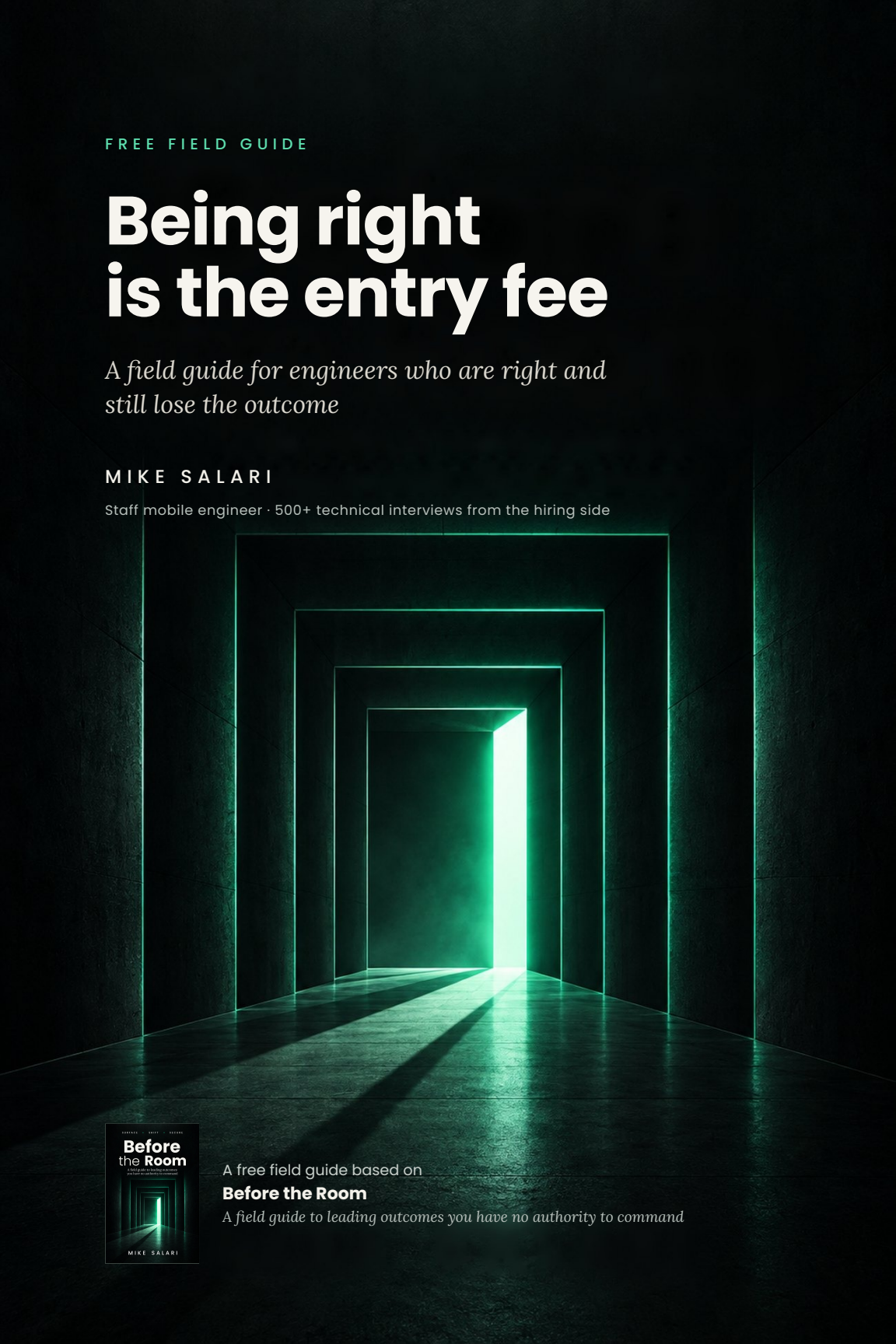
FREE FIELD GUIDE

Being right is the entry fee

A field guide for engineers who are right and still lose the outcome

MIKE SALARI

Staff mobile engineer · 500+ technical interviews from the hiring side



A free field guide based on
Before the Room
A field guide to leading outcomes you have no authority to command

THE DEBRIEF

The best engineer in the loop was downlevelled

> A composite scene from chapter 1: the pattern is real, the details are illustrative.

The debrief runs for twenty minutes and the candidate is not in the room. Five of us are.

He had interviewed for a staff role, and on the whiteboard an hour earlier he was the strongest engineer any of us saw that week. He caught a consistency bug in the design before the interviewer finished drawing it. He knew the failure modes of the datastore cold. When someone pushed on his caching call, he defended it with numbers, and he was right to.

Then the hiring manager says the sentence that ends it.

> *"He was clearly the best engineer in the loop. I just do not see him moving anything at this level."*

Nobody argues, and the loop downlevels him. The verdict had formed in one moment of the system design round, when the interviewer asked what he would do if the team owning the upstream service refused to make the change his design depended on. He said:

> *"I would explain why they are wrong."*

He meant it as strength. He heard a test of whether his design was correct. The question was whether he could move an outcome that lived inside a team he could not command, and his answer told us that when he hit that wall, he would push harder on the wall.

You are likely to meet that question at work long before you meet it in a loop. It rarely announces itself as a question.

From chapter 1 of *Before the Room*. Every scene in this guide is a composite from the book.

THREE WEEKS YOU MAY RECOGNIZE

Correct, prepared, and too late

The review that was already decided

A senior engineer spends three weeks on a proposal to consolidate two services. It is good. By Wednesday night, three of the five people who matter have privately settled their positions, none of them in a meeting. On Thursday she gives the best version of her case to a room that has already decided.

She prepared to win the meeting. The decision was being made before it.

The review that was won, then reopened

An engineer makes the case for a service boundary, fields the objections, watches heads nod, and starts building the next morning. On Thursday a principal who was not in the review posts a long, reasonable comment that reopens the question. Two of the people who nodded now nod along with him.

The meeting gave him an appearance of agreement. It did not give him the right to decide.

The status update that became a promise

A director asks whether the release is still on track for the fifteenth. The engineer sends a careful answer: yellow, trending green, two risks with mitigations. Ten days later the fifteenth is a hard commitment to a partner. He never agreed to one.

He answered the question. The decision underneath it was being made by someone else.

Three different weeks, one pattern. Each engineer was careful about the work and wrong about where the decision lived.

From chapters 3, 7 and 4 of *Before the Room*.

WHY GOOD INSTINCTS MAKE IT WORSE

The three moves that feel like integrity

When engineers first hit this wall, many reach for one of three tools. From the inside, all three feel like integrity. None of them touches the decision.

Be more right	More data, a cleaner diagram, a tighter proof. When the resistance comes from an incentive, a better argument can make the proposal look more threatening instead of more persuasive.
Work harder	Carry the outcome yourself when it will not move through the system. It scales until it does not, and it teaches the organization that nothing needs to change.
Private frustration	Decide the org is broken and the other teams are irrational, and retreat into being correct and unheard.

The first one deserves a closer look. Inside one team that owns its own trade-offs, a disagreement about facts can often be settled with better facts. Across a team boundary, the other person carries costs you cannot see: a roadmap already promised upward, an on-call load that is finally survivable, the memory of a previous failure. You keep sending knowledge, and the gap does not close, because what separates you is not knowledge.

> When a competent person resists a good technical proposal, they are often being rational from where they sit. The resistance is their reasoning, applied to costs and stakes you have not looked at.

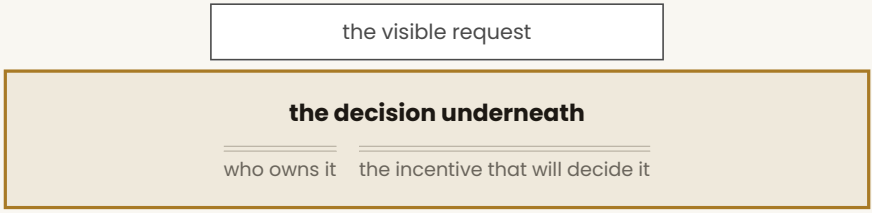
From chapters 1 and 6 of *Before the Room*.

THE IDEA THE BOOK RUNS ON

The decision is made before the meeting

Many consequential decisions are substantially shaped before the visible event. The architecture call, the migration go-ahead, the deadline, the launch: the meeting where they appear to be decided often sits on positions formed earlier, in hallways, review comments and, above all, incentives.

So most requests arrive as the top of a stack. “Can you confirm we are on track” sits on a decision about what counts as on track. “Review this design” sits on a decision about who owns the data model afterward.



the owner is often not you

The book turns this into three moves, always in this order.

Surface	Find the decision that is being made, who owns it, and the incentive that will decide it.
Shift	Change that decision with one artifact and the exact words to say, in front of the right owner, before the room.
Secure	Set the threshold that triggers escalation and the follow-up signal that proves the change held.

You cannot move a decision you have not located. What follows gives you Surface, complete.

From chapter 3. The diagram is figure 3 of the book (chapter 4), redrawn for this page.

THE TECHNIQUE

Three cuts, before you answer

When a request lands and you feel the pull to answer it well, run three cuts first. Each takes a sentence.

1. **What decision is this request scaffolding?** Ask what someone would have to decide for this request to make sense. Behind "can you review this design" often sits "who gets paged at three in the morning when this fails, and does that team know yet." Phrase it as a choice between named alternatives. If you cannot, you have not found it yet.

2. **Who owns that decision?** Name a person or a role, never "the team" or "leadership." Frequently it is neither the person who asked nor you. If you cannot name the owner, that is the most important thing you have learned.

3. **What incentive will settle it?** The roadmap someone is protecting, the on-call load someone is avoiding, the credit someone is guarding. State it without polish. You are trying to understand what the owner needs, never to find where to press.

Run the three cuts and you have the decision, its owner and its driver before you have argued, proposed or built anything. Until the cuts run in your head, write them down, and copy the request word for word: tidying it is where you smuggle in your own assumption.

YOUR FOUR LINES, BEFORE YOU REPLY

Request	*in its exact words*
Decision	*as a choice between named alternatives*
Owner	*a person or a role*
Incentive	*stated without polish*

From chapters 3 and 4. The book's version is template 1, the hidden-decision worksheet.

WORKED EXAMPLE 1

The status question at 4:12 on a Tuesday

DIRECTOR SPONSORING THE RELEASE

"Hey, can you confirm we are still on track for the fifteenth?"

DECISION	Whether to turn a soft internal target into a hard external promise.
OWNER	The director sponsoring the release, not the engineer confirming status.
INCENTIVE	How much the partner relationship matters to him this quarter, and what slipping would cost him.

BEFORE

"Yes, we are on track, two risks but both have mitigations, should be fine for the fifteenth."

AFTER

"Before I confirm, are you deciding whether to commit this date to someone outside the team? My confidence for an internal target and my confidence for a hard external promise are different numbers, and I want to give you the right one."

Once the director has to answer that, a status update can no longer be spent as a commitment the engineer never made.

It works when nobody is hiding anything, too. Asked for a feature by end of month, surface the trade-off instead of absorbing it: "*A month on search means the accessibility work and the two carryover bugs slip to next cycle. That is a fine trade if search matters more right now, but it is a trade, and I do not think it is mine to make. Whose call is that?*"

From chapter 4. The three rows come from the completed example of appendix template 1.

WORKED EXAMPLE 2

The team that keeps saying no

Take a version of the same refusal at work. It sounds like this, three times in three weeks:

> **STAFF ENGINEER ON THE OTHER TEAM**
> *"I just do not think we are ready to take this on right now."*

Each time, the proposing engineer answers with more: benchmarks, a cleaner diagram, proof the migration is smaller than it looks. Each answer makes the staff engineer more polite and less movable, and by week three the proposer is planning to route around him.

DECISION	Whether his team takes on a migration whose benefit lands on another team and whose cost lands on his.
OWNER	Him, answering to the director both teams report to.
INCENTIVE	Last year his team shipped a migration that slipped two quarters in front of that director. Another slip is a bet on his standing.

> **AFTER**
> Ask for his private math: "What would this migration cost your team specifically, in the next two quarters, given everything else you are carrying?" Then name the cost with respect: "You ate a migration that slipped two quarters last year, so of course you are wary of another one."

Now the redesign has a target: let his team go last, with a funded wave and a readiness gate. Expect a partial yes. A good outcome is that he still says in the review he would rather not be doing this now. That is enough to proceed on. If the answer is still no, take it to the person who owns both roadmaps, as a trade-off and never as a complaint about a person.

From chapter 6. The owner row is this guide's reading of the chapter's scene.

USE IT THIS WEEK

Sometimes the decision hides across three requests

In one week, three reasonable questions arrive:

MOBILE LEAD
Can the new endpoint tolerate older app versions?

SUPPORT
Can the old order screen stay for another quarter?

PRODUCT MANAGER
Can the widget work start before the backend is final?

READ TOGETHER
Is the organization going to maintain two versions of the order flow for the next six months, and who pays for that?

Answer each on its own and the two-version world arrives by accumulation, with no owner and no budget. Run the same cuts on the pattern: write the requests side by side, name the decision they share, and take it to the one person who can own it.

This week: pick the request that would commit someone to something, or that makes your stomach tighten as you read it. Write the four lines, hand the decision to its owner, and check in a week that the owner is the one carrying it.

WHERE THE MOVE IS WRONG

Not every request hides a decision. Treat every message as a buried commitment and you become the engineer who cannot answer a simple question without a discovery meeting. And the objection may simply be correct: in chapter 6, a warning about event ordering that an engineer read as an incentive turns out, once he replays a day of production traffic, to be a correctness bug. When an objection names a failure mode and a test, run the test first.

From chapters 4 and 6. The four lines are template 1 of the book's 16 artifacts.
See the book on Gumroad · on salari.dev

WHERE SURFACE STOPS

Finding the decision is the first third of the job

Surface tells you what is being decided and by whom. It does not move the decision, and it does not keep it moved. Run the consolidation proposal from the first of those three weeks again, with all three moves.

> **MONDAY**
> She names the decision: the director's ownership of the data model, driven by his need to keep his team essential.
>
> **TUESDAY**
> She brings him a one-page memo showing how consolidation could make his team the owner of the new platform. The staff engineer who had decided not to back her gets it early enough that his concerns can change the proposal instead of ambushing it.
>
> **WEDNESDAY**
> The director agrees to hand over the data model, but not this quarter. His team has a launch to land first.
>
> **THURSDAY**
> The sponsor decides in the open: approved, start date pushed a quarter. She writes the delay into the decision log as the price of the yes, with a threshold: if the director's launch slips, the date goes back to the sponsor instead of drifting.

Same engineer, same proposal. She won the choice and lost the date, and the duplicated on-call load she wanted gone stays. Shift got her the yes. Secure is there to keep the delay from turning into drift.

The harder questions start here. What do you put in the memo? Who do you see first, and what do you say to the one person who loses something? What stops Thursday's yes from reopening in a chat thread next month?

From chapter 3 of *Before the Room*.

THE REST OF THE BOOK

The next sentences are harder

Surface hands you the decision. The rest of *Before the Room* is about what you say once you have it, to whom, and what you put in writing so it holds. A few of the lines its chapters build toward:

> **WHEN THE REVIEW ENDS AND EVERYONE NODS**
>
> "*Before we call this closed, who owns this decision, and who could reopen it who is not in this room?*"
>
> Chapter 7 · Decision-rights map

> **WHEN ANOTHER TEAM KEEPS BLOCKING A SOUND PROPOSAL**
>
> "*This proposal moves ownership, and that is the real decision, not the architecture. I want your team to come out of it owning something that matters. Here is how that could look.*"
>
> Chapter 5 · Stakeholder and incentive map

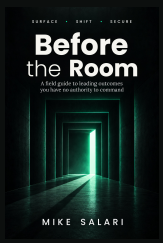
> **WHEN A DATE WAS PROMISED BEFORE ANYONE SIZED THE WORK**
>
> "*Engineering sizes the risks and recommends one. You own which promise we make. Which do you want?*"
>
> Chapter 12 · Three-option scope memo

> **WHEN EVERY TEAM REPORTS GREEN**
>
> "*The work is locally complete, but the launch path is not proven.*"
>
> Chapter 14 · Outcome evidence table and truth test

> **WHEN A RISK IS STUCK BETWEEN TWO TEAMS**
>
> "*If the endpoint is not owned and scheduled by Monday, I will take a brief to the sponsor to decide, with both managers consulted. I wanted you to hear it from me first.*"
>
> Chapter 15 · Escalation brief

A line works only when the work behind it is done. Each chapter supplies that work: the artifact that carries the decision when you are not in the room, the signal that tells you it held, and the boundary where the move stops applying. Three sections show the method failing, and what to do then.

Lines quoted from the book's operational index. 20 chapters, 16 artifacts, each with a completed example.



You can find the decision now. The rest of the book is about moving it, and making it hold.

The same discipline, through the proposal another team blocks, the review that reopens, the deadline no evidence supports, and the escalation that must not become a fight.

The difference: the book teaches the artifacts; the Field Kit gives you them as files to fill in on your next real decision.

Book

Current price on the store

Read the system

- The complete book, PDF and EPUB
- 20 chapters, 137 pages
- 16 artifacts, each with a completed example
- 12 figures and an operational index

MOST COMPLETE

Book + Field Kit

Current price on the store

Run it at work

- Everything in Book
- All 16 artifacts as fillable PDFs, A4 and US Letter, and as Word files
- Field Kit Workbook and Thirty-Day Repair Workbook
- Field Card, Problem to Artifact Map, When the Moves Fail, AI Companion

Buy on Gumroad | **See it on salari.dev**

All editions are on one page; you pick yours before you pay.
Buying for a team? A Team License for up to 10 readers in one organization is on the same page.

Mike Salari · Staff mobile engineer · 500+ technical interviews from the hiring side