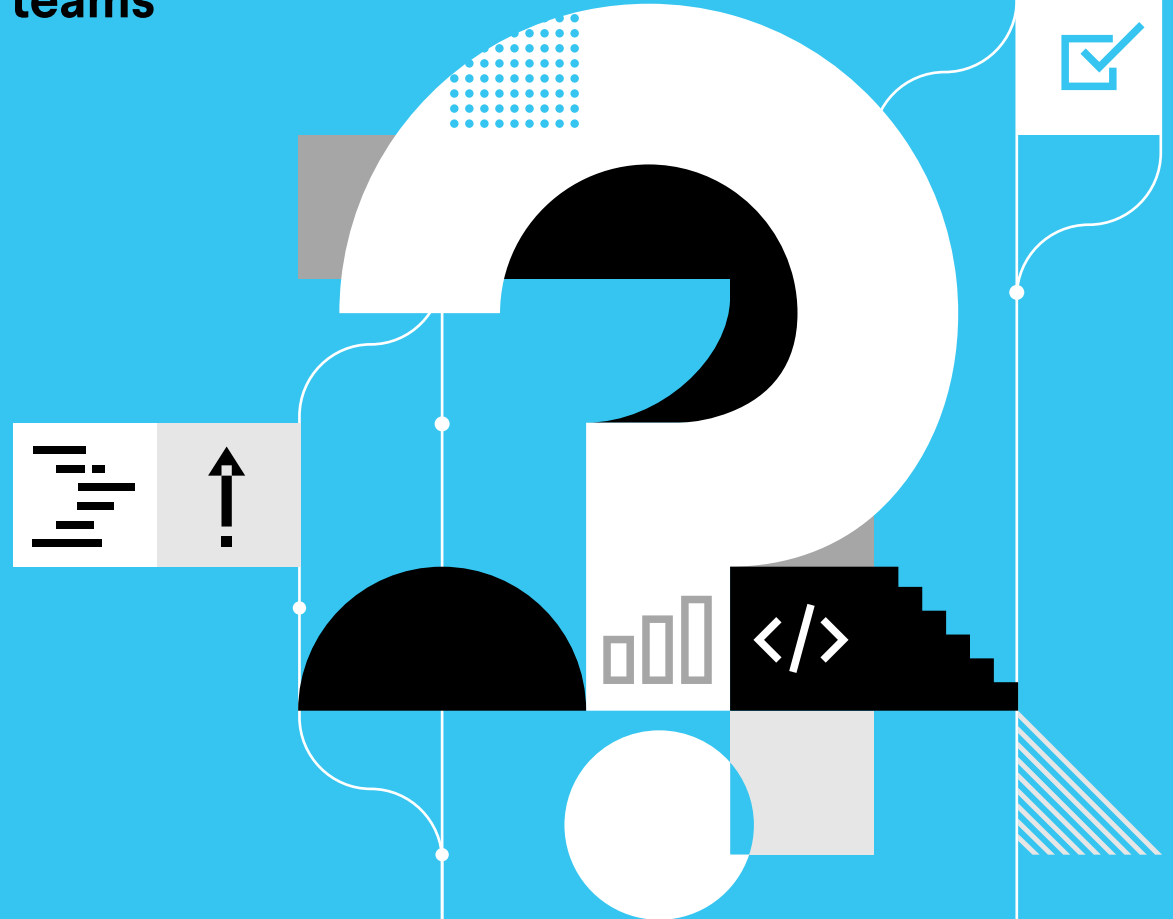# How Slack's own developers use Slack

**Best practice in our engineering teams**

**slack**

# How Slack uses Slack

**You may have heard Slack's origin story: It began life as an internal tool that helped a games development company collaborate better.**

Fast forward to today, and Slack is used by teams all over the world—and all over the enterprise, from marketing and sales to HR to customer support, not to mention engineering teams.

It's a collaboration hub that brings together people, data and applications. And it's highly adaptive, flexing to suit the structures and working styles of all kinds of teams.

To demonstrate the value Slack brings teams, it's often easier to show rather than tell.

As a company with engineering teams of our own looking to realize the full value of Slack in everything they do, we thought it made sense to turn the spotlight on ourselves.

**So here's the story of how Slack's developers use Slack every day.**

Every team is different, so what works for us won't necessarily work for everyone. But read on for some insights about our own engineering teams' use of Slack.
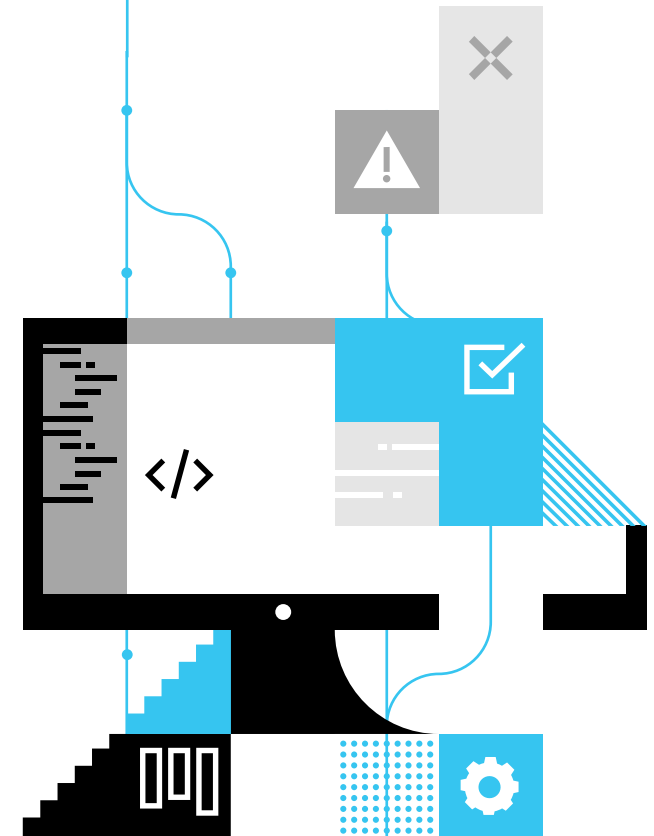
# Vertical and horizontal collaboration

**No two engineering teams are the same. Your setup will vary depending on your size, structure, industry, whether your engineering teams are building a core product, or supporting the wider work of the organization—or both. The list goes on.**

As a company with a product engineering focus, Slack has plenty of engineering teams.

Some organizations might have a traditional waterfall team structure, split according to development process chronology: architects, then developers, then testers, and so on.

At Slack we split our teams according to vertical pillars instead:

- **Platform**
  The team responsible for external dev-facing features like APIs

- **Core**
  The team that works on Slack as most users know it: the core product

- **Enterprise**
  The team that makes sure Slack scales for big businesses

- **Infrastructure**
  The team responsible for the back-end side of things

Each of these teams has its own goals and processes, but they all have to work together at different times, so collaboration has to be vertical and horizontal.

That's why we have cross-team channels that are **project or goal-specific**—for managing a feature launch or for end-of-quarter reporting, for example.

Then there are **function-specific** channels, like the one we use for front-end developers to share best practices across teams.

Let's take a deeper dive into one of these four pillars, the Platform team.

# How Slack uses Slack: the Platform team

Again, each team uses Slack differently (that's the beauty of it), but the Platform team will give you an idea of how we use our own hub.

Like all of the four main engineering teams at Slack, the Platform team has a **team channel** that serves as its home base.

This is where general team announcements happen. All team members are present, and the channel is open to employees from other teams who want to check in. Then each feature in development has its own set of subteam channels. For example, the Block Kit team has the following:

■ **#devel-block-kit**
Where day-to-day work is discussed and managed by engineers, project managers, designers, testers and others. You'd expect to see things like code merges, design updates and drafts of product specs.

■ **#feat-block-kit**
Content in this channel relates to discussion about the feature as a whole. Product, design and development agree on what is in or out of scope, nail down specific requirements, and make decisions as edge cases arise.

■ **#gtm-block-kit**
Channels with a "gtm" prefix are where the go-to-market strategy for a new product launch is discussed.

**"In the past you might have a load of email notifications about incidents, outages or code updates, but you can't really do anything with them. In Slack, you see the notification in context, and you can handle it right there."**

Mike Brevoort, director software engineering, Slack

We have triage and escalation channels for managing bug reporting, incidents, outages and more. The PagerDuty app posts incident alerts in an appropriate channel and lets you triage right within Slack.

Almost all our channels are open—even people from other teams with a general interest in what the Platform team is getting up to—but sometimes we do need private channels. We have a private **leadership channel** for management and for discussing how to cascade important comms to the rest of the team.

**PagerDuty app**

# How we use Slack apps

**When we say "Slack apps," we're talking about things like the PagerDuty app we mentioned in the previous section: an app that makes your critical software tools accessible and usable within your Slack workspace.**

Here are a few big engineering use cases where Slack apps help our teams work smarter.
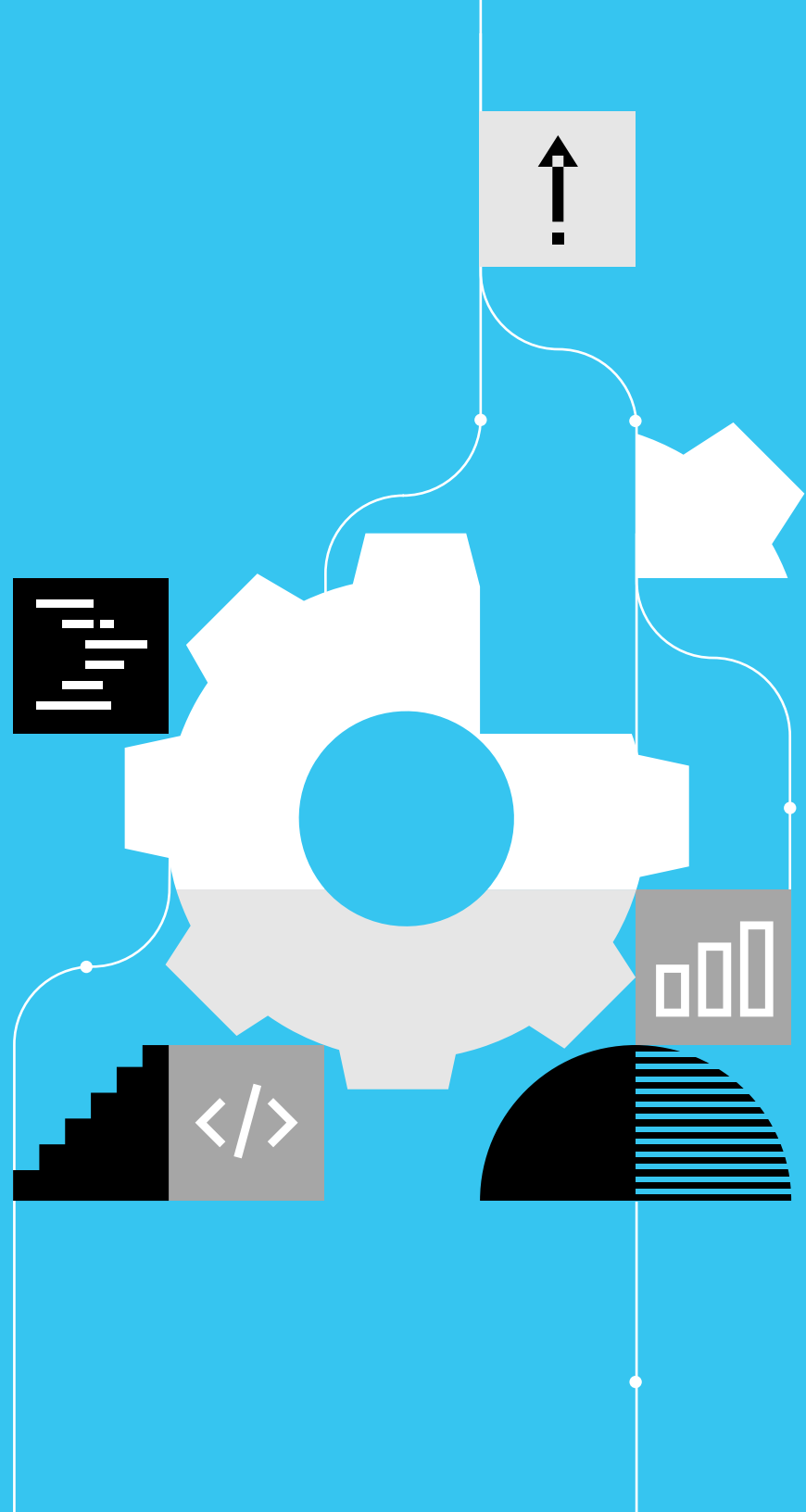
# Deployments

We use Slack apps that funnel push notifications from GitHub and other services. This automates and streamlines communication, leading ultimately to faster, safer code deployments.

**GitHub app**

# Bug tracking

**We can preview and address JIRA tickets in Slack, without having to find and open the URL.**

Form details are pre-filled, so tickets are automatically put into the right place for triage. And we use quick visual cues to speed up our work. For example, we use ⚪, 🔵 and 🔴 emojis to prioritize tickets, and a custom app organizes them based on those priorities.

When someone reviews an item, they use a 👀 reactji. Then a ✅ to show that it's resolved.

**Jira tickets**

# Customer support

Every Zendesk ticket that's created is funneled into a relevant Slack channel so it can be easily dealt with in collaboration with teammates.

"We use a lot of apps (both custom and publicly available) in our Slack workspace, but even built-in Slack functionality like reminders are a great way to rally the team and discuss what's happening today. That simple function has saved us from interrupting individuals with a daily meeting."

Brian Ngo, senior engineering manager, Slack

**Zendesk app**

# Graphs for quick insights

**Instead of logging into other software, you can pull handy graphs right into Slack, spot problems, and act quickly to correct them. For example, a graph might monitor your API's rate of 500 errors and give you an at-a-glance view of a recent spike that could signify a server issue.**

To learn more about Slack apps for DevOps—and for other use cases, like productivity, HR, customer service, security, sales and marketing—**check out this e-book about integrating your tools with Slack**.

**Some Slack apps we use for DevOps**
GitHub
BitBucket
SubVersion
Visual Studio
JIRA
Jenkins
Trello
Pivotal Tracker
Asana
PagerDuty

**Arc Report**
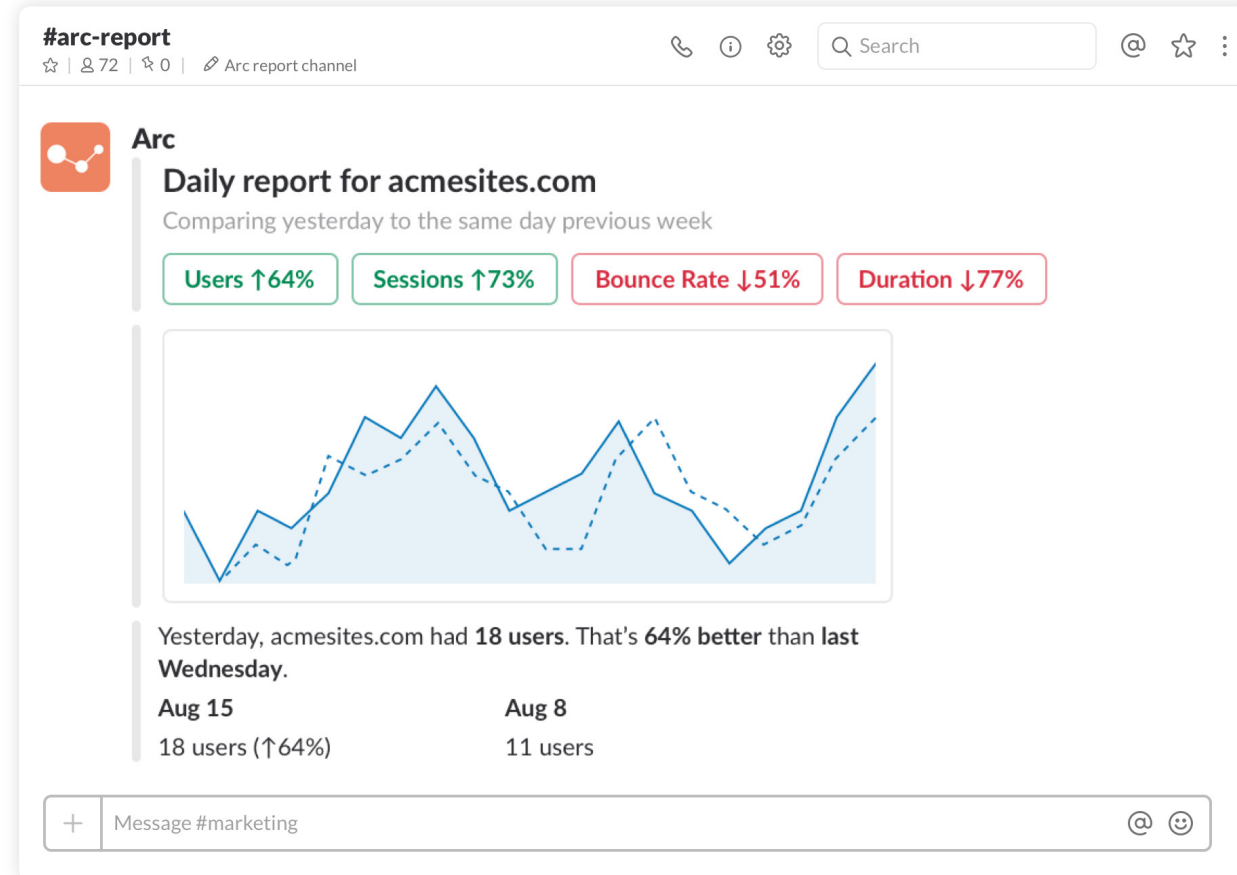
# Over to you

**So there you have it—a snapshot of Slack's own use of Slack.**

But it's just one example of many—even within our engineering teams, there are lots of different ways we use our own platform to work smarter.

Learn more about building bots, apps, tools and workflows for Slack **here**, or **contact sales** for a chat about how your engineering teams can make the most of Slack.
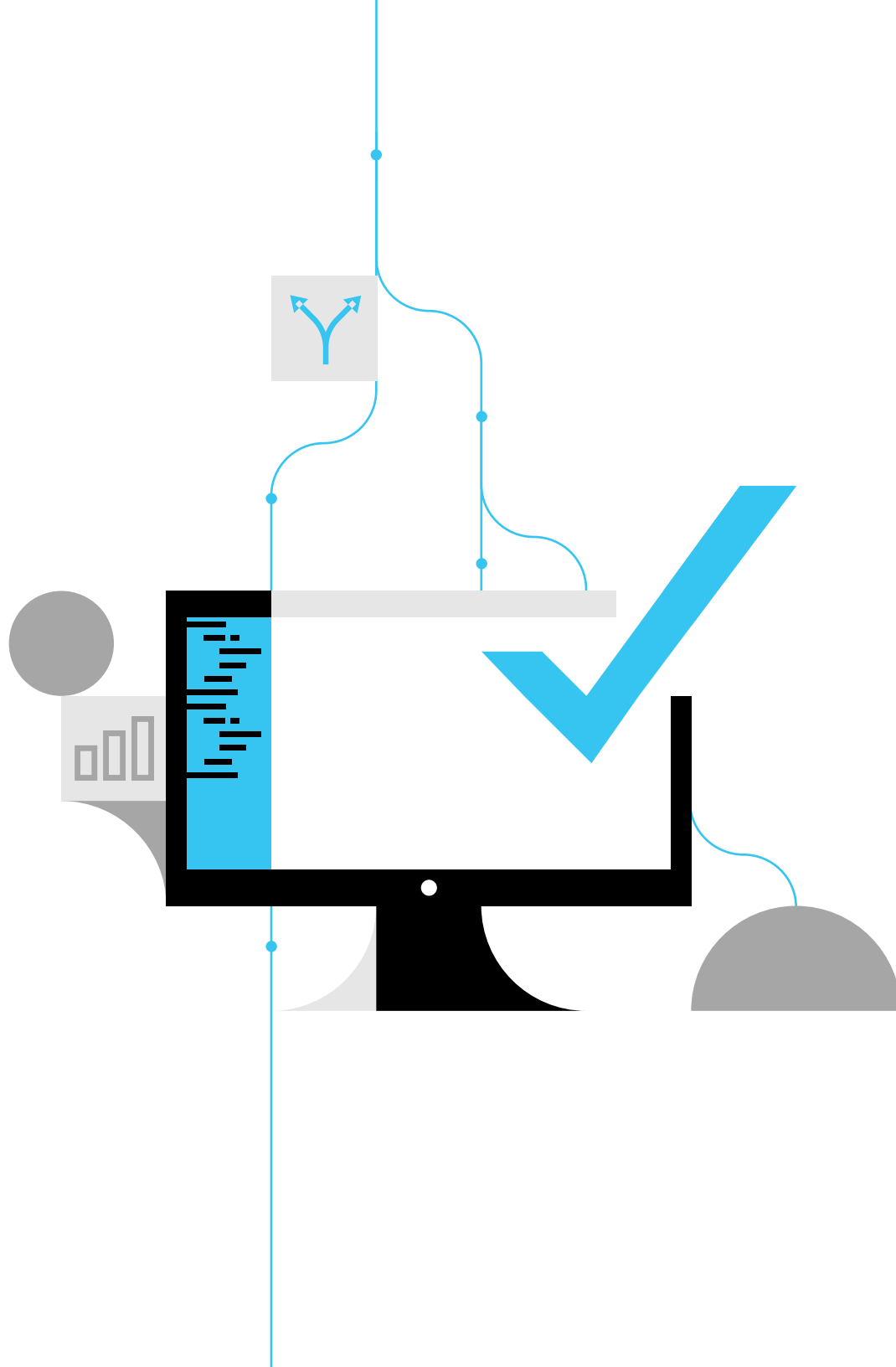
# Learn more

**8 questions every IT leader should ask**
Don't choose a collaboration hub before you've got answers to these questions

**The Slack App Directory**
Explore your integration possibilities

We should probably talk.
**Schedule time with our team.**

LET'S GO

# About Slack

Slack is a layer of the business technology stack that brings together people, data and applications—a single place where people can effectively work together, find important information, and access hundreds of thousands of critical applications and services to do their best work.

From global Fortune 100 companies to corner markets, businesses and teams of all kinds use Slack to bring the right people together with all the right information.

slack