Adam Donner

Assignment 1.3

Web 430

September 11, 2019

**The Principles Underpinning DevOps**

The Three Ways are the principles that all of DevOps are derived from. Each of the Three Ways lays out the values and philosophies of taking things from development to operations. By following anyone of the ways, the development of the product is more efficient and the customer experience is improved.

The First Way enables the fast flow from development to operations to the customer. This is done by limiting the Work in Process (WIP) and eliminating hardships and waste in the value stream. The Second Way enables the fast and constant flow of feedback from development to operations. This is done by the institutionalization of improvement of daily work. The Third Way enables a constant flow of from development to operations as well as developing and implementing best practices. This is done by institutionalizing the improvement of daily work, transforming local discoveries into global improvements and having leaders that reinforce a learning culture.

Work in Process is the amount of work you have started but is not yet completed. Limiting this is an excellent way to increase throughput in the software development pipeline. The theory of Work in Process is that people and processes cannot multi-task was well they think they can (Wentzel, 2014). When the amount of Work in Processes are reduced or limited, the number of bottlenecks that occur are reduce and the amount of time that is needed to move the project (software development) from start to finish is shortened. Limiting the amount of Work in Processes will also reduce the amount of time that is needed to get the kinks out of the system. Outcomes are what matters, not the process, controls or the work that you complete (Wentzel, 2014). A way to limit the amount of Work in Processes is by using a kanban board to manage work required for a project. Using the kanban board allows you to put Work in Processes limits for each of the columns. Setting limits does not allow any new work to be added until the current things on the board are completed or removed (Kim, Debois, Willis, Humble, & Allspaw, 2017). Work in Process limits also ensure that there is accountability for completion of tasks. Limiting the Work in Process also makes it easier to see problems that prevent the completion of the work (Kim, Debois, Willis, Humble, & Allspaw, 2017). Multitasking that is not productive can and usually result in problems.

Eliminating the hardships and waste in the value stream helps to speed up the development process and reduces the amount of time that is required to fulfill customer requests and increases the quality of work. Shigeo Shingo, one of the pioneers of the Toyota Production System, believed that waste constituted the largest threat to business viability. He described waste as “the use of any material or resource beyond what the customer requires and is willing to pay for” (Kim, Debois, Willis, Humble, & Allspaw, 2017). In terms of software development, waste and hardship is anything that causes delays for the customer, such as activities that can be bypassed without affecting the result (Kim, Debois, Willis, Humble, & Allspaw, 2017) Examples of waste and hardship includes partially done work, extra processes, extra features, task switching, waiting to complete work, the amount of effort to move from one work center to another, defects, manual work or when individual or groups are put into situations where they need to go above and beyond what is typically required. By making hardships and waste visible during the software development and reducing or even eliminating them, the process from start to finish flows faster.

In terms of DevOps and in Lean thinking, there are two types of customers, external customers and internal customers. External customs are the ones that are most likely paying for or receiving the service and the internal customers are the ones that are receives and process the work immediately after us (Kim, Debois, Willis, Humble, & Allspaw, 2017). Optimizing for the downstream work centers helps to identify design problems that may occur for the downstream work centers that may slow or prevent work flow. This can be done by designing for operations where architecture, performance or security (for example) are prioritized as high as the features of the users.

In the technology value stream, it is typical that fixing problems is avoided and workaround are the norm. This is why, in the technology value stream, it is so important if not imperative to improve daily work (Kim, Debois, Willis, Humble, & Allspaw, 2017). Improving daily work can be done by reserving time to reduce technical debt, fix any defects, and improve areas that are problematic. By addressing and fixing the problems in the daily work, the outcome is increased resilience in the system (Kim, 2015).

This is done by sharing not only the solution that are discovered but also the processes by which they were discovered or what was learned or how it was learned (Kim, 2012). Sharing and implementing the improvements ensures that everyone that is doing the same work, is doing it with the same cumulative and collective expertise of everyone in the organization (Kim, Debois, Willis, Humble, & Allspaw, 2017). This can also be looked at as best practices. It is important in the technology value stream to have global knowledge, mechanisms, code, documentation or strong internal groups, that are searchable and utilized by an entire organization.

A good leader will make all of the right decisions. A great leader will create the conditions so that their team can discover greatness together (Kim, Debois, Willis, Humble, & Allspaw, 2017). Leaders must show the importance and value of learning and disciplined problem solving. This must be fostered throughout the organization to improve the work flow.

In conclusion, the Three Ways are the underling principals of DevOps. When developers follow any one of the Three Ways, the development flow will be smooth and the customer experience will be increased.

# Referencers

# Kim, G. (2012, August 22). The Three Ways: The Principles Underpinning DevOps. Retrieved from <http://itrevolution.com/the-three-ways-principles-underpinning-devops/>

# Kim, G. (2015, January 24). DevOps Improvement Principles Behind Google (Randy Shoup Interview). Retrieved from <https://itrevolution.com/uncovering-the-devops-improvement-principles-behind-google-randy-shoup-interview/>

# Kim, G., Debois, P., Willis, J., Humble, J., & Allspaw, J. (2017). The DevOps handbook: how to create world-class agility, reliability, and security in technology organizations. Portland, OR: IT Revolution Press, LLC.

# Wentzel, D. (2014, May 20). DevOps: WIP and final thoughts. Retrieved from https://davewentzel.com/content/devops-wip-and-final-thoughts/