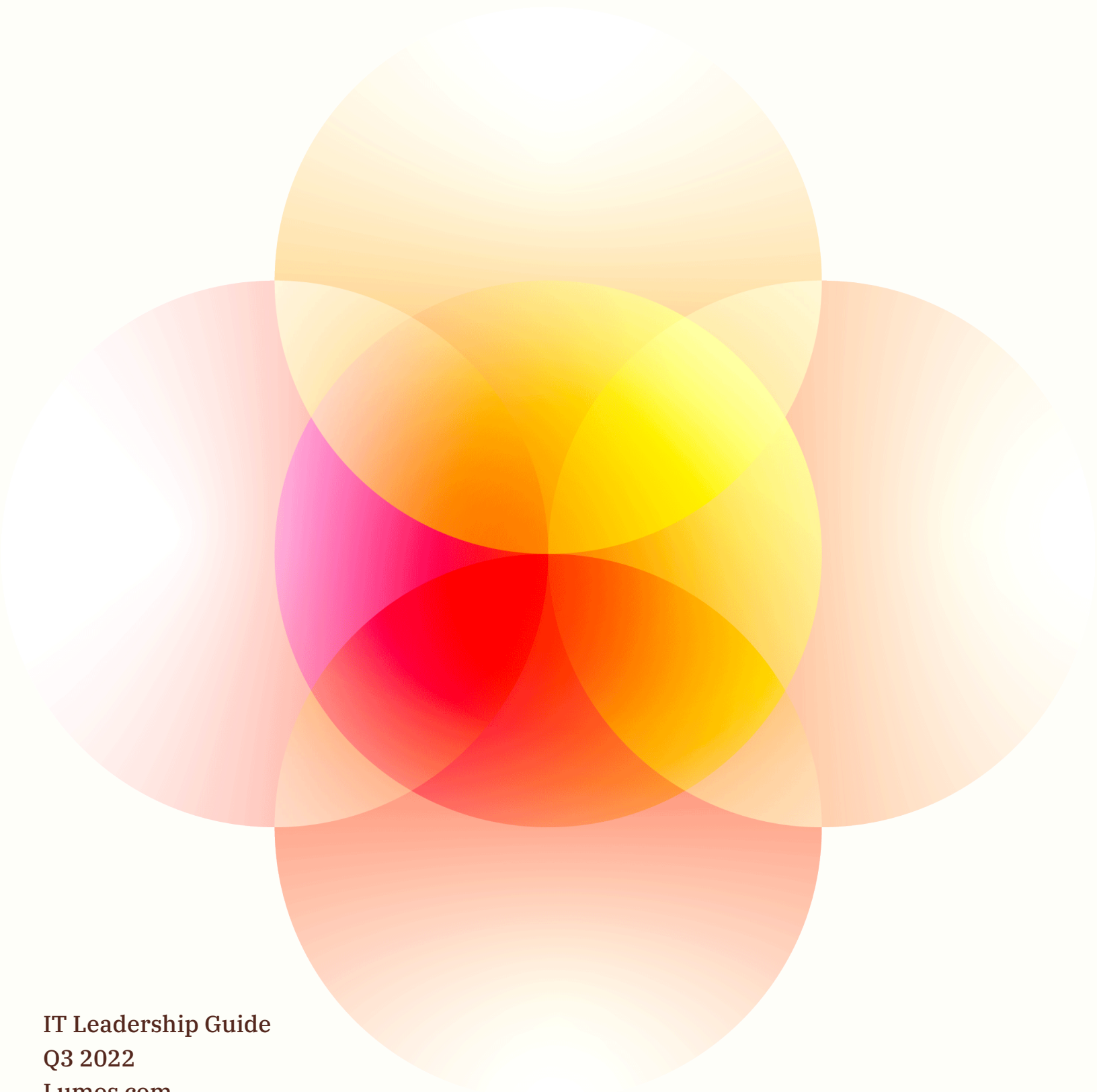


# Crowdsource IT Across Your Company

By Jeff Martin &  
Dillon Wheeler  
Senior IT Systems  
Engineers @GitLab



IT Leadership Guide  
Q3 2022  
Lumos.com

## Introduction

Jeff Martin looked up from his computer screen and smiled. At that time, he was a Senior Demo Systems Engineer in the sales division of GitLab and he was about to embark on his latest project that would not only have a huge impact on GitLab but with potential to help engineers and IT organizations across the industry.

As part of the GitLab company values, there is a strong focus on iteration and efficiency, particularly with using boring solutions, minimum viable change iterations, efficiency for the right group, and global optimization.

Many organizations find themselves using boring solutions, however it's important as IT leaders to look at how your team can build or enable the next iteration of the boring solution towards self service for users in addition or instead of administrator self service.

# Manage Change

It's crucial to enable a large group of engineers to be more productive. That's why one of Jeff's guiding principles is to strategically invest time to automate the most tedious tasks for hundreds of team members to solve economies of scale problems.

As IT organizations evolve and mature, Jeff suggests that the IT team's focus on resolving help desk tickets evolves or expands into elevating the whole company through automation. To do that, IT needs to embrace great ideas that come from all over the company.

For the last eleven years, Jeff has built an engineering career building impactful internal platform-as-a-service products for customer-facing departments (sales engineers, professional services, customer training hands-on labs, customer support, etc.). The irony is that Jeff didn't join the IT department until last year and has been in a solo or small team shadow IT infrastructure capacity for most of the last decade.

## Live By The Principle To Crowdsource IT Across The Organization

Smart administrators, engineers, and operators are spread across the whole organization – marketing operations, sales operations, DevOps and SRE, customer success, customer support, professional services, and many others. There is a general shared belief across most roles that automation makes things better, and many teams will seek to automate their day-to-day workflows where it makes sense. You'll find that most engineering departments have scratched their own itch with homegrown internal tools that can be leveraged too.

A huge part of IT is all about being an enabler, which means finding great ideas in the company and helping them flourish. No matter which company you work for, Jeff has three principles for you to follow to level up your IT organization:

- **Identify operational challenges solved by non-IT teams.** Find operators across different departments that are already building innovative solutions for existing niche department problems. Most likely, the leadership team isn't aware of some internal tools/initiatives. Open communication with individual contributors regarding their day-to-day workflow is imperative in discovering these grass root initiatives.
- **Create momentum through early 'a-ha moments'.** Get buy-in from leadership by demonstrating business value and early ROI that paves the way for more funding. Sometimes the buy-in actually comes from the individual contributors and crowdsourced adoption for useful apps, templates, tools, etc. Some of the best ideas are adopted by hundreds of users before they are recognized and/or adopted by leaders.
- **Enable cross-departmental teams.** Enable non-IT teams by making their side projects full-time initiatives. Take the time to learn how they are solving the problem and define lightweight guardrails. Encourage them to continue their work with the intent of expanding the scope of how the tool is used to benefit more team members in the organization.

## Identify Operational Challenges Impacting Non-IT Teams

Since IT is often a function of Finance and tasked to meet IT requirements for the whole organization, it rarely thinks about corner cases for specific departments. Jeff muses that most traditional functions of IT are based on a compliance-based charter. Instead of a “one size fits all” approach to IT, think about categorizing different needs based on the sensitivity of the data involved and what’s in-scope or out-of-scope for compliance.

This helps refine the focus on “what to care about” and “what to let others solve for.” At GitLab, the team uses a color coded Data Classification Standard that helps focus on “the important or scary stuff”. For the “not so scary” stuff, let departments solve their own problems and help enable them to be successful.

Jeff recommends getting directly involved with improving day-to-day life for a specific department. Remember, automation innovation happens across the whole company.

You don’t need engineers in IT to solve these problems. Your job in an IT role is to find creative solutions that different teams are already doing and add additional fuel by getting involved with upcoming iterations and project sponsorship. Build a habit of interviewing stakeholders at the individual contributor, management, and leadership levels in each department to see what their biggest technology-related problems are. Then, quantify the potential impact of automation or how they are already automating those problems. *When Jeff worked within sales, that’s what happened.*

One of the core problems that Jeff focuses on is the fact that provisioning ephemeral cloud infrastructure is hard – regardless of whether it’s AWS, GCP or other cloud providers. He changed the

face of how GitLab operates to this day with the HackyStack project by focusing on the green data (sandbox and dev/test) automation problems that were not being invested in. The goal is to make provisioning infrastructure easy and repeatable for users in any department.

Jeff states, “The average access request can take 30-60 mins. The average Terraform environment can take up to 5 hours to provision, even after you know what you’re doing. The reality is that most infrastructure requests require a lot of button clicking by administrators, and human administrators don’t scale. There’s a better way with self service provisioning.”

After reviewing the lessons learned from some security incidents, Jeff found a huge operational problem worth solving. And, this problem was totally out of the scope of IT in the traditional sense. Traditional IT comes with organization-wide responsibilities; however, this project was all about enabling members within the sales department. At least at first...

There are many projects that are built in the evenings or on Friday afternoons. Your job in IT is to find them, enable the operators to show early business value and then turn the side project into a full-time initiative. In that way, IT acts like a hub for internal innovation.

After finding innovative projects launched by non-IT teams, IT can play the role of an advocate for automation initiatives across the company. There is always a group of early adopters in an organization that love to try new tooling out to improve the modus operandi. Help operators in different departments find that group and let them test the product. Success will then become increasingly more visible to leadership and the whole organization.

## Putting This Principle Into Practice: Project HackyStack

Jeff started as an engineer in the sales organization at GitLab. In the early days, he received many requests from professional services and sales engineers to get access to their shared AWS account to perform experiments. At the time, they had a single AWS account that was shared by 100+ team members. It was a bit of a free-for-all where the hope was that each user prefixed their resources with their name. We all know how well that works.

As you can probably imagine, the need for a project like HackyStack was born out of a remediation from a security breach. The scary security breaches are the ones that affect production, staging, or customer and financial data (see GitLab's data classification standards for orange and red data). The security breaches that Jeff deals with are ephemeral dev and test environments (green data) that don't have any sensitive data affected, as long as the attacker can't move laterally across the organization or network.

Jeff recalls, "You know the drill. You might have 3, 10, or 100 people doing different experiments with different virtual machines. Everyone has an API key and occasionally someone will accidentally put their API key in their Terraform or script source code (which is open sourced on GitLab or GitHub). When a hacker finds it, they use the key or sell it on the black market that can be used to get the AWS account and provision a lot of crypto mining EC2 instances.

It happens to many organizations, and isn't a data breach that you usually hear about since it's in a dev/test environment that doesn't have any real data; the malicious actors are simply getting you to pay for their crypto mining activities.

When this happens in a shared account, you have to shut down everybody in order to clean up the mess. The problem we had was security blast radius for sandbox environments. I've been solving

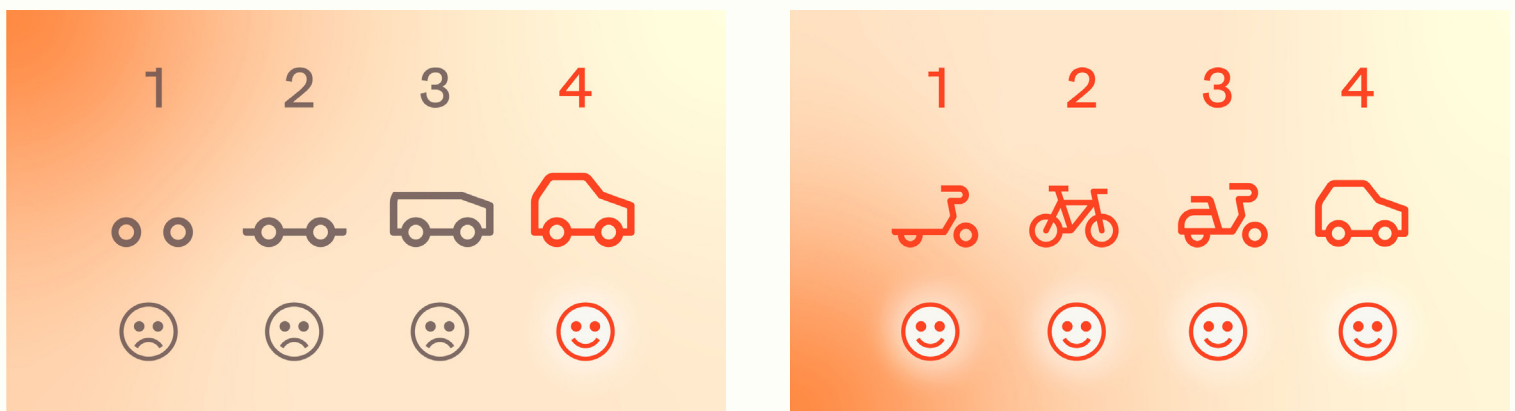


this problem for training hands-on labs for many years prior, however that was at a higher level of the infrastructure stack with hypervisors, VPCs, virtual machines, clusters, and containers.

They evaluated AWS tooling, however, there was no self service UI and limited API endpoint options at the time. Jeff didn't wait for permission and built the first iteration of HackyStack on Fridays over 2 months. The goal? Let team members sign in with Okta, including their department metadata, and use it to create an AWS account in their AWS organization in the appropriate organization unit to create an IAM user in the new AWS account using the AWS API.

## Create Momentum Through Early Iterations and Aha Moments

The team deployed HackyStack at GitLab as the Sandbox Cloud in November 2020. They found their early adopters among Solutions Architects and Professional Services. They advertised their initial version of HackyStack and already had 115 AWS accounts within 6 months.



Jeff recalls, “The proof in the ‘user experience’ pudding came from Dmitriy Zaporozhets (DZ), co-founder of GitLab, when he reached out to IT to ask how to get access to AWS for a sandbox experiment. IT knew about the project that was happening in the sales organization and he was referred to the getting started instructions on the handbook page that I created.

He shared a testimonial in December 2020: “I created an AWS account with the gitlabsandbox.cloud today. To be honest I did not expect it to be fully automated. I got my AWS credentials in 5 minutes without bothering anyone. That’s amazing!” With that, there was momentum behind the project.”

Shortly after Jeff and team automated AWS account provisioning, they iterated again and automated GCP project provisioning. This

significantly increased adoption by team members across the organization in engineering, customer support, DevOps/SRE, and the existing departments that they were already supporting.

By December 2022, the Support and Engineering Infrastructure team found HackyStack to be a better and easier-to-use solution than other homegrown and vendor tools that were available. The Director of Infrastructure announced that they would be adopting Sandbox Cloud for all ephemeral infrastructure moving forward. And, as an added benefit, the team has moved from a “total invoice” amount each month to having a breakdown of how much spending was performed by each user, and by each service. To get real-time costs by department, the user data is manipulated via pivot tables.

Upon further analysis, another one of the bigger efficiencies that the team needed to solve for was creating repeatable environments for ephemeral tests. To streamline this for all team members, Jeff and Dillon Wheeler (Senior IT Systems Engineer at GitLab) added Terraform GitOps environment provisioning to HackyStack that automated a lot of provisioning.

There are many tasks to set up infrastructure each time you want to start a new experiment. You have to create firewall rules and subnetworks, VMs or clusters, configure operating systems or images, etc.

Whether you are using Terraform and GitOps, or deploying infrastructure manually using the Web UI, there are a lot of security risks if not configured safely. It takes a long time to do so correctly so many engineers naturally take some shortcuts during initial setup. When you combine this with infrastructure-as-code syntax and manage this using streamlined CI/CD processes, there is a lot of trial and error to get it to work right and have it hardened with security

best practices. This is just the nature of using technology, however this can take several hours to several days depending on your experience and familiarity.

With standardized templates (for any system), users have an easy button and can get started quickly. They're able to focus on customizations specific to their needs rather than generic scaffolding, and can avoid having to recreate that scaffolding for every experiment. Some of GitLab's users have 20+ environments that they have created using their templates that allows them to avoid starting from scratch every time.

When this is used by Customer Success or Support engineers, this has a huge impact on time-to-resolution for answering customer questions or reproducing customer problems.

This addition saves 5+ hours for each environment created, and 20-30+ hours for first time users who are learning how it works. You can learn more about how Terraform environments work on the handbook page.

**Think about how users in your organization can benefit from templates and how IT can help the rest of the organization discover templates that already exist in different niches of your organization.**

You can learn more about how GitLab approaches iteration on the values handbook page and product principles handbook page.

## Embrace, Don't Eliminate Non-IT Projects

Looking ahead to his next projects, Jeff cautions organizations against underestimating the value of operators outside of IT that are building automation and improving business processes. Instead, he encourages teams to take a serious look at what can be leveraged to benefit more areas of the organization.

IT should be the central hub that binds people focused on operations in marketing, sales, finance, engineering and all other departments. The goal of IT should be to create an ecosystem of guardrails instead of a command-and-control structure that can result in unexpected and highly valuable innovations. In that way, IT can be crowdsourced.

As of July 2022, 491 users, 380+ Terraform environments, 311 AWS accounts and 275 GCP projects have been created and are managed by HackyStack. There are an estimated 750 team members that have job titles that are part of the “infrastructure community”, which means that they have 65% adoption of Sandbox Cloud.

To learn more about the GitLab Sandbox Cloud, GitLab's internally branded deployment of HackyStack and a summary of how HackyStack works with business and technical problems being solved, visit the public handbook page.

For HackyStack source code, visit the open source repository.

To see screenshots of how it works, check out the repository docs folder.



## About the Authors

**Jeff Martin** is a Senior IT Systems Engineer at GitLab. He moved from the Sales division to the IT department in June 2021 (after 10 years in a shadow IT capacity at several organizations). This transition happened after the second iteration to move from working on HackyStack on the side to working on it almost full time in an incubation engineering capacity.



**Dillon Wheeler**, another Senior IT Systems Engineer, joined the team and the duo now works on HackyStack almost full time to help manage most of the company's ephemeral/non-production infrastructure using HackyStack. It's an open source project that continues to be iterated upon with a focus on user experience and automation where it counts.

Connect with **Jeff** and **Dillon** on LinkedIn to learn more about what they're up to next.



## About Lumos

Employees shouldn't be admins for hundreds of web apps with excessive permissions. But then again, nobody wants to be stuck in IT ticket queues for days to get the right access. With Lumos, you make your company more productive and compliant with self-service app requests, access reviews, and license removals.

**[Click here to try Lumos out](#)**