

BUILD. BUY. PARTNER.

The AI-era communications platform playbook

Making the right communications platform decision in the age of AI

How to evaluate ownership, integration, AI readiness, employee experience, governance and long-term value — before you commit budget, headcount, or years of engineering time.

A practical decision framework for Communications, IT and Digital Workplace leaders

 Ownership

 Integration

 AI readiness

 Employee experience

 Governance

 Long-term value

Table of contents

01 The business case

Why employee communications is an operational investment, not a cost centre

02 Can we just build it ourselves?

What AI changed about building — and what it didn't

03 The five-year comparison

Build in-house vs. purpose-built SaaS, honestly compared

04 Why free code isn't the same as the right product

Typing, verifying, deciding — the three levels of building software

05 Patterns worth knowing

What happens when organisations choose to build — and what to ask before you do

06 Partner: what Staffbase + SCP delivers

Why the platform is also the knowledge of 500+ deployments

07 The decision framework

Six questions to put to your own evaluation

The business case

01

Employee communications isn't HR administration. It is the operating system through which strategy becomes behaviour — and when it works well, it directly moves the numbers that matter to your board.

The instinct in many organisations is to treat employee communications as a cost centre — something that exists to keep people informed, managed alongside email and intranet updates. This framing consistently leads to underinvestment, and it misses the actual economic stakes.

23%

higher profitability at companies with highly engaged employees

Gallup, 2023

81%

would work harder if they felt better recognised and informed

Gallup, 2024

50-200%

of annual employee salary to replace, depending on level

SHRM, 2024

70%

of change initiatives fail, most due to poor communication

McKinsey, 2024

The real ROI levers

These are the areas where the financial argument is strongest — and where a purpose-built platform outperforms anything built internally.



Employee retention

A platform that reaches every employee, including deskless workers, with timely information reduces churn — even a 5% improvement saves millions at scale.



Safety & compliance

Safety updates with 75% read rates within one hour are documented evidence of due diligence that reduces liability.



Change adoption speed

Poor communication extends transformation timelines by months — the cost of failed change is measured in hundreds of millions.



Deskless productivity

80% of the global workforce is deskless — mobile-first, native push delivery is operational infrastructure, not a comms upgrade.

"Highly engaged business units achieve 23% greater profitability, 10% higher customer loyalty and satisfaction, and 18% greater productivity."

— Gallup State of the Global Workplace, 2023

Can we just build it ourselves?

At some point in almost every evaluation, someone in engineering or IT raises the build option. This chapter gives you the evidence to address that conversation directly — not to dismiss it, but to make a genuinely informed decision.

"When you can easily turn any idea into reality, it is tempting to turn every idea into reality – and most things should not be produced."

— Deane Barker, Montreal CMS Conference, 2026

The hidden cost structure of building

The most common mistake in build evaluations is comparing the wrong numbers — the upfront development estimate against the annual licensing fee. This ignores the reality of software economics.



60–80%

of software lifecycle costs occur after release — maintenance, enhancements, operations (IEEE / O'Reilly)



70%

of IT budgets are consumed maintaining existing systems, leaving little for innovation (Gartner, 2025)



15–25%

of original build cost required annually just for maintenance, before AI model or compliance updates (Gartner TCO)



18+ months

minimum time to build a comparable employee communications app from scratch (Industry benchmark)

In practice: a \$500K build decision is actually a multi-year commitment. Applying Gartner's maintenance model — 10–25% of build cost in years 1–2, rising to 20–40% by year six — a \$500K build can require \$1–2M in maintenance over its lifetime, before the engineering team required to sustain it.

BUILD IT YOURSELF

**AI CHANGED THE BUILD.
NOT THE COMPLEXITY.**



AI accelerates
development



Integrations
multiply



Security and
compliance
remain yours



Data, reliability
and scale are
your responsibility



Teams, skills and
resourcing don't
go away

The five-year comparison

The honest comparison isn't "build cost vs. first-year SaaS fee." It's five-year build total cost of ownership vs. five-year SaaS total cost of ownership. Made honestly, buying a purpose-built platform is almost always more economical — and delivers results in months, not years.

Dimension	Build in-house	Purpose-built SaaS (Staffbase)
Time to first employee using it	18+ months minimum	Average 3 months from signature
Employee adoption rate	Typically 20–40%, no benchmark data	70% average across 500+ deployments
Push notification reach	Custom infrastructure, untested at scale	75% read within 1 hour, native iOS/Android
Multilingual support	Must be built and maintained	45+ languages with auto-translation
Compliance (GDPR, Works Council)	Your team builds and maintains	Maintained by dedicated compliance team
AI features	You own inference costs and model deprecation	Included and managed by vendor
App Store distribution	Approval process — your team's problem	Already live, proven in stores globally
HRIS integration	Build from scratch, test, maintain	Pre-built connectors; SCIM, SSO, SAML certified
Year 5 engineering cost	\$710K–\$1.1M/year to maintain the team	Included in platform pricing

Build cost estimates based on Pharos Production AI Development report 2026; Gartner TCO research. Staffbase figures based on internal data from 500+ customer deployments.

Why free code isn't the same as the right product

In 2026, code is effectively free. A team can describe an app in natural language, generate a working prototype in hours, and demo it within a week. This is remarkable — and it creates a seductive illusion: that the prototype is the hard part. It isn't. The hard part has never been typing the code. It is knowing what to build.

LEVEL 1

The typing

Syntax, code, implementation. AI has made this nearly free — but it's just the starting point.

LEVEL 2

The verifying

Testing, rejection, knowing what works. Comes from experience — much harder to replicate with AI.

LEVEL 3 – THE GATE

The deciding

What to build at all. Cannot be generated. It can only be earned.

"AI lowered the cost of Level 1. It did not touch Levels 2 or 3. The gate is exactly where it always was."

— Deane Barker, Montreal CMS Conference, 2026

A good platform codifies human nature, not just behaviour

Most internal builds optimise for what users do: they study existing workflows and replicate them. The best employee experience platforms are built for what employees *are* — time-pressured, mobile-first, multilingual, and fundamentally unwilling to seek out information. These truths emerge from watching thousands of deployments, not from a product specification.

"Good brand, legacy, taste and distribution matter more in an era when code is effectively free."

— Deane Barker, Montreal CMS Conference, 2026

Patterns worth knowing

Composite illustrations of dynamics that appear consistently across enterprise technology deployments — shared here as context for the questions worth putting to your own evaluation.



The "we'll build it ourselves" timeline

An 18-month internal build slips to month 26 after a key developer departs, an API changes, and a compliance review forces rearchitecting. It launches with 12% adoption — habits had already formed around WhatsApp and email.

Ask: what happens to the business case if year-one adoption is 20% rather than 60%?



The AI cost curve that wasn't modelled

An internally-built AI search feature hits 340% of its token budget in month one; the model is deprecated in month three; a prompt-injection vulnerability surfaces in month five.

Ask: is your build budget modelling context inflation, observability, guardrails and model deprecation? Hidden costs add 40–60% on top of the raw inference bill.



The technically excellent platform nobody used

A feature-rich internal platform ships with push delivery 23% lower than commercial alternatives and no concept of audience personas. Eighteen months on, the builders have moved to other priorities and monthly active users sit at 31%.

Ask: who owns this platform's product development in year two, three and five?

None of these outcomes result from bad intentions or poor engineering. They result from underestimating what it takes to build and sustain a platform employees actually use, at scale, in regulated markets, over years. The difference is almost always experience — not capability.

Partner: what Staffbase + SCP delivers

06

The comparison most organisations make is a technology comparison — features, APIs, security certifications. That's incomplete. Staffbase is also a consulting organisation that has productised what it knows across 500+ real deployments, in real markets, with real workforces.

An internal build gives you	Staffbase + SCP gives you
Code that does what you specified	A platform built around what employees actually do
Features your team designed	Features shaped by 500+ deployments you've never worked with
Engineers who built the platform	Customer success teams who launched it in your industry and market
A roadmap you own and fund entirely	A roadmap funded collectively by 500+ customers
Push infrastructure you engineered	Native push, 75% read within one hour
A platform that doesn't evolve after launch	A platform that improves continuously, including from your competitors' deployments

The 3-month deployment is a knowledge number, not an engineering number. Every integration pattern, Works Council requirement, App Store edge case, and HRIS sync failure has been encountered — and solved — before. An internal build encountering these for the first time has no playbook. The clock runs. The business waits.

The decision framework: six questions to ask

Before committing to a build evaluation, there is one question worth sitting with: is employee communications the domain in which your organisation creates unique competitive value — or the operating infrastructure through which your actual advantage is delivered? For most organisations, the honest answer points toward buying the best available platform and evaluating any partner against these six dimensions.



Ownership

Who owns the roadmap, the maintenance burden, and the risk when it breaks?



Integration

How well does it connect to your HRIS, SSO, and existing workplace tools?



AI readiness

Is AI native to the platform, or bolted on with untracked inference costs?



Employee experience

Will deskless and desk-based employees actually use it, from day one?



Governance

Does it meet GDPR, Works Council and industry compliance out of the box?



Long-term value

What does five-year total cost of ownership look like, not just year one?

"The cheapest solution on day one is rarely the cheapest solution over five years. The real question isn't 'can we build it?' – it's 'should we build it, and what does it truly cost us if we do?'"

— The Staffbase proposition

3 mo

average deployment from contract signature
Staffbase data

70%

average employee adoption across 500+ customers
Staffbase data

75%

of push notifications read within 1 hour
Staffbase data

45+

languages supported with auto-translation
Staffbase platform

IN CLOSING

Build the right thing.

The organisations that win the next decade won't be the ones that built the most software – they'll be the ones that built the right software, and bought everything else.

staffbase.com/demo