

CHATOPS EXPLAINED: HOW CHATOPS SUPPORTS COLLABORATION



Conway's Law: Any organization that designs a system (defined broadly) will produce a design whose structure is a copy of the organization's communication structure.—Melvin E. Conway

The goal of your company is to continue to advance—yet many organizations cite situational awareness as a primary concern. If success means finding 100 ways a light bulb cannot be created, then you don't want every team working on the same number 12.

Information silos exist between [development teams and operations, operations support, and personnel](#). These silos occur when one team in an organization has lots of information on which to act, while others, who are making similar decisions, do not have access to that information.

Progress will come sooner when the teams communicate what they have done, and what their results were. This is where ChatOps comes in. Let's explore:

- [ChatOps](#)
- [How it works](#)
- [Architecture](#)
- [Key Features](#)
- [ChatOps tools](#)
- [Chatbots](#)
- [Benefits](#)
- [Cultural change](#)

What is ChatOps?

Sharing is at the heart of ChatOps. ChatOps is a method of accessing and distributing in ways that are immediate and easy—no more chain-of-command information processing.

Information silos can occur because some teams outperform others. Some teams may happen to work with a client with a very unique set of problems, so they work through and create a unique solution. The silo happens because information and experiences are not shared among teams.

Despite the abundance of [collaboration platforms and project management tools](#), manual efforts to improve cross-departmental workflows are not enough to decrease the size of the information silos. This results in manual information processes that can:

- Slow down work
- Provide the wrong answers or context
- Negatively affect decision making
- Reduce overall productivity
- Waste money and time

If one research team is conducting trials and learns a test does not work, it benefits the entire company to share that information so other teams are not making the same mistakes or wasting money conducting the same trials.



How ChatOps works

In [software development](#) and [IT operations](#), ChatOps helps devs, ops, and support staff collaborate effectively. ChatOps uses tools and automation that promote easy communication within context, which is essential for successful collaboration.

When information is stored in chats, users don't need to communicate the status or context of the operational tasks with their team—the information is already visible in the chatroom. Every member can view the shared context of the situation, making decisions and taking actions accordingly.

The communication platform logs the progress as teammates work, and that progress is visible in real-time. It eliminates the silos and barriers between team members and does the same for cross-functional departments who work on the same projects.

ChatOps architecture

ChatOps is a transparent collaboration model that connects the [software development tasks](#) of:

- Communication
- Execution
- Operations

Think of ChatOps as a system that integrates the personnel, the existing work and technologies, processes, and communications into a unified conversational environment. The ChatOps communication structure allows users to execute actions via internal robots on tools that are integrated within the communication platform.

(If we look at the ChatOps architecture, it's loosely similar to the structure of [cluster orchestration](#) that [Kubernetes](#) is built on.)

Key features of ChatOps

Numerous features make ChatOps an invaluable tool for software developers. For example, users value its:

- **Real-time collaboration:** Teams can easily communicate within their group and with other teams through chat tools integrated with operational workflows. People can converse, make decisions, and take action in real time.
- **Automation of tasks:** Repetitive software development tasks, like deploying code, monitoring operations, and sending out patches and issue resolutions, are automated for speed and reduced errors.
- **Command execution from chat:** People can use the chat interface to directly run commands, enjoying immediate access to scripts and updates on status, along with system actions within the chat context.
- **Integration with DevOps tools:** ChatOps creates a centralized operations hub by integrating with Jenkins, GitHub, Nagios, and other IT management and DevOps tools.
- **Incident management and alerts:** Real-time incident detection gives teams an immediate alert when issues arise so they can effectively collaborate around a fast response.
- **Transparency and visibility:** Teams have a clear view of operations so they can document

conversation logs, decisions, and actions taken. Tracking progress and following up on tasks is easier.

- **Collaboration beyond IT:** ChatOps supports cross-functional operations to improve customer support, marketing, and other organizational functions.

Popular ChatOps tools

Common ChatOps tools include:

- Wiki pages
- Chat platforms, like Slack and Keybase
- Chatbots
- Automated notifications

Wiki pages are a great way for teams to contribute what they have learned to a single source. They have group knowledge, shared knowledge, and searchable knowledge. Each of these increases communication and transparency among teams. When a team member hypothesizes a new way for the light bulb not to work, they can research and find someone who has already discovered the method.

(ChatOps also supports and benefits from your [knowledge management processes](#).)

Chatbots in ChatOps

[Chatbots](#) are configurable through custom scripts and plugins to automate tasks. These chatbots can:

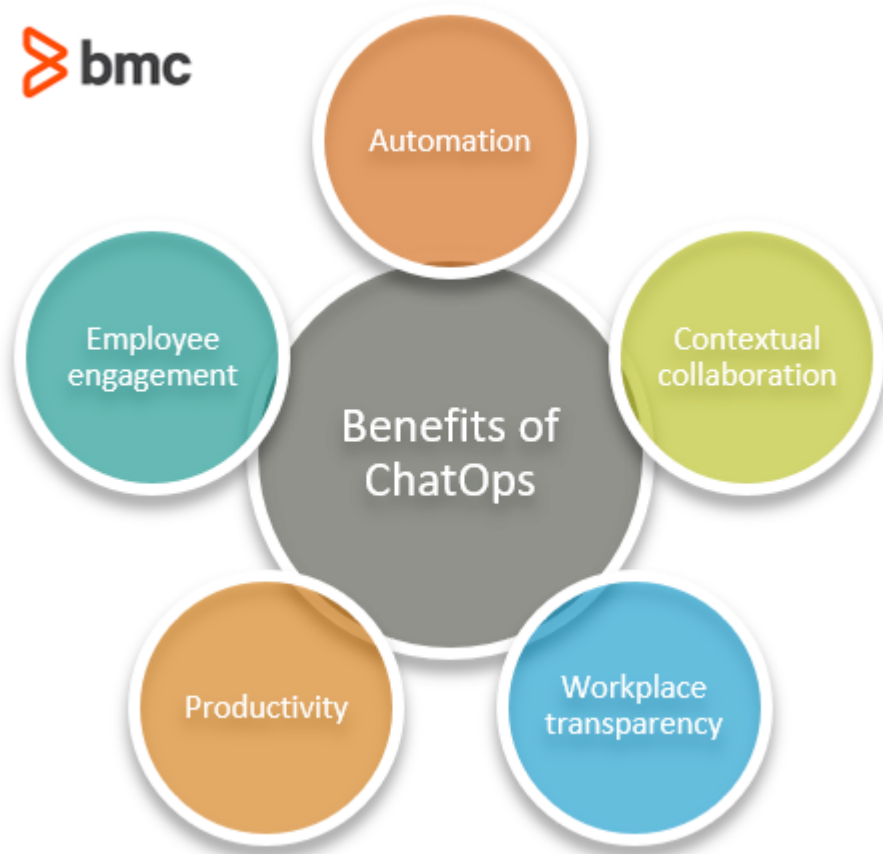
- Send messages when a process has completed
- Request information from other people
- Keep a channel of communication alive by highlighting when no one has chatted in the room some time

The collaboration and communication that takes place are the response to the operational tasks that are tracked and visible within the communication platform.

Hint: Create automated chat messages when new posts are made to certain high-value wiki articles.

Benefits of ChatOps

There are many reasons to implement ChatOps. Here's the five best reasons.



1. Automation

Set up conversations to automatically trigger actions. Instead of forcing every team member to maintain a repository of manual scripts, store and automate the code from a well-documented and centralized communication platform.

Configure the chatbot to:

- Pick up commands in real-time
- Execute actions
- Update the console to keep every team member aligned

Replace the error-prone practice of manual code execution and progress tracking with effective automation capabilities of intelligent chatbots. This saves users the time and hassle of executing repetitive manual operations tasks—an important step towards truly collaborative, effective teamwork.

2. Contextual collaboration

Silos are created when team members fail to communicate information in the right context. Due to the proliferation of tooling and complex processes involved in software development and IT operations, extracting and presenting accurate context with every conversation is a near-impossible task.

Stop relying on multiple channels to receive contextual notifications. Stop manually connecting the dots to make sense of the available information. Let ChatOps introduce information into an always up-to-date environment. This allows users to communicate constructive, concrete, and information-

driven feedback and actions between other users—without relying on assumptions.

3. Workplace transparency

Honest and transparent communication is critical in [incident management situations](#). In tense situations, organizations struggle to:

- Prevent the blame-game among employees
- Encourage them to collaborate effectively

In situations when team members are open and honest in their communications, even the correct and straightforward statements—“Well, it works on my machine”—do no favor to progress.

ChatOps allows the conversations to align with the reality of the project situations. If a build or feature is running on one system, not on others, the default reaction of blaming other users is replaced by collaborative efforts to identify the issues that prevent consistent results across all users and machines. This practice is easier when the associate information is accessible via a centralized and common communication platform that all team members can see and trust.

4. Productivity

ChatOps supports collaboration between distributed teams by making contextual information available to all users in real-time. Without a common platform to visualize and discuss project progress, users with limited access to siloed tools and technology processes may not be able to communicate the necessary information.

Since ChatOps operates as an automated environment to execute commands and trigger actions, users no longer have to rely on time-consuming manual scripts to perform the same actions. The time savings translate to convenience and resources for more meaningful job tasks.

5. Employee engagement

When all users are on the same page, it is easier for everyone to contribute to project discussions:

- Information-driven collaboration removes the bottleneck and delays in access to the right information to perform the appropriate actions.
- Users gain confidence as the build-measure-learn-iterate process keeps everyone informed regarding potential issues during the development lifecycle or project progress.
- Since users are relying on intelligent bots to share accurate information, the psychological barrier in requesting and reminding other team members for appropriate information/action is removed organically.

Ultimately, ChatOps enables team members to communicate on more important matters, such as devising strategies and making collective decisions—instead of requesting data and communicating incomplete information.

From a [security and compliance](#) perspective, ChatOps offers the added advantage of documenting IT Ops tasks and establishing a communication mechanism for [proactive issue resolution](#).

ChatOps is a cultural shift

Adopting ChatOps requires external tools to automate operational tasks—but that isn't nearly enough. Successful ChatOps requires cultural changes. Users across departments, functions, and teams must embrace the mechanism that involves real-time collaboration and progress on mutual project responsibilities.

Here's what to expect in the ChatOps culture shift:

- **A trustworthy structure.** Trust in the communication technology is required if it is to be accepted as a gateway to operational features that are previously only open to individual users in siloed console environments.
- **Additional investments in technologies.** You may need extra tooling or solutions to connect and integrate the distributed siloed tools in secure collaboration environments.
- **Humor as diffuser.** Humor may emerge as an integral part of the conversation, especially for operational tasks that were previously communicated via formal email and direct message statements. You'll start using emoticons, short statements, and emoji reactions to lighten the formal tone, ask questions, and project your understanding and awareness of the message.
- **Goodbye to tense war rooms.** Since any project's progress is carefully tracked within the communication console, the chatroom won't turn into a [war room](#). All necessary information is documented, updated, and accessible to the team members in real-time. The possibility of debates heating up to the point of no-return becomes less likely.

Chatbots are not ChatOps

Chatbots and centralized communication platforms have existed for years. These alone, of course, are not enough to ensure the benefits of real time context and collaboration.

Embracing ChatOps as a culture promises that information sharing will eliminate the barriers between siloed Devs, Ops and IT Support environments.

Additional resources

For related reading, explore these links:

- [BMC Business of IT Blog](#)
- [BMC DevOps Blog](#)
- [What Is "IT-Business Alignment"?](#)
- [6 Ways to Improve IT/Business Collaboration](#)
- [Best Practices for Enterprise Chatbots](#)