

Articles

DevOps Principles

- Overview
- History of DevOps
- Benefits of DevOps
- DevOps Culture
- DevOps Best Practices
- Agile vs. DevOps
 - Always-on services
- DevOps Frameworks
- DevOps Tools

Tutorials

- Automation
- Testing
- Security
- Observability
- Feature Flagging
- Continuous Delivery

Agile vs. DevOps

What are the differences and similarities between agile and DevOps?



TOM HALL

DevOps Advocate & Practitioner

When the agile methodology gained widespread adoption in the early 2000s, it transformed the way we develop software and other products. Yet, within a few years of becoming an industry standard, a critical oversight arose: the processes and requirements of the operations team who deployed and managed software products were left out of the revolution.

This led to [DevOps](#), an approach that aligned development and operations teams. So did DevOps replace agile? Or do they complement each other? It turns out there are both differences and similarities between the two. Agile and DevOps can be both exclusive and inclusive, with both existing within an organization.

What is agile?

[Agile](#) is an iterative approach to project management and software development that focuses on collaboration, customer feedback, and rapid releases. It arose in the early 2000s from the software development industry, helping development teams react and adapt to changing market conditions and customer demands.

In an agile approach, some planning and design is done upfront, but the development proceeds in small batches and involves close collaboration with stakeholders. Changes are incorporated continuously and a usable version of a product is often released quicker compared to products developed through the [waterfall methodology](#). This provides many benefits, with arguably the most important being that if software doesn't meet the needs or expectations of the customer, it can be remediated in real-time.

Agile is a collection of methodologies, not a single approach to development. It is an aggregation of scrum, eXtreme Programming (XP), and other systems of practice that developers used in years prior, and resulted from those practitioners coming together to unify these approaches into a single set of principles. The result of this unification effort was the [Agile Manifesto](#), which consists of 12 principles, based on four core values.



RELATED MATERIAL

Learn about the CALMS framework

[Learn more →](#)



RELATED MATERIAL

Learn about the history of DevOps

[See article →](#)

The Agile Manifesto's four core values

			
Individuals and interactions over processes and tools	Working software over comprehensive documentation	Customer collaboration over contract negotiation	Responding to change over following a plan

Enter DevOps

DevOps is an approach to software development that enables teams to build, test, and release software faster and more reliably by incorporating agile principles and practices, such as increased automation and improved collaboration between development and operations teams. Development, testing, and deployment occur in both agile and DevOps. Yet traditional agile stops short of operations, which is an integral part of DevOps.

The goal of DevOps is to help bring together developers who write application software and operations who run the software in production. Also, to build and maintain the infrastructure where it runs. DevOps replaces the old approach of development teams writing applications then throwing them over the wall to an operations team who deploys and manages the software with minimal visibility into how it was developed. In a DevOps environment, developers and operations teams work side by side throughout the entire process of developing, deploying, and managing applications.

Two common frameworks for understanding DevOps are “Three Ways” and “CALMS”, an acronym for Culture, Automation, Lean, Measurement, and Sharing. Culture refers to the cultural shift where development and operations work more cohesively. Automation increases velocity and ensures higher quality. The lean principles of continuous improvement and embracing failure are the foundation of an experimental mindset. Measure refers to the practice of measuring results to improve processes. Sharing emphasizes the importance of DevOps as a group effort and adopting best practices.

The DevOps Three Ways

		
---	--	---

Systems thinking Understanding that software applications are complex systems	Amplifying feedback loops Improve bidirectional communication between teammates	Cultural change Culture of continuous experimentation and learning
---	---	--

When do agile and DevOps work together?

DevOps can be thought of as an evolution of agile practices, or as a missing piece of agile. It's an effort to take the innovations of the agile approach and apply them to operations processes. At the same time, it's a missing piece of agile, because certain principles of agile are only realized in their most complete form when DevOps practices are employed. For example, there are multiple references to continuous delivery of software in agile documents, but because delivery pipelines encompass operations concerns, continuous delivery is usually regarded as a DevOps practice. Amplifying feedback loops requires improved communication across and between teams. Agile, specifically [scrum](#), helps facilitate this communication through its various ceremonies such as daily standups, planning meetings, and retrospectives.

What are the similarities/differences between agile and DevOps?

- Agile emphasizes collaboration between developers and product management – DevOps includes the operations team
- Agile centers the flow of software from ideation to code completion – DevOps extends the focus to delivery and maintenance
- Agile emphasizes iterative development and small batches – DevOps focuses more on test and delivery automation
- Agile adds structure to planned work for developers – DevOps incorporates unplanned work common to operations teams

The Agile Manifesto explicitly prioritizes individuals and interactions, working software, customer collaboration, and responding to change. These are clearly the same priorities of DevOps but extended beyond the development process and into the management of systems and running applications.

In addition, the [Twelve Principles of Agile Software](#) includes references to DevOps principles. For example, the emphasis on [continuous integration and delivery](#), working in small batches with frequent releases, and using automation are all referenced in the Twelve Principles of Agile Software.

In conclusion... agile AND DevOps

Ultimately the goals of agile and DevOps are the same: to improve the speed and quality of software development, and it makes very little sense to talk about one without the other. Many teams have found agile methodologies help them

tremendously, while others have struggled to realize the benefits promised by an agile approach. This might be for any number of reasons, including teams not fully understanding or correctly implementing agile practices. It may also be that incorporating a DevOps approach will help fill the gaps for organizations that struggle with agile and help them have the success they were hoping for.

Atlassian connects development, IT operations, and agile teams with automation that spans Atlassian products and third-party tools. To learn more visit [Atlassian DevOps](#).



TOM HALL

Tom Hall is a DevOps advocate and practitioner, voracious reader, and amateur pianist.

Among his accomplishments over the past 20 years are certifications from Novell, EMC, VMware, and AWS. He helped organize DevOpsDays in Atlanta in 2016 and in Austin, TX in years since.

SHARE THIS ARTICLE



NEXT TOPIC

[DevOps Engineer →](#)

Recommended reading

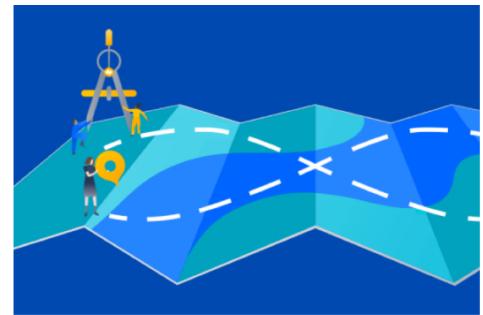
Bookmark these resources to learn about types of DevOps teams, or for ongoing updates about DevOps at Atlassian.



DevOps community



Simulation workshop



Get started for free

[Learn more →](#)

[Learn more →](#)

[Learn more →](#)

Sign up for our DevOps newsletter

Email address

[Sign up](#)

**PRODUCTS**[Jira Software](#)[Jira Align](#)[Jira Service Management](#)[Confluence](#)[Trello](#)[Bitbucket](#)[View all products](#)**RESOURCES**[Technical Support](#)[Purchasing & licensing](#)[Atlassian Community](#)[Knowledge base](#)[Marketplace](#)[My Account](#)[Create support ticket](#)**EXPAND & LEARN**[Partners](#)[Training & Certification](#)[Documentation](#)[Developer Resources](#)[Purchasing FAQ](#)[Enterprise services](#)[View all resources](#)**ABOUT ATLASSIAN**[Company](#)[Careers](#)[Events](#)[Blogs](#)[Investor Relations](#)[Trust & Security](#)[Contact us](#)

English ▾

[Privacy policy](#)[Terms](#)[Impressum](#)

Copyright © 2021 Atlassian

