

IBM

Building and deploying faster with Slack



VERTICAL

Technology



NUMBER OF EMPLOYEES

377,000



LOCATIONS

170



SLACK INTEGRATIONS

20+

No matter how much progress Thomas Lawless, a senior software engineer at IBM, or his team had made the previous day, they started every workday checking to see if there was code to review. “It was like we were remaking the process every morning,” he says.

Lawless is responsible for overseeing the production, deployment, and delivery of some of IBM’s largest intranet applications. “My scope spans development, continuous integration, test automation, deployment automation, and operations,” he explains, “which means that on any given day I’ll work with 40 or 50 different people, all on different teams.”

A major aspect of Lawless’s job is to introduce development techniques and services to improve operations and speed up delivery. He started using Slack with other teams within IBM over a year ago as a hub for team communications, but he soon discovered that Slack is also useful for gathering non-human communication, like system alerts and notifications from other services, as well.



“We have what we like to call an ‘end to end delivery pipeline’ that starts with source code and goes all the way through to production deployment. And now we have Slack integrated into all the key milestones in that process.”

THOMAS LAWLESS

Senior Software Engineer



Managing build and deployment processes in Slack

At IBM, teams lean towards public Slack channels (for example, #development-team) so team members can discuss issues openly and experts from other teams can easily join to add their input.

Team channels contain messages from people as well as system alerts from the numerous applications they're using. Say, for example, a developer submits a user story for review in the source code. The system triggers a notification in the team's Slack channel letting everyone know that there's new code up for review. The reviewer can then go into the system to review the code directly from that Slack message.

"Before we used Slack, a developer would have had to find the person they felt should be the reviewer, and then email them or launch a one-on-one chat with them," says Lawless.

With teams' workflows coursing through Slack, they can keep up the pace of their progress knowing they'll be notified about individual code reviews as needed, which they can then acknowledge and start to address from right within Slack.

Rallying teams around system alerts for better incident management

Here's a quick look at some popular team channels at IBM and the kinds of conversations and notifications that end up there:

#help-services: Collects pull request (PR) notifications to facilitate automated peer code reviews and alerts the team when the request has been approved, saving team members from having to switch applications to get the latest update. The team also relies on their Jenkins / Travis CI integration to notify the team when the status of a build changes.

#help-deployments: Team members are notified of failed deployments as code changes make their way through test automation.

#help-tasks: Posts notifications when there's a failure in batch processing jobs.

#starfleet-monitoring: For runtime monitoring and incident management, notifications from NewRelic, Splunk and PagerDuty are pushed here.



Lawless explains that when an alert about a failure or an incident pops up in Slack, relevant team members will strike up a new, incident-specific channel where they can discuss potential resolutions and where anyone can invite other experts as needed.

When the issue's been resolved, the team ends up with a record of the entire incident, including all the files, screenshots, error messages and alerts discussed as they worked towards a resolution.

"The channel serves as a kind of audit trail," says Lawless. "We use these incident channels as the root of our analysis for our post-mortems and what's great is there's no guesswork, because we have all that history right there."

More recently, Lawless and his team have started using Slack with providers of the various services they use. He's found that most companies are happy to share a private Slack channel across their teams so they can talk about problems and questions they might have, saving them a lot of back and forth.

Fewer extra steps make for big productivity gains

Lawless and his team have integrated Slack into every stage of the development process—from writing and testing initial source code all the way through to final deployment. Instead of spending mornings checking for new code to review, everyone picks up where they left off the day before.

"Anytime I've seen a Slack integration I've turned it on," says Lawless. "It's provided so much value and helped us save so many extra steps in our process."



Intrigued? Lots more at slack.com/enterprise.

