

CHRIS SHORT

**WHAT
IS
DEVOPS**

If forced to put it in one sentence,
I would describe DevOps as follows...

Brought to you by

DevOps'ish

Written by Chris Short



DevOps'ish



devopsish.com

What Is DevOps

contents

- 2 Definition**
- 3 Professional Practice**
- 4 Frequent, Continued, and Iterative Improvement**
- 7 High-Velocity Organization**
- 8 Measurable Changes**
- 9 Business Outcomes**
- 10 DevOps Culture**
- 11 What DevOps Isn't**
- 13 Where To Learn More About DevOps**
- 14 Credits**



Let's establish a definition that we can break down into smaller, more consumable pieces

Definition

If forced to put it in one sentence, I would describe DevOps as follows:

DevOps is the professional practice of frequent, continued, and iterative improvements through measurable changes, the goal of which is to become a high-velocity organization thus improving business outcomes.

Etymological Note: DevOps, the word itself, should be used in an active voice. The word itself is the combining of two words Development and Operations. The word itself indicates that there will be a removal of silos as intended throughout this document.

There's a lot to unpack in this definition. Let's dive into specific words and phrases to provide more color and further explain why this is my definition for DevOps.



Photo by [fauxels](#) from [Pexels](#)

Professional Practice

Professional Practice, in the case of DevOps, means that the folks doing the work are, in a similar vein as doctors and lawyers, practitioners. Much like how the underlying principles of medicine and law change over time, technology does too. Technology moves at a much faster rate of change though. Thus it's imperative to continuously hone the methods and processes through which business activity, or at the very least software development and operations, is conducted. This also requires thoughts on scientific methods for measuring systems as inputs impact outputs.



We're in technology, change is constant. Embrace that and use it as an advantage.

Frequent, Continued, and Iterative Improvement

Most DevOps practices embrace [Agile](#) as a method for managing work. This does not mean Agile will always work nor always be the best for every organization. Agile enables features and tools to be iterated on over many releases and developed faster than more traditional methods. When coupled with [Lean thinking](#) business principles, customers

can expect high-quality features, bug fixes, and modifications delivered at regular intervals.

These delivery intervals get shorter in length and more frequent as the organization embraces DevOps practices. Several deployments an hour is entirely realistic for more mature DevOps organizations. At some point along an organization's DevOps journey, a pipeline is built. This pipeline enables automation of changes and modifications to the system through controlled inputs. Whether this is completely automated from development to production in a continuous deployment model or something short of that is ultimately a business decision.

For years, we've been taught that [speed kills](#). While that may be true in a few facets of life, it is not the case in software delivery. DevOps Research & Assessment (DORA) has collected survey data for over five years from organizations and professionals delivering and maintaining software across the globe. In 2018, [Accelerate: State of DevOps Report](#) (ASDOR) described the highest performers as, "excelling at all aspects of throughput and stability." The Elite Performers are delivering software faster with shorter lead times for changes than the rest of the industry.

As counterintuitive as this might sound to some, when it comes to software, smaller, more frequent releases will

improve stability a lot more than a 900 line spreadsheet and detailed planning ever will. If a pipeline allows for safe, rapid, iterative improvements to a system, it becomes increasingly easier to deploy to fix issues that arise as they happen. This often leads to growth in capabilities as more work becomes automated and mind power is allowed to focus on more complex issues.



Photo by [ThisIsEngineering](#) from [Pexels](#)

High-Velocity Organization

A consensus is forming around the idea of organizations being technology organizations first and widget makers second. This means that a component of technology can improve everything between a business' logic and the market itself. Being able to integrate business processes more tightly with existing or new technologies can lead to improved outcomes. Time to market, lead time, mean time to resolution (recovery), change failure rate, and myriad other metrics can be gleaned from day-to-day operations. Focusing on improving the metrics that matter most to your business will lead to discussions about change, automation, and greater collaboration between otherwise disparate groups.

There are many environments and models for achieving higher velocity. No one model is better than another but, one model will be better for an organization and the people within it. Often strict adherence to any one model isn't possible either. If the business is maintaining commercial passenger jets doing multiple transatlantic crossings a day, several deployments an hour to production might not be workable (or wise). The point is, one size or style rarely fits all. Figuring this out and what metrics matter to your business are the pain people frequently described as a "journey."



Photo by [Mike](#) from Pexels

Measurable Changes

Measurable, in the sense of DevOps, can mean any of several different ways to track, measure, and analyze changes to the system. This translates to some disciplines such as version control ([git](#) being the most popular), monitoring, observability, record keeping, and analysis.

As work occurs in the system be it normal operation, routine maintenance, or full system upgrades all changes should be tracked, recorded, and measured. This allows for short or long-term analysis to determine the state of the system at a point in time or throughout normal operations. It's incredibly challenging to fix pain points when data can't back up what the problem is.



Photo by [Anna Shvets](#) from [Pexels](#)

Business Outcomes

Business Outcomes can be defined as top-level organizational goals and/or Objectives and Key Results (OKRs). These are the things that organizational leadership has determined are strategic goals of the business activity. DevOps must facilitate the desired outcomes to maximize the impact of the objectives and goals. By tying the purpose of the technology teams to the goals of the business the table is set for technology to better impact the business.

DevOps Culture

Organizational culture is often the biggest hurdle in achieving higher velocity. There is little the organization can do but change. But, human resistance to change is often very real. One of the most egregious cases of resistance to change was detailed in The Ops Mutiny section of [The Dark Side of DevOps](#). But, fear not, culture changes are happening across the globe every day as more and more organizations realize that the way business has been done is not the way it can continue to be done.

In [The Dark Side of DevOps](#) there is another story of a rebellious gang of developers and researchers I worked with. We used the existing restrictive processes and a team written white paper to accelerate cultural change. It worked. Ensuring the business is sitting at the table with the right members of the technology organization is critical to providing greater success. Iterative changes at an increasing velocity leading to improved business outcomes is the goal.

What DevOps Isn't



Lemi Orhan Ergin

@lemiorhan



Let's clarify:

- * Jira is not Agile
- * Open Source is not Free
- * Using cloud is not DevOps
- * Rituals is not Scrum
- * Developed is not Done

Heart 4,910 5:30 PM - May 1, 2017 · Turkey



Comment 3,687 people are talking about this



- DevOps is not something you can buy.
- DevOps is not a tool.
- DevOps is not using the cloud.
- DevOps is not ping pong tables and beer.
- DevOps is not Agile.
- DevOps is not Kanban.
- DevOps is not one team.
- DevOps is not one person.

Like Obi Won Kenobi talking about Darth Vader, “DevOps is more fluff now than actual outcomes. Its purpose has become twisted.” There is a long list of companies that are more than willing to help organizations embrace a component or two of DevOps. Some of those companies will sell you a tool with DevOps in the name. Others will sell consulting and process evaluation. Regardless, DevOps is not a one-time engagement, a standalone project, or a box that is checked. Much like law and medicine, DevOps professionals are practitioners and continuous learners.

Where To Learn More About DevOps

- DevOps'ish — A weekly newsletter covering DevOps, Cloud Native, Hybrid Cloud, Open Source, industry news, culture, and the ‘ish between.
- DevOps README
- DevOps For Dummies
- The Phoenix Project: A Novel about IT, DevOps, and Helping Your Business Win
- Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations
- DevOps Newsletters: “Continuous learning is a critical part of DevOps. Staying current is imperative. These are DevOps newsletters of note from several well regarded DevOps leaders.”

Credits

A very special thanks to:

- Emily Freeman
- Nathen Harvey
- Kim McMahon

Without your feedback and other works in the DevOps and cloud native spaces, this definition never would have come to mind. I sincerely appreciate your candid feedback and challenging my opinions. This is how we grow, by challenging the norms and finding innovations not just in the technology space but the humans spaces too. Thank you.

Sponsor DevOps'ish

Sponsor DevOps'ish and put your brand in front of thousands of highly skilled operators, maintainers, developers, and leaders from Amazon, Apple, Google, IBM, Intel, Microsoft, Red Hat, many of the Fortune 100, and beyond. Download the DevOps'ish Sponsor Prospectus now.

