

PERFORCE



Version Everything for
**Outstanding DevOps
Performance**



Over the last five years, the DevOps movement has brought Development and Operations teams together to meet the common goal of releasing better-quality software faster.

A key to DevOps success is the adoption of and adherence to a much advocated best practice: version everything.

““ You should use [source code] tools to version all service artifacts, including environment configurations. ””

— Amy DeMartine and Kurt Bittner, Forrester Research

VERSION CONTROL BOOSTS PERFORMANCE AND RESULTS

Versioning everything in both Dev and Ops is important and can lead to impressive results. Organizations that practice a DevOps approach with version control across the value chain will see improvements throughout their processes, leading to better and faster software products and services. Such organizations will see dramatic results overall as well.

In Perforce's October, 2014 webinar DevOps Done Right: The How and Why of Versioning Environment Artifacts, Author Gene Kim shared some interesting findings about his research and benchmarking of over 14,000 organizations. He and Continuous Delivery expert Jez Humble found that high-performing organizations:

- Do **30X more frequent** deployments
- Get from code commit to running in production **8000X faster**
- Are **2X more likely to succeed** with no service, security, or compliance issues, aka **50 percent lower change failure rates**
- **Recover 12X faster** from any issues

And beyond these stellar IT-centric metrics, these organizations:

- Were **2X more likely to exceed goals** for profitability, market share, and productivity
- Had **50 percent higher market capitalization growth** over three years (among publicly traded companies evaluated)



HOW DID THEY DO IT?

A number of factors separated high-performing organizations from the others, but when asked, “What was the one technical practice or cultural measure that was the best predictor of all the measures that we talked about: mean time to prepare, change success rate, lead time and deployment frequency?” the answer was:

Version control of all production artifacts by Operations

In fact, version control was a top predictor of both IT performance and organizational performance.

Top Predictors of High IT and Organizational Performance

1. Version control of all production artifacts
2. Continuous integration and deployment
3. Automated acceptance testing
4. Peer review of production changes
5. High trust culture
6. Proactive monitoring of the production environment
7. Win-win relationship (and outcomes) between Dev and Ops

WHY VERSION CONTROL MATTERS

“Infrastructure environments will change over time. To understand how these changes have affected the application, it is important to keep track of what these changes are and why they were made. Just like application source code, you should version these environments to enable rollback to previous versions, track where issues were introduced, or simply to keep as a record of the changes to help with future environment configuration.”

— Amy DeMartine and Kurt Bittner, Forrester Research

4 REASONS WHY VERSION CONTROL IS CRUCIAL TO BOTH DEV AND OPS



1 Both need the same single source of truth. When Ops uses the same version control system as Dev, anybody can reproduce the production environment based only on what is in version control. Dev and Ops can depend on version control to ensure that changes to the code and environment are continuously being integrated and deployed into the production environment, resulting in high IT and organizational performance.

2 Far more configurable settings can go wrong in the environment than in the code. The environment includes everything—except the code—such as database, OS, and configuration settings. To control this, all production configurations should be checked into version control, serving as the single source of truth that everyone can depend on for all environments (e.g., Dev, Test, and Prod).

3 Continuous delivery requires developers to create production-like environments on-demand via an automated build process. To ensure repeatable outcomes, the build mechanism should be checked into version control, as well as all its dependencies. This factor guarantees that anyone can re-create environments based only on what is in version control, as opposed to individual or tribal knowledge, or worse.

4 Version control helps foster mutual accountability and trust between Dev and Ops. If everything goes through the process and version control actually dictates the way development, test, and production environments are built, it creates mutual accountability. Just as version control is necessary to create mutual trust between developers, you need that control in the value stream among Dev, Test, and Ops, which creates accountability and a sense of visibility and trust in the whole value stream.

HOW CAN YOU DO IT?

Part of your DevOps approach, whether in place or just underway, needs to be the use of version control for Ops as well as Dev, and that version control system should be the same for both. Ops should version control all production artifacts, just as Dev version controls everything.

To obtain optimal service delivery, Forrester recommends that you¹ :

- **Optimize on cost** using tools and technologies that have emerged to enable repeatable processes and automatic handoffs as well as reduce the development and maintenance cost of scripting.
- **Attain delivery speed** by automating the handoff between phases and tools across greater parts of the lifecycle until the lifecycle is automated in its entirety.
- **Increase quality** with tools that can help develop and verify that processes are going to produce a quality result.
- **Remove manual processes** that are, by definition, error prone and use tools to automate your processes and achieve repeatable results.

Forrester also recommends designing operability requirements into the service from the beginning and using common tools to facilitate and enhance the processes.

“*Today’s need for speed relies on being able to skip any manual process or finger-pointing and proceed directly to the next phase or troubleshoot a problem. These tools become the foundation for modern service delivery as a single source of truth and trusted enabler of processes. As such, it is important to choose these tools together with development to encourage joint ownership and trust.*”

— Amy DeMartine and Kurt Bittner,
Forrester Research

7 DEVOPS PRACTICES FOR OUTSTANDING RESULTS

The actual practices of Dev and Ops are what lead to outstanding results. Expanding on the list in Table 1, the following approaches set the foundations for higher performance:

1. Version control of all production artifacts

Both Dev and Ops should use version control for everything and share the same single source of truth.

2. Continuous integration and deployment

Check in code every day and check into the trunk every day, as opposed to hanging onto private code branches and integrating only at the end of the release.

3. Automated acceptance testing

Stop the line not only when the build breaks but also when something breaks in an automated user test, an integration test, or a system test. This step keeps things in an always-deployable state.

4. Peer review of production changes

Use peer reviews for better quality based on their familiarity, shared goals, and mutual accountability, as opposed to external change approval, such as a change advisory board.

5. High trust culture

This outcome and practice result from a single source of truth, peer reviews, and shared goals.

6. Proactive monitoring of the production environment

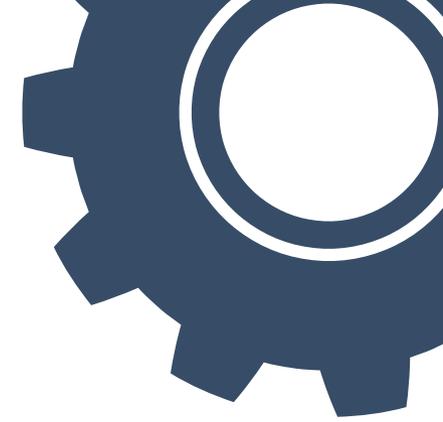
Monitor and communicate across the teams so everyone can see, understand, and affect end results and customer utilization.

7. Win-win relationship (and outcomes) between Dev and Ops

This approach counters the learned behavior that deployments hurt. By deploying code into production every day, you can change lives in Operations. Deployments don't have to be done at midnight on Friday with Ops working all weekend to get things running. When Ops employees are working the same hours as Dev, there is a sense of teamwork and joint accomplishment.

“ As a lifelong Ops practitioner, I know we need DevOps to make our work humane. In the past, I've worked every holiday, on my birthday, my spouse's birthday, and even on the day my son was born. ”

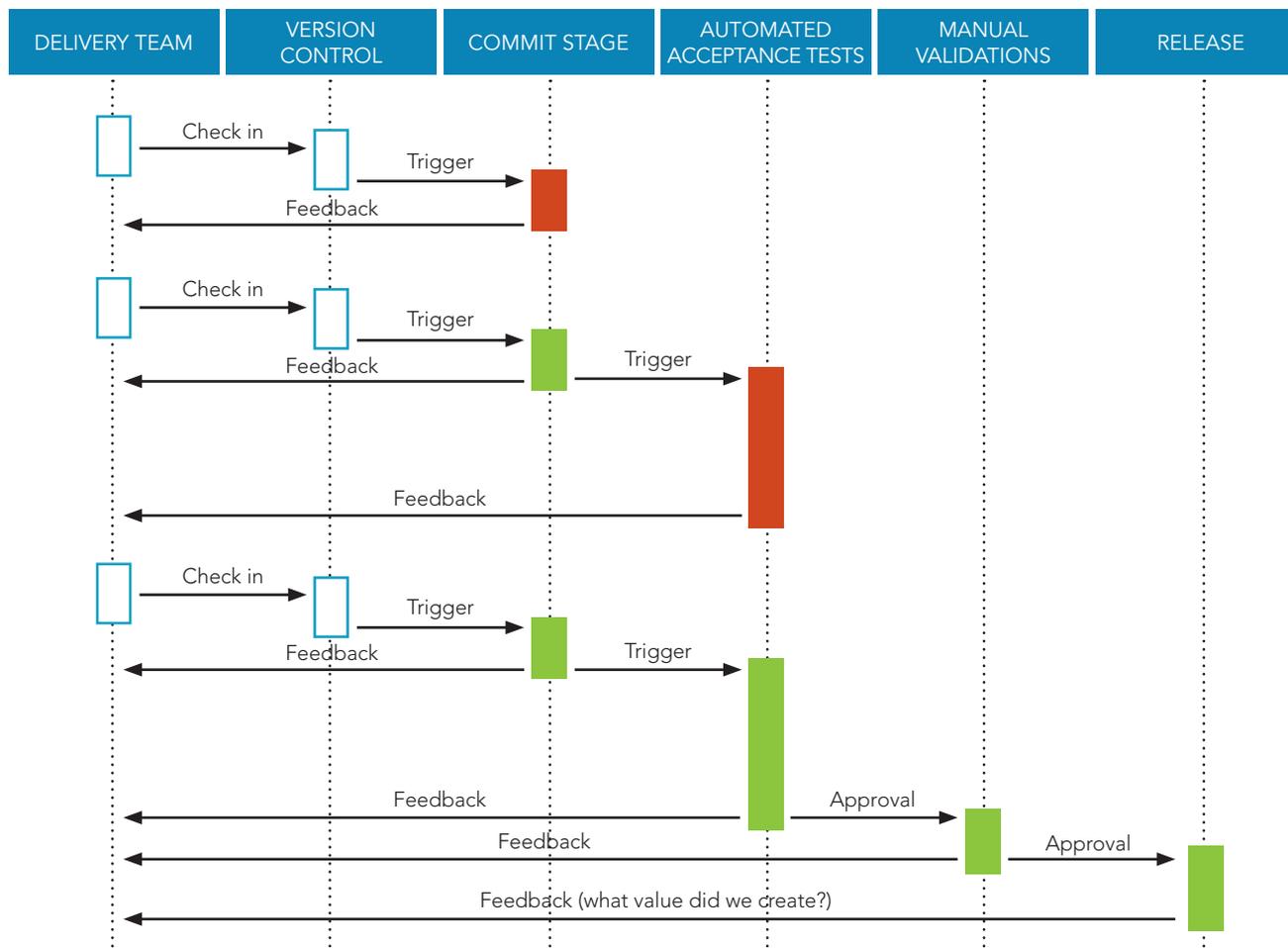
— Nathan Shimek, Engineering Manager,
New Context



WHAT GOOD LOOKS LIKE

The figure below shows what an ideal deployment pipeline with version control looks like. The only way things move along and get from one step to the next is through version control, the vehicle used to go from development to test and from test to production.

Version Control in a Deployment Pipeline²



“Continuous service delivery with frequent releases can only be achieved with seamless collaboration between application development teams and operations.”

— Amy DeMartine and Kurt Bittner, Forrester Research



Get the Forrester Report

info.perforce.com/service-delivery-forrester-report.html

² Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation, Jez Humble and David Farley, August 6, 2010.



GET STARTED

DevOps is having a big impact on organizations around the globe.

To make that impact even more profound, organizations should make sure they version everything –all elements across Dev and Ops, including environment artifacts. When Ops uses the same version control system as Dev, anybody can reproduce the production environment based on what’s in version control. Dev and Ops can depend on version control to ensure that changes to the code and environment are continuously being integrated and deployed into the production environment, resulting in high IT and organizational performance.

Having a single source of truth is essential and fostering collaboration and trust between Dev and Ops teams is fundamental to success. But trust isn’t just about trusting people. The success of any tool in an organization is also related to how much you can trust the tool. Employee happiness affects product quality. An unreliable tool can impact employee happiness. You want a solution provider you can trust with your data and IP. You want to ensure that you can version everything and maintain a single source of truth and trust.

ADDITIONAL RESOURCES

WEBINAR DevOps Done Right: The How and Why of Versioning Environment Artifacts, featuring Gene Kim

info.perforce.com/ondemand-webinar-version-control-devops.html

2014 State of DevOps Report

puppetlabs.com/sites/default/files/2014-state-of-devops-report.pdf

Forrester Report: Gear Up For Modern Service Delivery

info.perforce.com/modern-service-delivery-forrester-report.html

Contact your Perforce sales professional today for more information on how Perforce can help enhance your DevOps efforts to help your organization become an even higher performer, [**sales@perforce.com**](mailto:sales@perforce.com)