



DevOps Challenges and Version Control: The 2018 Report

Data Reveals the Correlation Between
Effective VCS and Achieving DevOps

November 2018



There are numerous reports about the ways DevOps can align development and operations – while fostering the necessary framework and culture for agility in the production pipeline. DevOps can remove silos in organizations by ensuring stakeholders have constant access and participation in software production pipelines.

But putting DevOps into practice is an ongoing experiment for many organizations. Developers often struggle with continuous integration/continuous delivery (CI/CD) performance, testing delays, and other bottlenecks.

Version control software (or VCS) has emerged as a way to solve these DevOps challenges. For many companies, the right version control tool is a critical foundation for achieving DevOps at scale.

Executive Summary

“Without version control, DevOps at scale is chaos... When you have many developers and content, the VCS gives you the single source of truth.”

***–Tim Russell,
Chief Product
Officer, Perforce***

DevOps.com produced a survey to learn how organizations’ development teams rely on their version control software for successful DevOps implementation. Perforce Software sponsored the study.

Most respondents said their organizations began their DevOps journey more than a year ago.

So how does version control fit into DevOps? “Without version control, DevOps at scale is chaos,” says Tim Russell, Chief Product Officer for Perforce. “Everybody is looking for a faster development cycle pace, and they look to DevOps as a means to get there. When you have many developers and content, the VCS gives you the single source of truth.”

The survey confirmed that the right version control software plays an integral role in helping organizations improve DevOps, and more specifically, their software development.

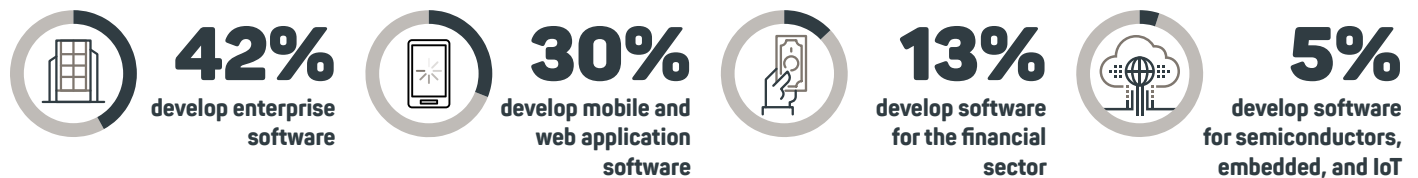
Survey Community

In total, 467 people completed the survey, which had 16 questions. The people who answered the survey are in roles ranging from DevOps Engineer and QA Analyst to CTO. The most common job titles were identified as:

- DevOps Engineer (21%)
- Software Developer (13%)
- IT/Software Architect (12%)

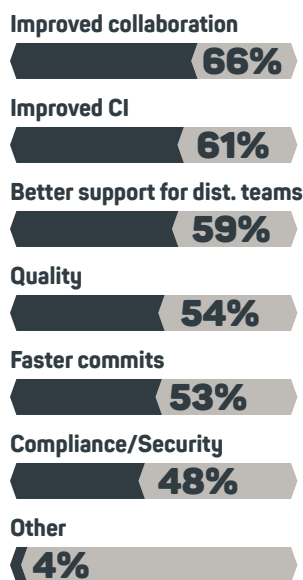
All questions and answers are provided at the end of this report.

Of the 467 respondents:



Why Is VCS Important?

Most important features of your current VCS



DevOps.com asked respondents to select from a list of features they wanted from VCS, as well as desired improvements. Not surprisingly, the majority of respondents (54%) said quality is important.

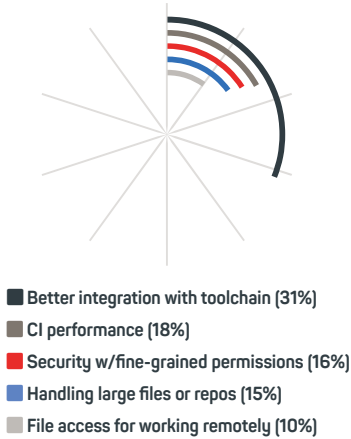
Compliance and security was also highly rated as something people want in a VCS – with 48% of respondents saying it’s important. People expect their VCS to improve DevOps, protect intellectual property, and help to promote compliance with security policies, regulations, and internal engineering standards.

Desired Benefits of VCS

Respondents also expect a VCS to improve various aspects of their DevOps process. Desired benefits of VCS include:

- Improved collaboration (66%)
- Improved continuous integration (CI) (61%)
- Better support for distributed teams (59%)
- Faster commits (53%)

Desired VCS improvements

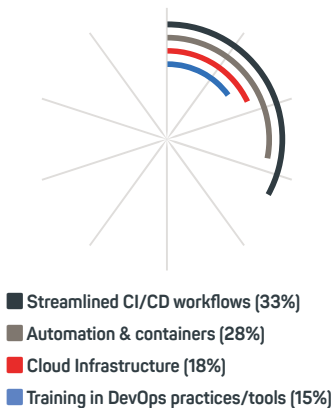


Desired Improvements for VCS

The survey also reported gaps in respondents' current VCS solution. Desired VCS improvements included:

- Better integration with the software development toolchain (31%)
- Improved CI performance (18%)
- Security with fine-grained permissions (16%)
- Handling large files or repos (15%)
- File access for teams in multiple locations or working remotely (10%)

Software development priorities for VCS



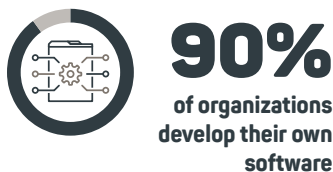
Among the software development priorities the respondents selected, streamlining CI/CD workflows topped the list (33%). Improved CI/CD is a key VCS feature. Other priorities included:

- Automation and containers (28%)
- Cloud infrastructure (18%)
- Training on DevOps practices and tools (15%)

Version control software is critical for achieving DevOps. And according to the survey, the features that development teams consistently look for show how instrumental the right VCS can be.

“Overall, the data points demonstrate that developers understand the challenges in DevOps, and the potential solutions from VCS today,” says Chuck Gehman, Technical Marketing Engineer at Perforce.

All Companies Are Software Companies



In the DevOps survey, 90% of respondents said their organizations develop their own software for internal or external use. This highlights the importance of software development in every organization. And it reiterates the importance of VCS and DevOps.

A good relationship between VCS and DevOps can break the silos that often exist between development teams, operations, and other stakeholders in the production pipeline.

“Every company is a software company. This means everybody has some competitive advantage or disadvantage tied to how well they do their software development,” Russell said. “And if you don’t have a version control system to manage that in a fast-paced way, then you’re not going to be competitive.”

VCS Solves the Challenges of DevOps

Common DevOps challenges

CI/CD performance issues

63%

Prioritizing my backlog and development projects

57%

Managing build artifacts and containers

47%

Managing projects, repos, and permissions

46%

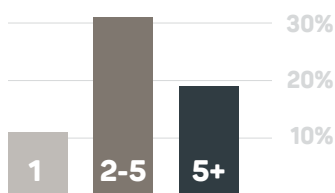
Slow commits and merges

27%

Working with open source software

27%

Software commits/day



We listed numerous potential challenges people may encounter in the DevOps workflow. Most of the problems respondents identified can be solved with the right version control software. For example:

- Continuous integration/continuous delivery performance issues (63%)
- Prioritizing my backlog and development projects (57%)
- Managing build artifacts and containers (47%)
- Managing projects, repos, and permissions (46%)
- Developer complaints about slow commits and merges (27%)
- Working with open source software (27%)

“These challenges are definitely things that the right VCS addresses,” says Gehman. “For example, at one time, the majority of transactions being served by VCS systems were user requests, such as cloning and pushing. Today, it is automation for CI.”

Deploy Fast or Die

More than 60% of the respondents said the developers at their organization make at least one software commit per day. Respondents said each developer at their company makes:

- 1 commit per day (11%)
- 2-5 commits per day (31%)
- More than 5 commits per day (19%)

At some organizations, this means there are 1,000s of commits happening simultaneously each day. And the largest organizations have tens of thousands of commits happening concurrently. In order to maintain that development cycle, organizations need a VCS that can keep up.

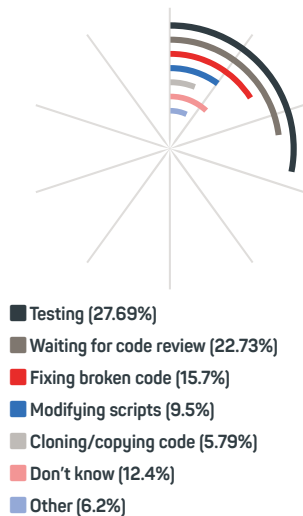
“The goal of Agile and DevOps is delivering value to the customer more quickly. You don’t want your developers sitting around waiting for builds to happen.”

A key practice in DevOps is continuous integration. Giving developers real-time feedback improves quality and empowers developers to own continuous integration. But this can cause bottlenecks on the VCS server – both in processing and I/O.

If your version control system isn’t fast and high-performing, it won’t be able to keep up. Then, you’re facing three equally unappealing options: invest in additional infrastructure, only support local builds with local teams, or slow your CI/CD pipeline.

“The goal of Agile and DevOps is delivering value to the customer more quickly,” Gehman says. “You don’t want your developers sitting around waiting for builds to happen. The whole reason you want to “shift left” is to achieve better quality, and to ship with greater frequency. Having high-performance automation is essential if you want to get to five commits per day.”

Biggest CI bottlenecks



Remove Bottlenecks

In the survey, we asked respondents where in the CI process they encounter the most delays.

People identified testing and waiting for code reviews as the two things that slow them down the most:

- Testing (28%)
- Waiting for code reviews (23%)

In many ways, time spent waiting for code reviews is a worst-case scenario for developers. They have to wait for feedback and reviews about their code before they can move on to their next task. If they start working on the next task while waiting for results, productivity can be negatively impacted by context-switching.

“The most expensive issue with development is non-optimal productivity,” Gehman says.

The right VCS combines speed and performance with built-in code review applications that alleviate the stress of testing and waiting.

Where does your development team work?



64%
are based in multinational locations



29%
all work in the same office



7%
work remotely

Organizations store their production-grade assets in:

Their version control system
28%

Artifactory
23%

Nexus
18%

A file system
16%

A separate VCS used specifically for binary assets
8%

One VCS for DevOps Around the World

We asked respondents about how and where their software developers were physically located. Their responses showed that development teams are often separated by long distances and are based in different time zones.

- 64% of respondents are on development teams that are based in multinational locations.
- 7% of respondents work remotely.

On a geographically distributed enterprise team, for example, there might be 100 developers in Minneapolis, 100 in San Francisco, and 100 in Bangalore. “The right VCS can intelligently distribute code so those developers all feel like they’re working on the same server,” Gehman says. “They are always confident that the work that they’re checking out is the correct version.”

These results underscore how VCS can improve collaboration and a DevOps culture among teams located in different locations. Using the right version control tool gives developers instant and constant access to contribute to a centralized production pipeline – regardless of their location.

Beyond Source Code

According to responses from the survey, organizations have varying opinions on where they prefer to store production-grade artifacts. For example, organizations store their production-grade assets in:

- Their version control system (28%)
- Artifactory (23%)
- Nexus (18%)
- A file system (16%)
- A separate version control system used specifically for binary assets (8%)

“This data reflects a very current and important requirement. The idea is that everybody agrees that the code is ‘perfect,’ and they opt to make it available with the binary. Keeping artifacts in the VCS is super convenient,” Gehman said. “All the scripts you’re writing for Jenkins (or whatever builds) and your other testing tools only have one system to interface with.”

Regardless of where you prefer to store artifacts, the data makes one thing clear. Being able to access artifacts and put them in a centralized VCS – which can be accessed by developers, regardless of their physical location – is essential.



81%
use Git as an
inexpensive
source for VCS

Scaling Git

Git is a great resource for developers and small development teams. It’s free and widely popular. Local operations are fast, and it works well for distributed teams. Given how ubiquitous Git is, it’s not surprising that 81% of the respondents’ organizations use Git as an inexpensive source for VCS.

Once Git is adopted, however, some of its limitations become apparent. “With Git, one of the main attractions to developers is all the open source tooling that the community has built around it. But pretty soon the DevOps teams says, ‘Oh, now I have to make this all work at scale,’” Russell said.

“A lot of the responses to this survey point out the challenges that organizations have with the VCS they choose,” Russell said. “Enterprise strategies are emerging that give developers the Git interface they’re familiar with. But they’re also getting scalability, global replication, and high-performance for moving files and performing builds. These are things you can’t get from an open-source Git server.”

Do You Need a New VCS?

Reasons for migrating to a new VCS

Scalability issues



License costs are too high



Poor cross-platform integration



Doesn't support DevOps



Too slow



Almost half of the respondents surveyed plan to replace their version control software. Their reasons for looking for a new VCS include:

- Scalability issues (18%)
- License costs are too high (18%)
- Poor cross-platform integration (16%)
- Doesn't support DevOps (14%)
- Too slow (9%)

Version control systems are tightly integrated in organizations’ development. So although migrating to a new VCS can seem challenging, the alternative may be worse.

“When you pay for something that is enormously expensive to license, and on top of that, it has these productivity issues, then you are actually hurting your full development effort,” Gehman says.

Teams who have challenges with their VCS but don't migrate end up paying for a tool that doesn't scale. They fail to achieve DevOps by using technology that slows their team, and is unable to integrate their other tools.

The right version control system can improve your scalability, performance, and CI/CD pipelines.

Conclusion

“Companies can make dramatic improvements in their efficiency by improving what they get from their VCS.”

Some observers might conclude VCS is an area where all the problems have been solved. But the data underscores the importance of CI/CD performance, managing permissions, testing, and IDE integration. “This makes it apparent that, just like everything else in DevOps, companies can make dramatic improvements in their efficiency by improving what they get from their VCS,” Gehman says.

There's a direct connection between an organization's version control system and their ability to achieve DevOps. Organizations who struggle to achieve DevOps may also struggle with:

- Continuous integration/continuous delivery performance issues
- Managing projects, repos, and permissions
- Managing build artifacts and containers
- Developer complaints about slow commits and merges
- Working with open source software

Since the right VCS can alleviate these challenges, the data shows that VCS can be a critical foundation for DevOps.

The Questions

DOES THE ORGANIZATION YOU WORK FOR DEVELOP THEIR OWN SOFTWARE FOR EITHER INTERNAL OR EXTERNAL USE?

Yes - 90.27%

No - 9.73%

WHAT TYPE OF SOFTWARE DOES YOUR COMPANY DEVELOP?

Enterprise software - 42.06%

Mobile and web applications - 30.56%

Financial systems - 12.70%

Semiconductors, embedded, and IoT - 5.16%

Video games and virtual reality - 0.79%

WHICH OF THE FOLLOWING BEST DESCRIBES YOUR COMPANY'S WORK ENVIRONMENT?

Teams are located in multiple offices around the world - 63.49%

All employees work in the same office - 29.37%

All employees work remotely - 7.14%

MY COMPANY BEGAN ITS DEVOPS JOURNEY:

> 5 years ago - 21.83%

2-5 years ago - 26.19%

1-2 years ago - 25.79%

< 1 year ago - 15.08%

Haven't started yet - 11.11%

IF YOU ARE DOING CI, WHAT IS THE FREQUENCY OF COMMITS PER DEVELOPER?

Several times a week - 26.03%

1 per day - 11.16%

2-5 per day - 30.99%

More than five per day - 18.60%

Other - 13.22%

MY TOP DEVOPS CHALLENGES ARE:

Continuous Integration/Continuous Delivery (CI/CD) performance issues - 63.10%

Prioritizing my backlog and development projects - 57.14%

Managing build artifacts and containers - 46.83%

Managing projects, repos, and permissions - 45.63%

Working with open source software - 27.27%

Developer complaints about slow commits, merges, etc. - 26.98%

WHICH OF THE FOLLOWING VERSION CONTROL SOFTWARE DO YOU CURRENTLY USE?

Git - 80.58%

SVN - 27.27%

TFS - 19.01%

VSTS - 11.57%

IBM Rational ClearCase - 9.92%

Perforce Helix Core - 6.20%

Hg - 4.13%

Our own internally developed solution - 4.13%

We don't use version control - 1.24%

Don't know exactly but we do use one - 0.41%

WHERE IN THE CI PROCESS DO YOU ENCOUNTER THE MOST DELAYS?

Testing - 27.69%

Waiting for code reviews - 22.73%

Fixing broken code - 15.70%

Don't know - 12.40%

Modifying scripts - 9.50%

Other - 6.20%

Cloning and copying code - 5.79%

WHAT ARE THE MAIN BENEFITS OF YOUR CURRENT VERSION CONTROL SOFTWARE?

Collaboration - 65.70%

Continuous Integration (CI) - 61.16%

Support for distributed teams - 58.68%

Quality - 53.72%

Fast commits - 52.89%

Compliance/Security - 47.52%

Other - 3.72%

IF YOU COULD IMPROVE ONE THING ABOUT YOUR CURRENT VERSION CONTROL SOFTWARE, WHAT WOULD IT BE?

Better integration with the software development toolchain - 30.58%

CI performance - 17.36%

Better security with fine-grained permissions - 15.70%

Handling large files or repos - 15.29%

Other - 11.16%

Faster file access for teams in multiple locations or working remotely - 9.92%

WHICH OF THE FOLLOWING VERSION CONTROL INTEGRATIONS ARE MOST IMPORTANT FOR YOUR DEVOPS WORKFLOW?

Build runner (e.g., Jenkins, TeamCity, Microsoft) - 73.14%

Issue tracking (e.g., Jira) - 60.74%

IDEs (e.g., Visual Studio, Eclipse, JetBrains) - 57.44%

Testing tools - 52.89%

Container management (e.g., Docker, Kubernetes) - 45.87%

Static code analysis tools - 41.32%

Other - 1.65%

IF YOUR TEAM OR COMPANY IS THINKING ABOUT MIGRATING TO A DIFFERENT VERSION CONTROL SYSTEM, WHAT ARE THE PRIMARY REASONS?

We are not thinking about migrating - 55.93%

License costs are too high - 18.22%

Scalability issues - 17.80%

Poor cross-platform integration - 16.10%

Doesn't support DevOps - 13.98%

Vendor no longer supports it - 10.59%

Too slow - 8.47%

Other - 5.51%

WHERE DO YOU CURRENTLY STORE YOUR PRODUCTION-GRADE ARTIFACTS?

In the version control system - 28.39%

Artifactory - 22.46%

Nexus - 18.22%

File system - 16.10%

In a separate version control system used specifically for binary assets - 8.05%

Other - 6.78%

WHAT IS YOUR TEAM OR COMPANY'S TOP SOFTWARE DEVELOPMENT PRIORITY?

Streamlining CI/CD workflows - 33.05%

Automation and containers - 28.39%

Cloud infrastructure - 18.22%

Training in DevOps practices and tools - 15.25%

Other - 5.08%