Jason Schriner

Module 1

6/1/25

DevOps is more than just a set of tools, it is a philosophy meant to close the gap between development and operations teams. The desired result is faster development and release of software. To understand the need for DevOps we need to first understand how software used to be developed and the problems that resulted from this.

Teams used to be separated into developers, those who would write code and develop new features, and an IT or operations team that would maintain the network and deploy new features. These teams had separate responsibilities and they often conflicted. This led to many issues. First developers were only concerned with innovation and a fast release cycle. They would just hand off code to operations teams with no concern for their priorities. The Operations team was more focused on keeping a stable network. Updates brought many challenges; first operations had to figure out how to configure deployments because they received no help from the developers. Secondly updates meant the possibility of introducing problems into the system, which is the opposite of the operations teams' goals. Having these two separate teams whose goals conflicted with each other only led to slower deployments.

A second contributing factor to slow development cycles was the waterfall method. This method involves a set of phases. Each phase must be fully complete before moving on to the next. This style of development was very rigid and did not easily allow for changes. Once a phase was completed, it was difficult and expensive to go backwards and make any changes. A new system was needed. A way for developers to introduce new products and features faster and also respond to customer feedback.

Developers started to figure out that instead of trying to meet six-month deadlines they could more easily break down tasks to hit shorter deadlines. According to Sacolick (2022), a group of developers who were no longer sticking to traditional methods sat down and came up with the Agile Manifesto in 2001. In it, they tried to establish principles to improve the current development process. This manifesto led to the agile method and a system of iterative and rapid development instead of using an outdated waterfall method.

A second approach was also developed. It was described in the book “Lean Software Development: An Agile Toolkit” (Poppendieck, Poppendieck, 2003). The Poppendiecks wrote the book “because they thought the waterfall method was inferior to the lean manufacturing process” (Herranz, n.d.). The lean manufacturing process is focused on bringing a product to market in the cheapest and fastest way possible. This new approach focused on efficiency and speed.

Following the introduction of the Agile Movement and the Lean development principles came the Continuous Delivery (CD) movement. The core goal of this movement is faster software delivery. One way of achieving this was making small frequent updates versus larger ones that would take more time to develop. This allowed teams to quickly address any issues while being able to consistently deliver value to customers. Something else CD introduced was an automated and streamlined process for software deployment. The use of automation helped speed deployment and reduce the chances of human error.

These new ideas and demands for faster and more frequent code deployment meant there needed to be a change to the way developers and operations worked together and how code was deployed to keep up with these other methodologies. This is where DevOps was created. It “changed the culture of software development organizations and evolved into a comprehensive set of practices and principles” (Smith, 2024). DevOps goes beyond improving deployment times; it is meant to improve collaboration between the development and operations teams. To create a cohesive work environment where code can be quickly written and deployed.

The Agile and Lean movements, followed by Continuous Delivery, highlighted the need for faster, more reliable releases. From this DevOps was born. It created a set of practices to streamline deployment while improving collaboration. DevOps allows organizations to deliver code more frequently and respond to customer feedback to stay competitive in the fast-paced world of technology.

References:

David Smith (January, 2024) The Evolution of DevOps. DevOps.com <https://devops.com/the-evolution-of-devops/>

Isaac Sacolick (April, 2022) A brief history of the agile methodology. InfoWorld <https://www.infoworld.com/article/2334751/a-brief-history-of-the-agile-methodology.html>

Mary Poppendieck, & Tom Poppendieck (2003) Lean Software Development: An Agile Toolkit. Pearson Education, Limited.

Raul Herranz (n.d.) What is Lean Software Development? ScrumAlliance <https://resources.scrumalliance.org/Article/lean-software-development>

IAN BUCHANAN. (n.d.). History of DevOps. Atlassian. <https://www.atlassian.com/devops/what-is-devops/history-of-devops>

Continuous Delivery: Basic Concepts and the GitOps Revolution. (n.d.). Codefresh. <https://codefresh.io/learn/continuous-delivery/>