

 **Atlassian Summit**

DevOps the Atlassian Way



ROGER BARNES | PRODUCT MANAGER | ATLASSIAN | @MINDSOCKET

Agenda

State of software


Steps to DevOps

DevOps behind the firewall

Q & A

**Software
is programming
the world.**

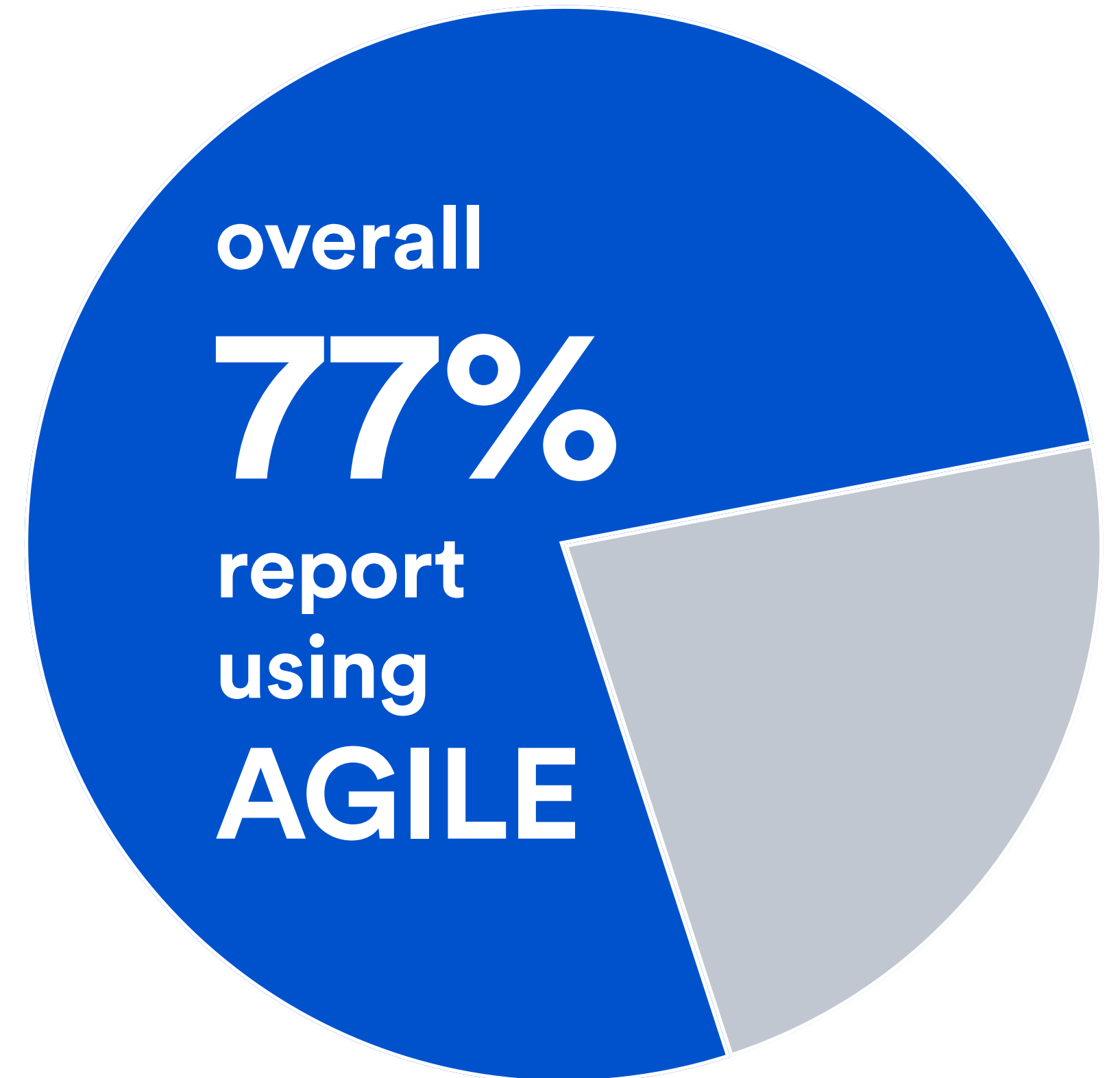
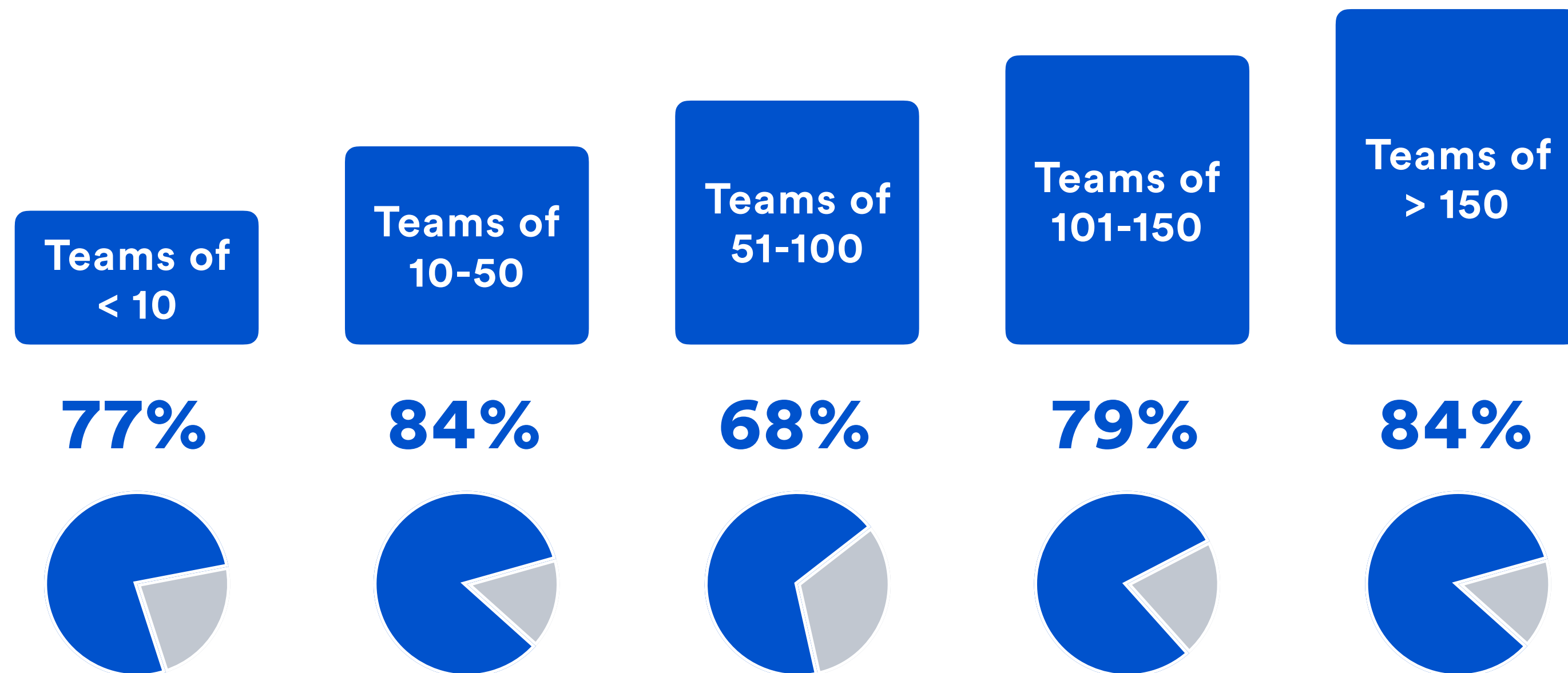


This is  software



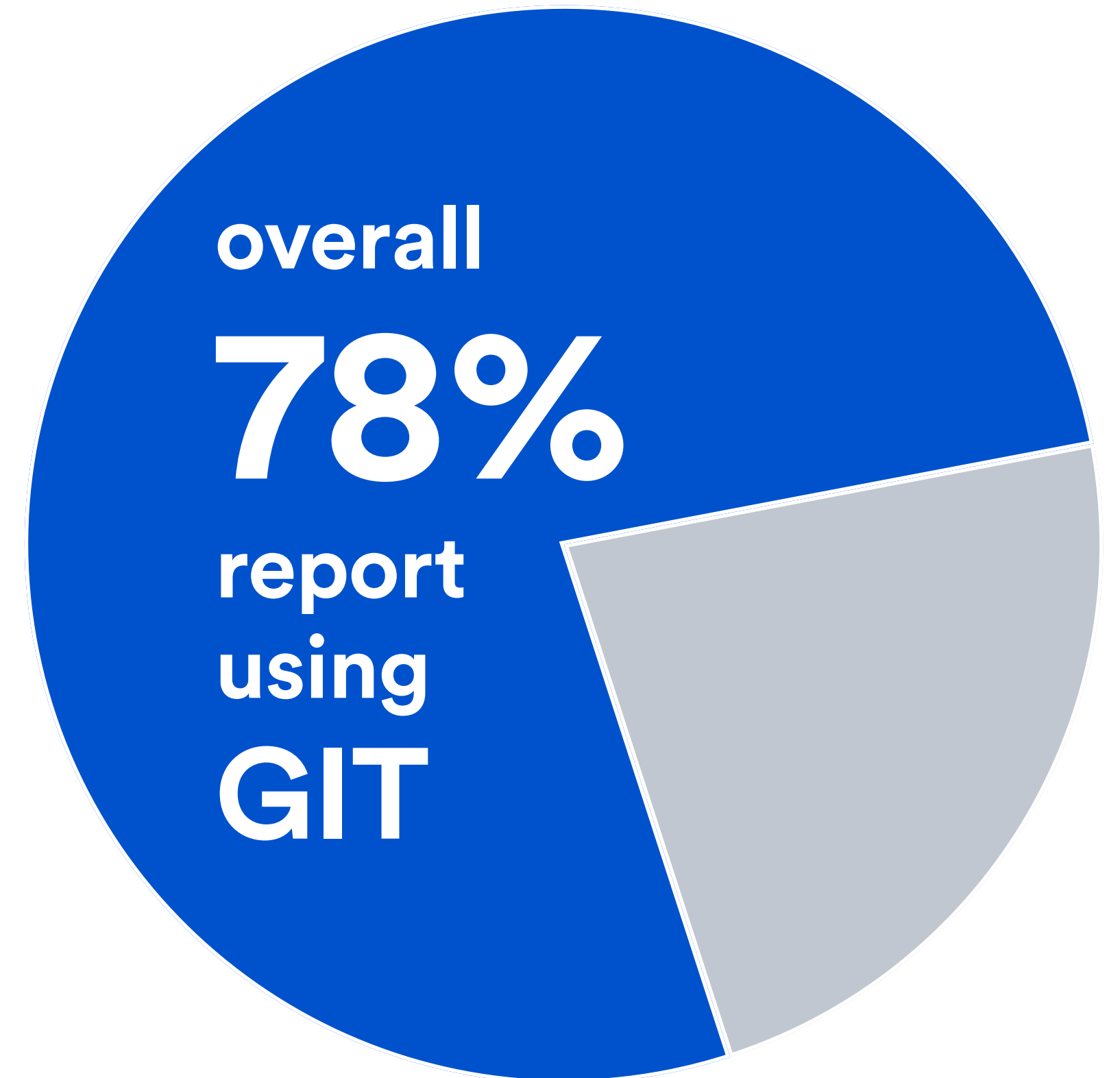
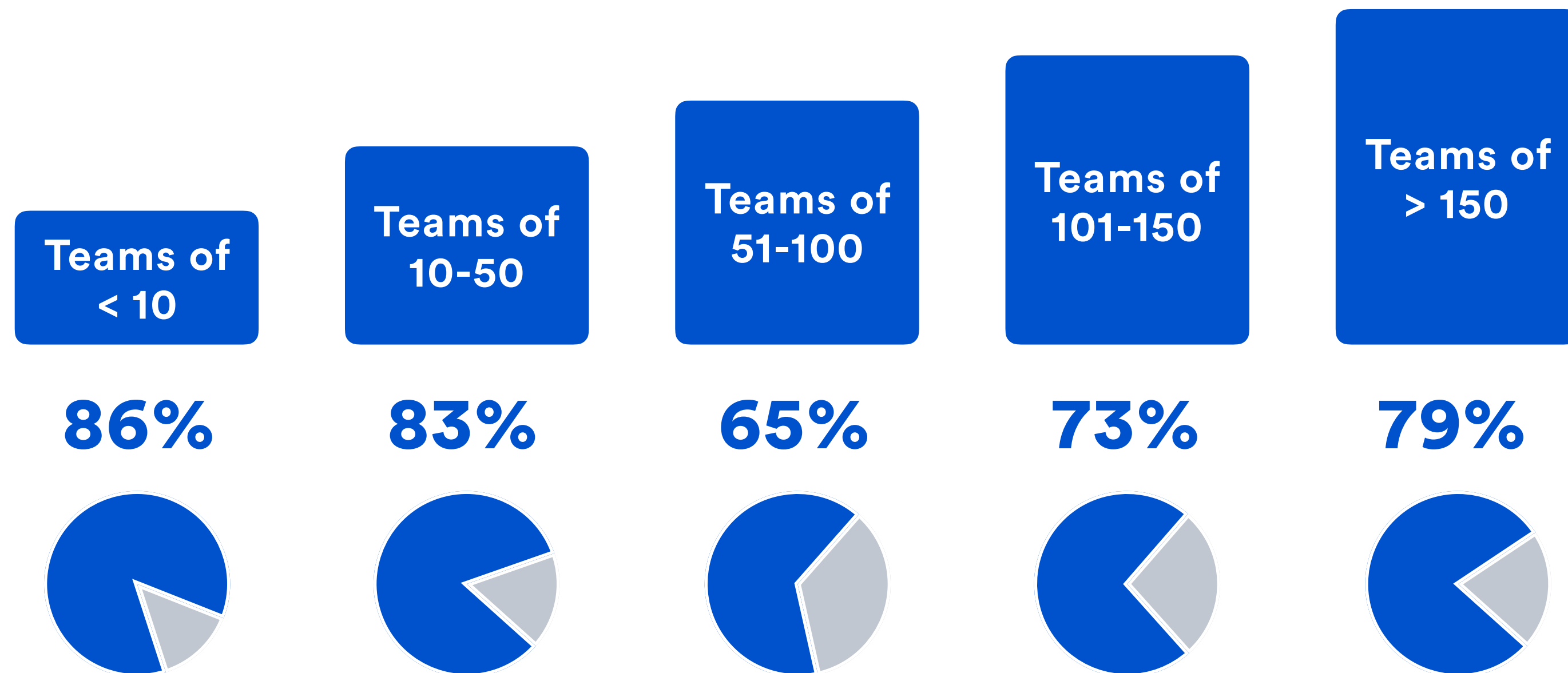
THE NEW NORMAL

Agile & Git



THE NEW NORMAL

Agile & Git



PULSE CHECK

- ▶ **Do incident response times often exceed SLAs?**
- ▶ **Is infrastructure always on fire?**
- ▶ **Is there friction between development and operations teams?**
- ▶ **Are releases slipping?**



Silos are still forming





What's **next**
after Agile?



DevOps

A culture where dev and ops collaborate to build a faster, more reliable release pipeline.

Rule of Three



Work flow

Visibility across groups



Amplify feedback

Swarming on incidents



Continuous Experimentation

Culture of learning



**DevOps is
everyone's
job**

High Performing Teams and DevOps

22%

less time on
unplanned work.

24x

faster recoveries
from failures.

200x

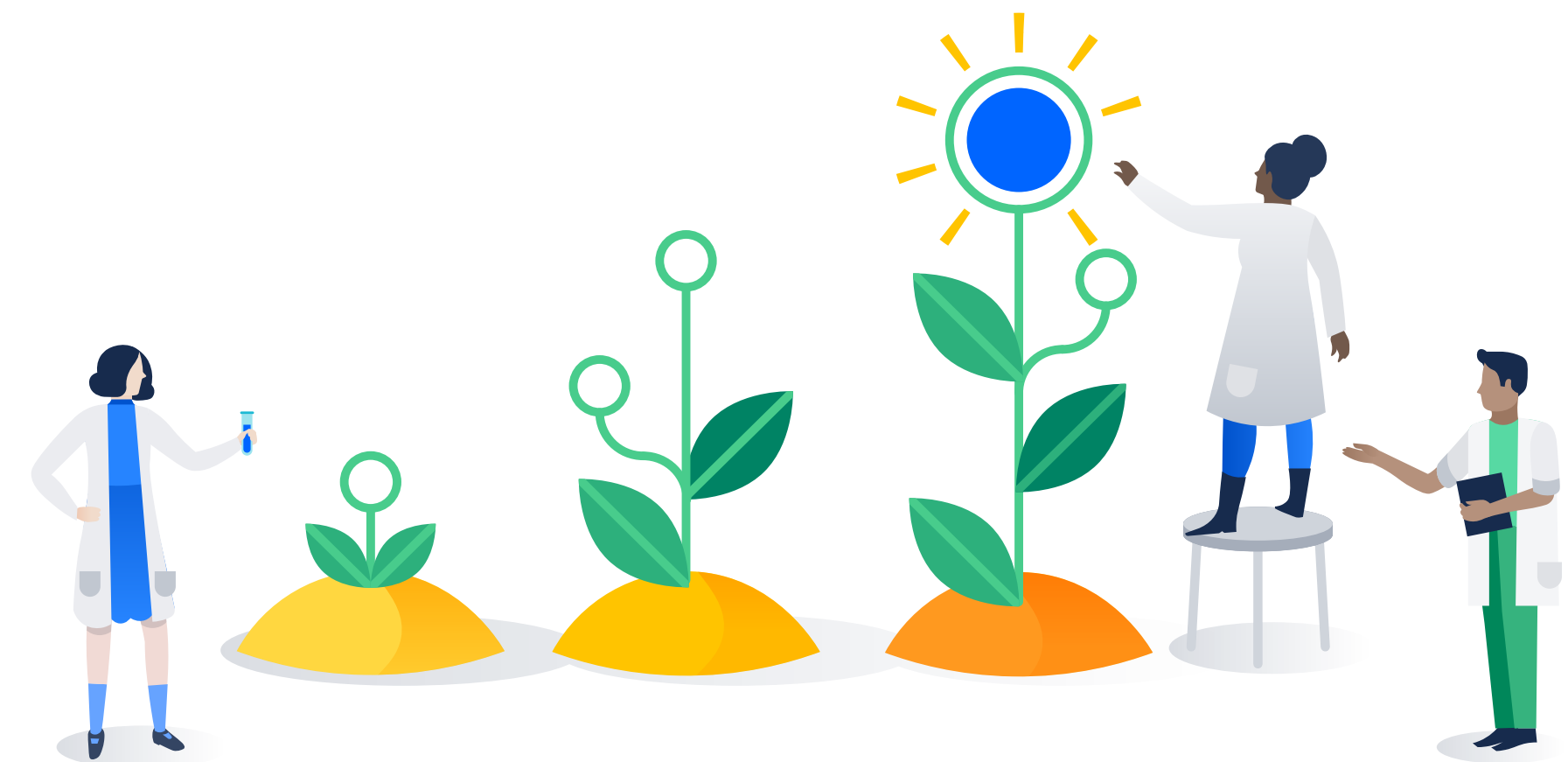
3x

lower change
failure rate.

2,555x

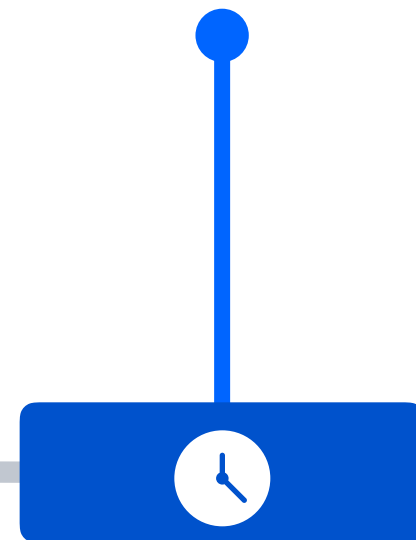
shorter lead
times.

more frequent deployments

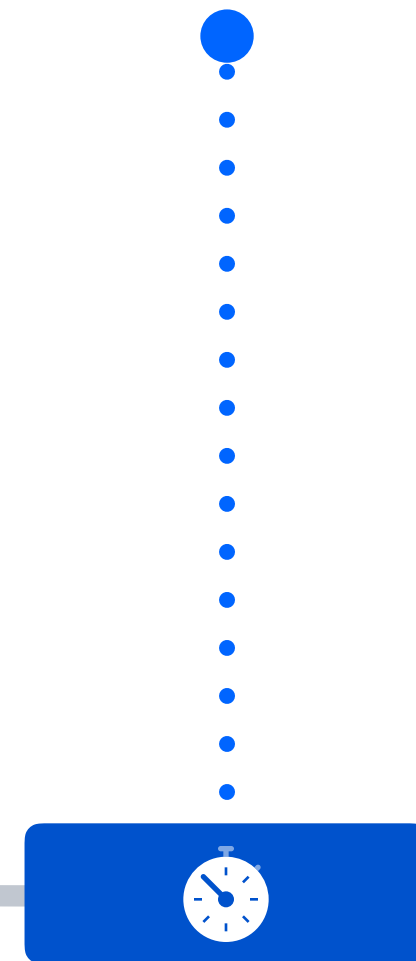


DEVOPS VALUE CHAIN

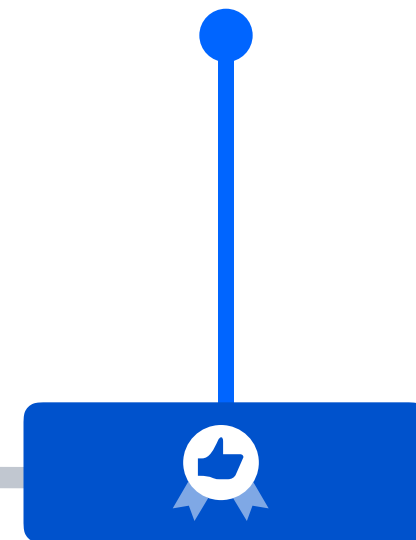
Faster time to market



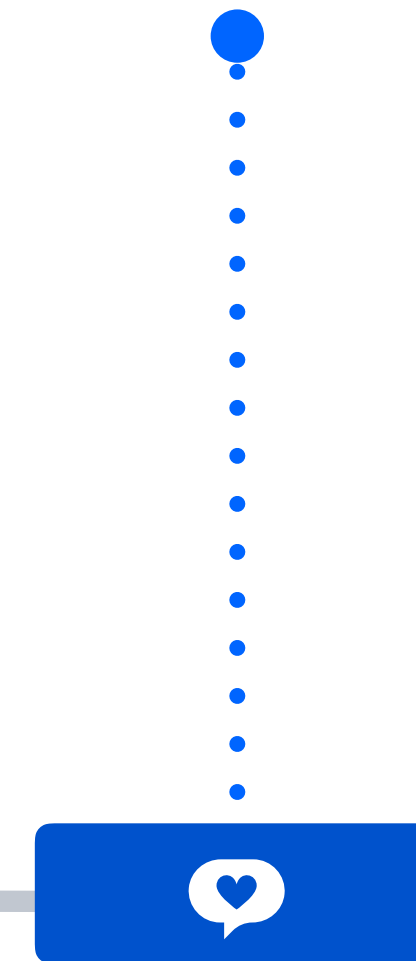
Faster feedback



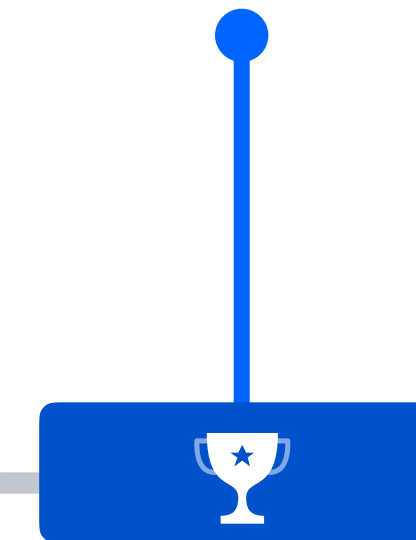
Higher quality product



Happier customers & teams



Competitive advantage



hayneedle



Agenda

State of software

Steps to DevOps

DevOps behind the firewall

Q & A

ATLASSIAN FOR DEVELOPERS

**Atlassian is the
culture and collaboration
layer of DevOps.**

STEPS TO DEVOPS



Culture



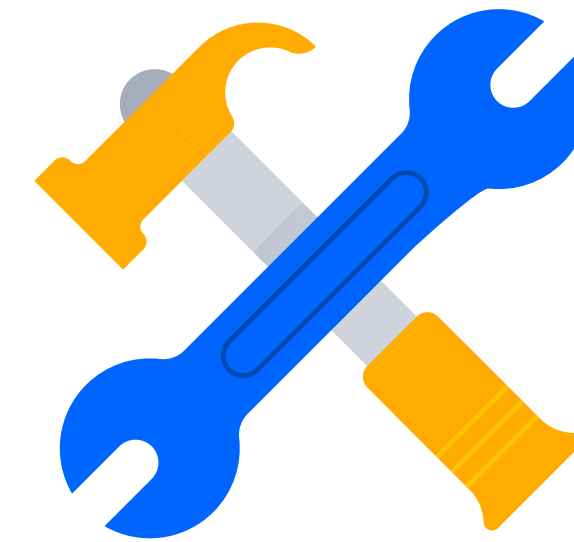
1



Practices



2



Tools

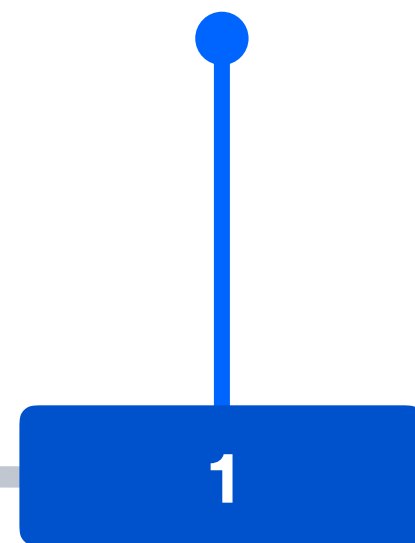


3

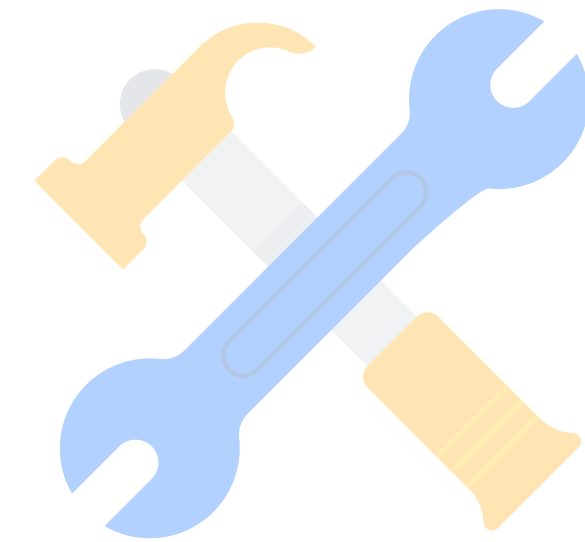
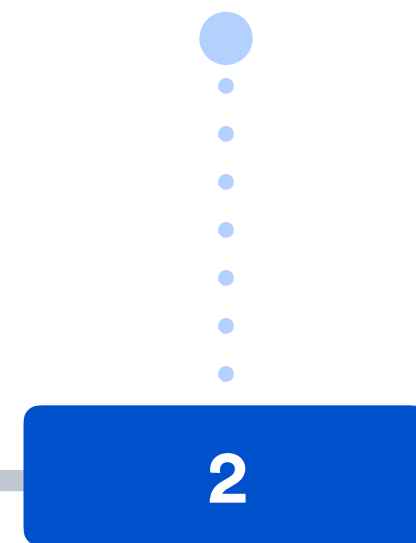
STEPS TO DEVOPS



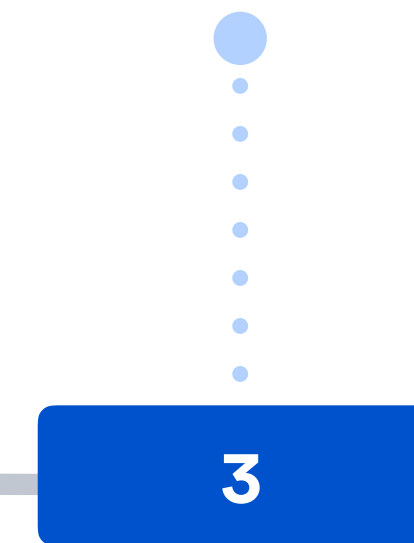
Culture



Practices



Tools



Building a culture of collaboration



Shared responsibility

Everyone shares in wins & failures



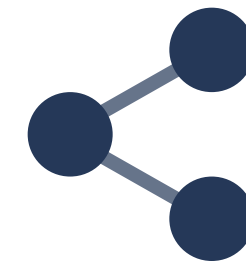
Encourage transparency

Information is readily available



Cross pollination of teams

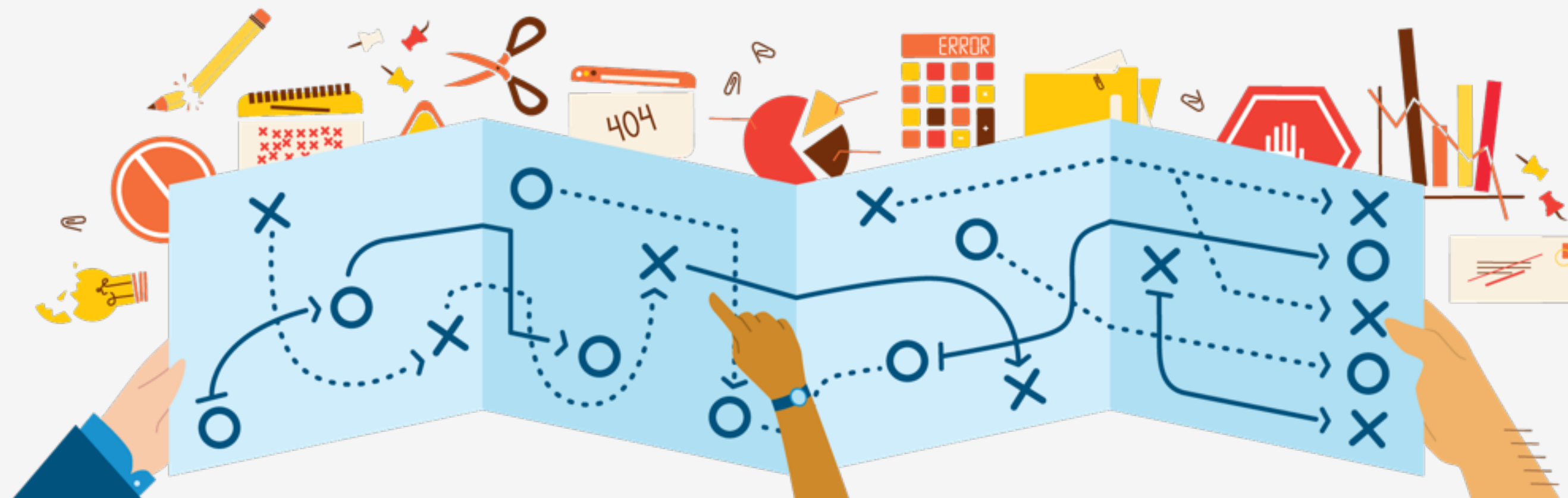
Build empathy & understanding



Effective communication

Teams talk to one another

www.atlassian.com/devops

[Products](#)[Teams](#)[Support](#)[Try](#)[Buy](#)[Team Playbook](#)[Health Monitors](#)[Plays](#)[FAQ](#)

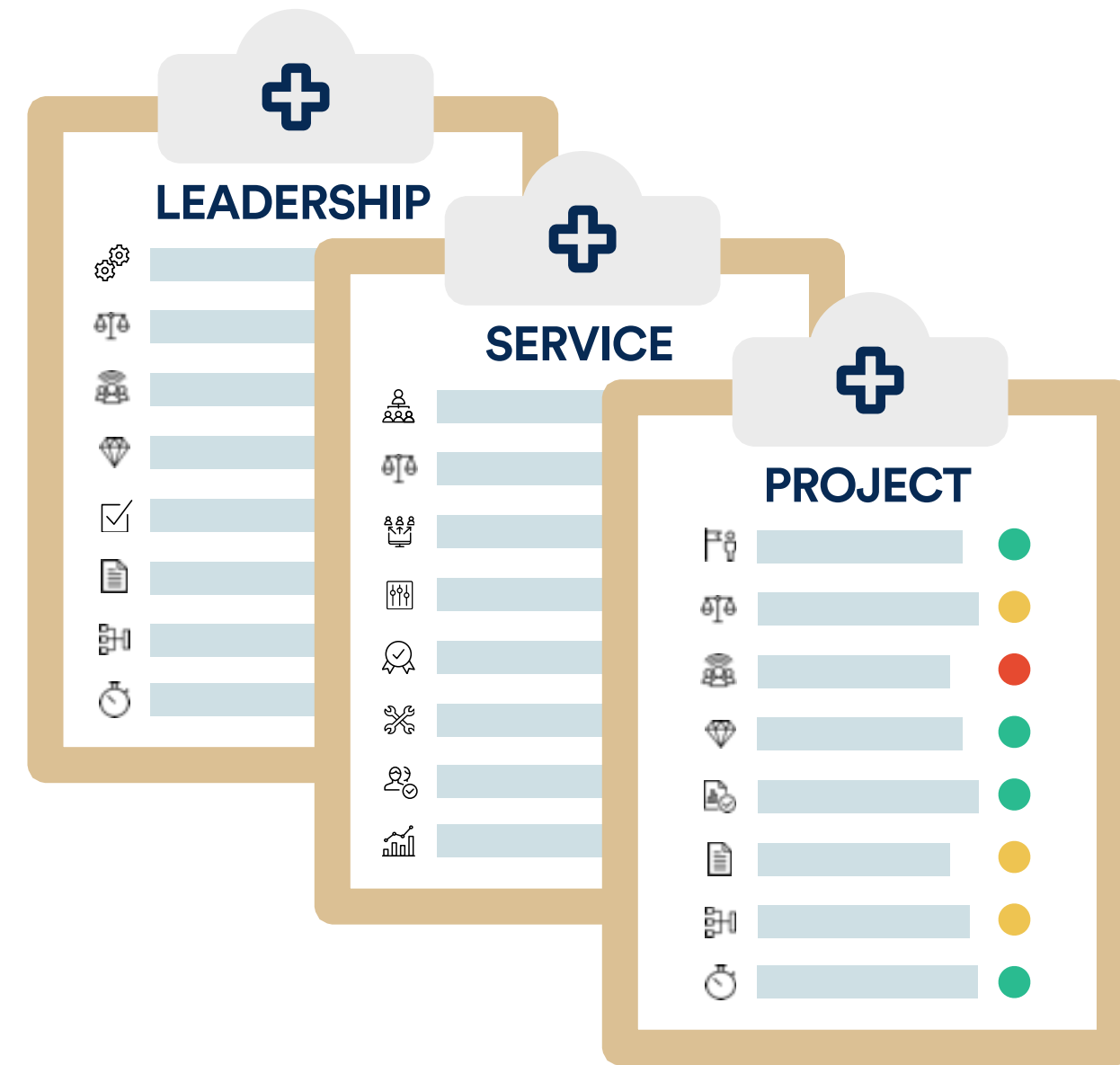
The Atlassian Team Playbook

OUR NO-BULLSHIT GUIDE TO UNLEASHING YOUR TEAM'S POTENTIAL

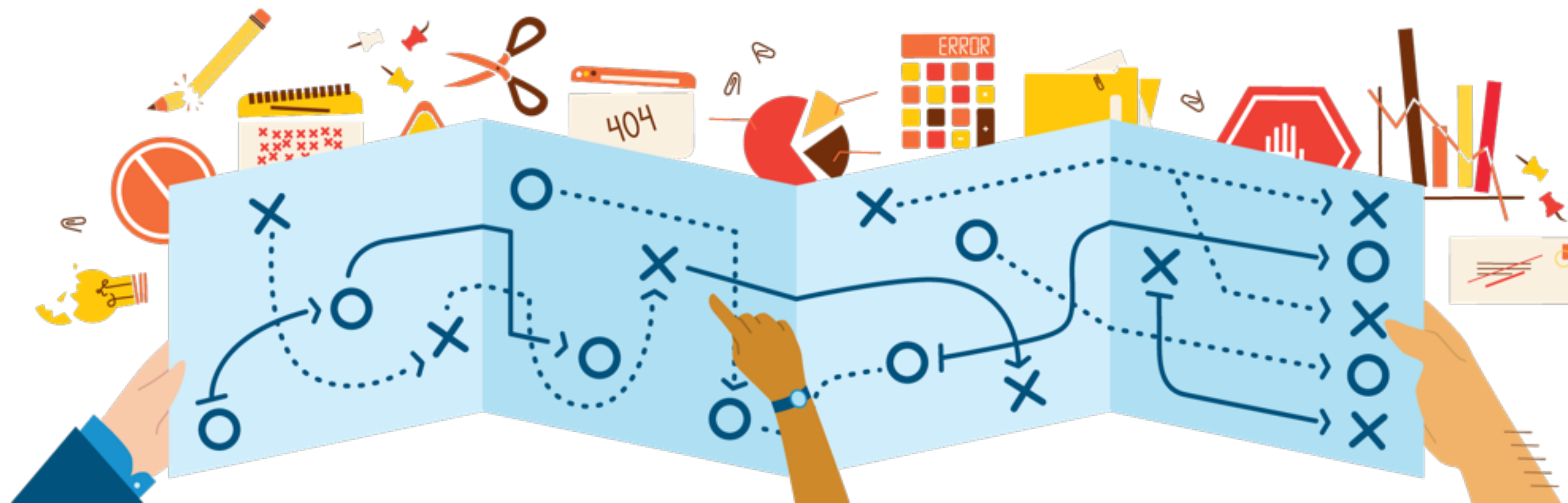
Here's a dirty little secret: working as a team is really hard, and tools alone won't fix that. Books wax poetic about Silicon Valley dream-teams, but they don't tell you how to start improving *your* team. For Atlassian, the mission to unleash the potential in every team starts at home. We've developed a playbook that changed the way our teams work. Now it's yours to try, too.

This ain't your CEO's ivory-tower management book. It's by teams, for teams. You'll find step-by-step guides for tracking your team's health, and plays that build your Get \$#!t Done™ muscle. Start with a Health Monitor workshop to get a reading on your vital signs, or just dive straight into the plays.

TEAM HEALTH MONITORS

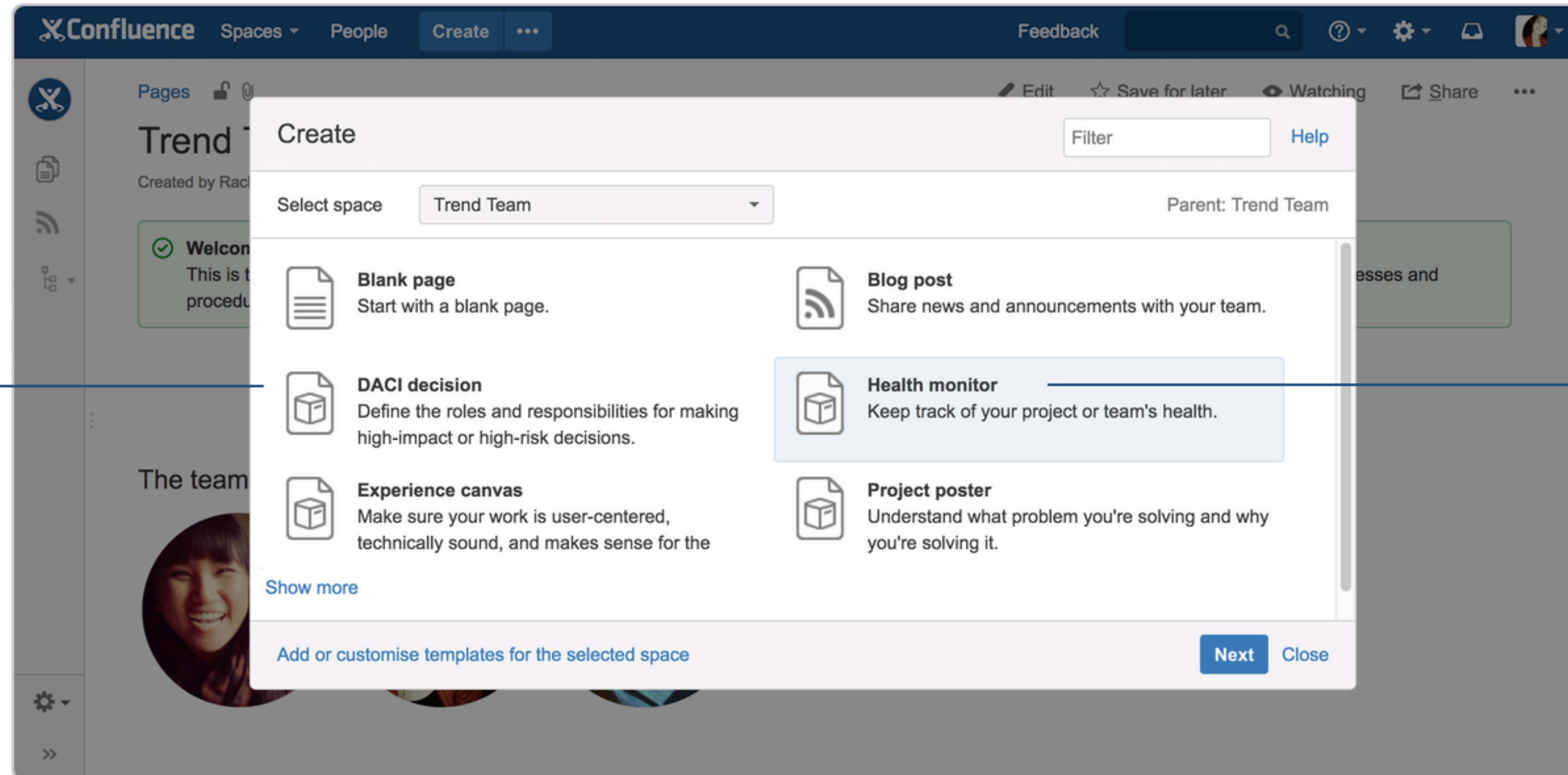


TEAM PLAYS



Playbook blueprints in Confluence Server 6.1

Make better decisions
Help your team rally behind the most important decisions



Stay on track
Check the health of your team or project

STEPS TO DEVOPS



Culture



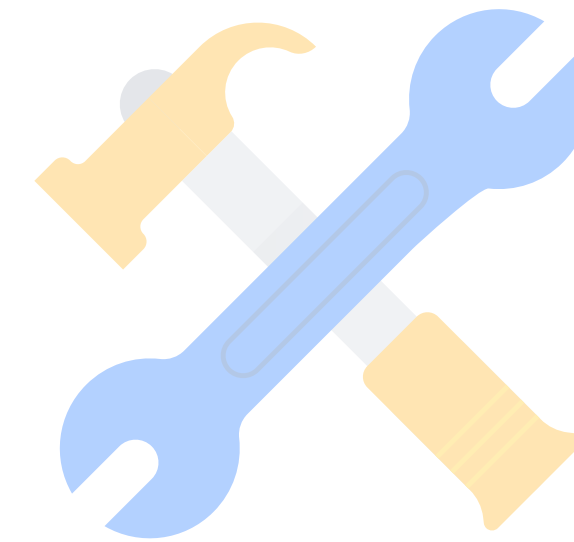
1



Practices



2

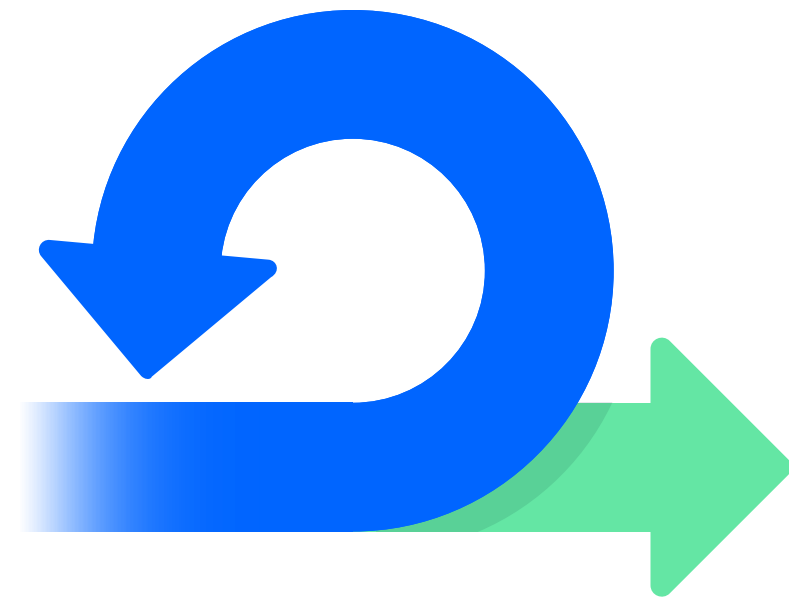


Tools



3

Practices



Agile



DVCS



CI/CD

AGILE

Aids in
culture shift






Quick reaction
to change

atlassian.com/agile

The Agile Coach

Our no-nonsense guide to agile development

What is Agile?

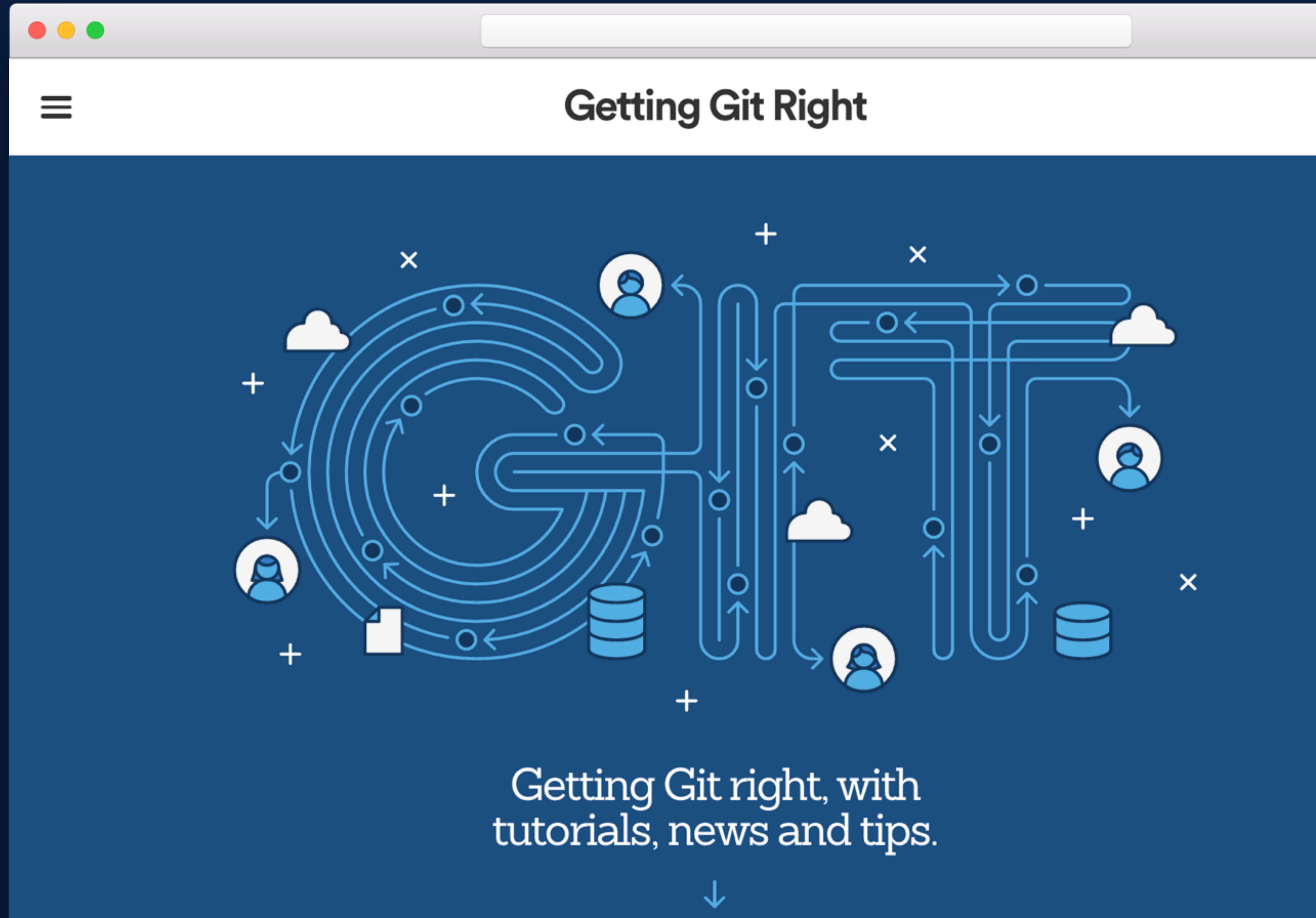
- **The Agile Developer**
 - Branching for agile teams
 - Secrets to agile estimation
 - Agile testing like a boss
- **The Agile Team**
 - A brief introduction to kanban
 - Four agile ceremonies
 - A brief introduction to scrum
- **The Agile Program**
 - Agile backlogs
 - Agile roadmaps
 - Agile requirements
- **The Agile Product Manager**
 - Your first week
 - Customer interviews
 - Requirements
- **Getting started with agile**
 - Set up a scrum project
 - Advanced scrum practices
 - Set up a kanban project

GIT

Quick iterations

Branching & merging

atlassian.com/git



The screenshot shows a browser window with the title "Getting Git Right". The main content is a complex diagram illustrating Git branching and merging. The diagram features a central "G" shape formed by multiple overlapping paths, representing different branches. Various icons are scattered around the diagram, including user avatars, clouds, databases, and plus (+) and minus (-) signs, indicating different components and actions in the Git workflow. The diagram is set against a dark blue background.

Getting Git right, with tutorials, news and tips.

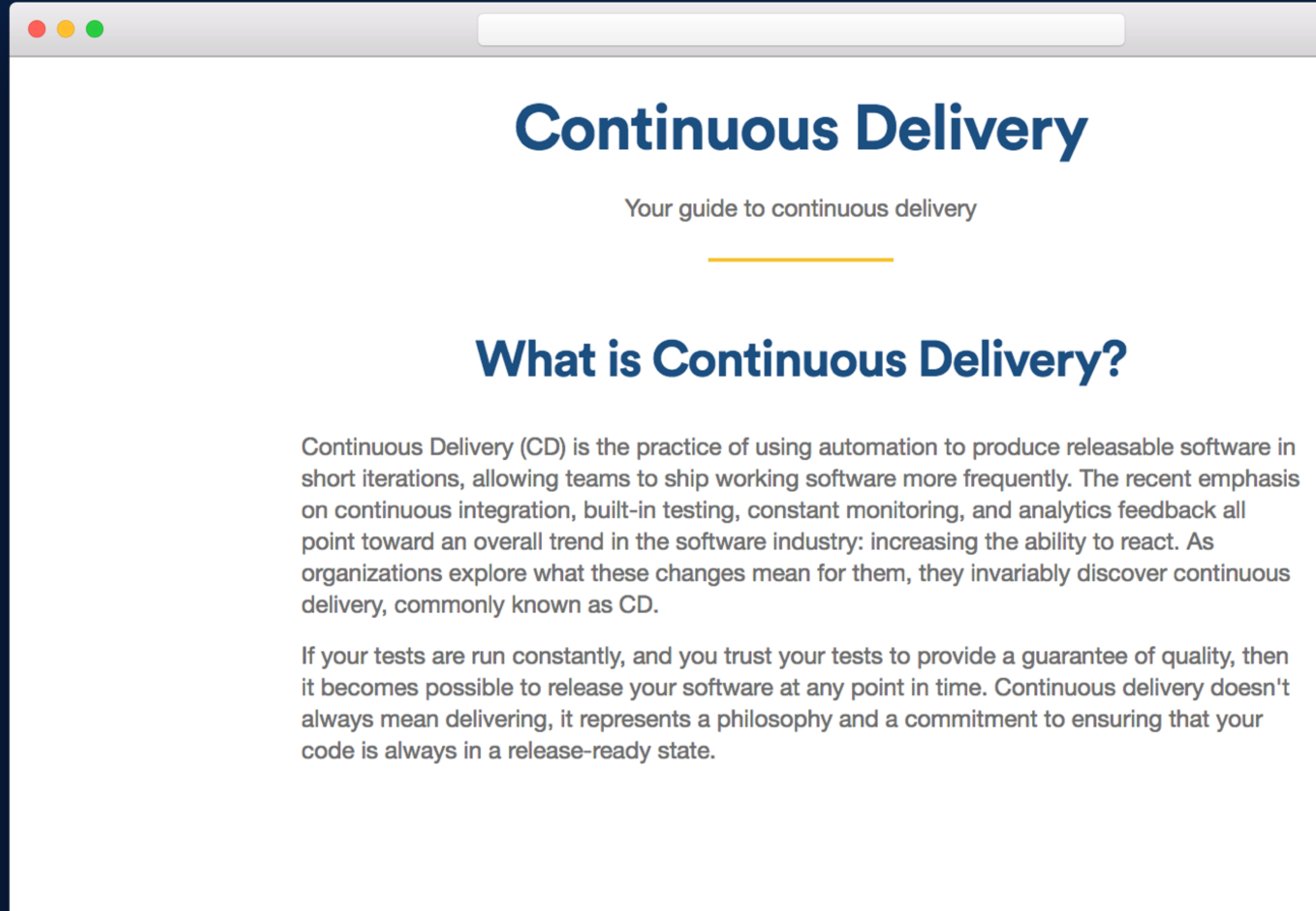
↓

CI/CD

Fast feedback

Automation

[atlassian.com/
continuous-delivery](https://atlassian.com/continuous-delivery)

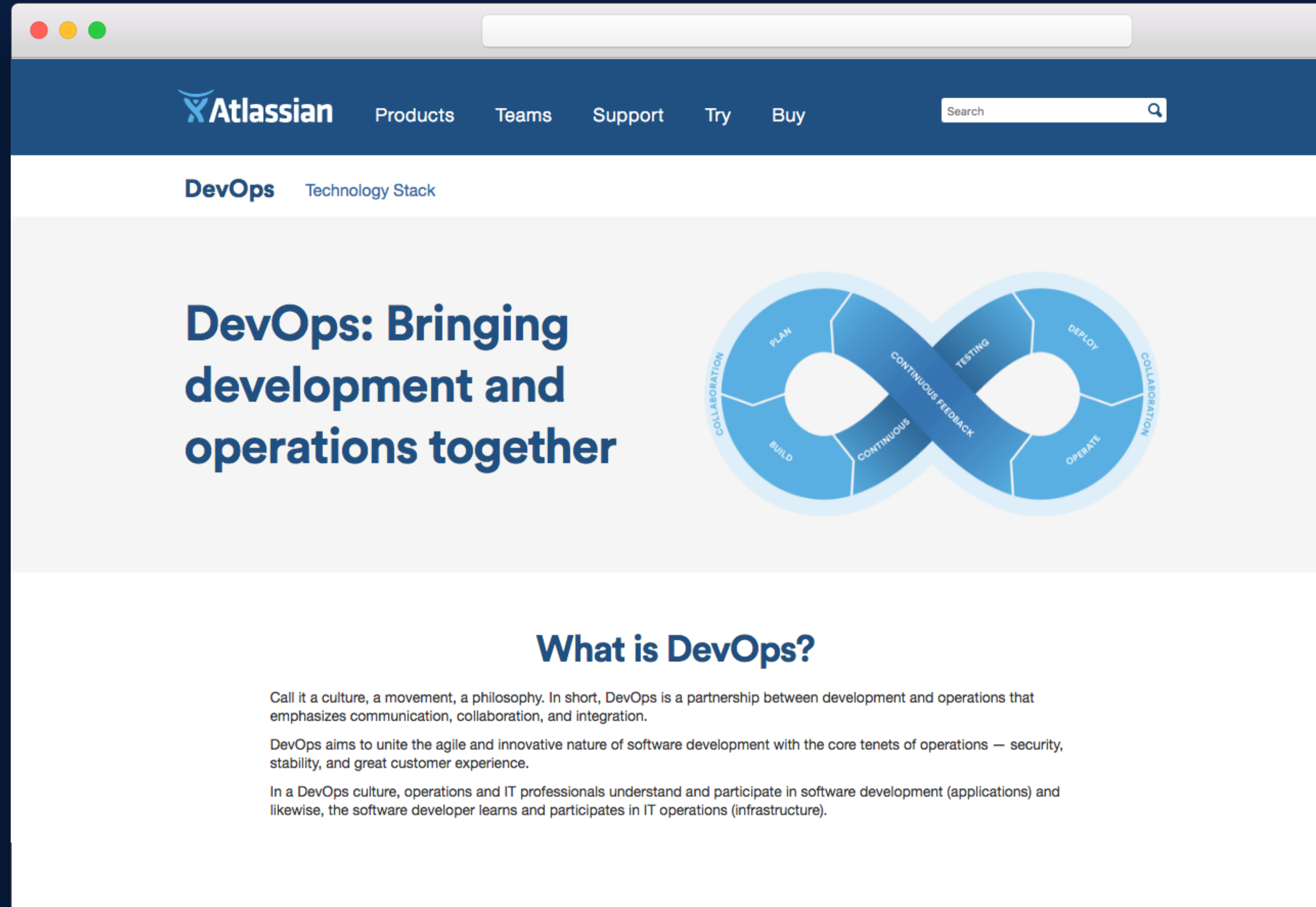


DEVOPS

Self-assessment

Tools & resources

[atlassian.com/
devops](https://atlassian.com/devops)



The image shows a screenshot of the Atlassian website's DevOps page. The page features a dark blue header with the Atlassian logo and navigation links for Products, Teams, Support, Try, and Buy. A search bar is located in the top right corner. Below the header, the page title "DevOps" is followed by "Technology Stack". The main content area has a light blue background and features a large heading: "DevOps: Bringing development and operations together". To the right of the heading is a circular diagram representing the DevOps cycle, which includes stages like PLAN, BUILD, TESTING, DEPLOY, OPERATE, and a central "CONTINUOUS FEEDBACK" loop, all surrounded by "COLLABORATION". Below this, a section titled "What is DevOps?" provides a definition: "Call it a culture, a movement, a philosophy. In short, DevOps is a partnership between development and operations that emphasizes communication, collaboration, and integration." It further explains that DevOps aims to unite agile software development with operations tenets like security and stability, and that in a DevOps culture, development and operations professionals collaborate and learn from each other.

STEPS TO DEVOPS



Culture



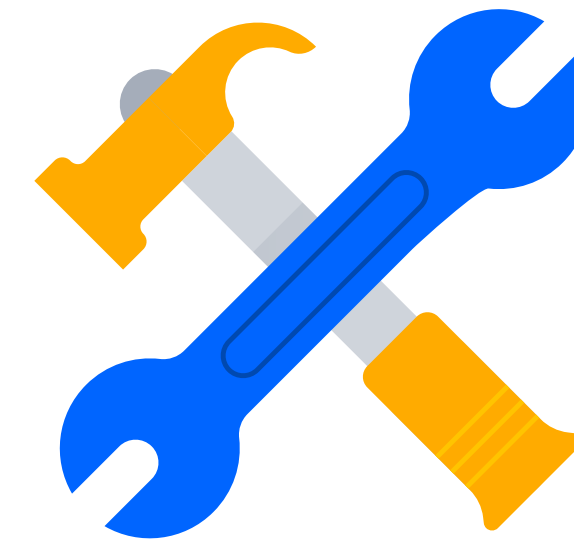
1



Practices



2



Tools



3

Agenda

State of software

Steps to DevOps

DevOps behind the firewall

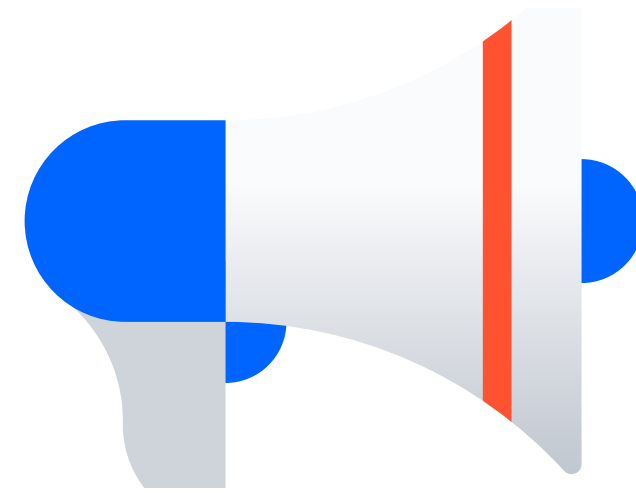
Q & A

DevOps behind the firewall - why not?



Flow

Make work visible



Amplify feedback

Swarming & CI



Continuous experimentation

Culture of learning

Focus areas

Flow

Feedback

Experimentation
vs control

Behind the firewall doesn't mean slow

Boost productivity through tight communication across teams and easy access to information

Measure success

Velocity, release frequency, defect rates, team happiness

Identify and remove pipeline friction

Apply consistent workflow automation within and between tools

Focus areas

Flow

Feedback

Experimentation
vs control

Fast feedback, and shared responsibility

Modern CI/CD practices help teams improve in all of these areas

Catch failures early

Give developers more control and ownership over builds

Integrations are key

Bring problems to the attention of developers earlier in the process by getting information where it's needed

Focus areas

Flow

Feedback

**Experimentation
vs control**

Compliance is not the enemy

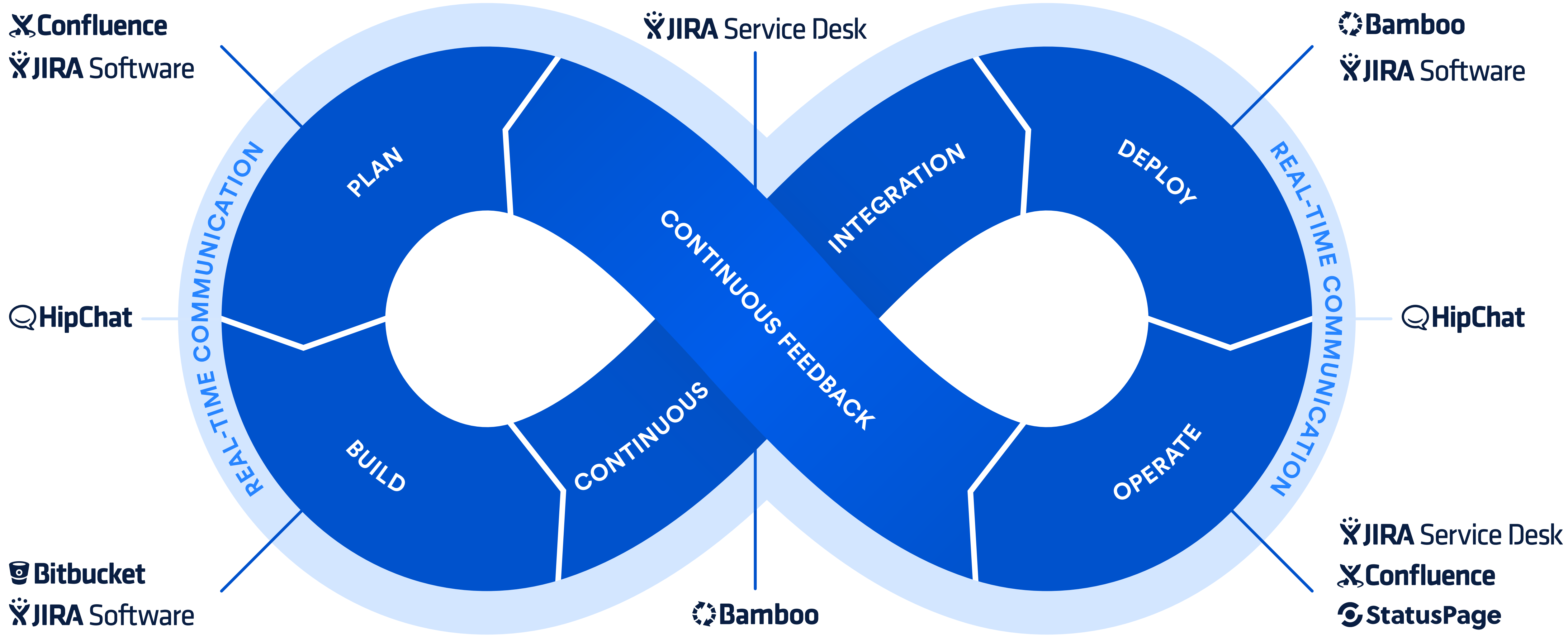
Compliance shouldn't keep you from adopting DevOps practices

Provide a safe place to experiment

Only apply control where it's needed

Install consistent guard rails

Repeat success at scale by applying common best practices across the organisation

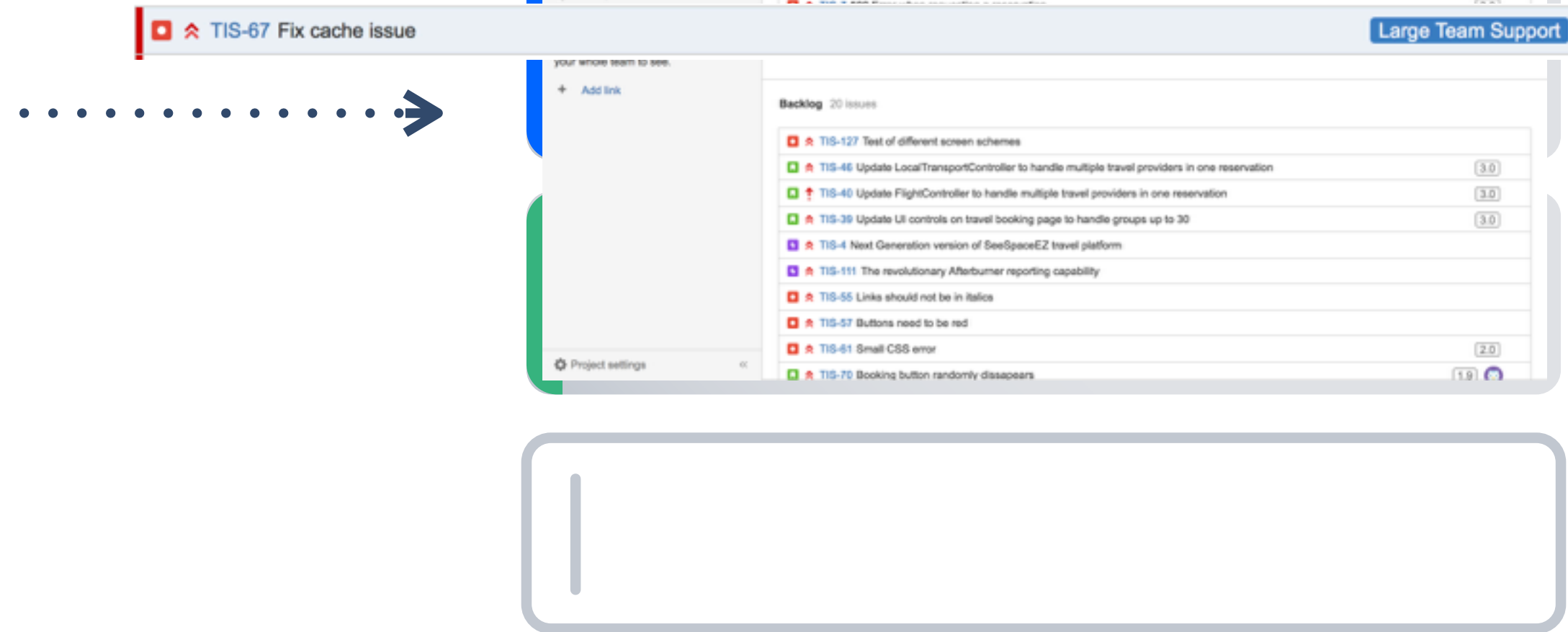




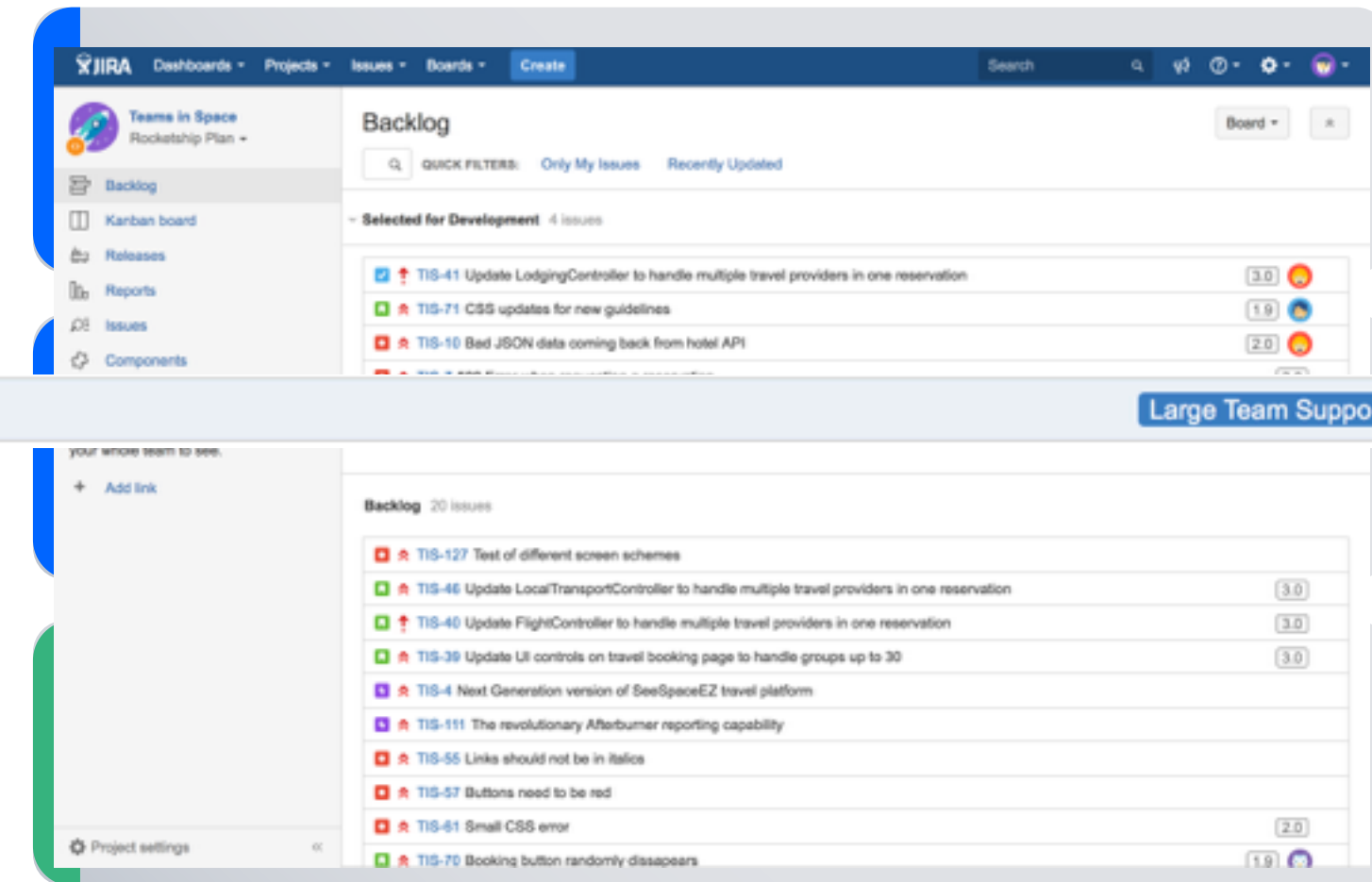
Teams In Space[®]

A DevOps Story

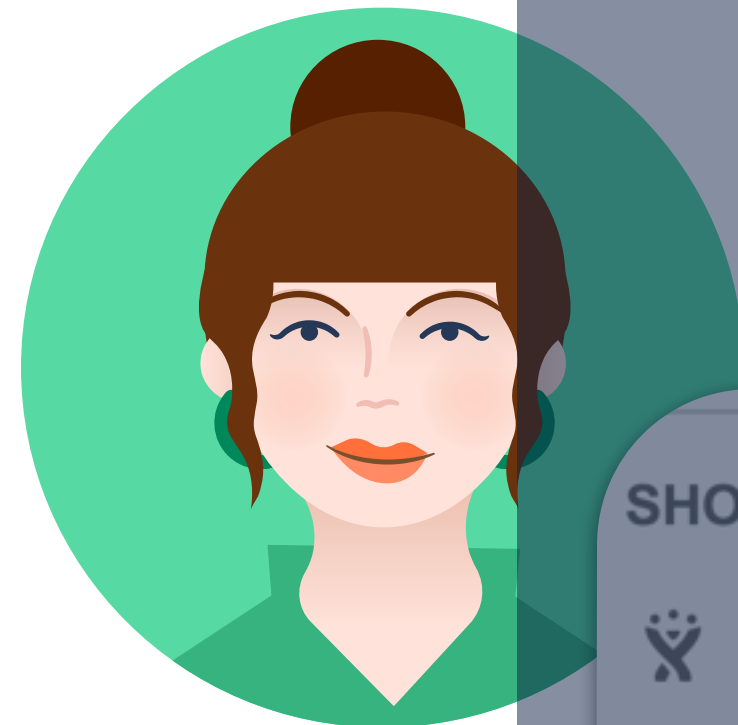
Tasks added to backlog



JIRA Software



Development



Jennifer
Developer

The screenshot displays an IDE interface with a file explorer on the left and a code editor on the right. The file explorer shows a project structure with folders like .idea, EIS Helpers, src, and java/ch/mibex/confluence/include. The code editor shows a Java file named ConfluenceNewCodePlugin.java with a diff view. A yellow tooltip labeled "Test coverage markers" points to a line of code, indicating "Covered (1 of 2 conditions)". The code includes comments and method implementations for Version and equals methods. In the bottom right corner, a Bitbucket pull request is visible, titled "77 fix cache issue".



Development

Configuration

Job details | Tasks | Requirements | Artifacts | Miscellaneous

Pages & jobs

Default Stage

Default Job

Branches

SL Quick Refer

for a Maven goal

ob's requireme











value that the

such as

Environment variables

Task types

- All
- Builder
- Tests
- Deployment
- Source Control
- Variables

	Sets up Docker Agent Image for Deployment Environment to run --		Drop Nexus Staging repositories
	Dump variables to log Prints current values of available variables. Useful for debugging builds.		Execute on Unicorn Executes a script on a Unicorn Instance by SSH
	Fastlane Execute a build using Fastlane		flaky-test-discovery Reads a flaky test file generated by Hallelujah for display in a web panel
	Grails Execute Grails commands as part of your build		Grunt 0.4.x Execute Grunt 0.4 task
	Gulp Execute Gulp task		Heroku: Deploy WAR Artifact Deploy a WAR artifact to Heroku

Get more tasks on the [Atlassian Marketplace](#) or [write your own](#)

Cancel



Jer
Dev

THE PAST



Build projects / Bamboo Specs Demo / Test My App

Configuration - Test My App

My first Bamboo Spec

Plan Configuration

Stages & jobs

Build Stage

Build My App

Branches

Plan details

Stages

Repositories

Triggers

Branches

Plan contents

Each stage within a plan represents a step within your build process followed by a stage for deployment jobs.

Build Stage

Build My App

+ Add job


Related deployment projects



You can now take the shared artifacts from the results of this build

NOW

```
Plan createPlan() {
    Project project = new Project()
        .name("Bamboo Specs Demo")
        .key("PROJ");
    return new Plan(project, name: "Test My App", key: "PLANK",
        .description("My first Bamboo Spec")
        .stages(
            new Stage( name: "Build Stage")
                .jobs(
                    new Job( name: "Build My App", key: "JOB")
                )
        );
}
```




 Build projects
Configuration as Code

 Project wallboard 

There are no plans in this project.

[+ Create a new plan](#)

Continuous integration powered by Atlassian Bamboo version 5.16.0-SNAPSHOT build 51602 - 27 Feb 17 (DevMode Enabled)

[Report a problem](#) · [Request a feature](#) · [Contact Atlassian](#) · [Contact Administrators](#)



Executable build definition

All the power of Bamboo's structured stages, jobs and tasks

```
PlanSpecsDemo testStage()
    .withKey("CASCI");
final TopLevelPlan plan = new TopLevelPlan(project, "CasC Demo", new BambooKey("CASCDemo"));

plan
    .withStages(buildStage(), testStage());

return plan;
}

Stage buildStage()
{
    return new Stage("Build (all the things)")
        .withJobs(
            new Job("Build Bamboo", "BAMBOO"),
            new Job("Build Bitbucket", "BITBUCKET")
        )
        .withDescription("This is my compile stage");
}

Stage testStage()
{
    return new Stage("Test Bamboo+Bitbucket integration RSP")
        .withJobs(
            new Job("Run here headless tests", "HEADLESS")
        );
}
```

```
Run PlanSpecsDemo
/opt/java/jdk/jdk1.8.0/bin/java ...
Executing request POST http://localhost:8080/bamboo/rest/1.0/project/10000/planSpecsDemo HTTP/1.1
OK
```



Status history data not available

▶ Run ▾

⚙️ Actions ▾

Plan Configuration

▾ Stages & jobs 3

Build (allthings)

☰ [Build Bamboo](#)

☰ [Build Bitbucket](#)

Test Bamboo+Bitbucket integration RSP

Run More Webdriver tests

▸ Branches 0

Job details
Tasks
Requirements
Artifacts
Miscellaneous

Tasks

A task is a piece of work that is being executed as part of the build. The execution of a script, a shell command, an Ant Task or a Maven goal are only few examples of Tasks. [Learn more about tasks.](#)

You can use [runtime](#), [plan](#) and [global variables](#) to parameterize your tasks.

94 agents and 3 elastic images have the [capabilities](#) to run this job

[How to use the Script task](#)

Script ✕

Final tasks Are always executed even if a previous task fails

Drag tasks here to make them final

Add task

Script configuration

Task description

Disable this task

Interpreter

Run your script with /bin/sh or cmd.exe

Script location

Script body*

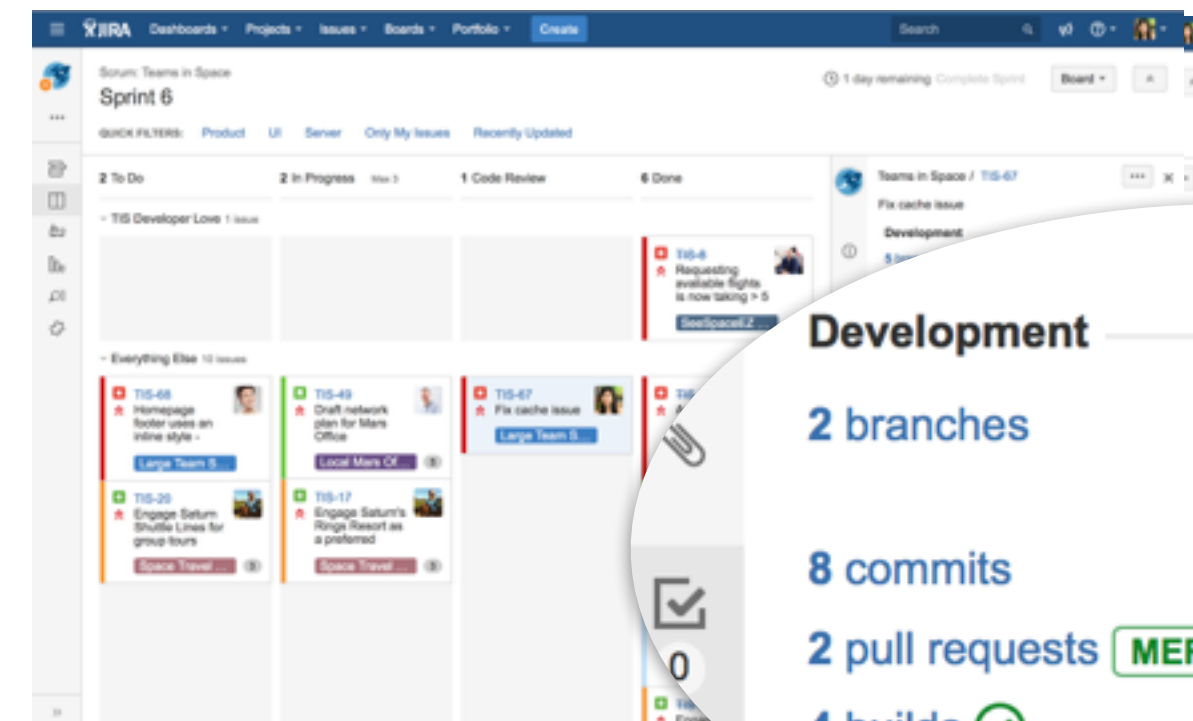
```

1 echo Success
    
```

Development



Jennifer
Developer

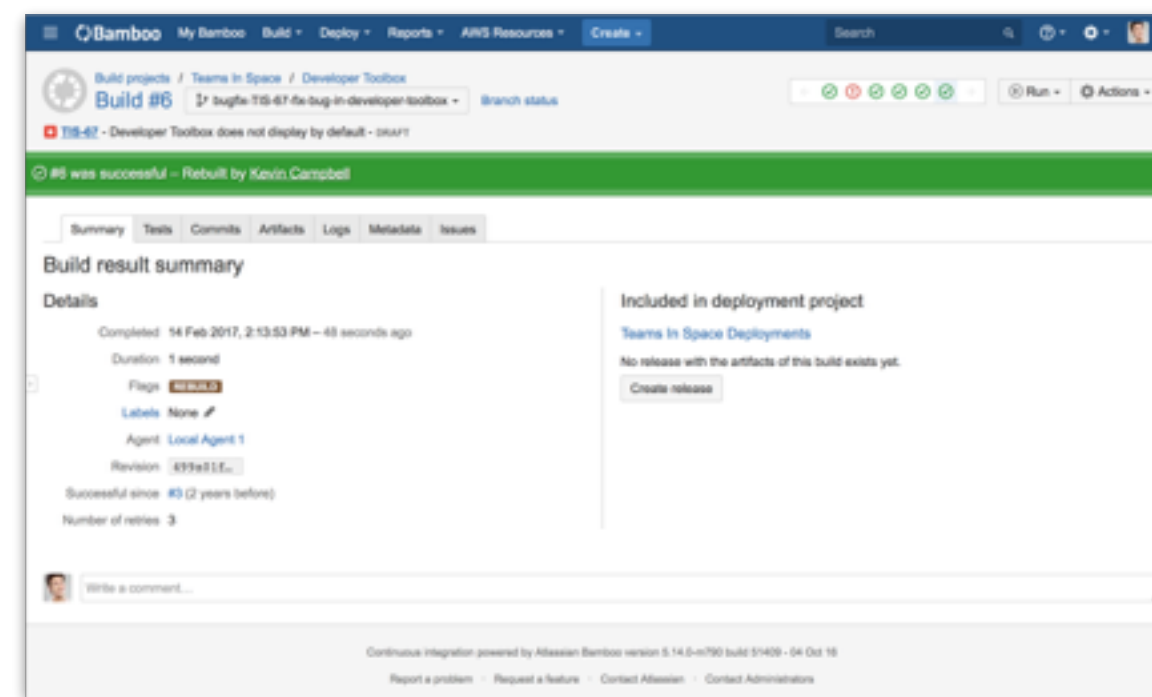


Development

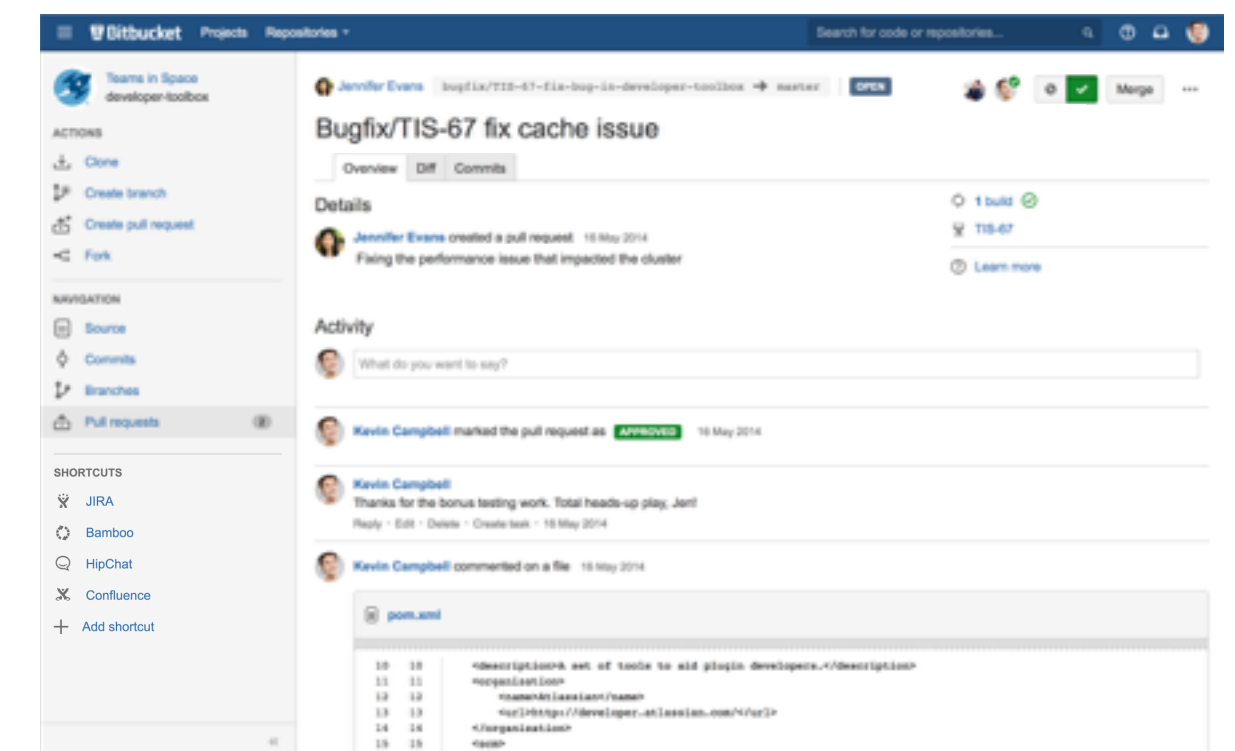
- 2 branches
- 8 commits
- 2 pull requests **MERGED**
- 4 builds ✓

Updated 16/May/14 4:30 PM
Latest Just now
Updated Just now
Latest 29/Oct/15 5:00 PM

JIRA Software



Bamboo



Bitbucket

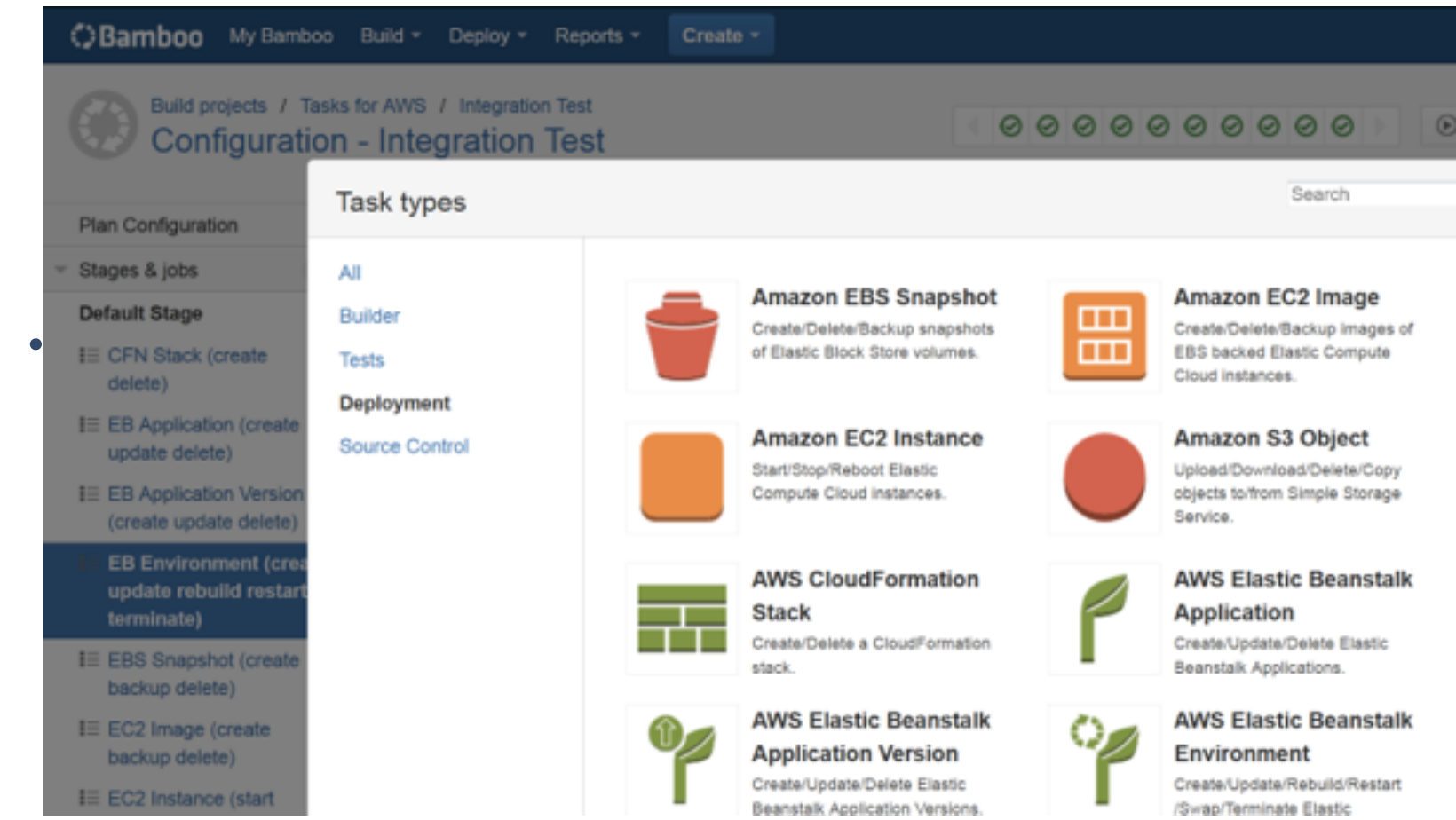
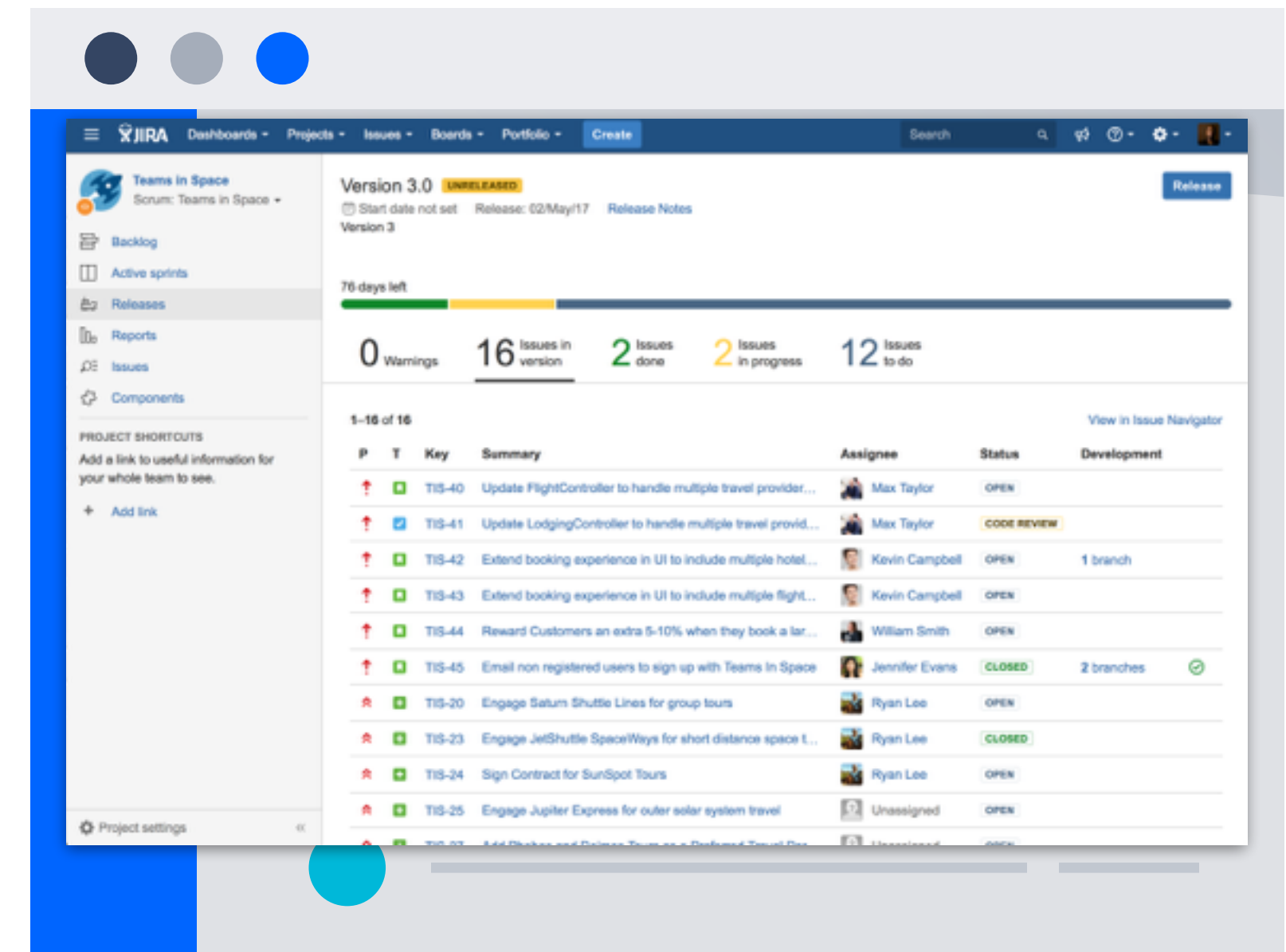
Release



John
Release Manager

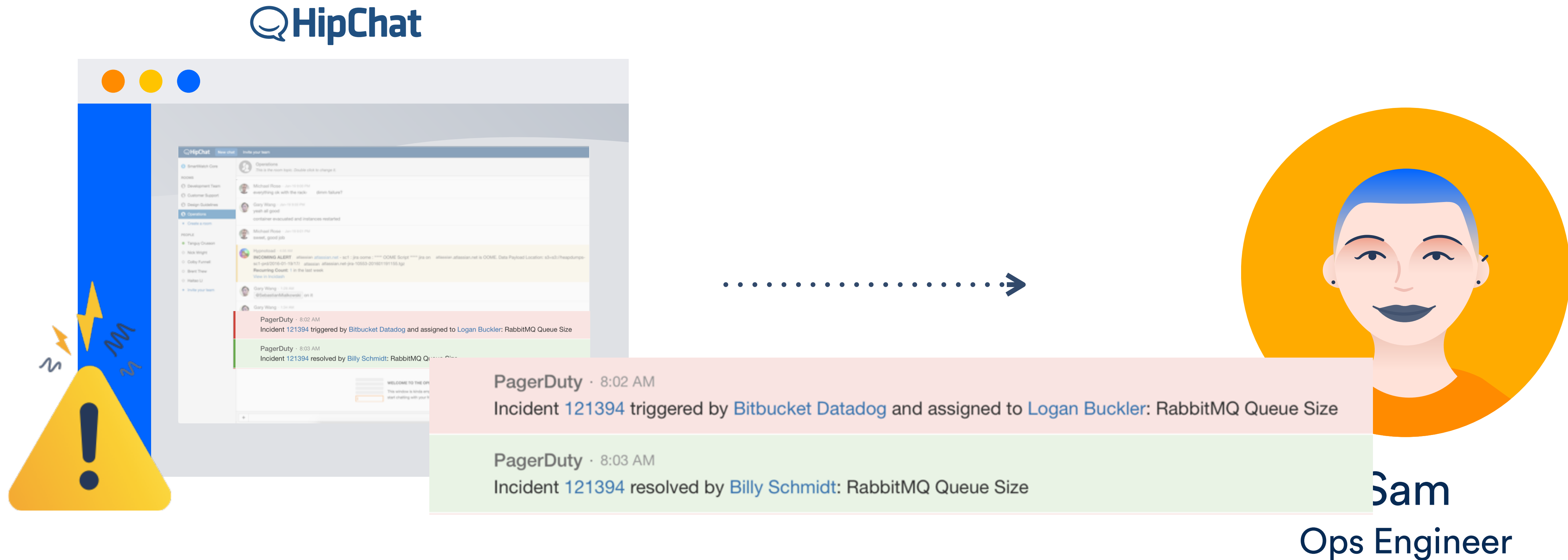


JIRA Software

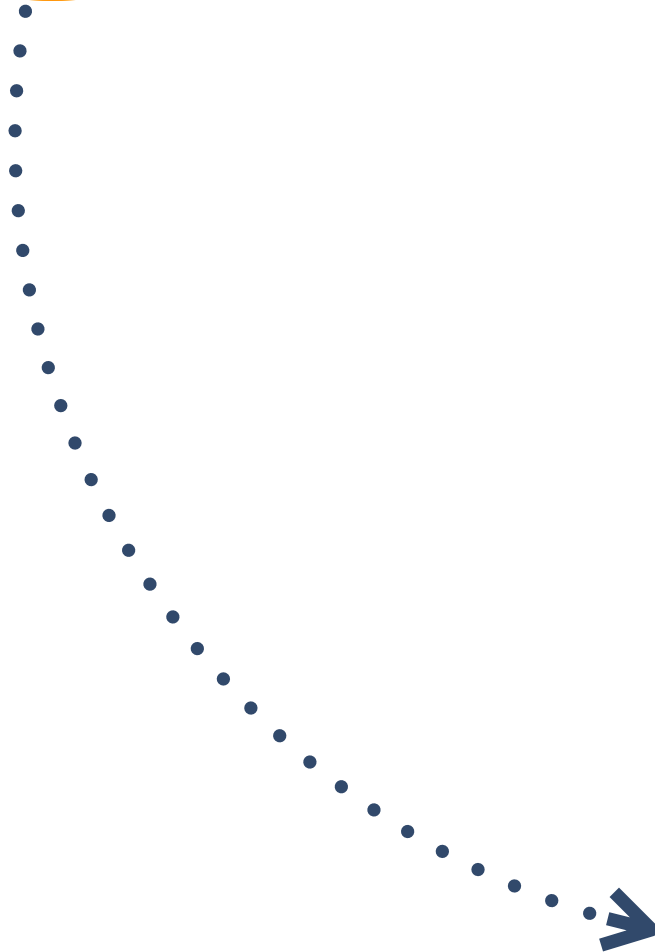


Bamboo

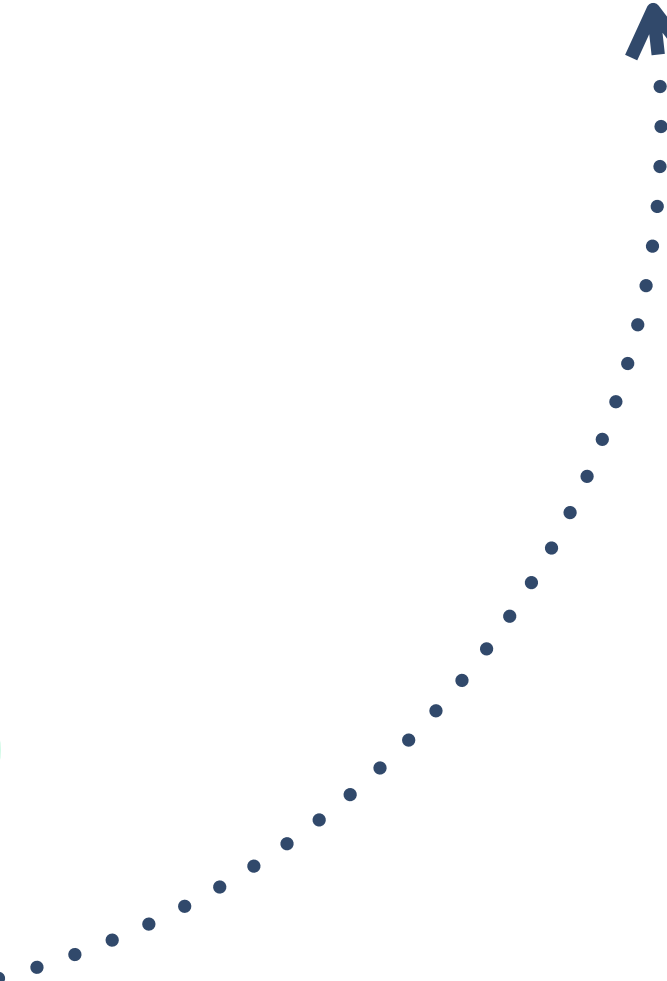
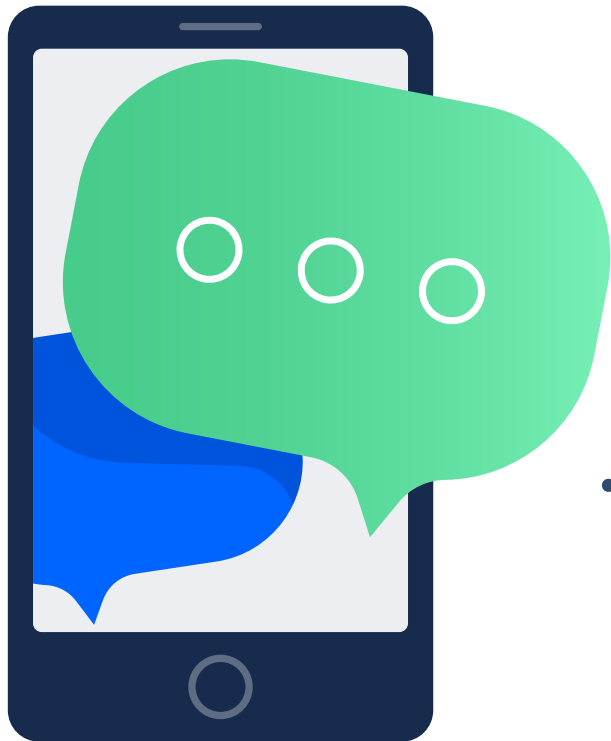
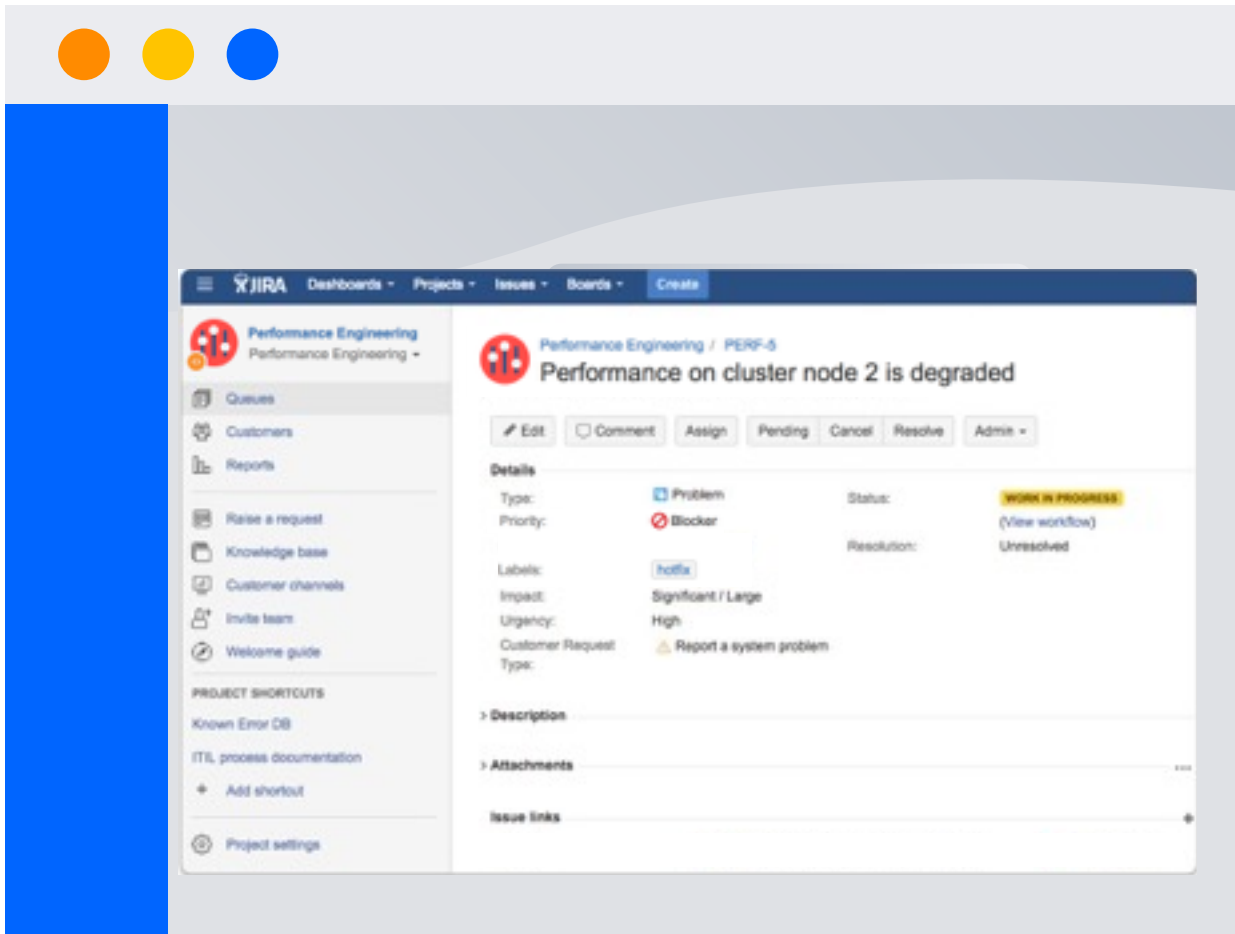
What if an incident occurs?



Devs are notified



JIRA Service Desk



Sally
Developer

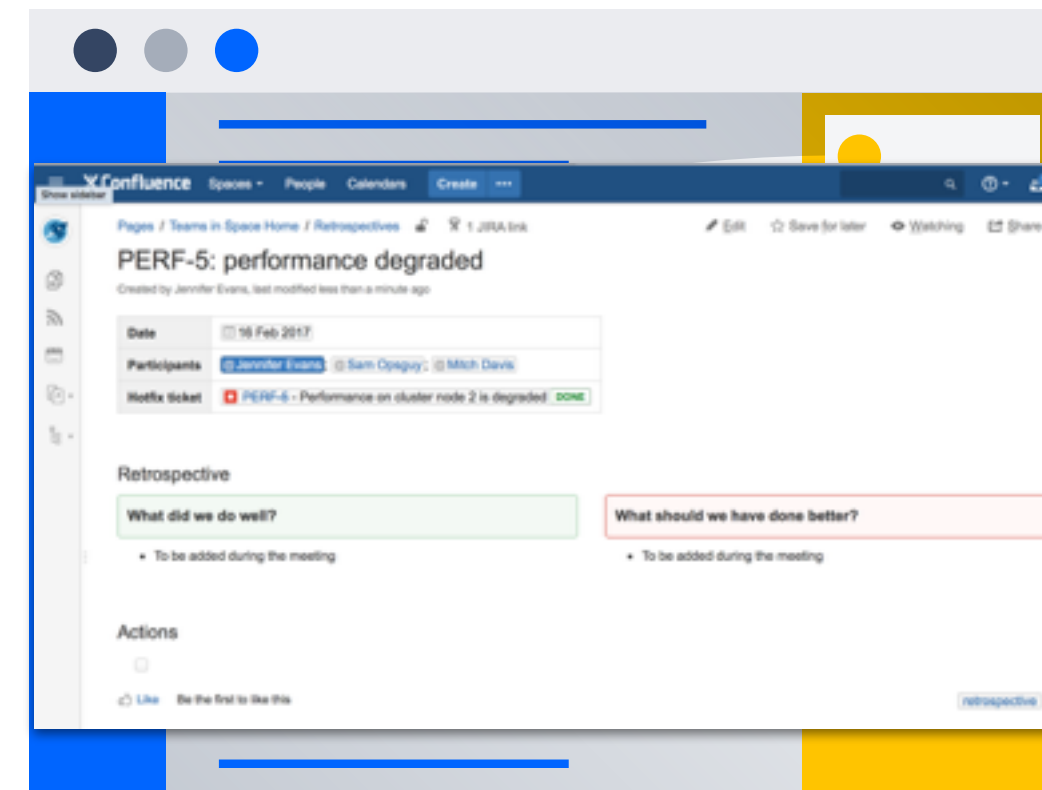


Swarming begins



Incident post-mortem

Confluence



New in JIRA Software Server/Data Center



Backlog for Kanban projects

The space and freedom to plan and prioritise, and less clutter on boards when you're in the zone

New in Bitbucket Server/Data Center



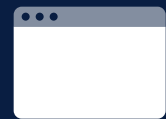
Iterative review

Bring greater speed and confidence to code reviews



Repository links

Give core teams and contributors easy access to the tools that complement the code



In-browser editing

Make simple changes on the fly



Verify committer hook

Establish trust in code authenticity without slowing teams down

New in Bamboo



Config as code

Build configuration that can be composed and versioned by the developers, who know their code best



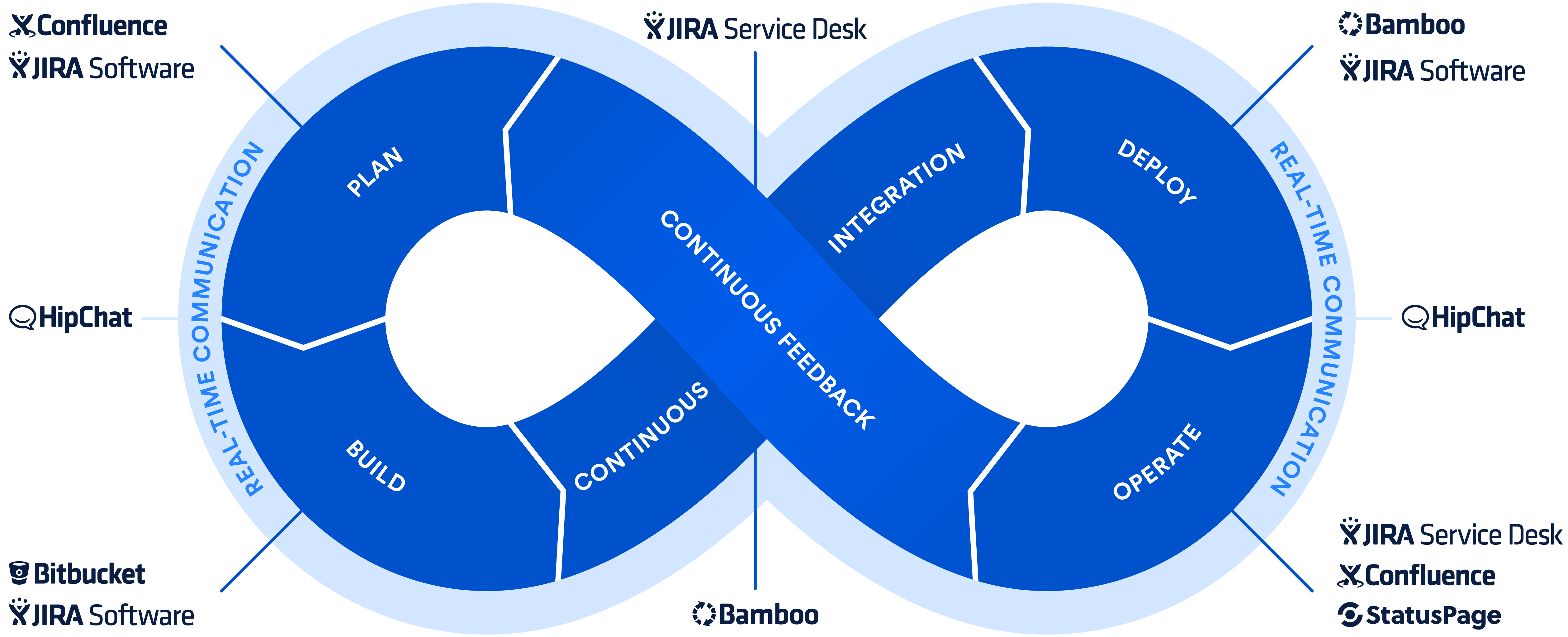
Pull request aware builds

Build the right set of changes at review time



iOS and MacOS build support

Use fastlane to automate building and deploying your apps



INTEGRATIONS

**Atlassian
integrates with
key DevOps tools**



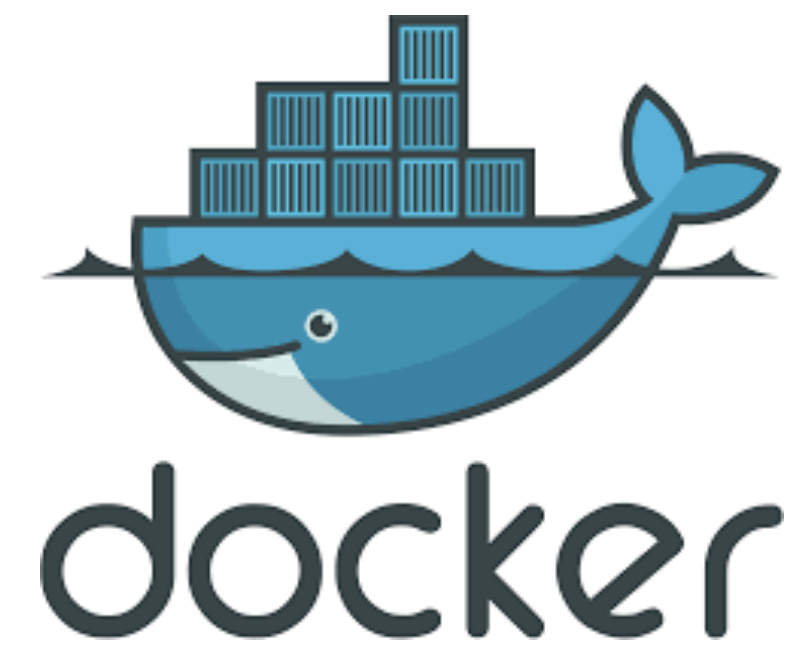
INTEGRATIONS - CODE QUALITY, COMPLIANCE AND SECURITY

sonarqube 

 **BLACKDUCK**

[:] SourceClear

INTEGRATIONS - CONTINUOUS INTEGRATION



INTEGRATIONS - CONTINUOUS DEPLOYMENT



INTEGRATIONS - APPLICATION MONITORING AND MANAGEMENT

APPDYNAMICS

(x) matters®

splunk® >

pagerduty

Atlassian Partners

You don't have to
go it alone



Atlassian Partners



Hands-on assistance



**Solution
customization**



**Best practices &
experience**

atlassian.com/partners

ATLASSIAN FOR DEVELOPERS

**Atlassian is the
culture and collaboration
layer of DevOps.**

SUCCESSING WITH DEVOPS

atlassian.com/devops



Atlassian Summit

Q & A



ROGER BARNES | PRODUCT MANAGER | ATLASSIAN | @MINDSOCKET