You found the perfect finance platform. You can't get it approved. Here's why.

Evaluation and approval are two different games with two different rulebooks. Most champions only ever learn the first.

Most finance leaders have lived this. You evaluate a platform, and it clicks. It solves a problem that's been costing your team nights and weekends — the multi-entity close, the intercompany mess, the reconciliation that never reconciles. You're convinced. You bring it forward.

And then nothing happens. Not a no. Just a slow, procedural death somewhere between your desk and the people who sign. Three months later the platform you loved is a browser tab you keep meaning to close.

The frustrating part is that the platform wasn't the problem. The *approval* was. And approval runs on a completely different logic than evaluation — one most champions never learn, because they assume a good product sells itself. It doesn't. Someone sells it, or it dies.

This is written for two people: the finance leader trying to get their platform approved, and the vendor who wants their champion to actually win. Same problem from both sides.

Why good tools fail to get approved

The reasons are rarely about the product. They're about the shape of the decision.

- **You solved your problem, not the company's.** You feel the pain daily — the close eating your month-end. The person who signs doesn't feel it; to them it's an abstraction on a slide. Framed as “this makes my life easier,” it competes with every other function's life-easier request and loses. Approved tools are framed as company problems — risk, audit exposure, headcount avoided, a close blocking the board timeline.

- **There's no number attached.** “It'll save us time” is a feeling, not a business case. Approvers sign against numbers: hours reclaimed × loaded cost, the FTE you avoid hiring, the audit finding you avoid. No quantification, no approval — because they've nothing to defend the spend with when their boss asks.
- **You're the only person who wants it.** A single champion is a single point of failure. Deals that get approved have a coalition — IT comfortable with the security posture, an ops peer who benefits, a finance peer who'll nod. One voice is a preference. Three voices is a mandate.
- **Procurement and security were an afterthought.** You fell in love with the product; you didn't check whether it clears vendor-security review, data residency, the SOC 2 gate. The deal hits a wall you didn't know existed and stalls in a queue with no urgency behind it.
- **It looks like a new line item, not a swap.** Every new tool is net-new spend — unless framed against what it replaces: the spreadsheet army, the contractor you rehire every close, the module you're overpaying for. Substitution gets approved far more easily than addition.
- **The timing is invisible.** “Whenever” is not a reason to act now. Without a deadline the company already cares about — the audit, a new entity going live, the fundraise, a migration — the decision has no forcing function and drifts to next quarter forever.

How to actually get it approved

The fix isn't a better demo. It's re-shaping the decision so it survives contact with the people who sign.

- **Translate your pain into their language.** Rewrite the problem from the approver's seat. Not “the intercompany close is killing my team” but “we have a manual process that's a material-weakness risk and a bottleneck on board reporting.” Same tool, different problem — theirs, not yours.
- **Bring one number they can defend.** Not a forty-line ROI model — one credible, conservative figure they can repeat upward without embarrassment. “Roughly

two FTE-weeks per close, across twelve closes, is most of a headcount we're about to have to hire.”

- **Build the coalition before the ask, not during it.** Get IT to pre-bless the security posture. Make the peer who also benefits a co-sponsor. By the decision meeting, the answer should already be socialized — you're confirming consensus, not starting an argument.
- **Clear the procurement path early.** Ask the vendor for the security package — SOC 2, data residency, DPA — on day one and hand it to security before you're committed. A tool that's already cleared review can move the moment budget says yes.
- **Frame it as a swap.** Name what it replaces, in dollars — the contractor, the overtime, the redundant module, the spreadsheet risk. “We're not adding cost, we're moving it from something worse to something better” is the sentence that unlocks budgets.
- **Attach it to something already urgent.** Don't create urgency — borrow it. Tie the tool to a deadline the company already owns. The decision stops being “should we buy a tool” and becomes “how do we hit this date,” which is a question that gets answered.

For the vendor: your job is to arm the champion

If you sell to finance, your champion will walk into a room you'll never see and win or lose it without you. Your real product, past the demo, is *making them win that room.*

That means handing them the business-case number, pre-built and conservative. The security package, ready before they ask, so procurement never becomes the reason. The one-paragraph internal email that reframes the problem in CFO language, the swap framing with line items named, the two peers most likely to co-sponsor. The champion who loves your product but walks in unarmed will lose to a worse product with a better internal seller. Your job is to make sure the best tool is also the best-sold one — from the inside.

The uncomfortable truth

The tools that win aren't always the best tools. They're the best-*sold* tools — sold internally, by a champion who understood that evaluation and approval are two different games with two different rulebooks.

The good news: the second game is learnable. Reframe the problem as theirs. Attach one defensible number. Build the coalition early. Clear procurement before you're committed. Frame it as a swap. Borrow urgency that already exists.

Do that, and the platform you love stops being a browser tab you keep meaning to close.

> *If you've ever lost an approval for a tool you knew was right, it probably wasn't the tool. It was the case. And the case is the part you control.*