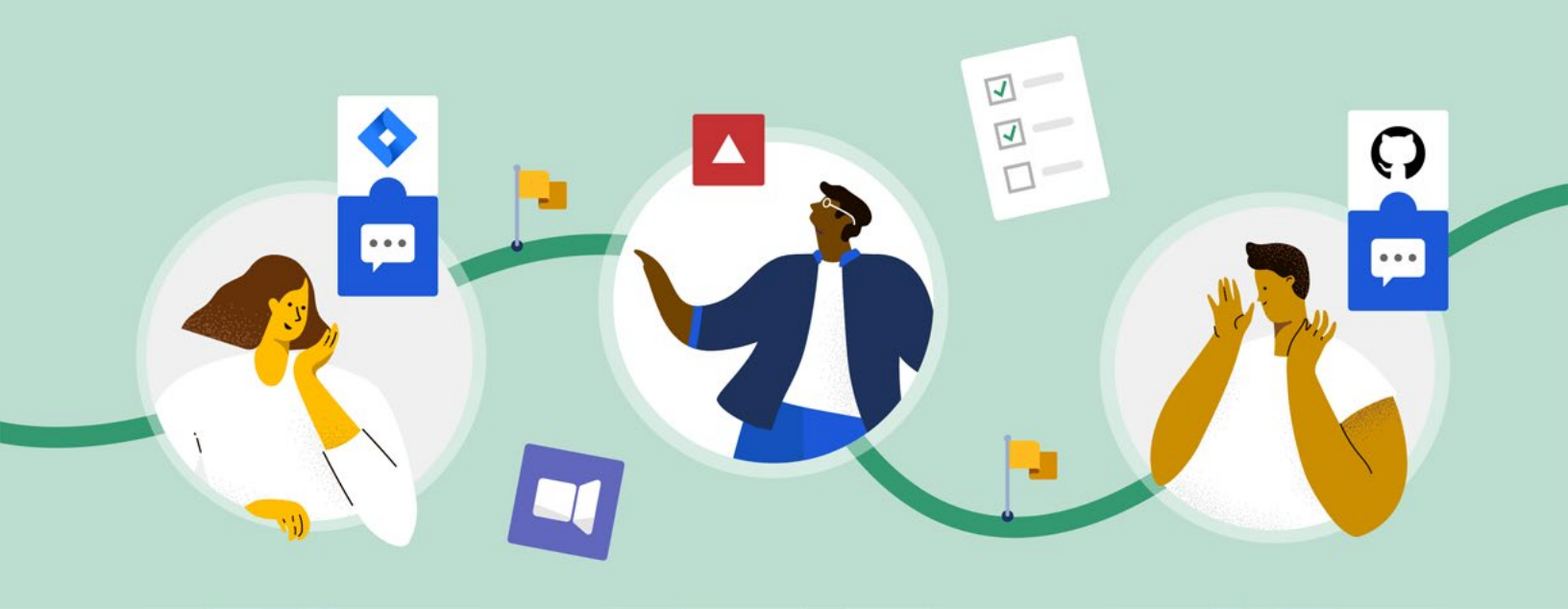




Steps to ChatOps for Enterprise Teams

Everything you need to know about setting up ChatOps for your organization.



Introduction to ChatOps

From staying aligned on the next big product milestone and coordinating logistics around your latest launch to responding to a major incident, the success of any software development team hinges on great communication and collaboration.

But developers often work across a wide range of tools, and constant context switching can create an unfocused work environment that increases the risk of critical communications getting lost in the shuffle. What's more, the rise of remote and hybrid teams have made strong digital collaboration tools and processes even more essential to organizational success.

ChatOps is a collaborative, conversation-centric way of working that brings people, discussions, bots, tools, and files together in one central location. It's more of a methodology than a

specific technology — a collective approach to running DevOps workflows and building a collaborative team culture. Chat and messaging applications are now at the center of collaboration workflows for many teams; ChatOps makes your chat tool work harder for your team.

If you're looking to explore the potential of ChatOps for your organization, then you're in the right place. In this guide, we'll walk through the progressive steps towards developing a mature ChatOps practice, no matter how your team, tools, and processes are structured.

It can be hard to know what ChatOps looks like without real-world examples. We'll show you how teams use ChatOps every day at each step. No matter how simple or sophisticated your ChatOps practice, your team will work better together, and your organization will benefit from increased efficiency and productivity.

Let's get started!

Benefits of ChatOps

- Gathers all the required people in one place, whether on-site or remote
- Provides a shared console or command line for easy collaboration
- Shortens the feedback loop and enables faster response times
- Reduces the mean time to restore service after system failure
- Captures the history of team activity, system events, problems, and solutions

STEP

1

Start the Conversation

You can't have ChatOps without chat! If you're already using a messaging tool to collaborate, then you're already on your way to ChatOps. If not, then making chat a central part of your team's collaboration culture is critical. Invite your people to join a team channel and get the conversation started. Encourage them to post screenshots or code snippets. Think of it like a cooperative board game — just getting everyone gathered in a channel, discussing timely topics, and solving problems together is half the battle. Soon, your entire team will be making progress together daily as collaboration becomes easier and more fluid.

Developing a ChatOps practice also augments your team's collective expertise. Instead of relying on communication siloes like email threads or meetings, everyone brings their own domain knowledge into a team channel, making it easy to share and learn from each other. It also gives team members a shared knowledge repository of conversations, files, and everything else they'll need in the future.



Pro Tip

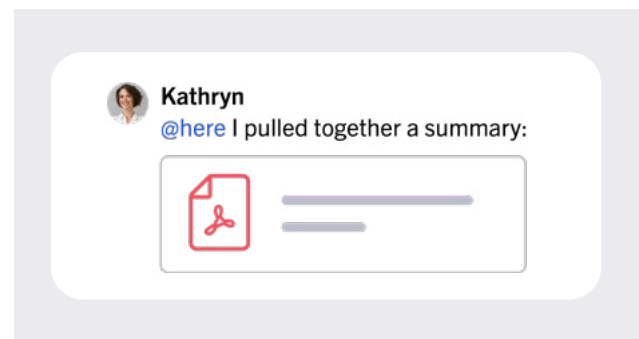
The more, the merrier! Don't be afraid to over-add people to your channel in the beginning. It's better to have more eyeballs in the channel and let people opt out if the conversation isn't relevant to them.



CHAT IN ACTION

Manual File Sharing

File sharing is an early “win” for ChatOps implementations. Files often get lost in endless email threads and shared folders. Bringing file sharing into your team chat ensures that files



are easily accessible to the whole team, whether you need to loop in a new hire or dig up an old PDF a few months after it was first shared.

CASE STUDY



How the U.S. Air Force Uses Chat to Share Mission-critical Documents

ChatOps has helped streamline pre-flight preparations for the USAF Air Mobility Command. Flight and mission paperwork must pass through four or five different offices for approval before every flight, which can be a slow and time-consuming process. In the past, aircrews received this information only about 30 minutes before flights, giving them limited time to review everything.

The AMC has begun using Mattermost Channels to share files securely via chat. The flight manager or command can share flight plans and mission packages directly to aircrew on their electronic flight bag approximately two hours before every flight, giving flight crews additional time before the mission to study and prepare. As a result of this streamlined process, they've been able to reduce flight-related fuel consumption and costs.

“For us, using Mattermost offers a huge benefit in mission velocity and safety. We found that also gave us a lot more time to fix cargo and fuel problems before the crew even shows up at the airplane,” says Major Justin Poole, Phoenix Spark Innovation Cell Director at Travis Air Force Base.

[Learn more about ChatOps at USAF »](#)

STEP 2

Automate Simple, Repeated Processes



Once your team is comfortable with centralizing collaboration in the team channel, it's time to branch out. Channel-based communication is key to [many DevOps workflows](#), including incident response and code deployment. Consider creating separate channels for specific topics, such as security, performance, built systems, war rooms, or whatever makes sense for your organization. Your goal is to lay the foundation for more powerful automation as your ChatOps practice matures.

As you get started with automation, remember that you don't need to start with super powerful bots to solve every problem. Whatever chat platform you're using, there are a range of ready-to-run plugins that can help you start building your ChatOps practice one step at a time. You can use them to run daily standups, kick off retrospectives, return data from monitoring systems, and handle just about anything else you can imagine. By automating a few repetitive tasks, you'll get your first taste of the power and potential of ChatOps. Once you get started, you'll find that the possibilities are (almost) endless.



Pro Tip

ChatOps is all about helping your team work more efficiently. Try to identify what's creating the most drag on your team's time, and ask yourself, "What if we didn't have to do this every day?"

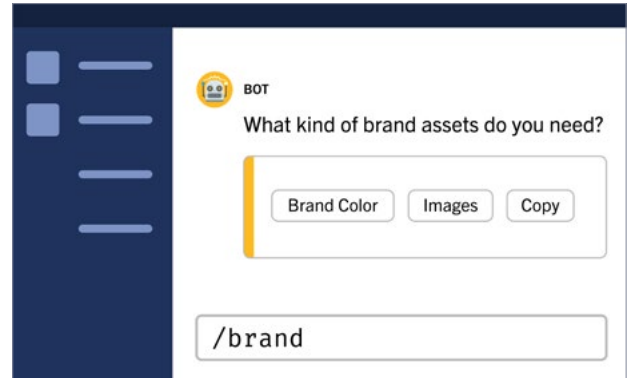


CHAT IN ACTION

Triggering Common Workflows with Slash Commands

Slash commands are a great way to build in automation quickly. Slash commands can trigger outgoing webhooks, allowing you to integrate external applications into your chat platform easily. Chat clients often have built-in slash commands that are ready to use for com-

mon workflows and platform shortcuts, from starting a video call or inviting someone to join a channel to spinning up a new environment for testing or deploying a build. Building custom slash commands that integrate with your other systems allows you to trigger actions on those external systems.



CASE STUDY



How n8n Automates Repeated Release Tasks with ChatOps

The team at n8n makes a new release almost every week, and all that work can add up fast. Fortunately, they're also workflow automation pros – and they've found lots of ways to automate common release tasks to keep the team moving quickly.

For example, they've created a custom slash command `/graphic`, which triggers a workflow that automatically generates a unique graphic to promote the release on their blog, social media, and more. And if the team needs more information for a release promotion, brand assets like their brand color, logo, and marketing copy are just a `/brand` slash command away.

[Learn more about how n8n uses ChatOps »](#)

STEP 3

Automate More Complex Workflows



By now, you've probably started identifying other opportunities to [automate your DevOps workflows](#). If you haven't already begun connecting other tools in your team's workflow into chat, now is the time to get started. Integrating essential tools lets your team initiate workflows and check information without switching between windows, minimizing context switching and helping them stay focused.

There are various ways to integrate tools into chat, from [plugins and webhooks](#) to [API calls](#). The most popular tools usually have pre-built integrations ready to start using right away. Open source tools offer even more flexibility and extensibility to build a custom solution that fits your team's needs.



Pro Tip

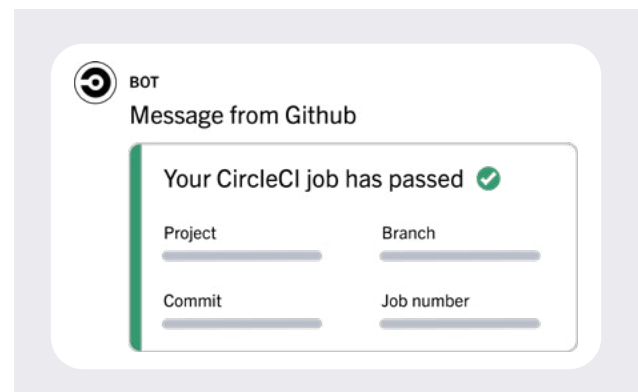
There's no one-size-fits-all way to approach ChatOps. The key is finding the right points of connection and integration for your organization. Keep an eye on how your organization responds to incidents, requests, and other events. What happens to response times when alerts are delivered to a topical channel? How do conversations change when everyone has the right information at their fingertips?



CHAT IN ACTION

Using ChatOps to Streamline Builds

Streamlining and automating your build processes is a great way to take your ChatOps practice to the next level. Connecting your CI



builds to your chat server allows you to monitor pipelines, trigger workflows, receive notifications about your builds, and more – all without leaving your chat window.

CASE STUDY



How Crossover Health Uses ChatOps to Automate Internal Support

IT teams often spend a lot of time fielding the same kinds of requests over and over – and even small requests end up eating up a lot of time in the long run. By taking a ChatOps approach, Crossover Health has dramatically sped up turn-around time for support tickets. The Crossover Health DevOps, technology, and support teams use Mattermost daily, making it the perfect forum for providing immediate support and issue resolution to all 1,250 users in the company.

“We use webhooks to automate support in Mattermost,” says Daniel Gover, IT System Admin for Crossover. “If someone has an issue, we’re able to move faster with less risk of something getting lost in the shuffle.”

Previously, Crossover Health primarily used a standalone ticketing system for support. The response time for tickets could take hours. Now that they’ve integrated support into Mattermost, many issues are reported, handled, tracked, and resolved in a matter of minutes.

[Learn more about how Crossover Health uses ChatOps »](#)

STEP 4

Start Automating Command and Control



Do your DevOps or SRE teams have access to reports that your engineers do not? Not everyone may be on the same email threads or connected to the same systems. Once you've started automating some processes via ChatOps, it's time to start centralizing notifications and collaboration in one place by pulling important data in from external sources.

Take stock of where in your stack information is stored – whether that's outage reports in PagerDuty, bug tickets in Jira, or somewhere else. Make sure those tools are connected to chat so your team can spend less time hunting down the answers to their questions and more time solving problems. Using webhooks to route information from external sources to the right channels can help. [Building interactive Playbooks](#) that increase visibility into the status of workflows while they're in flight can keep stakeholders looped in at every stage.

The key is that automation keeps people from making mistakes when running robust and potent queries that may otherwise harm your infrastructure. In addition, it allows non-developers to run commands themselves. By automating command and control, you empower more people to be self-sufficient.



Pro Tip

Focus on creating more signal and filtering out noise. Chat can feel like a firehose of information, especially during an incident. Bringing everything together into your chat window makes it easier for your team to stay focused on resolving an issue. Creating role-based views and [permissions schemes](#) that keep the right people in the loop when it matters most helps distill that flood of information into what's most important.



CHAT IN ACTION

Bringing Data Sources Together for Increased Visibility and Autonomy

DevOps teams regularly get hit with common requests, like provisioning environments or running reports. It takes time to fulfill those requests — time better spent on more important tasks. The next step in your ChatOps journey is to enable team members to fulfill those requests themselves. Using slash commands, combined with outgoing webhooks, you can set up self-provisioning through your messaging interface and allow anyone in your organization to query your system and retrieve the data they need.

What's more, self-provisioning can be done in a public channel. This allows others to see what queries are being run and what data is being returned, driving broader information sharing and greater transparency within your organization. It also allows others to offer support if needed.

CASE STUDY



Using ChatOps for COVID-19 Contact Tracing

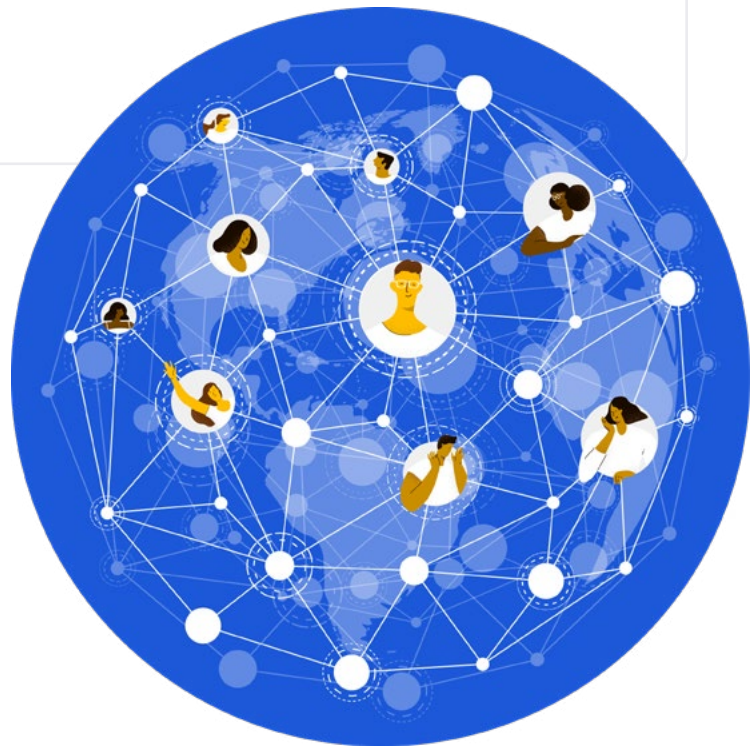
Belgian non-profit VCLB Ghent was tasked with contact tracing for schools in Ghent, Belgium, during the early months of the COVID-19 pandemic. As case numbers started rising, the spreadsheet they used to track and manage tracing became too difficult to use. To scale contract tracing, they needed a way to get essential information into the hands of tracers as quickly as possible.

The team connected a webform for new infection reports to a Mattermost Channel with an incoming webhook and used custom emojis to track the current status of the case. Tracers were able to see who was working on other cases, and identify schools where multiple infections were cropping up quickly and easily.

CASE STUDY continued

“Mattermost helps our team to easily get a view on the status of each case and who worked on it,” says Tom de Moor, IT Manager for VCLB Ghent. “It also improves the communication and knowledge-sharing between the contact tracers, helping us combat the virus. The contract tracers have set up a channel to chat with each other, and they use the primary channel we created for them just for managing the cases.”

[Learn more about this story »](#)



STEP 5

Bring in the Bots!

Most people have a hard time remembering more than a few slash commands and all their parameters. Why not move that heavy lifting over to a bot? Running a bot gives users a friendly interface and makes it easier to bring automation to a wider user base. Bots allow people to bypass slash commands and use natural language instead. They also allow integrations or plugins to privately message individuals instead of posting to public channels, which is useful for sending notifications relevant only to individual team members.

Bots can be dynamic and engaging, too. Interactive messages allow your team to take actions or initiate workflows directly from chat — from allowing them to minimize context switching and stay focused.

You can use bots to connect to a range of DevOps systems and bring information into relevant channels. There are many [great existing bot interfaces](#) to get you started. Connect to Jenkins to retrieve server logs and results for various test servers in your QA channel, or connect to Grafana to display data visualizations in your analytics channel. Ultimately, the goal of introducing bots is to uplevel the ChatOps experience by making workflows as easy and efficient as possible.



Pro Tip

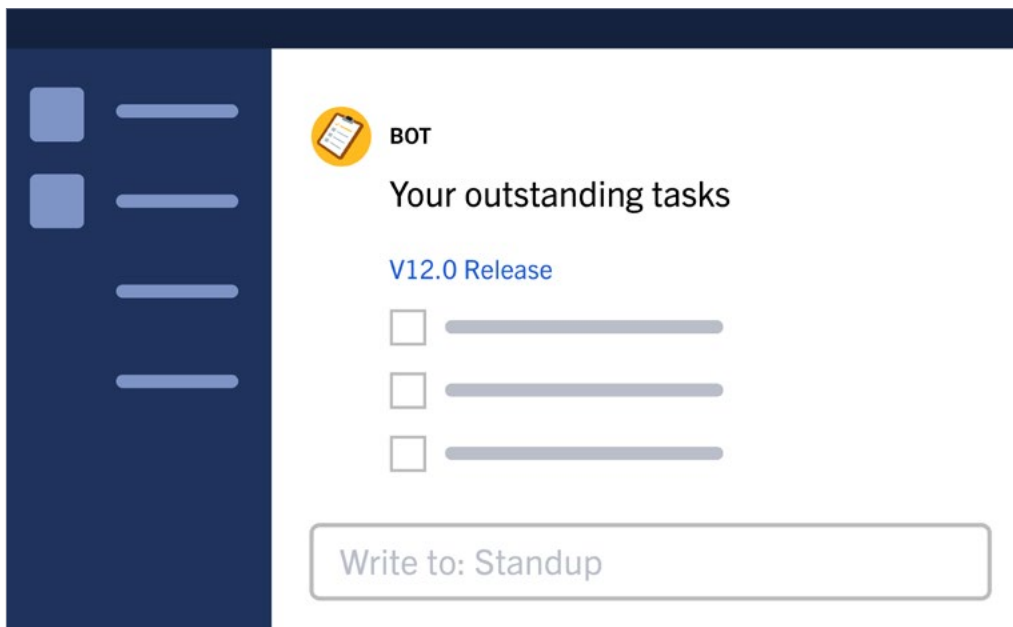
Your bot humanizes ChatOps for your organization, so give it some character. Make it fun! Your bot can be angry, ecstatic, goofy, nerdy — whatever you think will engage your users most and drive ongoing use. You can even build a story around your bot and integrate it into your team culture. “The angry bot did it, not me!” Blame the bot, not people.



CHAT IN ACTION

Run Your Team Standups with ChatOps

Every team has likely felt the administrative pain of gathering and disseminating updates for team standups. Using a bot to help manage your daily standups from your chat platform is a great way to experience how ChatOps can remove friction from your team's daily workflows by making it easier for them to share daily updates right in your team channel.



STEP 6

Build Smarter Bots and Workflows



At this point, your ChatOps practice has become pretty sophisticated — but there's much more room for growth. So far, your ChatOps has largely lived in your chat tool; now, it's time to extend it to build more complex, customized workflows that cross and connect the tools in your workflow.

Plugins go beyond slash commands and enable more deeply integrated functionality between systems. A plugin-powered bot can drive common DevOps tasks in a shared workspace where the team is most active, helping everyone minimize context switching and stay productive. Bots can be smart (and customizable) enough to alert specific users via private channels or even direct messages when appropriate, such as alerting a user that tests failed for their feature branch.



Pro Tip

Make your bots your own! Bots are better when they're customized to fit your workflow. Fortunately, there are many open source projects, libraries, and frameworks to get you started so you can design your perfect bot.



CHAT IN ACTION

Capturing Workflows with Playbooks

Every team has processes and workflows that can feel a little unruly. Knowing what gets handed off to whom (and when), and what worked (or didn't work) last time can be a lot of information to remember, especially for processes that aren't frequently run.

ChatOps can be a powerful mechanism for guiding your team through repetitive tasks and processes. Building shared playbooks that can be run inside your chat platform ensures

Playbook Details

Participants 

Checklists

-
-

-
-

that every stakeholder knows what's going on at every stage. And because process documentation happens in the same place the team is running those processes, this approach also helps reduce friction to iterating them. The end result? Better visibility, greater consensus, and everything you need to continuously improve your workflows.

CASE STUDY



How LAIKA Uses a Chatbot to Keep a Global Team on the Same Page

The LAIKA team had built a chatbot for IT usage on Jabber and migrated it over to Mattermost using the platform API and web sockets. The bot listens for commands and helps users accomplish a range of tasks faster. It also collects messages in the IT alerts channel and delivers them to a repository channel, which provides a scrolling timeline of things that have happened throughout the day. A Wolfram Alpha extension enables the bot to parse user questions through the knowledge engine and try to find the best answer. Users can even ask the bot what to have for lunch.

[Learn more about how LAIKA uses ChatOps »](#)

STEP 7

Give Your Bots More Autonomy



Your ChatOps practice is humming along nicely with rich integrations, smart bots, and a vibrant messaging-based team culture. Why not find ways to let bots handle more automation *automatically* to reduce the need for manual intervention? Bots have great potential to improve your workflows, keep your team focused on what matters, and capture a fuller picture of what's happening.

Giving your bots more autonomy, often called robotic process automation (RPA), adds that final layer of agility that will help your organization realize its full potential, freeing your team up to focus on creative work, complex problem solving, and other tasks best done by humans. Empowering your bots to respond to outages or degraded service can help reduce time-to-fix, while notifications in chat keep your team informed about changes in real time.



Pro Tip

Look for use cases in your organization where one or more bots can step in and help humans do their job better. It may be watching a message stream, performing rapid actions, or handling massive amounts of data.



CHAT IN ACTION

Automating War Rooms

When exactly *do* you want bots taking over automation? One common use case is a war room. During an outage, a war room has so many messages flowing in (thanks to all those systems integrations you've set up!) that it might be impossible for human team members to keep up with them. Instead, bots can watch the war room, pull out key messages, and duplicate them in channels where people can easily interact with them. The bots in those channels push messages from team members back to bots in the war room channel, almost like a message bus. The key is that humans and bots are interacting in natural human language.

Get Started with ChatOps Today

Whether you're just getting started with ChatOps or looking to take the next step, we hope you found something that sparked inspiration for you in this guide. Have more questions, or want to chat with other teams who are on their own ChatOps journeys? Check out the [Mattermost Community Server](#) to see ChatOps in action and connect with other folks.

▼ Why Mattermost is Great for ChatOps

✓ **Designed for R&D Workflows**

Mattermost is built for developers teams with features that help you collaborate when coding, sprint planning, deploying, and during outages and support escalations.

✓ **Open Source and Fully Customizable**

Whatever tools are in your dev stack, Mattermost is designed to fit your workflow, not the other way around. Build your own integrations if they don't exist already.

✓ **Baked-in Project Management and Playbooks**

Mattermost brings chat, Kanban-style project management, and customizable playbooks into the same platform so you can plan, organize, and run software development projects with minimal context switching.

About Mattermost

Hundreds of thousands of developers around the globe trust Mattermost to increase their productivity by bringing together team communication, task and project management, and workflow orchestration into a unified platform for agile software development.

Founded in 2016, Mattermost's open source platform powers over 800,000 workspaces worldwide with the support of over 4,000 contributors from across the developer community. The company serves over 800 customers, including Samsung, Nasdaq, SAP, European Parliament, and the United States Air Force, and is backed by world-class investors, including Redpoint, YC Continuity, Battery Ventures, and S28 Capital.

To learn more, visit www.mattermost.com.



Mattermost

[Mattermost.com](https://mattermost.com)

