Continuous Delivery
A methodology that allows you to deliver software faster and with lower risk

Too often, agile processes stop with Continuous Integration (CI) practices. However, CI does NOT equal Continuous Delivery (CD).
True Continuous Delivery is present at every stage of the lifecycle.

DZone surveyed **500+** IT professionals...revealing insights into devops practices and CD adoption.

Three key traits are evidence that actual Continuous Delivery practices are in use:[1]

Software is in a shippable state every time new code is added

Stakeholders have instant visibility into production readiness

Teams can perform push-button software deployment on-demand

HOW MANY PROFESSIONALS DO CONTINUOUS DELIVERY?

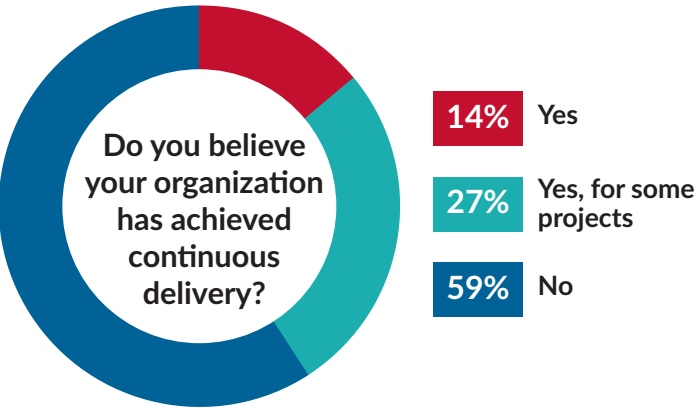**41%** **THINK** they are using Continuous Delivery

26% **ACTUALLY** are performing basic CD

Only **8%** **MEET** the three key traits

Achieving Textbook Continuous Delivery is Hard
For many, Continuous Delivery is still a work in progress.

Do you believe your organization has achieved continuous delivery?

14% Yes
27% Yes, for some projects
59% No
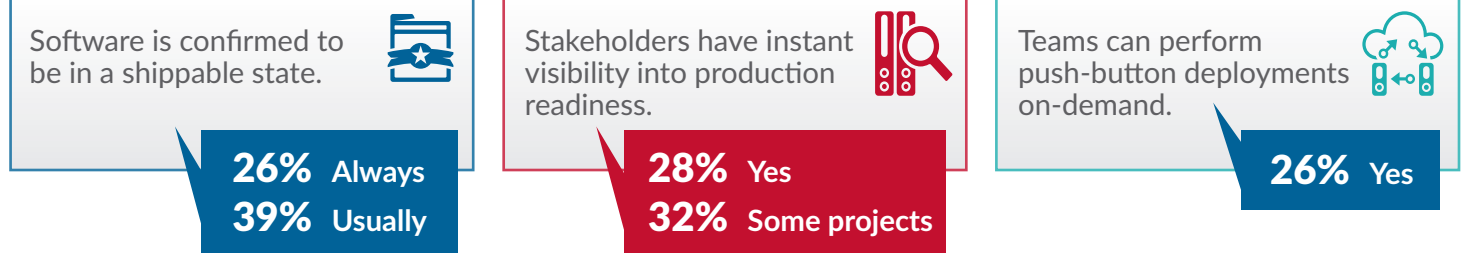
Respondents were considered to be using basic CD practices if they:
> Frequently confirm software updates are shippable
> Use version control, issue tracking and resource monitoring
> Have automated >30% of infrastructure changes

Only 26% of all respondents met these criteria.

When respondents were further filtered by the three key traits of CD, only 8% met all three criteria.

Software is confirmed to be in a shippable state.
26% Always
39% Usually

Stakeholders have instant visibility into production readiness.
28% Yes
32% Some projects

Teams can perform push-button deployments on-demand.
26% Yes

Continuous Delivery Still Primarily in Development
Continuous Delivery isn't a hard sell for developers.

Is continuous integration extended into production deployment process?
62% NO **38%** YES
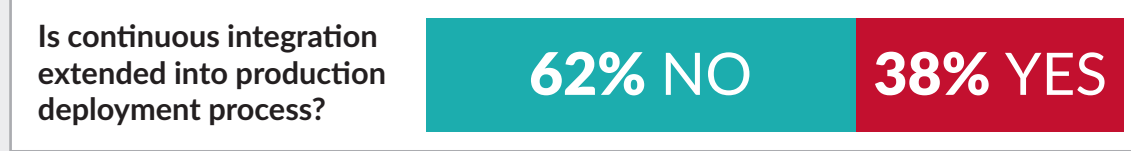
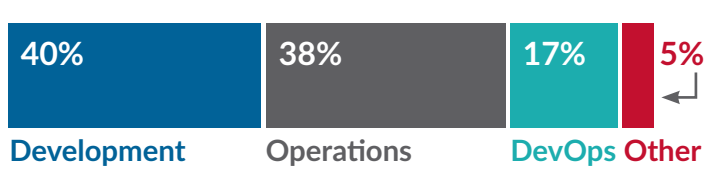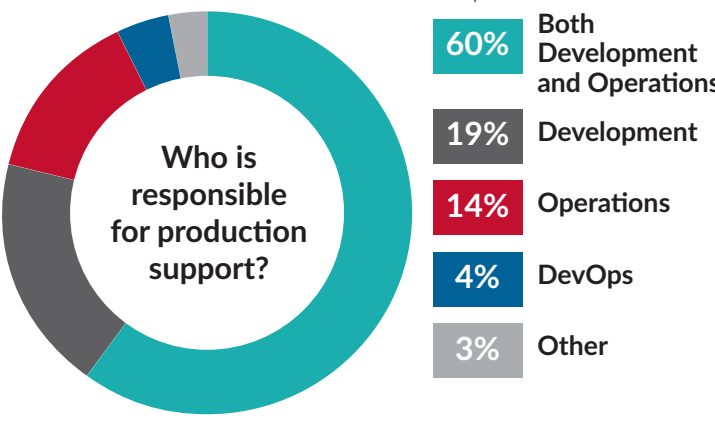## The DevOps Way is Catching On
Continuous Delivery supports the DevOps philosophy of better collaboration across the organization.

30% have an officially designated DevOps team.

60% said development and operations were responsible for production support.

Who Actually Deploys Code?

40% Development
38% Operations
17% DevOps
5% Other

Who is responsible for production support?

60% Both Development and Operations
19% Development
14% Operations
4% DevOps
3% Other

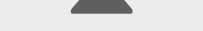## Treating Infrastructure as Code: The DevOps Way
The good news:

49% use an infrastructure configuration management tool.

puppet labs

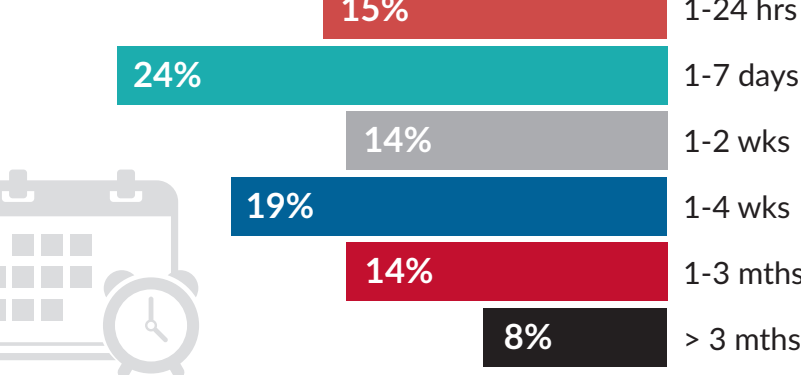**48%** use a version control system for infrastructure changes and system definitions.

The not-so-good news:
73% of respondents still have to use manual scripts for at least half of their infrastructure changes.

ARE ENVIRONMENT CHANGES, INFRASTRUCTURE CHANGES OR SYSTEM DEFINITIONS CHECKED INTO VERSION CONTROL?
48% Yes
52% No

Deployment Times Leave Room for Improvement
Once a code change is committed, how long does it take to get to the customer?

6% < 1 hr
15% 1-24 hrs
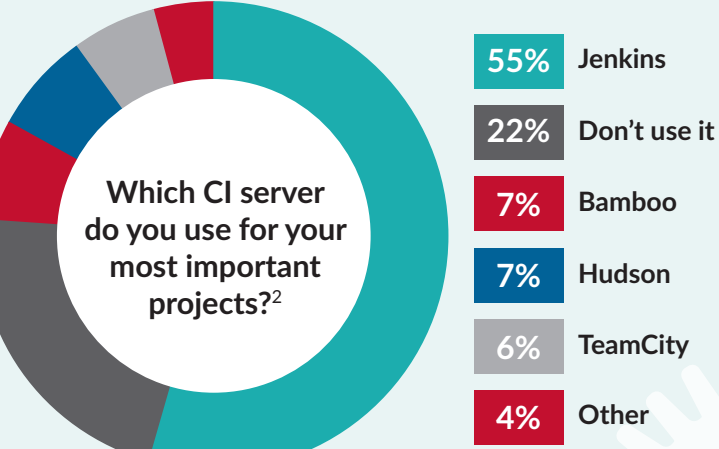24% 1-7 days
14% 1-2 wks
19% 1-4 wks
14% 1-3 mths
8% > 3 mths

Culture and Lack of Time: The Major CD Barriers

66% Lack of Time

66% Company Culture

Which CI server do you use for your most important projects?[2]

55% Jenkins
22% Don't use it
7% Bamboo
7% Hudson
6% TeamCity
4% Other

Continuous Delivery
Starts with Continuous Integration

CI is a jumpstart to enabling CD and is definitely an important component to achieve CD. Teams already using Jenkins for CI can more readily extend their software delivery processes through to CD.