Brian Preston

Module 1.3

CSD-380

6/1/2025

**The History of DevOps**

When I think about how far software development has come, I truly believe that DevOps has been one of the most game-changing movements in tech. As I have progressed in my software learning journey these past few years, I can certainly tell that it has grown out of decades of trial, frustration, and the need to build and deliver software better, faster, and with fewer headaches.

The Lean Movement which originated in manufacturing. The idea I felt was simple: eliminate waste, shorten lead times, and improve flow. I feel that Lean laid the groundwork for what we now consider smart, scalable systems.

In The DevOps Handbook, the authors define Lean as focusing on “creating a smooth and even flow of work” by using small batch sizes and minimizing rework (Kim et al., 2016, p. 8). These same ideas were later adopted into software processes, shaping how teams think about delivering value.

The Agile Manifesto in 2001. It flipped the traditional Waterfall model on its head. Agile stressed collaboration, responding to change, and delivering working software quickly. I have personally seen how Agile helps teams feel more involved and motivated because it focuses on people and progress, not just process. As Amarachi Iheanacho writes, “Agile significantly improves how we create software and offers advantages like faster time-to-market, testing and superior quality, and risk reduction” (Iheanacho, 2023).

Agile still left a gap between Development and Operations. This is where the Continuous Delivery Movement came in. It emphasized getting code into production quickly and safely through automation and collaboration. DevOps built on this, breaking down the walls between dev and ops teams. I believe this shift made a huge difference in how organizations work. Instead of throwing code “over the wall,” teams now work together throughout the whole lifecycle.

One of the most important concepts that helped shape DevOps is The Three Ways, introduced in The DevOps Handbook. These are the guiding principles that underpin all DevOps practices.

• The First Way is about improving the flow of work from Development to Operations. As the authors explain, “We increase flow by making work visible, by reducing batch sizes...and by building quality in” (Kim et al., 2016, p. 13). I really connect with this idea because it’s about creating momentum and reducing the friction that slows teams down.

• The Second Way focuses on feedback. This means building systems where issues can be spotted and fixed fast. “We must create fast feedback and feedforward loops wherever work is performed,” the book notes (Kim et al., 2016, p. 29). I feel this is the heart of what makes DevOps reliable and resilient—when something breaks, everyone knows fast, and everyone can learn from it.

• The Third Way is all about continual learning and experimentation. It promotes a culture where failure is seen as a chance to improve, not something to hide. That’s powerful. DevOps thrives when people feel safe to take risks and share knowledge across teams.

Knowing where DevOps came from also helps me appreciate why it matters today. As Iheanacho puts it, “DevOps was a grassroots effort… not a product, a specification, or job title. It is an experience-based movement that is decentralized and open to all” (Iheanacho, 2023). That resonates with me because I believe DevOps isn't just a method—it’s a mindset.

In the end, DevOps has changed the game by blending Lean, Agile, and Continuous Delivery into one powerful framework. I feel that the real win with DevOps is not just faster software—it’s happier teams, smarter systems, and stronger results.

**References**

Iheanacho, A. (2023, February 26). *A brief history of DevOps and its impact on software development*. EverythingDevOps. <https://www.everythingdevops.dev/blog/a-brief-history-of-devops-and-its-impact-on-software-development>

Kim, G., Humble, J., Debois, P., & Willis, J. (2016). *The DevOps handbook: How to create world-class agility, reliability, and security in technology organizations*. IT Revolution.

p. 8 – Lean and value stream

p. 13 – First Way (Flow)

p. 29 – Second Way (Feedback)