks.

STABLE: everything that would be identical next week — audience, format, tone, what to cut, which decisions are mine. Write this as a standing brief I can save and paste unchanged every time.

VARIABLE: only the parts that are genuinely specific to this week's run.

Then tell me what share of the original was stable, and flag anything I've written as variable that could actually be promoted to stable.

2 · The weekly run

`COPY-PASTE · WHAT YOU RUN EVERY MONDAY`

[paste your saved STABLE brief]

This week's material:

[paste this week's raw notes, threads, exports — nothing else]

That's the whole weekly ritual. If you find yourself re-explaining the format or the audience in the second block, that sentence belonged in the stable half — move it and it's fixed permanently.

Fix the brief, not the output

This is the habit that makes a saved brief get better instead of going stale. Run it any week the result isn't quite right.

3 · Upgrade the standing brief

COPY-PASTE · THE IMPROVEMENT LOOP

Here's my standing brief: [paste STABLE block]
Here's what it produced this week: [paste the output]
Here's what was wrong with it: [describe in plain language]

Don't rewrite the output. Instead, tell me the smallest change to the standing brief that would have prevented this — and show me the edited line.

If the problem was actually this week's material rather than the brief, say that instead. Don't add instructions to cover a one-off.

Two rules for the loop

DO

Make one small edit per week and keep the brief. Mine is on version eleven and barely needs touching now — that's what a compounding asset looks like.

DON'T

Bolt on a new instruction every time something's slightly off. A brief that grows a rule per week becomes an unreadable rulebook nobody, including the model, can follow.

The Monday roll-up, `split`

Forty minutes every Monday, five tabs open, mostly copying. Here's the same job after the split — and the ratio that makes the case.

The stable half — saved once

SAVE THIS · PASTE UNCHANGED EVERY WEEK

You're helping me write my Monday roll-up for Priya, my director. She reads it on her phone before a leadership call and cares about blockers, not activity.

Format: one bottom-line sentence, then three bullets maximum. Anything that is only activity gets cut. Keep the one item that needs someone to decide something, and say plainly what the decision is.

Your call vs mine: pick the wording and the ordering. Don't invent dates or commitments, and don't soften a slipped item to make it read better.

The variable half — this week only

PASTE FRESH · THIS WEEK'S MATERIAL

This week's material:

[Tuesday's shipped deck thread, the vendor migration status, my rough notes]

THE RATIO

Stable: 104 words, written once. Variable: 12 words, retyped weekly. Roughly nine tenths of the thinking is now a saved asset — and the run takes about ninety seconds.

Three ways this unravels

01 Rebuilding it from memory anyway

The default failure, and it happens by accident.

You open a blank chat because it's faster than finding the file. The result is slightly worse every week and you never notice, because each week only differs a little from the last. Put the brief somewhere that takes one click, not one search.

02 Editing the output instead of the brief

Feels efficient. Guarantees you do it again next week.

Fixing the draft in front of you takes two minutes and saves you nothing. Fixing the line in the brief that caused it takes the same two minutes and you never see the problem again.

03 Keeping it to yourself

The difference between a trick and a system.

A saved brief you share means the roll-up still goes out the same way when you're on holiday, and it's how you become the person who improved how the team works rather than the person who types quickly.

RECAP

You now have

- 1 The stable-versus-variable split, and a table of what belongs in each half.
- 2 A prompt that performs the split on a brief you've already written.
- 3 The weekly run, and the improvement loop that upgrades the brief instead of the output.
- 4 A worked example with the ratio that makes the case for doing it at all.

briandoai.com/resources