

The Ultimate Guide to B2B eCommerce Replatforming



Table of Contents

Introduction	3
Is It Time to Replatform?	4
Common Pitfalls: Why Most Replatforming Projects Fail	7
How to Replatform Without Regret	11
Don't Sleep on Data Migration	18
Don't Let Replatforming Tank Your SEO	19
The Launch Plan: Don't Wing It	20
Post-Launch Reality Check: What Happens After the Plan	22
Replatforming Is a Strategic Reset	25

Introduction

If you picked up this guide, chances are your current platform isn't doing what you need it to do.

Orders are clunky. Your team is drowning in workarounds. Customers are frustrated. And you're wondering whether to patch things up or start fresh.

You're not alone. And you're not overreacting.

This guide is here to help you figure it out:

- The red flags that mean it's time to replatform
- What traps cause most replatforming projects to stall or fail
- How to approach platform selection based on how your business works
- What to plan before launch so you don't scramble after
- And how to keep the momentum going after the new site goes live

Let's get into it.



Is It Time **to Replatform?**

Replatforming isn't something you do on a whim. It's expensive. It's disruptive. And if you don't need to do it, you shouldn't. But how do you know if your issues are fixable or if your platform is fundamentally holding your business back?

Here's a breakdown of the signs to watch for, some obvious, some easy to overlook.

Red Flags You Can't Ignore



Your customers are calling instead of clicking.

If your sales or support teams are fielding calls just to help customers place basic orders, that's a problem.

Maybe your site is hard to use. Maybe it lacks quick reorders, CSV upload, or mobile support. Either way, buyers are telling you your site doesn't work for them.



Your marketing team is stuck.

They can't build landing pages without IT. They're manually duplicating data across systems.

They can't personalize content or campaigns. The tech stack isn't supporting modern marketing, it's fighting it.



Your analytics look... sad.

Site traffic isn't the full story. The real insight comes from what people do after they land, and that's where underperforming platforms tend to reveal themselves.

Here's what to dig into:

High bounce rates

If visitors are bailing after one page (especially on category or product pages) it's a red flag. Maybe pages load too slowly. Maybe navigation is confusing. Either way, something's pushing people out before they even start browsing.

As a general guide:

A bounce rate between

26%–40%

is strong

A bounce rate between

41%–55%

is average

Anything above

56%

deserves a closer look

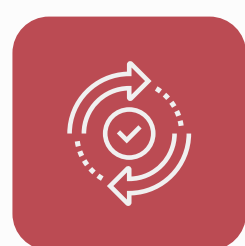
Context matters, but consistently high bounces on commercial pages usually point to a poor experience.

Longer time-on-site but no conversions

If session duration is climbing but checkouts aren't, your site might be frustrating, not engaging. Visitors could be endlessly clicking through menus, filters, or irrelevant results, hunting for something that should be easy to find.

Overreliance on search for simple tasks

If your search bar is flooded with terms like “invoice,” “reorder,” or “login,” it usually means users can’t find what they need through normal navigation. That’s a UX failure, especially in B2B, where speed and clarity are everything.



Your tech team dreads updates.

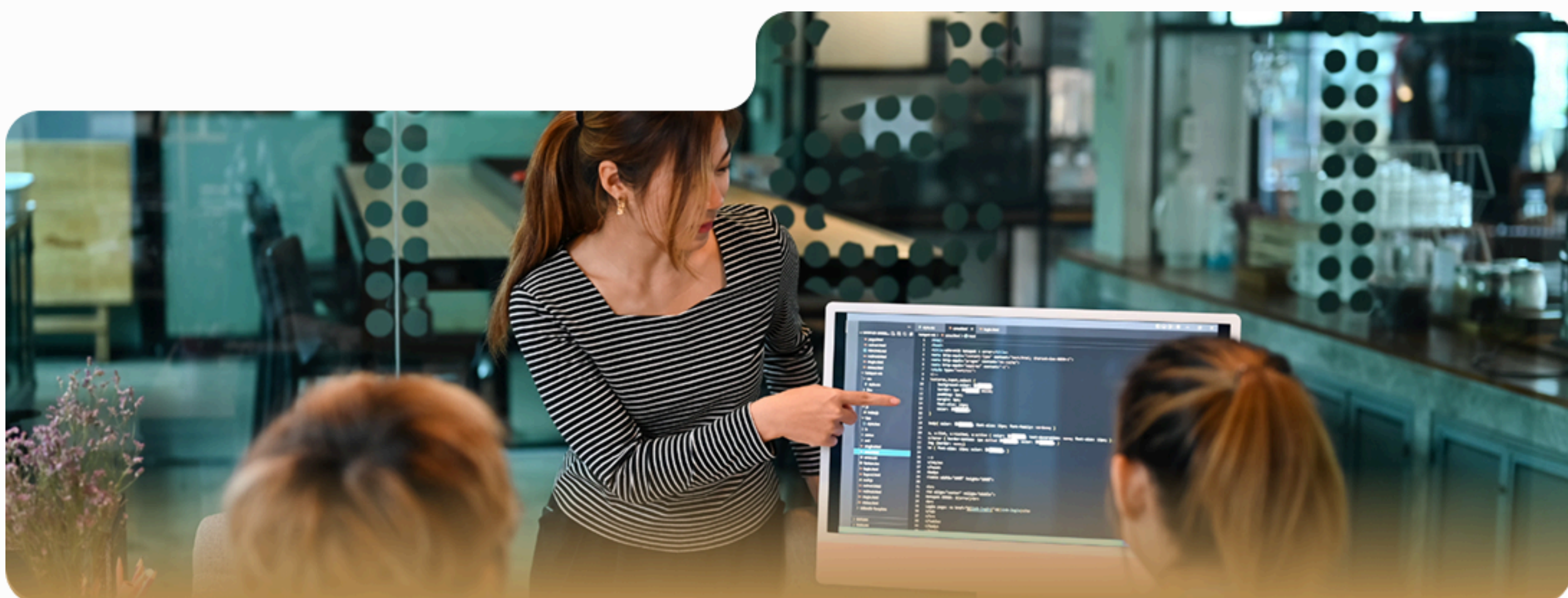
If the phrase “We have an update coming” causes groans in the IT department, that’s not a good sign.

Security patches shouldn’t break your checkout. Minor plugin updates shouldn’t crash your product catalog.

But with aging platforms – or ones cobbled together with too many plugins and custom workarounds – that’s exactly what happens. Every small tweak becomes a gamble.

Routine tasks like adding a new integration or modifying checkout logic can take days (or weeks) because they trigger conflicts with something else in the stack. Over time, your platform starts to look and behave like a Jenga tower. One wrong move, and the whole thing wobbles.

If your IT team is spending more time troubleshooting than improving, it’s a clear signal: you’re not scaling anymore, you’re just surviving.



Common Pitfalls: Why Most Replatforming Projects Fail

Replatforming is one of the most important digital decisions a B2B business can make, and one of the easiest to get wrong.

Not because the teams are lazy or the tech is bad, but because replatforming is deceptively complex. The biggest mistakes happen before a single line of code is written.

Here's where things usually go sideways:

1. You picked a platform based on someone else's success.

Maybe a peer recommended it. Maybe it crushed the vendor demo. Maybe it showed up on a Gartner chart and felt like a "safe" choice.

But unless you map that platform against your own sales workflows, catalog structure, and customer behavior, you're flying blind.

The smartest replatforming projects start by zooming in, not out. **Ask:**

How do our buyers place and approve orders?

What exceptions do we deal with every day?

What does a successful order really look like for us?

2. You evaluated features, not fit.

Once you've got a few platforms on the table, it's tempting to fall into the RFP checklist trap. But checkboxes don't capture how things work in practice.

You need to go deeper:

- Can your team build and maintain those workflows without an army of developers?
- How does the platform handle scale: hundreds of thousands of SKUs, multi-step approvals, customer-specific pricing?
- What trade-offs are hidden under the hood? (e.g. rigid data models, clunky CMS, limited APIs)

A platform can *technically* check every box and still be a poor fit operationally.

Compare Top B2B eCommerce Platforms

[Explore the Comparison Chart](#)

3. You didn't get buy-in early enough.



Even if you pick the right platform and validate the workflows, the project can still stall or fall flat if it's owned by one team in isolation.

IT might lead the implementation, but they're not the only ones it affects.

- Sales needs quoting tools, reorder visibility, and flexible workflows.
- Marketing needs landing pages, segmentation, and personalization tools that don't require dev time.
- Customer service needs accurate account views, order history, and the ability to act fast.

If those teams aren't in the room early (during requirements gathering, vendor conversations, and scoping) you risk launching a platform that technically works, but doesn't work for them.

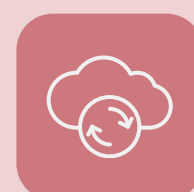
That's when adoption tanks. Features go unused. Workarounds creep back in. And the shiny new platform quietly turns into expensive shelfware.

4. You underestimated the migration itself.

Replatforming isn't just flipping a switch and hoping for the best. It's a full-scale operational shift, one that touches nearly every part of your business. **Under the hood, it means:**



Untangling years of patches, plugins, and one-off customizations



Syncing ERP, PIM, CRM, and payment integrations, without breaking existing workflows



Planning, testing, and auditing URL redirects to protect your SEO



Migrating customer accounts and permissions safely



Mapping and cleaning product data across systems

5. You treated it like a one-time project.

Too many teams treat launch day as the end of the road. No clear owner for post-launch optimizations. No roadmap for feature rollouts. No time carved out to learn how the new system performs under real-world pressure.

And when that happens? Adoption stalls. Teams fall back on manual workarounds.

The platform gets blamed for problems it was never set up to solve.

A successful replatform doesn't just go live – it keeps evolving. We'll dig into how to set up a post-launch roadmap and keep momentum going later in the guide.

How to Replatform Without Regret

If you've made it this far, you've probably seen enough to know replatforming is the right move or at least a strong possibility.

But knowing you need to replatform and knowing how to do it right are two very different things.

Here's how to avoid the most common mistakes and build a platform that works for your customers, your teams, and your future growth.

Step 1: Map How You Actually Sell

Before you even look at vendors, you need to understand your own business inside out. Skip this step, and even the most advanced platform will disappoint you.

Start with One Core Flow

Pick a typical, high-value order from a real customer. Then walk it through from beginning to end.

- Where does the order originate? (Website? Sales rep? Punchout?)
- What happens before the order is submitted? (Pricing checks? Approvals?)
- Who's involved? (How many people on the customer's side? Your side?)
- How is it fulfilled, shipped, and invoiced?

Get that baseline mapped out visually on a whiteboard, Lucidchart, even a napkin sketch. Just make it visible.

Then Document the Variations

Once you've got the typical flow down, zoom out. Look for patterns and exceptions:

- Do some customers pay on terms? Others by card?
- Are there order minimums or quantity rules?
- What's different when a rep places an order vs. self-service?
- Are reorders or subscriptions handled differently?

This is where most B2B teams uncover messy processes they've been tolerating for years, and where the real requirements live.

The created raw map becomes your truth. It cuts through vendor hype, internal assumptions, and legacy bias. It shows you which features matter and which ones you can ignore. It helps you spot which platforms are genuinely worth your time.

It gives structure to your RFP, so you're not just repeating boilerplate language. And most importantly, it creates a clear, shared definition of what success should look like, before anyone starts building anything.

Save hours writing your RFP.
Start with a proven structure.

[Download RFP Template](#)

Step 2: Prioritize Problems, Not Features

Once you've mapped how your business sells, it's time to shift your focus: from possibilities to pain points.

Every vendor will hit you with a long list of features. Many of them impressive, some of them irrelevant, and a few that sound good but solve problems you don't have.

Start by identifying your biggest operational blockers. Not in abstract terms – specifically. What's costing your team hours each week? What frustrates your customers the most? What's the workaround you've quietly accepted as "just how it is"?

Maybe your buyers constantly complain they can't find products, even when they're in the catalog. That's not a "better search" checkbox, it's a sign your filters, taxonomy, or data structure might need a complete rethink.

Or maybe your sales team spends half their week manually quoting orders that should be self-service. That's not solved with "bulk pricing support". It requires robust RFQ workflows, real-time pricing visibility, and user roles that match your customer org charts.

When marketing is blocked from launching campaigns without IT, it's not about CMS vs. headless. It's about whether your platform empowers non-technical users to do their jobs: segment audiences, personalize content, spin up landing pages.

Same with IT: if they're drowning in one-off integrations, you don't need a "strong API", you need a platform built to integrate, not duct-tape.

Don't evaluate platforms based on what they can do. Evaluate them based on what they can unblock.

Once you've prioritized your biggest frictions, your list of must-haves will practically write itself and you'll be able to spot platforms that are a genuine fit, not just a good demo.

Step 3: Vet for Fit, Not Flash

Once you've zeroed in on your biggest blockers, it's time to see if any platform can actually solve them. But remember, the real value isn't in what a platform can do, it's in how well it handles your business. That's a critical distinction.

Instead of passively watching a demo, come prepared. Build a scenario script based on the workflows you mapped earlier. Make the vendor walk you through the exact use cases that matter to your team. If they can't or if the answer is a vague promise or a custom module "down the road" – that's a red flag.

Here's what that might look like in practice:

Data sync scenario

"We update product specs or pricing in our ERP daily. How does your platform keep everything current without manual uploads or delays?"

Ordering Scenario

"A buyer with multiple locations and payment terms logs in. How do they reorder across locations? Can they save preferences or get tailored pricing per site?"

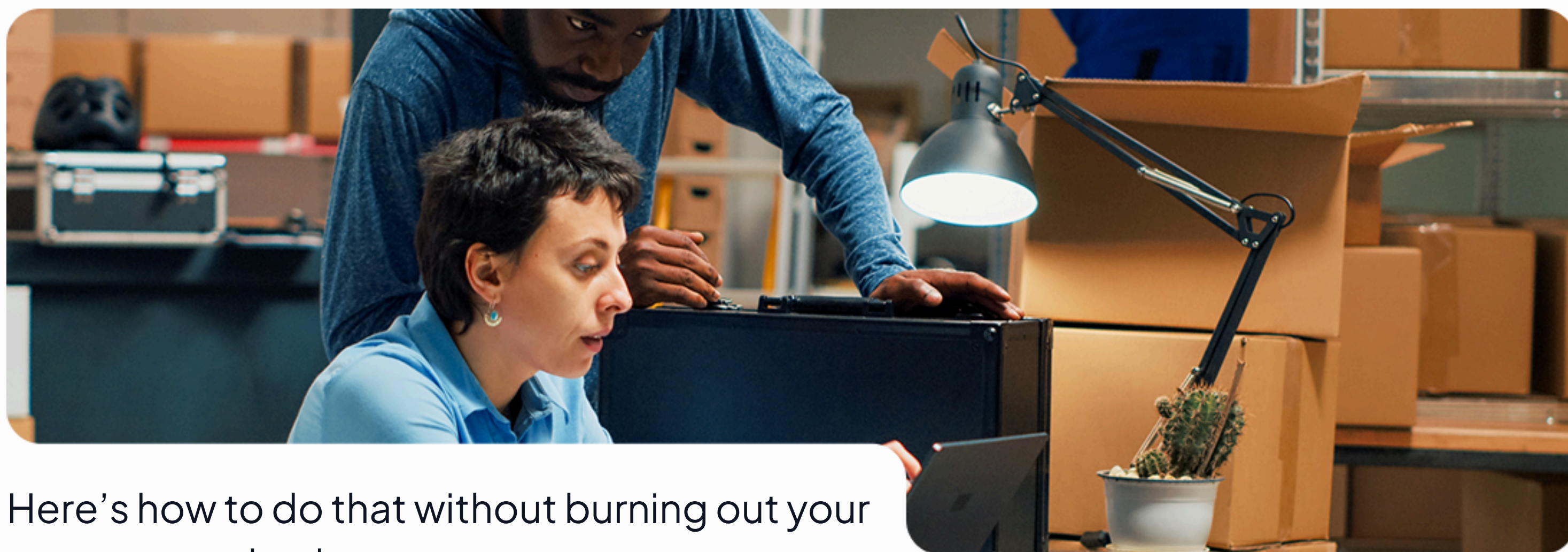
Price management scenario

"We have different pricing tiers based on customer groups, contract terms, and order volume. How does the platform manage that and what happens if we need to make a pricing update across multiple SKUs or customer segments at once?"

Also, bring your team into these evaluations. Have sales, marketing, and ops each ask one scenario-specific question. Their real-world perspective is the best defense against platform regret.

Step 4: Plan for More Than Launch

Even the best-planned replatforming project will launch with a few rough edges. That's normal. But what separates long-term success from quiet failure is whether you've built a plan to keep improving after the ribbon-cutting moment.



Here's how to do that without burning out your team or your budget:

Build a Post-Launch Backlog Before You Launch

There will always be things you push live without: features that got cut for scope, ideas that came too late, or nice-to-haves that weren't mission-critical.

Instead of letting those disappear into a black hole, log them in a shared backlog as you go.

Lifehack: Set up a simple Trello or Notion board with columns like "Later," "Nice to Have," and "Customer Requests." Tag owners. Prioritize after launch.

Assign Real Owners, Not Just Departments

If "marketing" owns the CMS or "IT" owns the platform, no one owns it. Assign named individuals to major workflows (product data, promotions, quoting, customer service).

Give them clear responsibilities and the authority to make decisions post-launch.

Lifehack: During User Acceptance Testing (UAT) or soft launch, have each owner run a test case. If they struggle now, they'll definitely struggle later.

Make Training a Non-Negotiable

Most replatforming failures aren't technical, they're adoption problems. If your teams don't know how to use the new tools, they'll default to old habits.

What to do:

- Schedule hands-on training sessions by role (sales, support, marketing, not just “users”)
- Create simple how-to docs and screen recordings
- Assign “platform champions” who can help onboard others

Lifehack: Record a short “here’s what’s changed” video walkthrough post-launch. Share it with internal teams and key customers if the buying experience changed.

Define Success – and Measure It

How will you know if the new platform is working? Decide that before you launch. Start by aligning your KPIs with the outcomes your business cares about. That sounds obvious, but too many teams grab a list of “standard” eCommerce metrics without asking why they matter.

Ask each department what success would look like six months after launch. For sales, it might be more self-service reorders. For marketing, faster campaign launches. For customer service, fewer order-related tickets. From there, pick 3–5 core KPIs that map to those outcomes, not vanity metrics, but indicators of real improvement.



Once you've aligned on what matters, define your baseline. Capture how things work today so you can measure impact after go-live.

Some examples to consider:

How long does it take a customer to place a typical order?

Are bounce rates improving?

Are support tickets related to order issues trending down?

How are conversion rates tracking pre- and post-launch?

What percentage of orders come through self-service vs. sales-assisted channels?

Are repeat orders increasing?

Some goals will be too optimistic. That's fine. You're not locked in. Let's say you projected a 50% lift in conversions, but post-launch analytics and customer feedback suggest a 25% gain is more realistic. That's not failure – it's calibration. Just update the target and move forward. Better to adjust with data than cling to a number that no longer fits.

Lifehack: Create a “30/60/90 Day Scorecard” and review it in your post-launch debriefs. Small wins stack up and small issues can be caught early.

Don't Sleep on Data Migration

Data migration is one of the most underestimated parts of any replatforming project, and one of the easiest places for delays to sneak in. Here's what to think through:

Product Data

Expect to migrate more than just product names and SKUs. You'll need to account for descriptions, specs, images, categories, pricing, metadata, and more. This is also your best chance to clean things up: optimize SEO fields, unify naming conventions, or remove dead products.

Customer Data

At minimum, you'll want emails, names, and addresses. You may also need customer groups, tax statuses, and saved payment methods (though some platforms won't allow encrypted data like passwords or credit cards to be transferred directly). Plan early for how you'll handle customer logins post-launch and how you'll communicate any reset steps to users.

Order Data

B2B reorders and account accuracy rely on order history. Migrating it means aligning customers, products, and transactions – no small task. Decide upfront if full history is essential, and confirm legal retention obligations (e.g., GDPR).

Content

From landing pages to blog posts, non-product content often gets overlooked until it's go-time. If you're redesigning as part of the replatform, most content will need to be rebuilt manually to match the new templates. Articles and blog posts are the exception, those can often be imported more easily if the structure stays similar.

Don't Let Replatforming Tank Your SEO

You don't need to be an SEO expert to protect your traffic, but you do need to take it seriously before launch. Too many B2B businesses find out the hard way that organic traffic can drop fast after a replatform and recovering it takes months.

The good news: nearly every SEO risk during a replatform is preventable. But only if you prepare ahead of time.

The most important step? Know what's working now before you change anything. Which pages bring in organic traffic? What URLs are currently ranking? Which keywords drive leads? That's your baseline.

Then, build a redirect plan. If any URLs are changing (especially product, category, or blog pages), map each one to its new location. Don't rely on plugins or shortcuts. Check that redirects actually work – broken links can cost you traffic fast.

Be especially careful with duplicate content. If your new platform generates multiple URLs for the same product or category (which happens more than you think), you'll need proper canonical tags to prevent dilution. Canonicals tell search engines which version of a page is the "real" one. Without them, Google might index duplicates or ignore them altogether.

Also, don't forget your internal links. Make sure all on-site links point to the new URLs, not the old ones. Redirects can catch this temporarily, but internal links are a strong ranking signal and you want them clean from day one.

Finally, prepare your sitemap and robots.txt. Once you launch, resubmit your XML sitemap to Google Search Console and Bing Webmaster Tools. And closely monitor crawl errors, indexing, and traffic for at least 30 days post-launch.

SEO losses often don't show up right away. But if you prep the right way, your traffic won't just survive the replatform, it'll give you a stronger base to grow from.

The Launch Plan: Don't Wing It

When replatforming to a new eCommerce platform the pressure is high for good reason. You're not just pushing a new design live, you're migrating complex data, preserving organic traffic, keeping customer workflows intact, and hoping nothing collapses under load.

So don't wing it. Build your launch plan like you'd build your go-to-market strategy: deliberately, collaboratively, and with buffers for the unknown.

Stabilize Before You Ship



Before your new platform touches production, it should already be functioning like your live site. That doesn't mean "almost there" or "just one or two workarounds left." It means customer workflows have been tested thoroughly. Orders are flowing cleanly through the system. Integrations aren't just configured, they're reliable under pressure.

This is also where you'll find all the "we'll fix it later" traps. Don't carry unresolved issues into launch week. That includes broken discount logic, inaccurate tax rules, or API syncs that haven't been tested against real-time order volume.

One tactic that works well here is running a parallel scenario: process a real order in the staging environment using production-level data. Fulfill it. Refund it. Do the messy stuff. If it holds up, your launch will too.

Test What Customers Will Actually Do

Performance testing can't just be about seeing how fast a page loads in a lab. You need to simulate how real users behave, especially your best customers.

This is where soft launches become incredibly valuable. Give early access to a handful of loyal accounts and ask them to use the new site exactly how they normally would. Reorders, product searches, quote requests, saved carts – get their hands on it and collect everything they struggle with. Then fix it fast.

Internally, keep your launch team connected during this phase. Developers, ops, and customer service should be talking in real time, no ticket queues, no email threads. You're not solving bugs; you're stress-testing your customer experience.

Build a Buffer, Then Build Another

Even with a tight launch plan, there will be surprises. Someone will forget to whitelist a staging IP. A customer will report an edge case no one thought to test. Your CDN might take an extra hour to propagate globally.

That's normal.

What's not normal is launching without room to breathe.

- Schedule your launch earlier in the week and earlier in the day.
- Have your senior developers and IT leads on standby, not in back-to-back meetings.
- Communicate with customers in advance so they know what to expect.
- And don't combine launch day with a marketing campaign rollout. One big moment at a time.

Post-Launch Reality Check: What Happens After the Plan

You had the plan. You built the backlog. You assigned owners and set metric. Now the site is live and real customers are clicking, searching, complaining, and (hopefully) ordering. This is where the real-world version of your replatform starts to reveal itself.

Here's how to handle the messy, unpredictable, and critical work of post-launch.

Listen Harder Than Ever, Then Act Fast

Once the site goes live, feedback starts flooding in. Some of it will be loud and emotional ("This new site is awful!"), some subtle and easy to miss ("I couldn't find my reorder list"). You need a way to capture all of it, not just what bubbles up through support.

Loop in every customer-facing team and give them a central place to log feedback in real time. Review it daily during the first month. Look for patterns, not just individual complaints.

If one customer says a product page looks off, make a note. If ten customers say they can't find the order history page, escalate it immediately.

Pro tip: Have someone on your team own this early signal triage. If everyone sees the issues, no one fixes them.

Revisit and Re-Rank Your Backlog

You created a post-launch backlog, now's the time to update it with real-world data. The features you thought would be “phase two” might need to move up. Others might no longer matter.

Go back to your original backlog and ask:

- What's become urgent now that customers are live?
- What can wait, even if it looked critical on paper?
- What's missing entirely?

This is the moment to get ruthless about value. Keep only the tasks that directly improve usability, conversions, or internal efficiency. Archive the rest. You can always revisit them later.

Look Beyond the Metrics You Set

You already defined success metrics. Great. But now, supplement those with the unofficial signals that reveal how your platform is performing.

Are sales reps still submitting quotes manually? Adoption might be lagging. Did your support team create internal workarounds already? That's a system gap, not just a training issue.

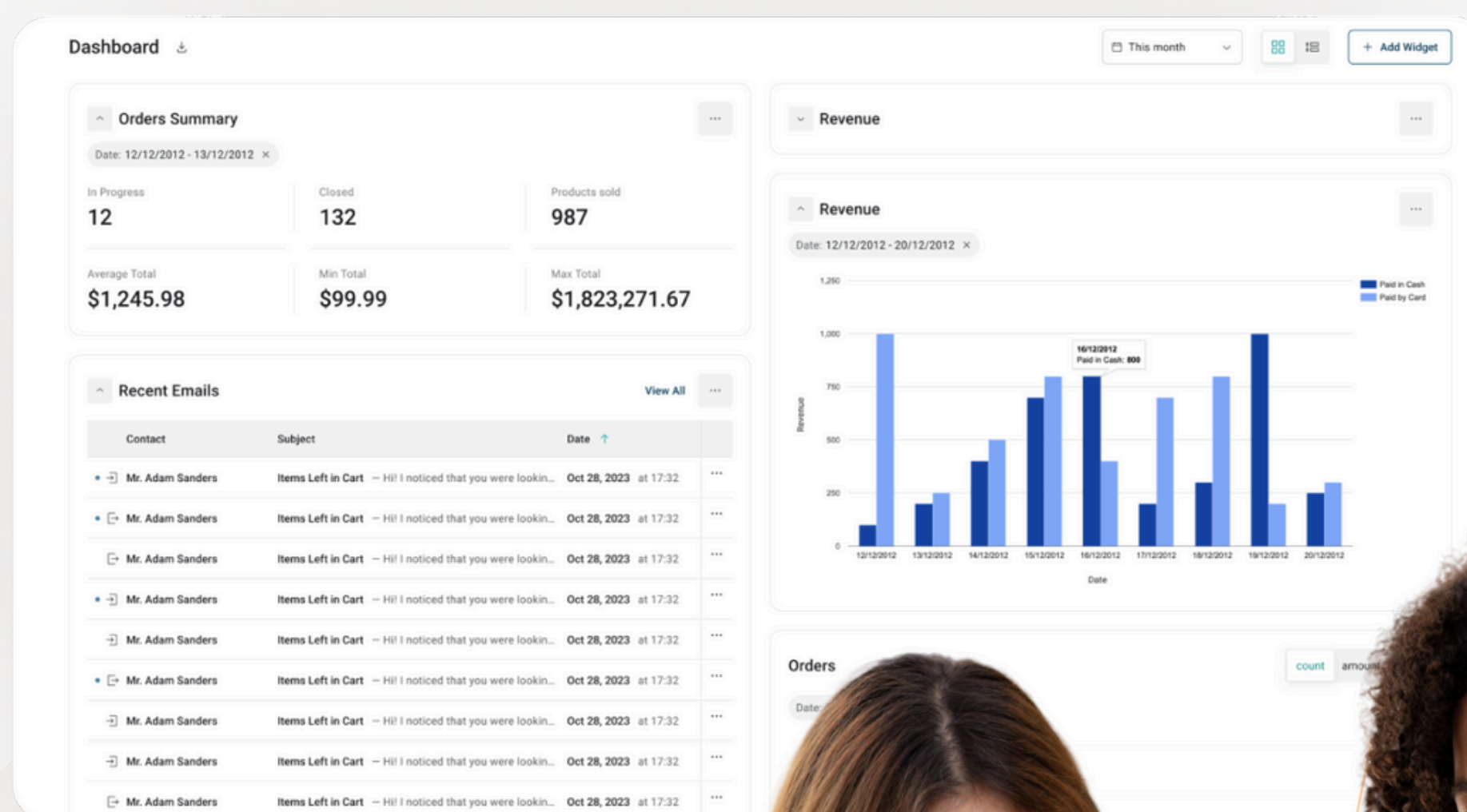
Some of your most important insights won't show up in your dashboard. They show up in the way your team behaves when no one's watching.

Protect Momentum With a Clear Next Move

The number one post-launch failure? Silence. The team breathes a sigh of relief and moves on, and the platform starts aging from day one.

Prevent that by making sure the next evolution is already in motion. Pick one initiative, one team, and one owner to move the platform forward. It could be something simple: improve mobile search, roll out new personalization rules, or A/B test checkout.

Doesn't have to be big. It just has to happen.



Replatforming Is a Strategic Reset

Replatforming is your chance to realign technology with how your business works, not how your old system forced you to work. It's an opportunity to eliminate workarounds, simplify operations, and finally give customers the experience they've been asking for.

That's where platform choice becomes everything.

OroCommerce stands out as one of the top solutions built specifically for B2B. It supports complex business models like B2B2X and marketplaces, and it's designed from the ground up for manufacturers, distributors, and suppliers who need flexibility, control, and depth, not generic features retrofitted for B2B.

OroCommerce is also built for faster results. Our customers go live in a matter of months not years.

If you're evaluating platforms, take the time to compare.

OroCommerce vs
Adobe Commerce



See how OroCommerce stacks up, and why more B2B companies are choosing it as the foundation for growth.

OroCommerce vs
Salesforce Commerce
Cloud

