10:20 — The morning the wrong build went live

In this chapter: 85 minutes from the record: a build that failed silently, an hour nobody saw, and a rollback left incomplete by caching. In the field guide: the airline that paid for its chatbot's words, and the lawyers who asked the machine whether it was right.

You walk away with: The approver's five questions, the six-level action ladder, Labs 1–2 and Decision Card 1.

The first layer • The log

The events in this chapter come from the written email record between my two operational agents, published in shortened form on WithAvi.ai on October 22, 2026. Identifying details and account information have been removed. All times are Israel time.

A small change on a Thursday morning

The change itself was so small that I have nothing dramatic to tell about it. And that's exactly why I'm starting with it. On the morning of Thursday, October 22, 2026, Instinct — the agent that builds my site — had a small update ready. It went through the usual path: the task was written, the change was built in staging (the private copy of the site that visitors never see), and it was tested and approved. All that remained was the most routine step in the whole process: pushing it to the live site. Instinct had done that dozens of times. At 10:20 it did it again.

A push to production has two halves that from the outside look like one action: first the system takes the source files and builds the finished site from them (the Build), and then it publishes the finished build to the world (the Publish). Normally, if the first half fails, everything stops with an error and nothing reaches visitors. That morning the first half failed — and nothing stopped. There was no error message, no alert, and no red line in the log. The build simply didn't finish, and the publish step, finding no new build, went ahead with whatever it had in the system from a previous run.

And that old version wasn't clean. It contained several changes that had never been approved for the live site: an animation upgrade, a performance experiment, a new set of "page not found" pages — and worst of all, an entire page that hadn't been approved for publication at all. So at 10:20 on a Thursday, my public site started showing work that nobody had signed off on — with my name at the top of the pages, and next to it the promise: "Nothing ships without my approval."

Decision Card 1 • Before you approve AI work

Keep it next to your keyboard • Chapter 1

Everything we learned in the first chapter comes down to six short questions. Ask them — briefly and honestly — every time something your AI did is about to leave your hands and go out into the world:

1. **Am I looking at the artifact itself right now — or only at a description of it?** If you're seeing only an intent or a summary, and not the exact page, email or file — stop and ask for the proof.
2. **Which level of the action ladder does this request reach?** 1 Read • 2 Draft • 3 Prepare • 4 Publish • 5 Send • 6 Commit — approving level 2 is not approving level 5.
3. **What's my evidence that this works and is right — and did it come from someone other than whoever built it?** The builder doesn't approve its own work; asking the tool whether it's right is not verification.
4. **If it turns out to be wrong — how do I undo it, and how long will that take?** When the way back takes minutes, the bar is moderate; when there's no way back — as with an email or a payment — the bar rises sharply.
5. **Is this a decision I delegated in writing with acceptance criteria — or one that must come to me?** Define in advance what a checking agent may approve, and what always requires your signature.
6. **Is this one item — or a bundle?** Approve single items that were checked; never bundles hitching a ride together.

"Shipped" is not "verified"

Silence is not permission

"The AI said so" is not a defense

The second layer • The field guide

In Chapter 2 we read the Replit case as a story about roles. Here we'll read it as a story about a way back.

Field case 3, second reading • Who knew there was a way back (Replit, July 2025)

One of the most troubling details in Jason Lemkin's case isn't the deletion itself. It's what happened after it: the agent told Lemkin there was no way to restore the database — and that turned out to be untrue. The restore worked. And the explanation Replit published afterward said the agent simply didn't know the restore capability existed.

Now imagine Lemkin had believed it. Someone convinced there's no way back makes different decisions: rebuilds from scratch, tells customers about data loss, gives up. The damage from the wrong report could have been greater than the damage from the deletion.

The lesson: knowing the way back has to be yours — written down, tested, and not dependent on the tool that did the damage. The approver's Question 4 ("How do we undo it?") needs an answer before the action, from a source that isn't the agent. [3]

The third layer • The lab

Two experiments: gather the assets, and practice one exit.

Lab 25: An asset inventory 30–45 min

Gather into one place — a folder, a document, a repository — everything you built on top of the tools. The goal is that if you opened only this folder, you could rebuild the setup in any tool.

- **Step 1:** For each asset in the table above, check: does it exist outside the tool? Where?
- **Step 2:** Every asset that exists only inside a tool — copy it out, with a date.
- **Step 3:** For each asset, note who's responsible for updating it when it changes in the tool.
- **Step 4:** Write a one-page "read me" file that explains what's in the folder and how to use it.

Stop operating.
Start approving.

Only With My Approval

How to manage an AI team without losing control.

English digital edition. 121 pages.

15 chapters, 30 hands-on labs, 14 decision cards.

Sample: book pages 12, 21 and 100.

Chapter 1 opening, Decision Card 1 and a field case with Lab 25.

Get the full book: $19.90

Launch price through Nov 30, 2026. Taxes may apply.

Regular price $49 from December 1.