



The benefits platform RFP playbook:

What actually matters
(and what doesn't).

A practical guide for enterprise HR, Reward and Benefits leaders



Table Of Contents

The RFP process is broken but you can fix it

02

Why implementation is where RFPs succeed or fail

19

What you're actually buying

04

Final-stage evaluation: where decisions should actually be made

24

What actually matters in an evaluation

09

How to measure success

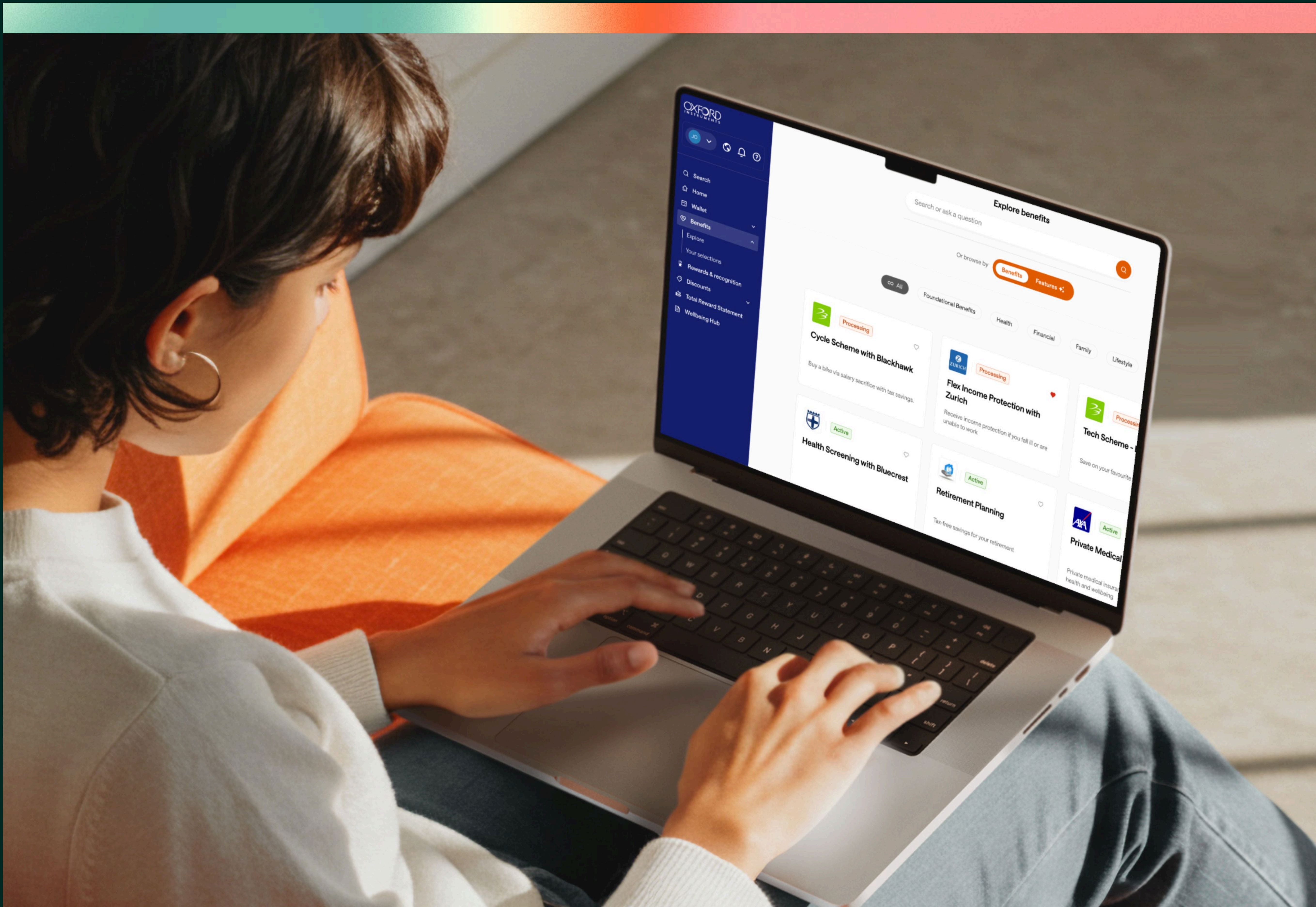
28

The questions most teams get wrong

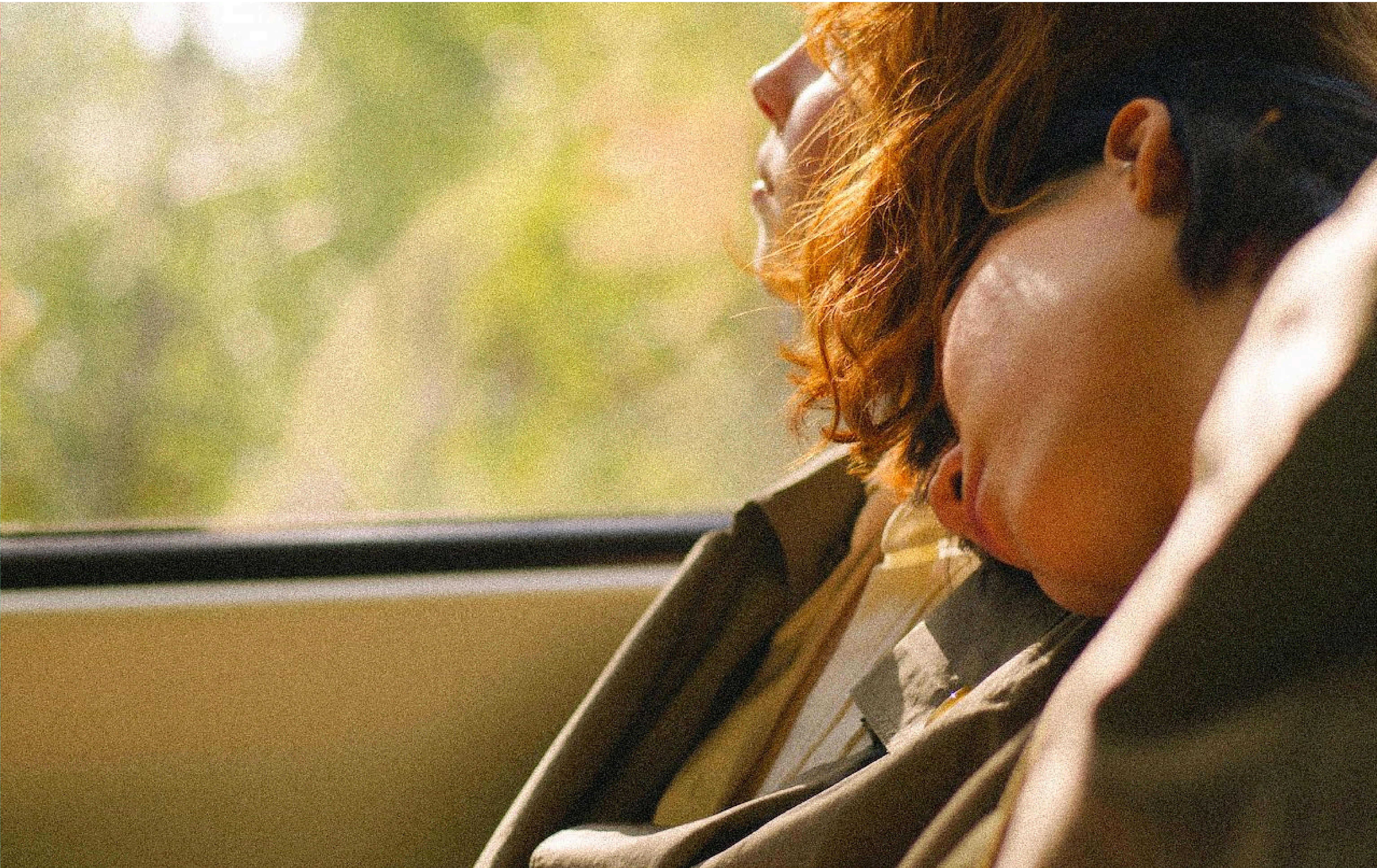
16

A better way to run a benefits platform RFP

32



The RFP process is broken



Most RFPs gather information. Fewer help you decide.

Most benefits platform RFPs are built to produce a shortlist, not a decision.

They gather information, compare feature sets, and score presentations.
But they rarely answer the questions that actually matter:

1

Will this platform reduce the operational complexity my team deals with every day, or will it quietly create more of it?

That gap between what RFPs measure and what actually determines long-term success is why so many enterprises launch a new benefits platform only to discover that it can't do what was promised, changes take months to implement, and the team is spending more time managing the tool than managing benefits.

This guide is designed to help you run a smarter process. One that cuts through polished sales narratives, forces vendors to prove capability rather than describe it, and gives your internal stakeholders — HR, Reward, Payroll, Finance, Operations — a shared framework for making a decision with confidence.

2

Will this platform deliver the outcomes we actually need, and will it do so without creating more burden than it removes?

It's written for organisations that are serious about getting this right. That means enterprise businesses with complexity: multiple countries, mixed workforces, evolving benefit programs, and accountability for financial accuracy and compliance.

The structure follows the natural arc of an RFP process: from framing what you're actually buying, through evaluation and final-stage due diligence, to defining what post-launch success looks like. Each section surfaces the questions most teams either ask poorly or skip entirely.

What you're actually buying



Start with your business case

Benefits platforms all look similar from the outside.

Every vendor has a clean interface, a compelling demo, and a roadmap full of features they're about to ship. The real differences (the ones that determine whether your team is relieved or exhausted two years from now) sit underneath all of that.

Before you define evaluation criteria, define your desired outcomes. Operational complexity is a real driver for many organisations, but it's not the only one — and it's not always the main one.

For example, you might be:

- Looking to replace a legacy platform that's hit its ceiling
- Trying to reduce headcount through better self-service and automation
- Focusing on improving how employees understand and value their benefits
- Consolidating a fragmented supplier landscape

The outcomes you're trying to achieve should shape every part of your evaluation. They inform the questions you ask, the scenarios you test, and the metrics you'll use to measure success after go-live. That's where you need to start.



Interface vs. infrastructure

There's a distinction that most RFPs gloss over too quickly: the difference between what employees see and what your team depends on.

The interface — the employee portal, enrolment journey, and mobile experience — matters. Engagement is real, and employees will abandon a confusing product. But most decent vendors have invested in creating a solid interface, so you need to dig a little deeper.

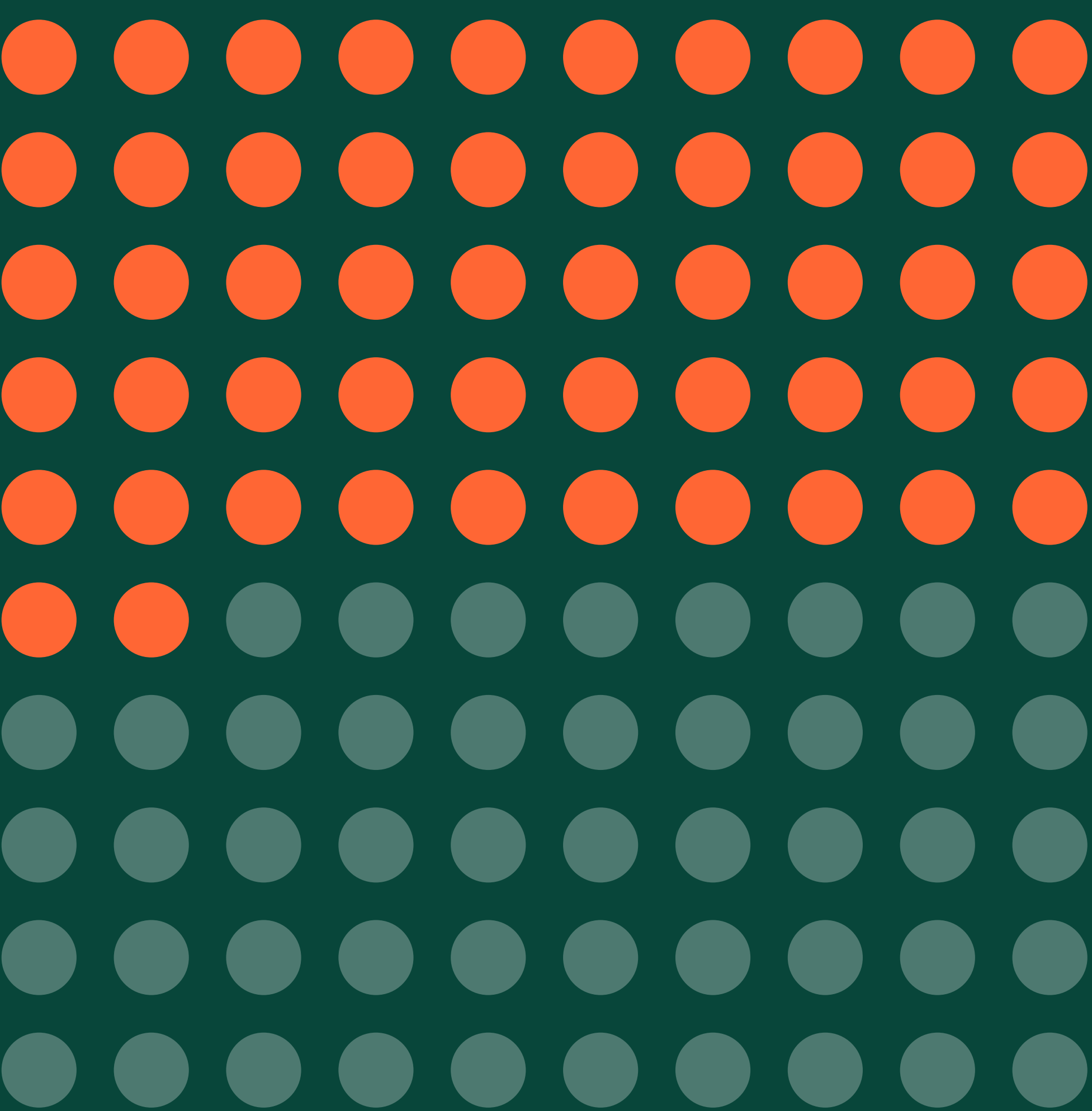
Infrastructure is where the decisions get made.

This means:

- **How the platform models data**
Does it use a single unified data model or stitch together separate systems for different markets?
- **How payroll and finance workflows are handled**
Can you trust outputs without manual reconciliation?
- **How eligibility logic is structured**
Can the platform handle the complexity of your workforce or does it require workarounds for edge cases?
- **How multi-country configuration works**
Is it a genuine global system or a UK platform bolted into other markets?

**You're not buying a product.
You're buying how
complexity is handled.**

The platform that looks best in a demo isn't necessarily the one that performs best when it's running your benefits program across fifteen countries.



>50%

**of UK employees don't fully
understand their benefits**

Source: Drewberry Employee Benefits Survey 2024

The hidden cost of operational debt

The risk most organisations underestimate isn't a feature gap.

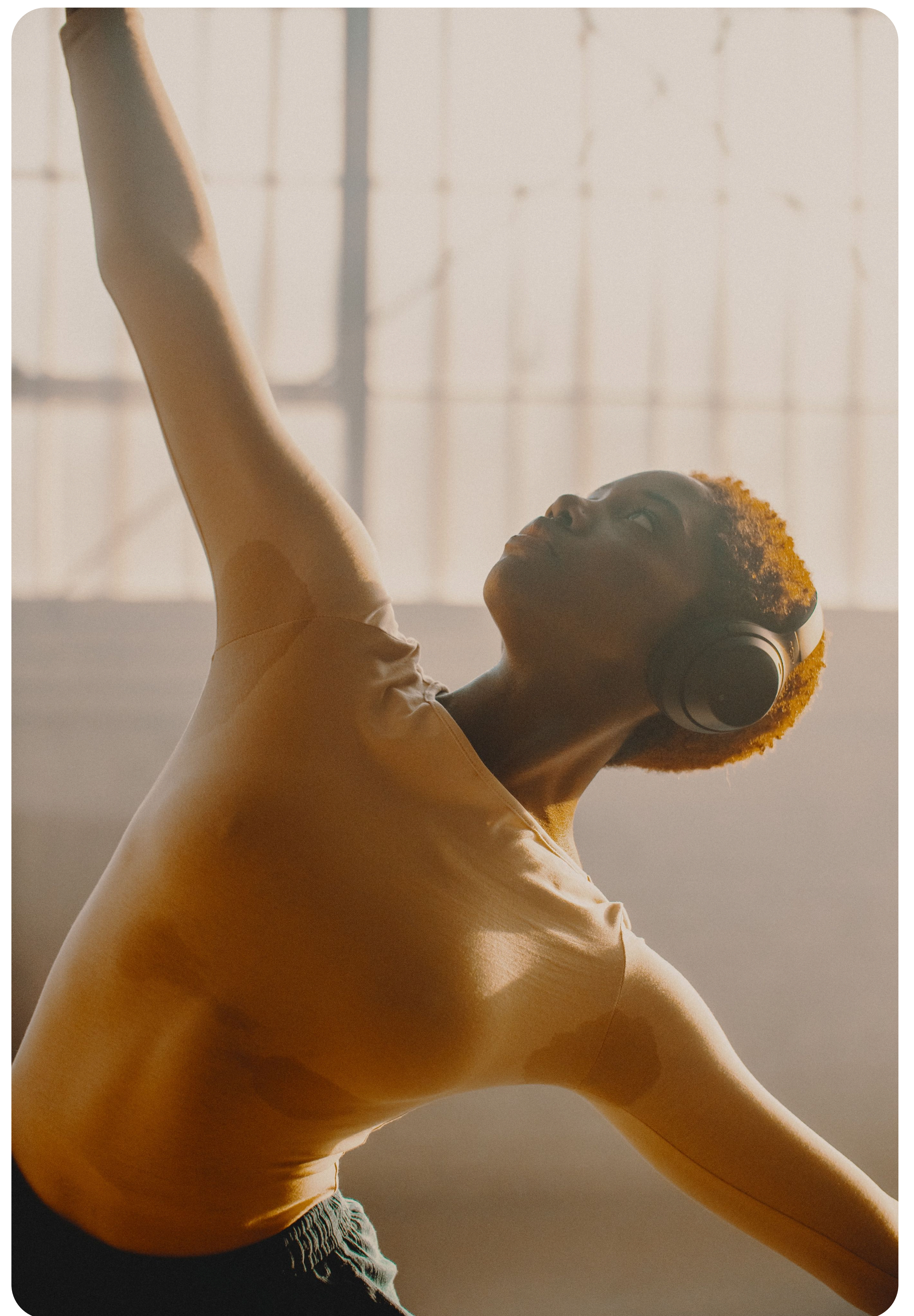
It's operational debt: the accumulation of manual processes, workarounds, and coordination overhead that builds up when a platform absorbs complexity poorly.

This could show up as:

- A payroll reconciliation your team does every month because the platform outputs can't be trusted directly
- A change request queue with a six-week lead time
- A multi-country rollout that stalls because what worked in the UK requires re-engineering for Germany
- A member list that goes out to carriers and comes back different, and no one knows why

None of this shows up in a demo. The RFP process is your best, and sometimes only, opportunity to surface it before it becomes your team's problem.

Many organisations feel a strong pull toward well-known names. When organisations don't buy benefits technology very often (and most don't — a platform decision might happen once every ten years), they naturally default toward large, established providers because it feels like a safe decision.



- **This guide is designed to help you evaluate based on evidence rather than brand recognition.**



“

Familiarity isn't the same as fit. The vendors with the biggest sales presence aren't necessarily the ones whose architecture handles your complexity well.

What actually matters in an evaluation



A well-structured evaluation focuses on seven areas.

Most RFPs cover some of these superficially. The organisations that make good decisions treat each one as a genuine test, not a box-ticking exercise.

1. Architecture: is it truly unified?

The first question to ask of any platform is whether it's a single system or a collection of connected tools. This distinction matters more than it sounds.

A fragmented architecture

(multiple systems integrated at the surface)
means inconsistency:

- Eligibility rules defined in one place have to be replicated in another
- Reporting pulls from different sources and needs reconciliation
- Changes in one market can have unexpected effects elsewhere
- When something breaks, no one can immediately tell you why

A unified architecture

(a single data model, centralised logic, consistent configuration) scales:

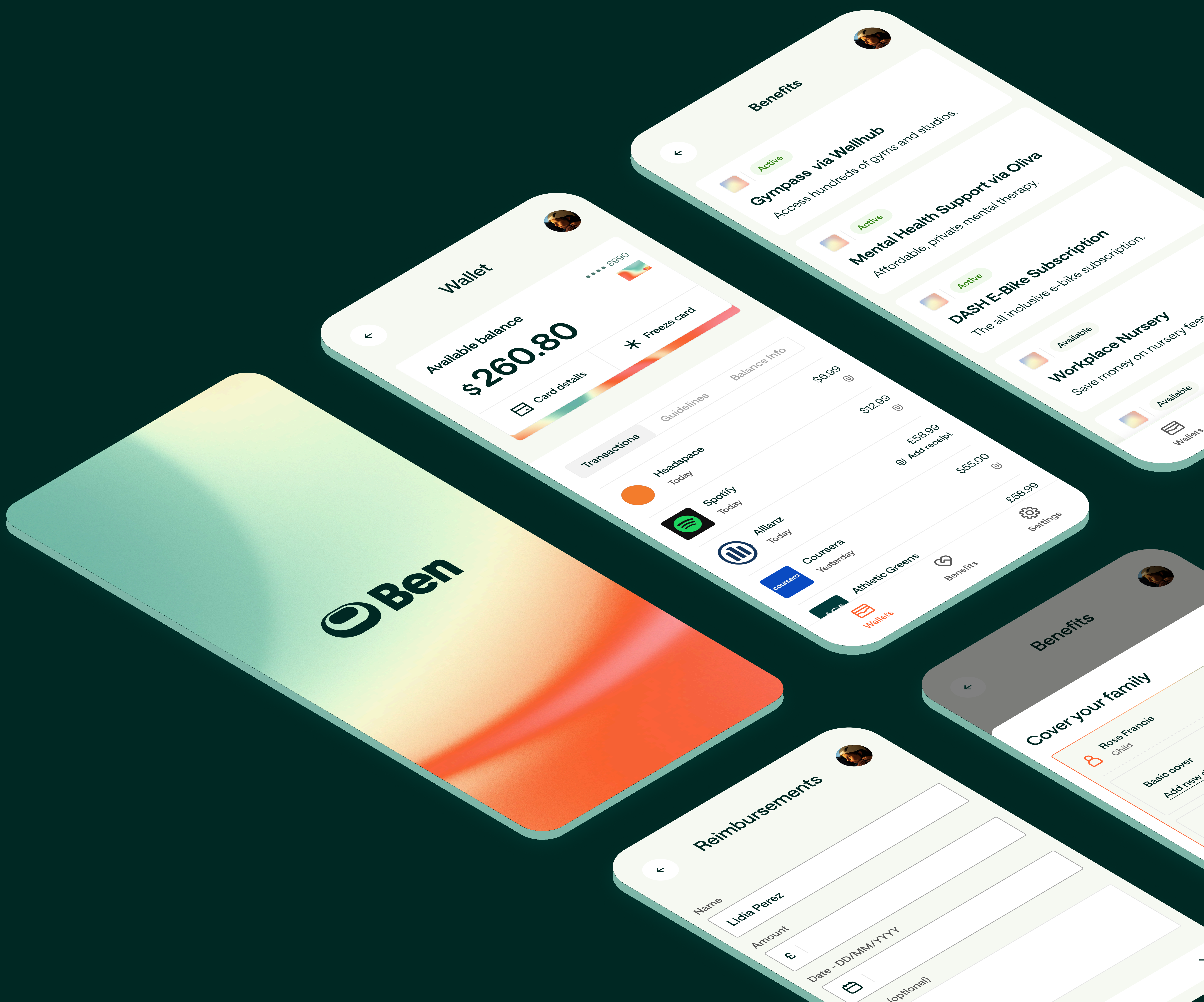
- You can trust the reporting
- You can make changes confidently
- Your platform behaves the same in Singapore as it does in London
- When something breaks, no one can immediately tell you why



Ask every vendor

Is this a single global system,
or have you integrated
separate regional tools?

Then ask their existing enterprise
clients the same question.



2. Configuration and control

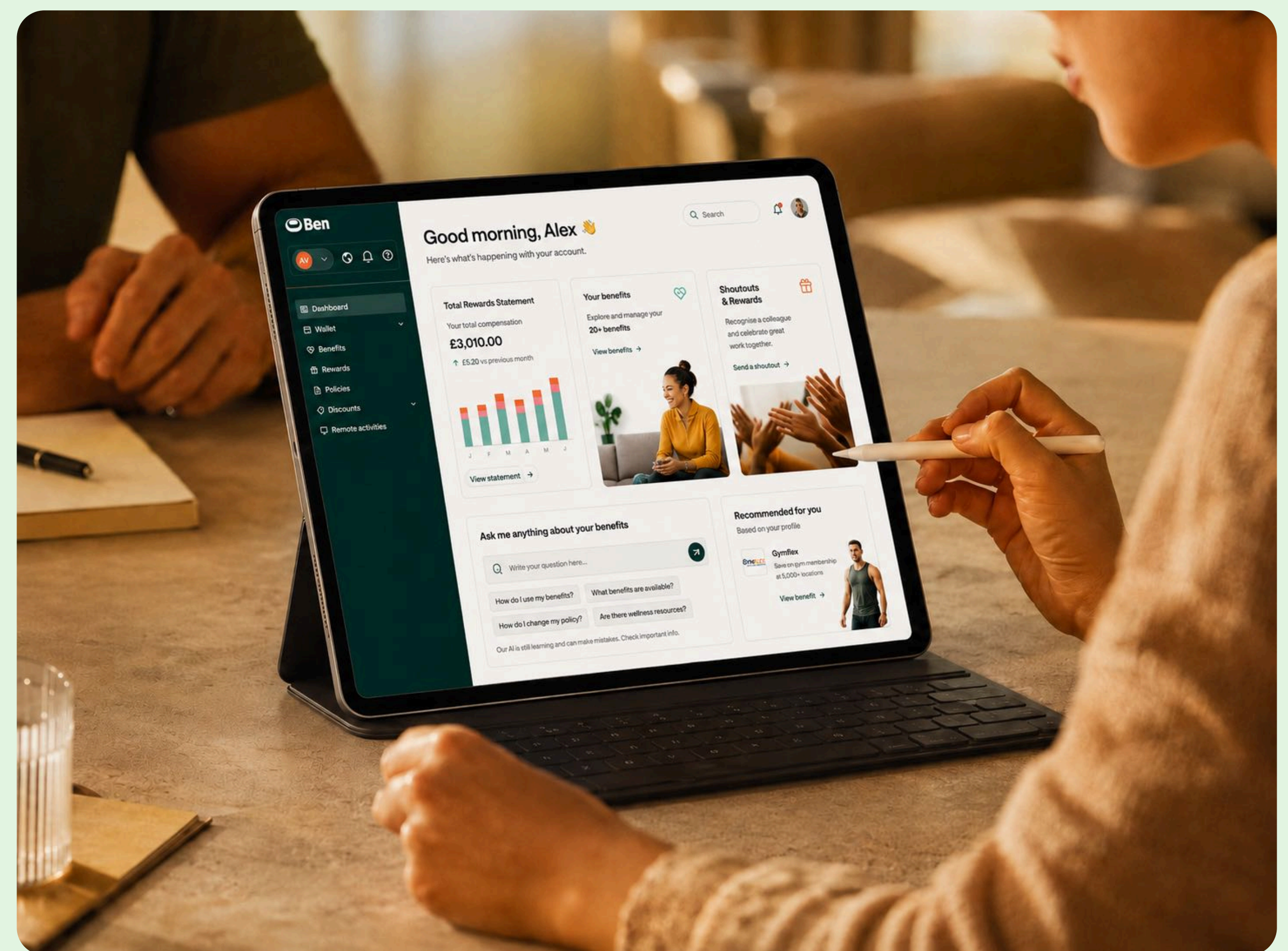
One of the most revealing questions you can ask a vendor is: who owns the configuration, and what happens when we need to make a change? Their answer tells you a great deal about what the next five years will look like.

If every change — opening a benefit window, updating plan information, adjusting eligibility criteria — requires a formal request to the vendor, with a lead time measured in weeks and a cost measured in percentage points of your annual license, you've bought a platform that constrains you rather than enabling you.

The right model gives your administration team control over the changes they should own, and gives the vendor a clear, efficient process for changes that need their expertise. Both of these matter.

You're looking for controlled flexibility

- A good platform gives your team meaningful autonomy within a governed structure, not one that either locks everything down or gives you an open-ended system that's easy to break.



Ask every vendor

Who owns the configuration,
and what happens when we
need to make a change?

3. Data integrity and financial accuracy

Can you trust your platform's outputs?

This is a simpler question than it sounds. And the honest answer from many platforms is: mostly. When you can't trust those outputs, it leads to:

- **Payroll files that need to be checked manually**
- **Member lists that don't quite match**
- **Invoices that require reconciliation against what was actually processed**

For enterprise organisations, this is more than a minor inconvenience: it's a compliance and financial risk. Every manual reconciliation step is a point where errors can accumulate, deadlines can be missed, and audit trails get muddled.

Ask vendors to walk you through exactly how payroll outputs are generated, how member list reconciliation works, and what happens when there's a discrepancy. Then ask their clients whether they've experienced payroll errors or reconciliation issues, and how those were resolved.

4. Change management and agility

The benefits landscape changes constantly, with evolving regulation and shifting market expectations. Your workforce changes, too, with new markets, new employee populations, and new priorities. Your benefits program will need to respond to these changes.

Modern platforms built on configuration rather than hard-coding can update benefit parameters, add new plans, and adapt eligibility logic through the platform itself. Platforms built the other way — where customisation is achieved through code — require development resources for changes that should be administrative.

The commercial dimension matters here, too. Many platforms charge change fees, sometimes significant ones. It's not unusual to see enterprise clients paying 20% or more of their annual license in additional configuration costs that weren't clearly disclosed at the time of signing a contract.

It's crucial that you understand the full picture before you sign anything.

The key test

Can you make meaningful changes without engineering involvement?

The question isn't whether a platform can handle your current configuration. It's whether it can handle what your configuration looks like in three years, without requiring expensive reengineering each time.

5. Real-world behaviour

Demos are highly curated.

They show you what's known as 'the happy path': clean data, standard scenarios, features that work exactly as described. But as we all know, this doesn't always reflect real life.

If you haven't interrogated how the platform handles complexity, you haven't properly evaluated it. This is the principle that should govern your final-stage assessment, which we'll return to in detail later in this guide.

The platform you choose will need to deal with situations that don't fit the standard eligibility criteria:

- Benefit windows that need to open and close at short notice
- Payroll files that have to accommodate an acquisition mid-year
- A regulatory change in a market that needs to be reflected in the system within weeks

6. Total cost of ownership

Ask vendors to explain the costs in detail. Request a three-year and five-year projection that includes likely change request volumes, multi-country expansion costs, and any fees associated with additional integrations. Then ask their existing clients whether the costs they're experiencing match what they were quoted.

The license fee is the number vendors lead with. But it's rarely the number that determines whether a platform is good value.

The real cost includes:

- Implementation costs (which vary enormously — from £150k to close to £1 million for a comparable global deployment, depending on the vendor and their approach)
- Ongoing change request fees
- Support costs
- The internal resources required to manage the platform day to day



Ideally, you're looking for a platform that's clear and transparent, with no hidden costs. Asking these questions will reduce the risk of experiencing any unexpected surprises.

7. Reporting quality and actionable insights

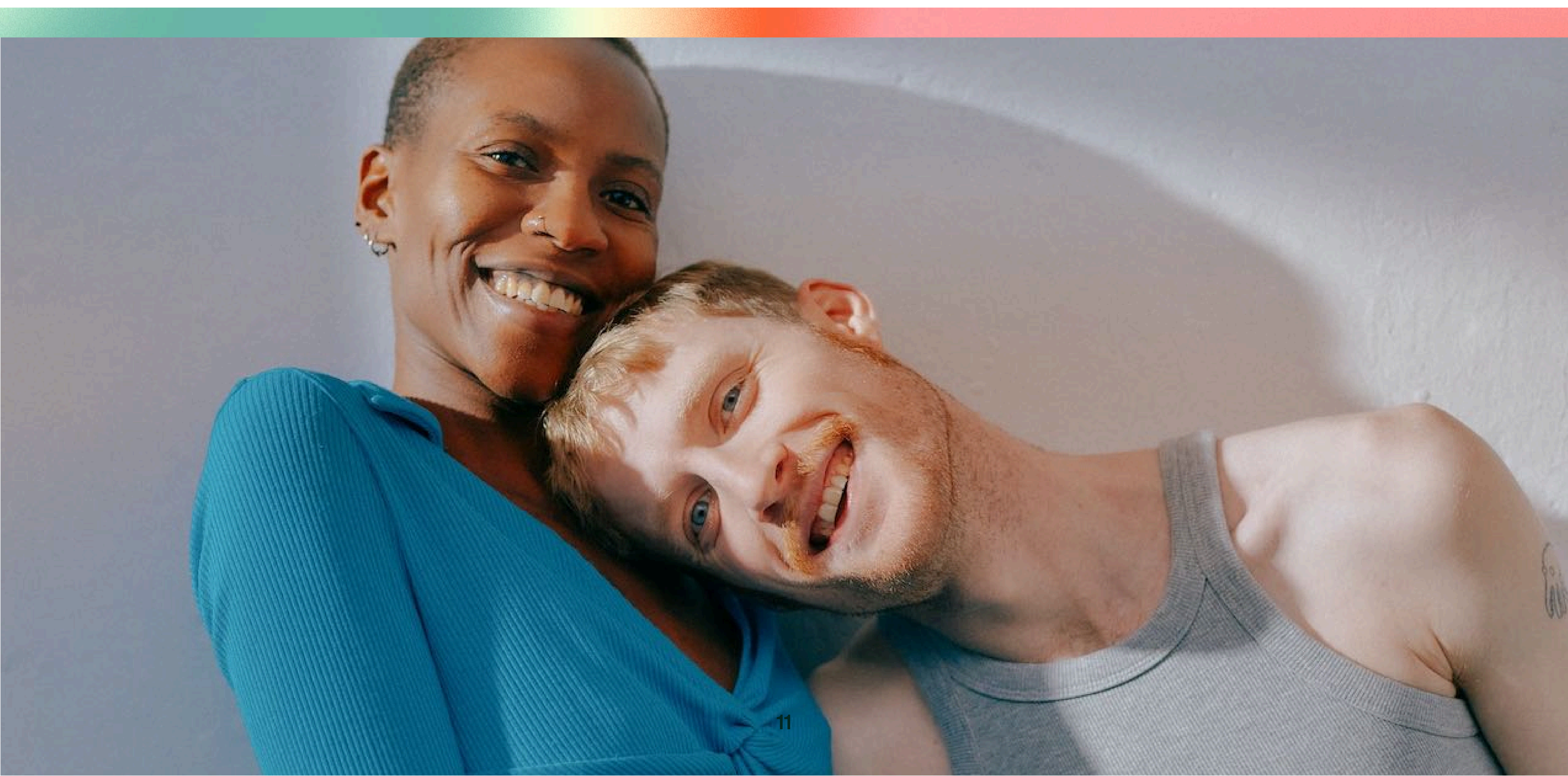
Reporting is one of the most frequently-cited differentiators by experienced buyers, and one of the most commonly glossed over in RFPs. Most people ask whether the platform has a reporting suite. The better question is whether it gives you insights you can actually act on.

Many platforms provide periodic data extracts, like a monthly report, or a spreadsheet on request. This isn't the same as getting real-time visibility into benefit utilisation, cost, and employee engagement.

If you're managing benefits across multiple countries and populations, you need to see what's happening now — not what happened last month. And you need to be able to act on it without waiting for a vendor to generate a report.

Here's how to get the full picture on reporting:

- **Ask vendors to show you a live reporting dashboard, not a screenshot or mockup**
- **Ask how data is refreshed and at what cadence**
- **Ask what alerts or triggers the platform supports, so your team is notified of issues rather than discovering them during a monthly review**
- **Ask whether reporting access is included in the license or priced separately (in some platforms, accessing in-depth analytics comes with an extra cost)**



The questions most teams get wrong



Why standard questions produce standard noise

The standard set of RFP questions is well-intentioned but largely ineffective. It produces polished, rehearsed answers, and very little signal about how the platform actually performs.

The process usually goes something like this:

- 1 Teams ask high-level capability questions
- 2 Vendors confirm they have the capability
- 3 Everyone moves forward

**The problems only emerge
after you've gone live.**

By then, it's too late.

What teams typically ask — and why it produces noise

What people ask	Why it doesn't work
What features does your platform offer?	Vendors can truthfully claim to offer almost any feature by carefully wording their response.
What's on your product roadmap?	Roadmap commitments aren't contractual. Features often get delayed or descoped.
How engaging is the employee experience?	Engagement metrics from vendor case studies aren't comparable across different implementations.
Do you have a mobile app?	Every vendor says yes. The real question is whether the mobile experience is fully functional or a stripped-down portal.
Can you integrate with our HRIS?	The answer is almost always yes. The question is how, how reliably, and at what cost.



What you should be asking instead

The questions that surface real differences are the ones that put specific, operational scenarios to vendors, and ask them to prove rather than describe.

1 Where does complexity sit — in your system or in our team?

This frames the central tradeoff. Every platform manages complexity somewhere. The question is whether it's absorbed by the technology or transferred to your administrators.

2 Walk us through exactly what happens when we need to make a change. Who does what, how long does it take, and what does it cost?

The answer to this single question reveals more about a platform's architecture and commercial model than almost anything else. Make sure to push for specifics.

3 What manual work still exists behind the scenes after implementation?

There will always be manual work with any platform. The question is how much, and whether your team or the vendor's team carries it. Ask for a detailed answer, not a reassurance.

4 How do you handle exceptions and edge cases in eligibility?

Standard scenarios always work. You'll learn more by asking vendors to walk through a complicated eligibility scenario: someone on a temporary contract, mid-year joiners, or cross-border employees.

5 Can you show us a payroll file from an existing client and explain how it was generated?

This isn't always possible due to confidentiality, but you'll always learn something from a vendor's response. Vendors with clean, reliable payroll outputs are usually comfortable discussing the mechanics.

6 What would a three-year total cost look like, including all change requests, integrations, and multi-country expansion?

Procurement teams are increasingly asking this. Vendors who can't model it, or who are evasive about the variables, are usually carrying costs they'd rather you discover later.

You're not evaluating a pitch

You're evaluating a system that your team will depend on for the next five to seven years. The questions should reflect that reality.

Why implementation is where RFPs succeed or fail



When implementations fail, it's rarely the product.

Implementation is the most common blind spot in benefits platform RFPs.

Teams focus on product selection, leaving implementation planning until after the contract is signed. This is where decisions that looked good on paper start to unravel.

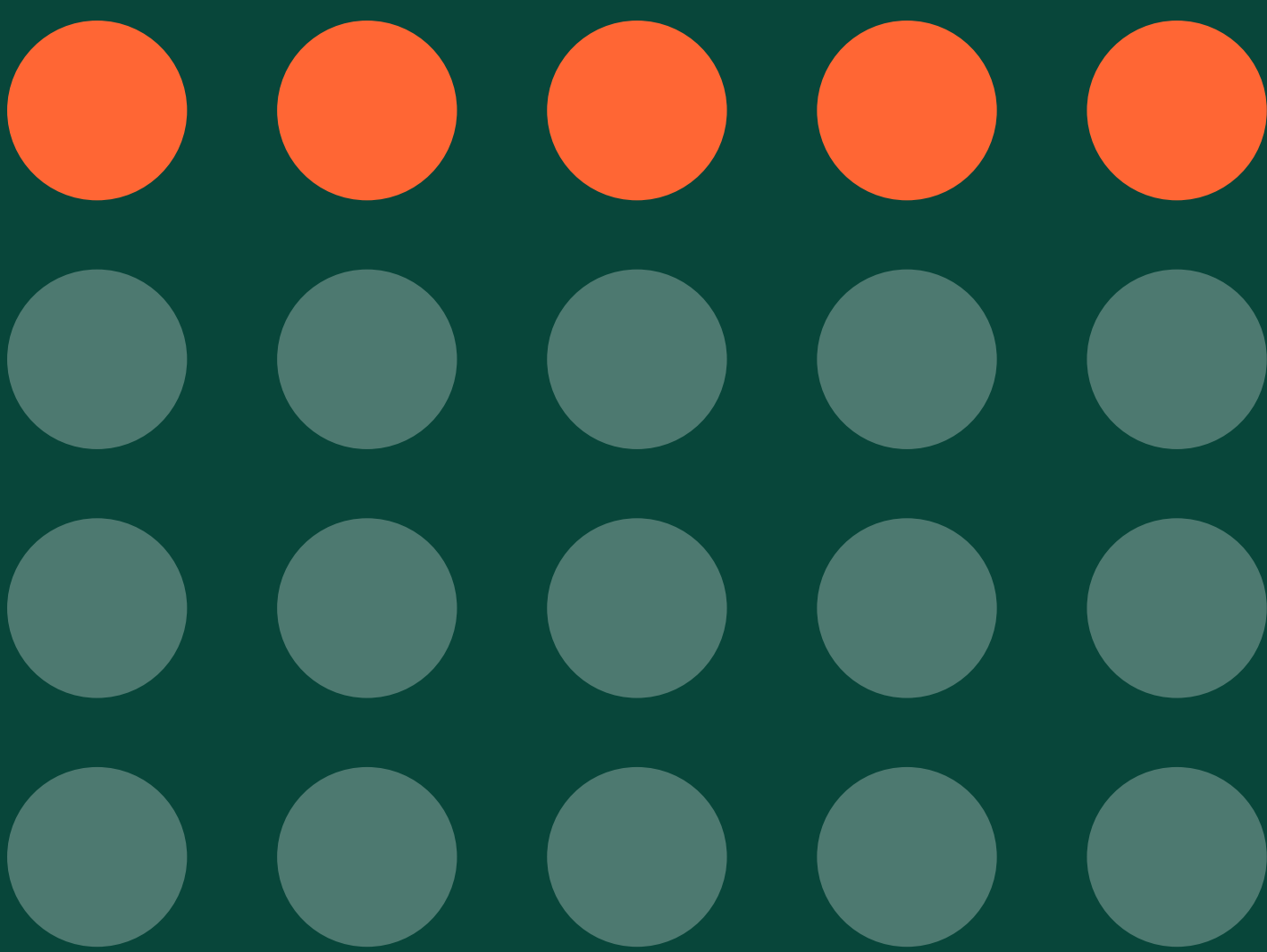
Why implementations fail

When implementations go wrong, the causes are almost never product limitations or unrealistic timelines alone. They're usually:

- **Lack of clear ownership on the client side**
No one person or team accountable for decisions and signoff
- **Poor governance**
A process for making decisions that's unclear, slow, or involves too many stakeholders without a mechanism for resolution
- **Misaligned expectations**
The vendor assumes the client's data is clean, the client assumes the vendor will handle the complexity — neither assumption is correct
- **Testing treated as discovery**
UAT (user acceptance testing) surfaces things that should have been defined months earlier, causing delays that compound

The mistake isn't always a bad product choice.

Usually, it's a failure to understand what implementation actually involves, and evaluating vendors on that basis before making a decision.



1 in 4

organisations report that new HR tech implementations fail to meet adoption expectations.

Source: Sapient Insights Group survey, [reported by SHRM](#)

Common myths about implementation

The myth	The reality
It will take too long	Timeline overruns are almost always governance failures, not technical ones. Having a clear decision-making structure and defined ownership dramatically compresses timelines.
Testing always drags	Testing drags when it becomes discovery — when UAT is the first time that edge cases and exceptions are surfaced. Define these upfront.
Customisation requires engineering involvement	Modern platforms built on configuration rather than code can make most changes through the platform itself. This is a fundamental architectural question worth asking explicitly.
The vendor handles the complexity	Good vendors absorb significant complexity. But they need your team to own certain decisions, like data definitions, eligibility rules, and approval workflows. Clarify the division of responsibility before you sign.

What to evaluate during the RFP

Your implementation evaluation should be as rigorous as the product evaluation.

Specifically, you should explore:

- **The implementation model**
What methodology does the vendor use, and what does the client need to provide?
- **Roles and responsibilities**
Who owns what, and how are decisions escalated when there's disagreement?
- **Data requirements**
What does the vendor need from you, in what format, and how do they handle data that arrives in the wrong shape?
- **Testing approach**
When does UAT happen, which scenarios are covered, and what's the process for issues that emerge?
- **Support model post-launch**
What changes after go-live, and what does the ongoing relationship look like?

If you don't evaluate implementation during the RFP, you're deferring risk — not removing it.

The costs and complexity that weren't visible in the process don't disappear. They just become your problem later.

Vendor project management: what good looks like

Poor vendor project management is one of the most common causes of implementation overrun, and one of the hardest things to identify before you've signed a contract.

Organisations have found themselves building bigger internal teams simply to compensate for a vendor's inability to run a structured project: chasing milestones, filling gaps in communication, and managing escalations that should have been handled by the implementation partner.

During the RFP, ask vendors for a sample project plan — not a high-level timeline, but a working document with named milestones, dependencies, and owner assignments.

- Ask them to walk you through how they handle a milestone that slips
- Ask what their escalation protocol is when something critical is at risk, and who on their side has authority to make decisions quickly

These aren't aggressive questions. They're the questions that responsible buyers ask, and confident vendors will have no problem in answering them.

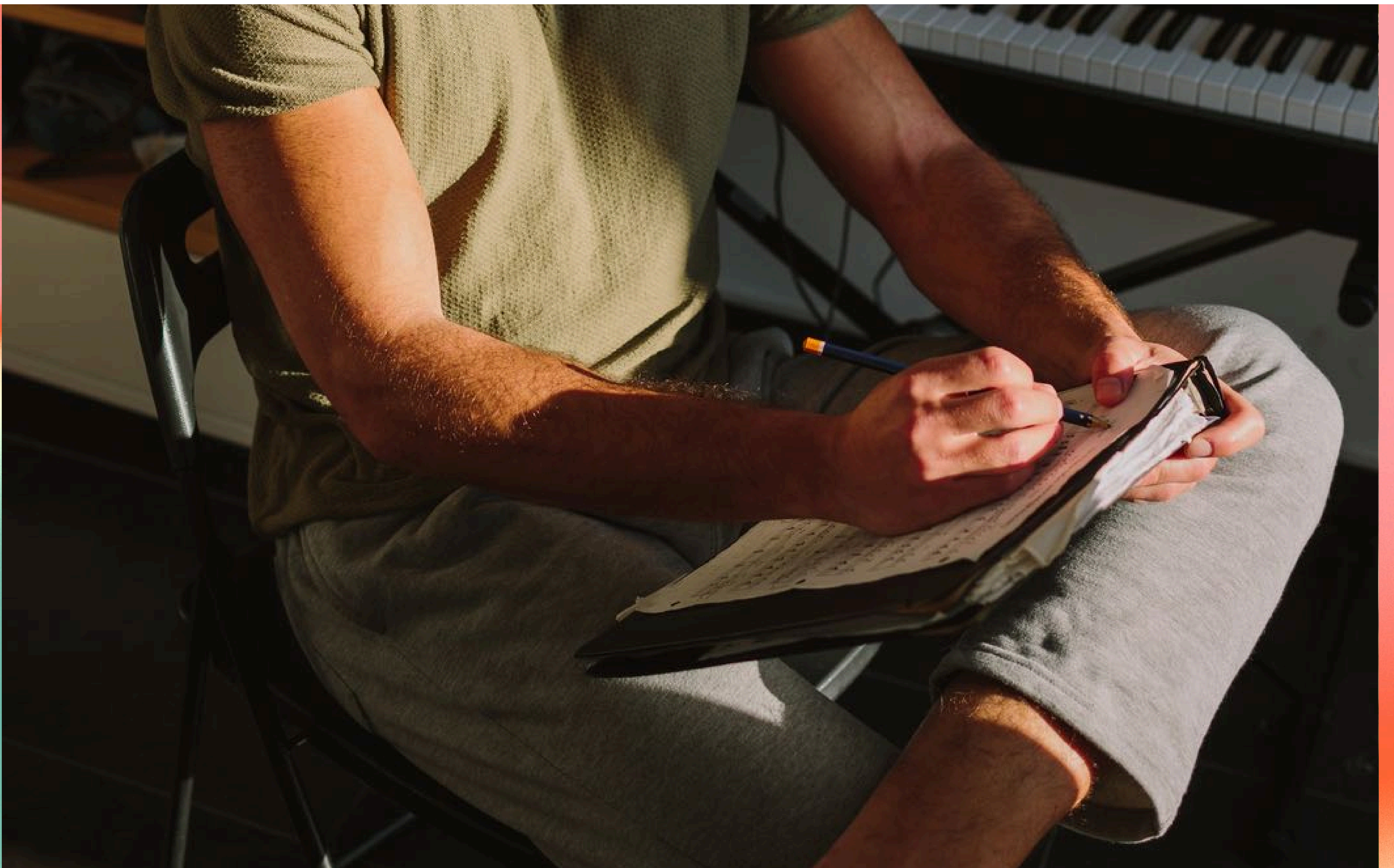
When speaking to reference clients, ask specifically about project management quality:

- Did the vendor arrive at each phase with a clear plan, or did the client have to drive it?
- Were problems surfaced early or discovered late?
- Was there a named point of escalation, and did it work?

These questions cut through polished sales references and surface the experience that you're likely to have.

Final-stage evaluation:

Where decisions should
actually be made



What matters at the final stage

By the time you reach a shortlist of two or three vendors, you've gathered a lot of information. What most teams do at this stage is ask for more information: reference calls, extended demos, additional Q&A. What you should be doing is shifting from information gathering to validation.

There's a big difference. Information gathering asks vendors to describe what they can do. Validation asks them to prove it in your context.

Your final evaluation should test three things:

- **Real workflows, not ideal scenarios**
Your most complex eligibility logic, messiest data, and edge cases
- **Real data, not sample data**
Or at minimum, data that accurately represents the complexity of your workforce
- **Real outputs, not demos**
Payroll files, member lists, reporting that reflects how the platform actually performs



The principle is simple: if a vendor can't show it working in your context, it doesn't count. Roadmap promises and capability descriptions aren't evidence that you can rely on.

The technical due diligence session

The most effective final-stage evaluation format is a structured technical due diligence session.

This is an in-person working session where vendors configure and demonstrate the platform using your actual scenarios. This matters for two reasons:

1

It exposes the true architecture of the platform

A system built on configuration can be adjusted and demonstrated live. A system built on hard-coding can't. The vendors who struggle with this session aren't failing to prepare adequately — they're revealing a platform limitation that would have become your problem post-implementation.

2

It gets the right people in the room

HR and Reward leaders see the product from an employee and program perspective. Finance and Payroll see the outputs they'll depend on. IT and HR Systems see the integration architecture. Operations see the admin workflows. In a standard demo, no one sees the full picture.

Who should be involved

→ HR / Benefits / Reward

They're program owners who know what the platform needs to support

→ Finance and Payroll

They'll depend on the outputs and need to trust them

→ Operations / HR Systems

They'll manage the platform day to day and integrate it with existing infrastructure

→ Procurement

They'll need to validate the commercial model and total cost picture

What to prepare

- Real scenarios from your benefits program, including both standard cases and edge cases
- A representative dataset, or a synthetic one that reflects your data's actual complexity
- Specific questions for each stakeholder group — payroll output format, eligibility configuration, reporting capability
- A list of the changes you expect to need in the first twelve months (ask the vendor to demonstrate how they'd make those changes)

What good looks like

- **Clear, consistent outputs**
Payroll files that match what was configured, and member lists that reconcile cleanly
- **No manual workarounds**
The vendor shouldn't need to do anything outside the platform to produce the outputs
- **Transparent logic**
You should be able to follow exactly why the platform produced what it produced
- **Confidence across your stakeholder group**
HR trusts the employee experience, Finance trusts the numbers, Operations can see how they'd manage it

This is the moment where storytelling stops and systems are exposed.



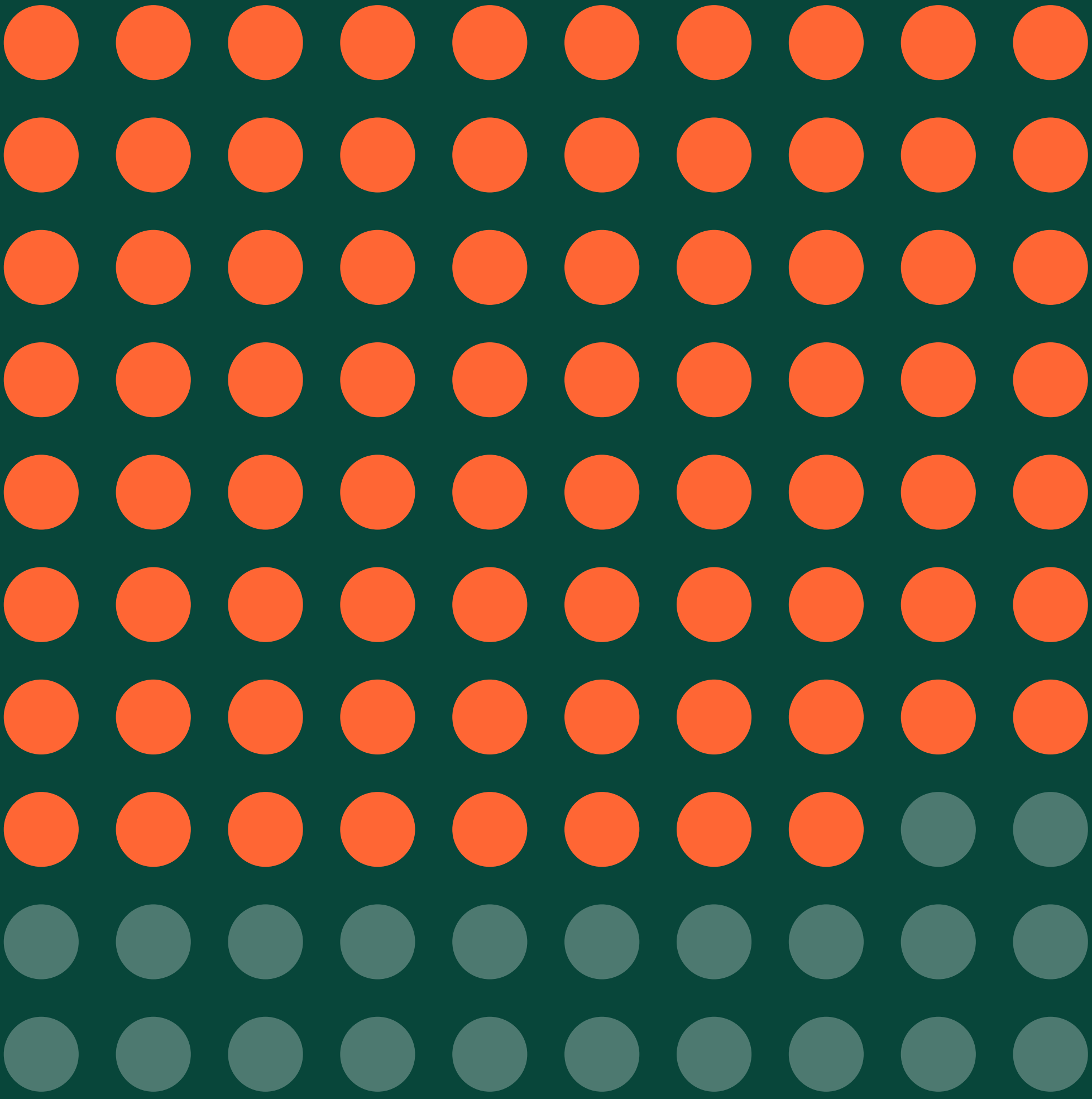
How to measure success



Going live is not success — it's the start of a much longer relationship

The metrics that matter for the first six months — login rates, enrolment completion, positive employee feedback — aren't the metrics that determine whether you made a good decision.

It's important to reframe success before you buy. The most effective way to hold a vendor accountable is to agree on what you're measuring, and why, before the contract is signed.



78%

of companies don't measure the engagement rate of their benefits

Source: [Ben](#)

What most teams measure

The standard post-launch metrics — logins, engagement rates, enrolment numbers — are visible and easy to report. But they're also incomplete. A platform can produce healthy engagement numbers while quietly creating significant operational overhead for your team.



**Organisations lose 2-4% of
total labour spend to payroll
leakage, with nearly 2 in 5
companies reporting \$1m-\$5m
in annual payroll losses**

Source: UKG and KPMG research on [HR Tech Cube](#)

What actually defines success

1

Utilisation with context

Are employees using their benefits meaningfully — not just logging in, but making selections, engaging with new offerings, and accessing support? Looking at enrolment rates by benefit type, and across different employee populations, gives you a more useful picture than aggregate engagement.

2

Financial control

Are payroll outputs accurate, first time? Is forecasting reliable — can you see what benefits are costing before the invoice arrives? Have reconciliation issues reduced or disappeared? These are the metrics that finance teams care about, and they directly reflect the quality of your platform.

3

Operational efficiency

How much manual work does your team still do? How many change requests have been submitted, at what cost, and with what turnaround time? How long does it take to launch a new benefit or update an existing one? These numbers should be improving over time, not staying flat or growing.

4

Risk reduction

Are you confident in your compliance position? Is the audit trail clean? Can you produce accurate reporting without a reconciliation exercise? Risk reduction is rarely measured explicitly, but it's one of the most significant returns that a well-implemented benefits platform delivers.

Success isn't go-live. It's sustained control, clarity, and efficiency.

Define what this looks like for your organisation before you sign the contract, and build it into the vendor relationship from day one.

A better way to run a benefits platform RFP



Evaluate the platform, not the pitch

Organisations that make good decisions in this process share a common characteristic: they're sceptical of polished demos and roadmap promises, and they insist on seeing the product work in their context before they commit.

This guide has been built around five principles that reflect that approach.

1 Start with complexity, not features

Define the operational complexity your platform needs to handle, like your workforce profile, countries, benefit structures, and payroll and finance requirements. Then use that as the anchor for every evaluation conversation. Features are only valuable if they address your actual complexity.

2 Evaluate infrastructure, not just interface

The employee experience matters, but it's the infrastructure — data model, payroll handling, configuration architecture, multi-country capability — that determines whether your team is empowered or burdened. Spend at least as much time on the latter.

3 Assess implementation, not just product

The quality of the product at the point of evaluation is only part of what you're buying. The implementation approach, governance model, division of responsibility, and support structure post-launch are equally important (and often more revealing).

4 Validate using real scenarios

Make sure vendors demonstrate capability using your data, scenarios, and edge cases. Describing a capability isn't the same as proving it. Make the technical due diligence session a non-negotiable part of your final-stage process.

5 Define success before you buy

Agree on the metrics that will determine whether this investment has worked, considering financial accuracy, operational efficiency, and risk reduction. Build those metrics into the contract and the ongoing vendor relationship. Go-live is the beginning, not the destination.

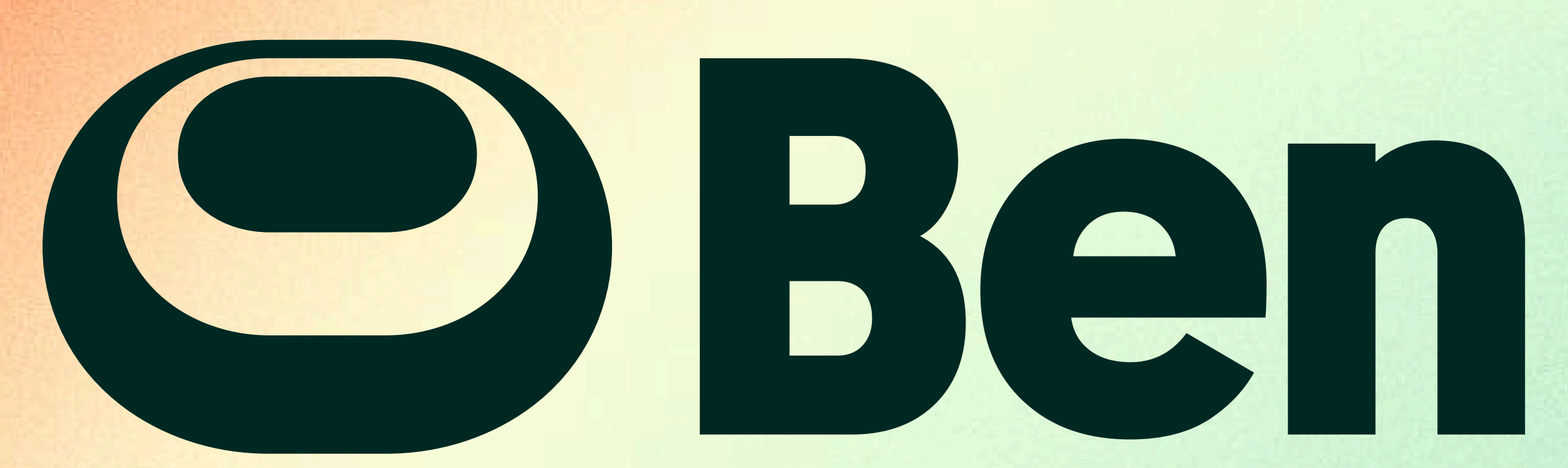
About Ben

Ben is built for organisations that need benefits technology to handle real complexity — across countries, workforces, and a benefit landscape that keeps changing.

Want to talk through how your RFP process is structured, or explore whether our product could meet your needs?

We'd love to hear from you.





thanksben.com