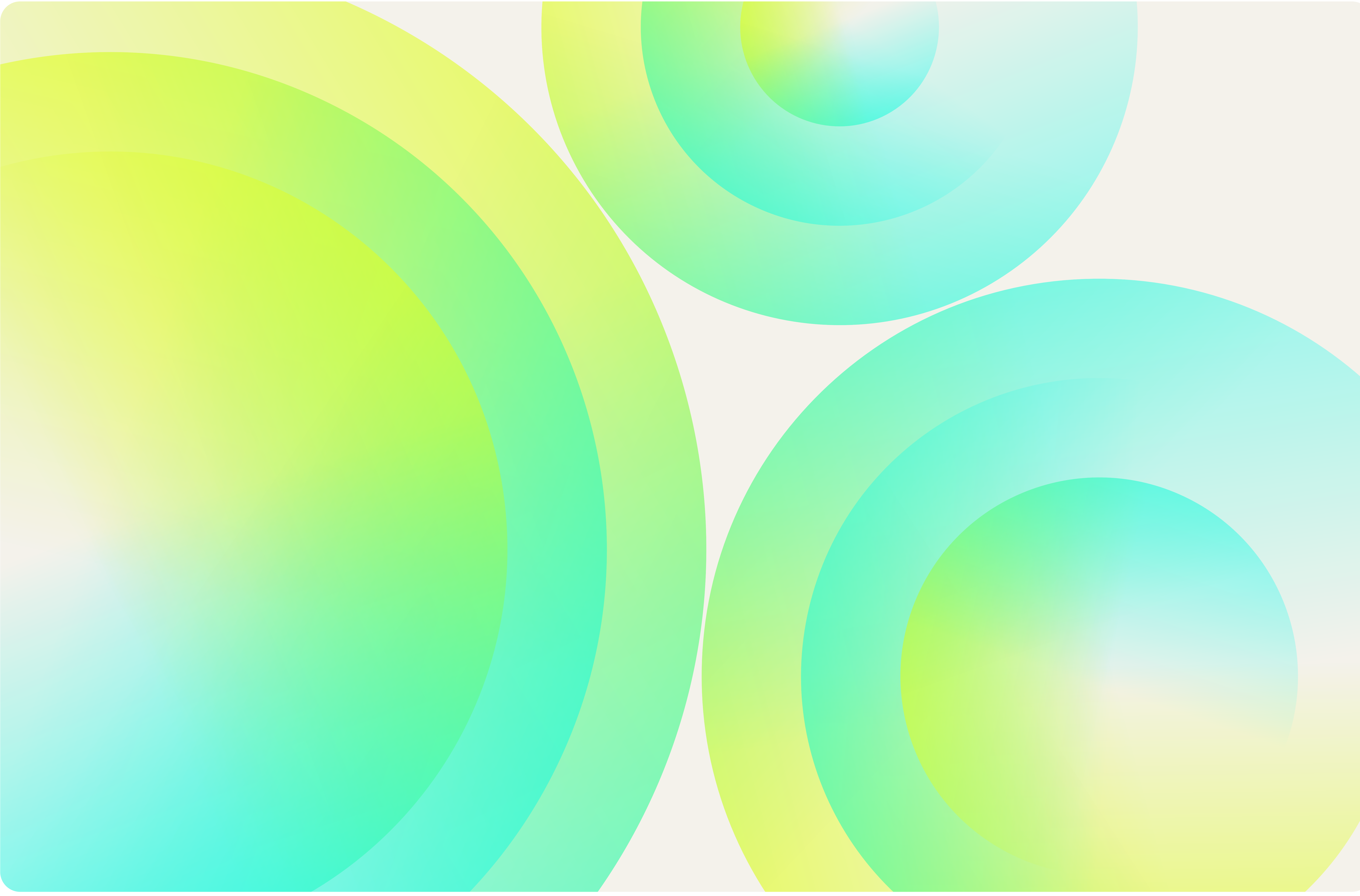
Highnote



Built For Ownership:

The Build, Buy, Or Partner Decision Framework For Embedded Finance Leaders

Most executives believe they must choose between ownership and speed. Build it yourself and own everything, but wait. Partner with a platform and launch fast, but surrender control. This is a false choice. The right infrastructure partner gives you both.

Featuring perspectives from:
Drew Glover | Co-Founder & General Partner, Fiat Ventures
David Galvan | Head of Issuing Sales, Highnote

The Decision Every Leader Faces

Every company building an embedded finance program reaches the same inflection point: do we build this capability ourselves, acquire a company that has it, or partner with a platform that can get us there faster? These are not equivalent choices, and most leadership teams approach them with an incomplete picture of what each path actually requires, and what it costs to get it wrong.

The landscape has changed enough to make all three paths look more attractive than they are. API-first infrastructure has compressed integration timelines. AI coding tools have lowered certain development barriers. Fintech consolidation has made interesting assets more available. What has not changed is the compliance burden, the banking and network relationships, the operational overhead, and the engineering opportunity cost of building payment rails instead of the product that wins your market.

BUILD	ACQUIRE	LEGACY PLATFORM	HIGHNOTE
Own everything from day one	**Purchase existing capability**	**Launch fast, on their terms**	**Launch fast, keep full control**
Timeline: 18+ months **Cost:** High, front-loaded **Ownership:** Complete, if successful	**Timeline:** 12-24 months to integrate **Cost:** Very high **Ownership:** Inherited, with prior constraints	**Timeline:** Weeks to months **Cost:** Low upfront, expanding **Ownership:** Gradually eroded	**Timeline:** Weeks to months **Cost:** Low upfront, transparent **Ownership:** Complete and preserved
Architecture: Custom from day one	*Architecture: Inherited constraints, aging core*	*Architecture: Batch-era cores; opaque, shared infrastructure*	*Architecture: Modern cloud-native; real-time API; full program management included*

Most frameworks present three paths. There are four. The distinction between the third and fourth is the one that changes everything: most embedded finance partners give you speed in exchange for ownership. The fourth path, a modern ownership-first platform, gives you both. That difference is the subject of this playbook.

> Every aspect of the business needs its own special, custom-tailored fit. One has high arches, one is flat-footed. We are not looking for one solution anymore. We need a partner that has the flexibility to service every dimension of the business."

Drew Glover Co-Founder & General Partner, Fiat Ventures

The True Cost of Building

The case for building is compelling: own the infrastructure, own the economics, own the roadmap. For companies with the right team and long enough runway, the economics can eventually work. The problem is that most build decisions are made on a partial cost model. The direct costs are visible. The indirect costs are not.

An embedded finance card program requires bank sponsorship or principal network membership, BSA/AML and OFAC monitoring, PCI DSS certification, and Regulation E compliance for debit and prepaid products or Regulation Z for credit. These are not engineering problems. They require specialized expertise, dedicated headcount, and ongoing operational investment that does not decline after launch. Add fraud operations, cardholder servicing, and dispute resolution, and you have a second organization inside your first. It is one your engineering estimate almost certainly did not include.

AI coding tools have changed certain development economics, and executives who have watched engineering teams use them effectively arrive at the build conversation with new confidence. That confidence, in embedded finance, is frequently misplaced. An AI-assisted team building a card program still needs a bank sponsor and licensed principal network relationships. It still needs the legal and operational organization that keeps the program compliant after launch. AI accelerates the code. It does not replace the infrastructure.

“AI is like the most beautiful bait sitting in the water. I want to chomp it. But it can hurt you if you don't have the right infrastructure.

When companies think they're going to build one thing, they open up Pandora's box and realize they actually have to build 50 things. And then they have this MVP that's only a portion of what they wanted, and they're back to the question of whether they should have partnered instead.”

Drew Glover Co-Founder & General Partner, Fiat Ventures

Every month I sit across from a leadership team with a clean build plan. The engineering estimate is solid, the timeline is 18 months, and the internal narrative is that they can do this. Then we work through the full cost model together: bank sponsorship, BIN fees, program management, BSA/AML operations, dispute management, cardholder servicing, PCI compliance, and the ongoing regulatory change management that never stops once you are live. By the time we are done, the cost has doubled and the timeline has grown by a year. That is not a Highnote pitch. That is just what building actually requires when you account for everything.”

David Galvan Head of Issuing Sales, Highnote

The True Cost of Acquiring

The acquisition path is the least discussed of the four options. A company with an existing payments capability represents a seemingly direct route to the infrastructure, regulatory standing, and technical architecture that would otherwise take years to build. In a market where fintech valuations have compressed and interesting assets have become more available, the argument is real. It is also frequently misunderstood.

Acquiring a fintech asset does not bypass the hard problems of embedded finance infrastructure. It purchases them. Every licensed or sponsored entity carries its own regulatory obligations, compliance posture, and banking relationships. None of these transfer cleanly under new ownership. The technical infrastructure reflects decisions made at a specific moment in time; acquiring it means inheriting those constraints alongside the capabilities. And the engineers who built and maintain the systems often leave during the transition, taking the institutional knowledge with them. This is a well-documented pattern in technology acquisitions, and it is particularly acute in financial infrastructure, where the cost of losing that knowledge is measured in regulatory exposure and operational failure, not just slower development.

Change-in-control approvals from banking regulators and card networks, where required, can add months to the integration calendar after close. The executive who models acquisition as a faster route than building should test that model against the date when the acquired capability is fully operational inside their organization, under their compliance posture, on their technology stack. That date, not the close date, is when the acquisition delivers on its premise.

The Hidden Cost of Legacy Partnership

The third path, partnering with a legacy platform, is the one most companies have already taken or are actively considering. It is the most marketed option, the most available, and the one that generates the most regret. Understanding why requires a clear definition of what legacy actually means, because the word is often softened in platform marketing until it loses its meaning.

Legacy, in this context, is architectural. These platforms were built on batch-processing cores designed before real-time APIs became the standard for financial infrastructure. Data moves in scheduled intervals rather than continuously. Subscribers interact through aggregate reporting dashboards rather than programmable, transaction-level API access. Compliance infrastructure is shared across the platform's subscriber base rather than configured to each program. The economics are opaque because the platform, not the subscriber, owns the ledger. None of these are configuration choices that can be changed with a support ticket. They are foundational decisions built into the platform's design, and they determine the ceiling of what a subscriber can do with their own program regardless of what the contract says about ownership.

Within that model, the platform captures a meaningful share of program economics, controls the roadmap, and owns the data relationship. The subscriber gets access to the infrastructure but not control of it. Roadmap decisions require waiting. Data access means receiving reports rather than querying a ledger. Pricing that looked reasonable at signing has a way of expanding at renewal as switching costs accumulate.

> “The story I hear most often is a company that launched well on a legacy platform, hit their first real growth milestone, and then started running into walls they did not know were there. They want to add a credit product. It is on the roadmap. They want real-time transaction data. They get a reporting dashboard. They want to adjust their reward structure. That requires a contract amendment. None of these feels fatal in year one. In year three, they all add up to a program that cannot move at the speed the business needs to move.”
>
> **David Galvan**
> Head of Issuing Sales, Highnote

The hidden cost of legacy partnership is not a single decision. It is an accumulation. The first year feels productive. By year three, the constraints have become operational reality. By year five, a migration has been scoped, estimated, and deferred because the switching cost has grown to match the cost of fixing the underlying problem. The companies in the most difficult infrastructure situations are not the ones that made obviously bad decisions. They are the ones that made reasonable early decisions, accepted the constraints, and found themselves with a platform built for the business they were rather than the business they had become.

> I have had prospects tell me they did not know what was happening inside their own program. They had aggregate reporting but not transaction-level visibility. They could see the totals but not the individual decisions being made on their cardholders' behalf. That is not a reporting gap. That is an ownership gap. And once a business understands that distinction, the conversation about which platform they should be on changes very quickly."

David Galvan Head of Issuing Sales, Highnote

The Decision Framework

The four paths have been examined on their own terms. What remains is how to actually make the decision, with incomplete information, under time pressure, and with a credible sales argument behind every option.

Start with the right question

Before evaluating features, timelines, or pricing, ask one question: is embedded finance your product, or is it infrastructure that enables your product? If it is your product, the case for ownership is strong and the partner evaluation must be rigorous about control. If it is infrastructure, the case for the fastest, most capable partner is clear and the build argument requires an unusually high bar.

From there, build an honest cost model over five years, not twelve months. Include the full compliance and operational build, the engineering opportunity cost, and a realistic probability that the build succeeds on schedule. A model that includes all of these inputs tends to shift the answer.

> "The companies that evaluate us well are the ones that evaluate the control model before they evaluate the feature list. They ask what they can configure without filing a change request with us. They ask who owns the transaction data. They ask what happens to their roadmap if our product priorities ever diverge from theirs. Those are the right questions. And they are the questions that reveal quickly which platforms are built to be controlled by the subscriber and which platforms are built to control the subscriber."
>
> **David Galvan** Head of Issuing Sales, Highnote

Evaluate the control model, not the feature list

The most important question to ask any platform partner is not about features or pricing. It is about control: what can you do with your program, on your timeline, without requesting a change from them? A platform built around its own priorities will describe a roadmap. A platform built around yours will describe a set of controls you own.

> **I wish more executives made this decision based on speed and a fearlessness to fail. Failing doesn't mean launching to your entire user base and failing epically. It means launching with a cohort of a hundred, learning what that best-tailored fit is, and doing it in ways that don't cost you the whole company to find out."**
>
> **Drew Glover** Co-Founder & General Partner, Fiat Ventures

The Partnership That Preserves Ownership

The choice between ownership and speed is false. The fourth path delivers both. What ownership actually means in an embedded finance program is specific: controlling your roadmap, owning your program economics, accessing your transaction data, and designing the subscriber experience your customers actually want. Not one of these is a contractual claim. Each is an architectural reality determined by how the platform is built.

A platform built for ownership gives you programmable primitives: the ability to configure and compose at the level of individual business logic, not pre-built modules. It gives you direct ledger access with real-time, transaction-level visibility. And it is backwards-compatible, meaning new capabilities are additive. The investment you made at launch compounds.

The compliance and operational infrastructure that makes a full card program real: bank sponsorship, BIN management, BSA/AML and OFAC monitoring, PCI DSS compliance, dispute management, and cardholder servicing. All of it is managed by Highnote at platform scale. This is the complete program management layer that the build path requires a company to construct from scratch, and that legacy platforms provide on their own terms rather than yours. On Highnote, it operates as a foundation you depend on but do not have to operate, so that your organization builds the product your customers see rather than the compliance infrastructure they never will.

"What we built at Highnote starts from a different premise. The legacy assumption is that the platform owns the infrastructure and the subscriber owns the experience. Our assumption is that the subscriber should own both. That means full transaction-level access through a direct API, configurable program parameters that belong to your business, and a double-entry general ledger that is yours, not ours. When a Highnote customer wants to know what happened inside their program, they do not submit a request. They look at their own data."

David Galvan
Head of Issuing Sales, Highnote

The backwards-compatible architecture matters commercially in a way that is easy to underestimate at the start of a relationship. When we release a new capability, you adopt it on your timeline without rebuilding. Your existing integration stays intact. That means the investment you made at launch compounds over time rather than requiring a new implementation project every time the platform evolves. The cost of staying current on Highnote is close to zero. That is not the experience companies have had on legacy platforms."

David Galvan Head of Issuing Sales, Highnote

THE UNIFIED PLATFORM ADVANTAGE

Highnote's unified platform includes issuing, acquiring, credit, and money movement on a single purpose-built, modern cloud-native architecture with a real-time double-entry general ledger that belongs to the subscriber, not the platform. Full program management is handled by Highnote at platform scale: bank sponsorship, BIN management, BSA/AML and OFAC monitoring, PCI DSS compliance, dispute management, and cardholder servicing. Subscriber teams focus on the product rather than the compliance back-office. A program launched on Highnote can expand into acquiring, credit, or money movement without replacing the infrastructure it launched on. The GraphQL API gives subscribers programmable, transaction-level access to every platform layer. The self-serve sandbox lets teams build and test before touching production. The investment compounds. The ownership is preserved across every expansion.

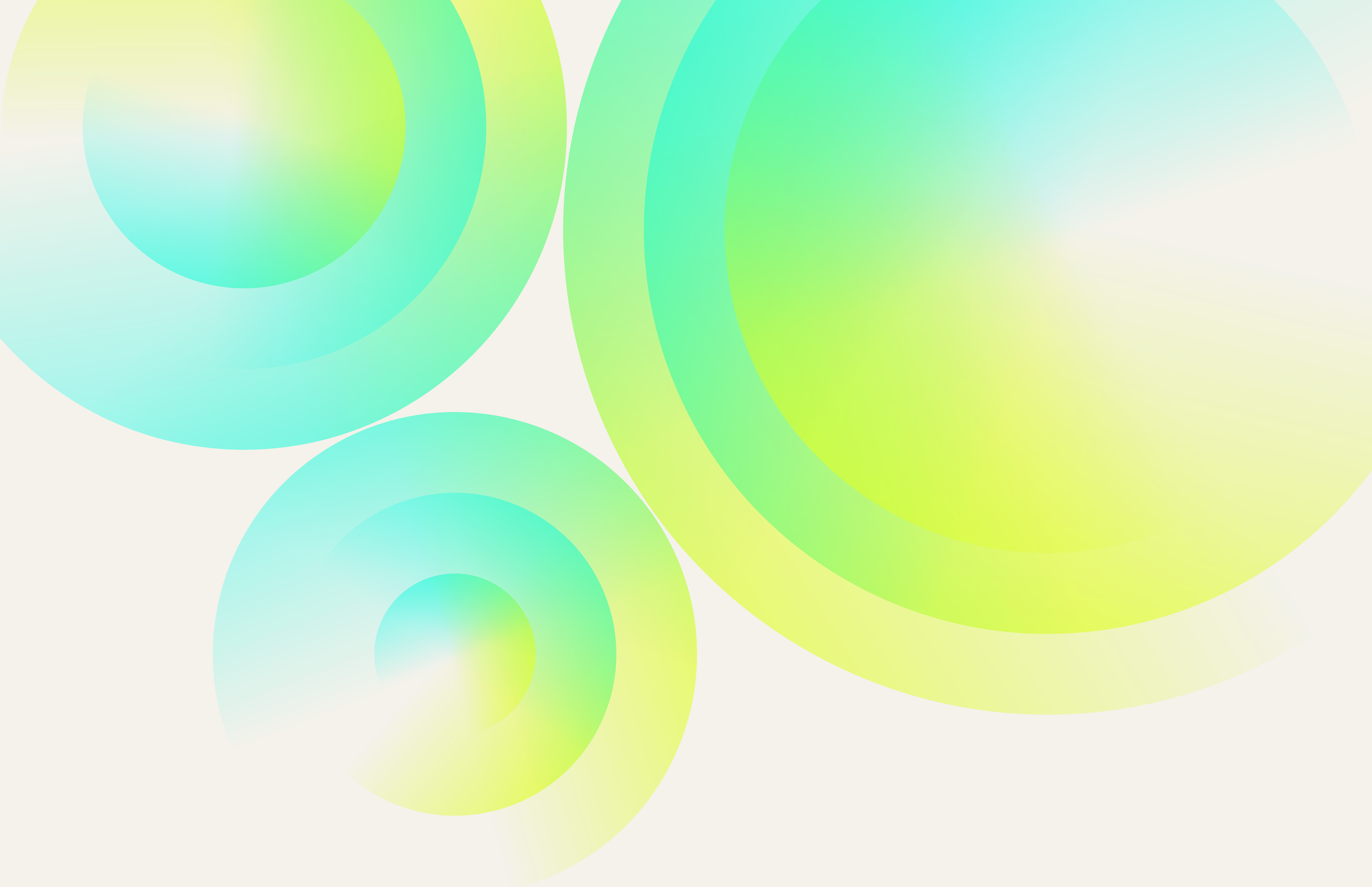
"For a sprinter, being 1% better is the difference between a world record and finishing seventh. One to two percent improvement over five years means a catastrophically different business, in a good way."

Drew Glover Co-Founder & General Partner, Fiat Ventures

The right partnership does not ask you to choose between moving fast and staying in control. It treats ownership as the baseline, not the premium.

"The executives who get the most out of Highnote are the ones who came in asking what they could build, not what we would sell them. They wanted to design their program, not select from a configuration menu. That is exactly what this platform is built to support. And the ones who built that way are running the most differentiated embedded finance experiences in their categories right now."

David Galvan Head of Issuing Sales, Highnote



Built for Ownership

Launch on a unified platform, design your program your way, expand into new capabilities without rebuilding, and keep the data, economics, and roadmap control that belongs to your business.

Talk to an ownership specialist →

See how customers are building →