Forest Admin

Operations

When Tech Invests in Making Ops Autonomous

The secret of four
Fast-Scaling Startups

When Tech Invests in Making Ops Autonomous

Editorial: when Tech makes Ops autonomous

Here's an interesting question for CTOs: What if operations team could handle 80% of their tooling needs without submitting a single engineering ticket?

Most tech leaders think this sounds too good to be true. They've seen ops teams struggle with clunky dashboards, endless spreadsheets, and manual processes that don't scale. The natural response is to keep building internal tools and fixing ops problems one by one. But there's a different path.

The tech teams at fast-scaling ops-intensive companies have discovered a smarter way. Instead of staying involved in every ops request, they're investing upfront to make ops completely self-sufficient. The result? Engineering time gets freed up for product work, and ops teams become faster and more agile than ever before.

We went and asked four leaders at fast-growing startups, Belvo, Givebutter, Motto and Swan, to share their ambitious way of thinking about this. Here is the gist of it, and we kept some of their best quotes too.

I hope this helps you challenge your organisation for the best and helps you unlock the next level of operational excellence... while freeing up engineering time.

Guillaume
Head of Marketing
Forest Admin

Why Ops Autonomy Matters (for Ops & Tech)

The Hidden Cost of "Just One More Fix"

The ops team comes to Tech with another request. They need a new dashboard. Or a way to segment customers. Or access to transaction data. It's always "just a quick fix."

Each request seems small. But they add up fast.

Soon, the engineering team is spending 30% of their time on internal tooling requests. Product development slows down. New features get delayed. The best developers become glorified report builders.

Or the Ops team keeps waiting and begging. Maybe they use spreadsheets. Not an ideal situation for them and for the customers whose cases they are handling.

As Mark Rummel from Givebutter explains:

> "Engineering time is probably one of the most scarce, the most sought after thing that everybody at the company wants. So, we're always fighting for those engineering resources."

This is the ops dependency trap. And it gets worse as you scale.

Why Linear Scaling Breaks

Most companies think scaling ops means hiring more people. Need to handle twice the volume? Hire twice the team.

This approach falls apart quickly.

At Motto, they realized early that managing a bike-sharing fleet couldn't scale with headcount alone. They had to improve their ratio of customers handled by customer service rep.

Antoine Bouttier, the CEO and Cofounder of Motto explains:

> ""When we started Motto, we reached our 450 bikes in the street and that was the ratio that we established for our customer service. We were saying every 450 bikes we need to hire someone. And today (...) we have 2,500 bikes and we still have one people at the customer service,"

And how to improve productivity? Through continuous review of tasks and automation.

> "After three years of operating the business you basically realize that you had automatized 75% of the tasks that at first you were thinking were gonna need a human."

What Autonomy Actually Looks Like

True ops autonomy isn't about giving your team better spreadsheets. It's about letting them build and modify their own workflows without touching code.

This means ops can:

1. Set up new workspaces and dashboards instantly
2. Create custom data views and segments
3. Build approval workflows and dispatching rules
4. Connect to external APIs and tools

All without engineering involvement. And with the capacity to make changes.

"Being able to be no-code where we're able to set up inboxes and segments and workspaces and all that no-code is huge (...) to be able to go in and set up workspaces and set up (request) inboxes without having to submit a single ticket to engineering for those to function is amazing and so so appreciated," notes Mark Rummel, about using his back office tool built on Forest Admin.

The New Tech-Ops Pact - What Autonomy Looks Like in Practice

The Upfront Investment

Of course, making Ops autonomous requires Tech to do more work upfront. Not less.

They need to set up the data connections. Configure user permissions. Build the initial integrations with your core systems. This takes time.

As Jonathan Araujo from Belvo puts it:

> "Invest time to make time because the thing is, even if you have the smartest people, you need to invest a lot of time in educating your own team so they can later on free you from time by not needing you. And the same thing happens for improving processes."

The payoff comes later. Once the foundation is solid, ops will be able to handle everything else.

What Tech Does Once vs What Ops Does Daily

Here's how the division of labor works in practice:

Tech Team - One-Time Setup:

- Database connections and data sync
- Initial workspace configuration
- Security and access controls
- Core API integrations

Ops Team - Daily Management:

- Creating new dashboards and views
- Setting up customer segments
- Building workflows and approval processes
- Connecting new external tools

At Givebutter, the ops team can set up entire workspaces without submitting engineering tickets. Their engineering team focuses on product features instead of internal tooling requests. And this enabled them to scale very well.

Mark Rummel explains:

> "Having Forest Admin has been a great part of our ability to scale and grow (...) And it has scaled so well for us, when we go from just smaller record of customers or smaller records of transactions to now millions and millions of those records and transactions all living in Forest Admin where our team can track and understand."

The Forest Admin Advantage

Most companies try to build these capabilities in-house. This usually takes months of engineering time and creates ongoing maintenance overhead.

Forest Admin plugs directly into your existing database. No data migration. No rebuilding your data models. Your engineers set it up once, then ops takes over.

"Most of our tech stack is based on JavaScript technologies, and there was pretty much nothing else out there that could even compare to what Forest Admin offers and also the support that Forest Admin offers behind the solution," says Jonathan Araujo from Belvo.

The tool grows with your team. As your data gets more complex, Ops can handle most of the complexity without additional engineering work.

From Reactive to Scalable - What This Unlocks

Beyond Efficiency, Strategic Capacity

Autonomous Ops teams don't just work faster. They work differently.

Instead of waiting for engineering to build solutions, they identify problems and fix them immediately. They experiment with new workflows. They optimize processes in real-time.

Antoine Bouttier of Motto tells a good anecdote on the topic:

> "When you put the tech team and the ops team very close, basically in the same room and they get to talk every day on the problems that our users have that our teams have, you create this synergy that is very healthy."

.This creates a feedback loop that drives continuous improvement.

The Innovation Effect

When ops teams can build their own tools, they become innovation drivers instead of cost centers.

Swan's operations team rarely asks their product team for new features. They build most of what they need using no-code tools and flexible back-office systems.

"We are very autonomous today in Swan (...) And we very rarely ask for additional stuff from our product and dev team," says Maxime de Juniac from Swan.

This frees up the product team to focus on customer-facing features while ops handles internal optimization and necessary adaptations.

Maxime from Swan adds:

> "I think being agile and being autonomous in Ops allows us to go much faster. (...)On operations, we have processes that have a lot of variation depending on the country, depending on the partner, depending on the type of end users. That changes a lot also because we see some fraud, we see some money laundering. We have scale, we have new partners and we need to adapt very fast."

Automation That Actually Scales

The goal isn't to replace humans with robots. It's to let humans focus on high-value work.

At Motto, customer service used to spend hours tracking bike repairs. Now that process is completely automated. Customers report issues through the app, and the operations team sees everything in their dashboard automatically.

"The customer service team doesn't need to look at the repairs of any rider it's been completely automatized," says Antoine Bouttier.

The customer service team can focus on complex problems instead of data entry.

Making the Case Internally - Your CTO Conversation Guide

Need to convince your CTO to get onboard? We've got your back.

The ROI Framework

Here's how to make the business case for ops autonomy:

Current State Costs:

- Engineering time spent on internal tools (estimate hours per month)
- Ops team efficiency limitations due to manual processes
- Delayed product features due to internal tooling requests

Investment Required:

- Initial setup time (typically 2-4 weeks of engineering work)
- Tool costs (significantly less than building in-house)
- Training time for ops team

Expected Returns:

- 70-80% reduction in ops-related engineering tickets
- 2-3x improvement in ops team efficiency
- Faster product development cycle

Addressing the "Build vs Buy" Discussion

Most CTOs want to build everything in-house. This makes sense for core product features. It doesn't make sense for internal admin tools.

Building a flexible ops platform takes months of engineering time. Then you need to maintain it. Add new features. Fix bugs. Handle security updates.

Forest Admin gives you enterprise-grade functionality without the engineering overhead. Your team gets back to building products instead of maintaining internal tools.

"Forest Admin is a pretty strong enabler. We have been trying to focus on supporting our operations team. Not on many tools, but for example, Forest Admin. We have been

learning a lot about the product, trying to centralize as much as we can to support our operations team through Forest Admin," explains Jonathan Araujo.

Timeline and Success Metrics

Month 1: Foundation

- Database integration and initial setup
- Core team training and access configuration
- Basic workflows and dashboards

Month 2-3: Independence

- Ops team building their own workflows
- Reduced engineering ticket volume
- Process optimization and automation

Month 4+: Scale

- Ops team fully self-sufficient
- Engineering team focused on product development
- Continuous process improvement without engineering involvement

Track these metrics:

- Engineering tickets from ops (should drop 70-80%)
- Time to implement new ops workflows (should decrease significantly)
- Ops team satisfaction and capability growth

"We have all these people with great mentality, great experience, but maybe not all of them super tech savvy to get them to be tech savvy. And that took of course quite some time. But now we are there. We are there. And you can definitely see how it paid off," reflects Jonathan Araujo on the transformation process.

Conclusion

The path to scalable operations isn't hiring more people. It's making your existing team autonomous and equip them like superheroes.

This requires an upfront investment from your engineering team. But the payoff is substantial. Engineering gets back to product development. Operations become faster and more agile. Both teams can focus on what they do best.

The companies that figure this out early have a massive advantage. They scale efficiently while their competitors risk getting bogged down in operational complexity.

The question isn't whether you'll eventually need autonomous ops. It's whether you'll build this capability soon enough.

Thanks to the teams who shared their story

Big thanks to the people who joined us on the podcast and provided the insights for this guidebook.

Mark Rummel, VP Customer Experience at Givebutter, an all-in-one fundraising platform for non-profits.

Jonathan Araujo, Senior Engineering Manager at Belvo, the leading Open Finance and payments platform in Latin America

Antoine Bouttier, CEO & Co-founder at Motto, a hassle-free e-bike subscription service.

Maxime de Juniac, Chief Service Officer at Swan, a European Banking-as-a-Service Fintech

Listen to the full conversations: check out our "Mission: Ops Possible" podcast.

Annex A: 90-Day Autonomy Roadmap

Week 1-4: Assessment and Setup

- Audit current ops workflows and pain points
- Identify key data sources and integration needs
- Set up Forest Admin database connections
- Configure initial user permissions and team structure

Week 5-8: Foundation Building

- Create core dashboards and data views
- Set up initial workflows for common processes
- Train ops team on platform capabilities
- Establish success metrics and tracking

Week 9-12: Handoff and Optimization

- Ops team takes over workflow creation
- Engineering support shifts to consultation only
- Measure ticket reduction and efficiency gains
- Identify additional automation opportunities

Annex B: Autonomy Assessment Checklist

Current Dependency Indicators

- ☐ Ops submits 5+ engineering requests per month
- ☐ Manual data export/import processes
- ☐ Spreadsheet-based workflow management
- ☐ Limited real-time access to operational data
- ☐ Delayed response to operational issues

Readiness Indicators

- ☐ Clearly defined operational processes
- ☐ Ops team eager to learn new tools
- ☐ Engineering leadership buy-in
- ☐ Budget allocated for tooling improvements
- ☐ Measurable operational pain points

Success Benchmarks

- ☐ 70%+ reduction in ops-related engineering tickets
- ☐ Sub-24 hour turnaround for new workflow creation
- ☐ Ops team satisfaction score improvement
- ☐ Measurable process efficiency gains
- ☐ Engineering team time reallocation to product work