Austen Rhyce Erickson
CSD 380: DevOps
Assignment 1.3: The History of DevOps
1/12/25

In the field of software development, the sub-field specialty of DevOps can be thought of as a set of practices, tools, and philosophies of combining development (Dev) and information technology operations (Ops). The purpose of studying the Dev-Op relationship is to expedite or accelerate the software development lifecycle in a way that preserves quality and reliability of systems. This is desirable because the resulting efficiency leads to higher customer satisfaction, market competitiveness, and improved morale of engineers – and happy engineers perform better. Furthermore, caring about the wellbeing of employees is an ethical practice, reduced waste is important in a world with an increasing demand of scarce computing resources, and efficiency is lucrative. Yay DevOps.

The set of practices, tools, and philosophies that constitute DevOps has a history. Notable trends in DevOps are the Lean movement, Agile manifesto, and continuous delivery movement. The lean movement is a set of principles for improving efficiency, eliminating waste, and delivering value that originated in the field of manufacturing, namely the Toyota Production System. It was around the 2000s when the Lean principles began seeing adoption by the software industry which subsequently led to the Agile manifesto which was created by expert software developers. Seventeen developers came up with four core values and twelve principles which include frequent incremental deployments, an emphasis on customer collaboration, and adapting to changing requirements. This was different from the rigid waterfall method which was the standard at that time. As a result, Agile philosophy was pivotal in the evolution of the standard of software development.

The next evolutionary stage of DevOps as we know it today is called the Continuous Delivery movement and it is one of the cornerstones of DevOps. Continuous delivery refers to

the streamlined automation of building, testing and deploying software. The idea of a “deployment pipeline” emerged in which software can be released reliably to production at any time. Initial pipeline setup can be complex but it is arguably worth it. Along with the deployment pipeline, continuous delivery makes use of version control and various deployment strategies such as canary deployment which slowly shifts production traffic from the old version to the new one, and a blue-green deployment in which both the old and new versions run simultaneously until there is confidence that the old version can be shut off.

New initiatives in the DevOps landscape include Infrastructure as Code (IAC) which entails keeping deployment requirement details within the codebase itself, and Site Reliability Engineering (SRE) which are specialized DevOps roles dedicated to monitoring availability, performance, and reliability in a way that treats operations like a software problem. As one can see, DevOps is a continually evolving field that shapes how software is built and delivered. By understanding Lean, Agile, and Continuous Delivery, we gain a historical context for where DevOps is headed. It will be interesting to see how recent trends in artificial intelligence use may be applied to the field of DevOps.

Sources:

Kim, Gene, et al. *The DevOps Handbook, Second Edition*. I

Scrum Alliance. “The Key Values and Principles of the Agile Manifesto [0.25 SEU].” *Resources.scrumalliance.org*, resources.scrumalliance.org/Article/key-values-principles-agile-manifesto. T Revolution, 30 Nov. 2021.

8 Deployment Strategies Explained and Compared. 25 Jan. 2023, www.apwide.com/8-deployment-strategies-explained-and-compared/.

“Infrastructure as Code : Best Practices, Benefits & Examples.” *Spacelift*, spacelift.io/blog/infrastructure-as-code.

---. “Love DevOps? Wait until You Meet SRE.” *Atlassian*, www.atlassian.com/incident-management/devops/sre.