Adam Donner

Web 430

Assignment 5.3

October 9, 2019

Case Study-The Launch and Hand-off Readiness Review at Google

Ben Treynor Sloss coined the term “Site Reliability Engineers” (SRE) at Google. This term referred to the Ops engineers. Treynor Sloss has described the SREs as “what happens when a software engineer is tasked with what used to be called operations.” The Site Relability Engineers keep production running smoothly, while also committing to developing new features and operational advancements. The SRE is made up of 50% software engineers and 50% system administrators. This allows for the development work to stay in line with the goals of the organization, at the same time automating solutions and promoting code quality improvements (Thrope, 2019). There are not many SREs in Google. They are only assigned to product teams that have the highest importance to the company or the projects that must comply with regulatory requirements. Even if a new product becomes significant enough to be assigned an SRE, it must be self-managed by the developers for six months before being assigned to the SRE team (Kim, Debois, Willis, Humble, & Allspaw, 2017).

In order for the self-managed product teams to still benefit from the experience that the SREs have, Google created two sets of safety checks for two critical stages of releasing new services. These are called the Launch Readiness Review (LRR) and the Hand-Off Readiness Review (HRR). The LRR must be performed and signed off on before it can be available to the public customers. It is a self-reporting checklist the products teams conduct before the service goes live to customers (Thrope, 2019). The HRR is performed when the service is transitioned to an Ops-managed state. This usually occurs months after the LRR. The checklist that goes with the HRR is much more rigorous and has higher acceptance standards than the LRR. The LRR is self-reported by the product teams. There is an SRE assigned to any of the HRR or LRR product teams to help the understand and achieve all of the requirements the checklist before the products can be released. This process makes the Development team experience the work of the Operations team and allows for an improvement of the standard the service transition makes from creation through launch (Thrope, 2019) (Kim, Debois, Willis, Humble, & Allspaw, 2017).

Every time Google releases a new product, something, whether it is good or bad, is learned. Through the evolution of the release checklists, Google has created an organizational memory. Going through this type of release process, the Development team gets to see how the Ops team does things. When the LRR and HRR are involved, the transitions are easier and more predictable for the organization. (Kim, Debois, Willis, Humble, & Allspaw, 2017).

Case Study-Auto-Scaling Capacity at Netflix

Netflix had developed a tool called Scryer to address some of the shortcomings of Amazon Auto Scaling (ASS). Scryer allows Netflix to provision the right number of AWS instances needed to handle the traffic of its customers (Jacobson, Yuan, Joshi, 2017). It does this by predicting what customer demands will be based on historical usage patterns and provisions of the necessary capacity (Kim, Debois, Willis, Humble, & Allspaw, 2017).

There were three problems that Scryer addressed that AAS could not. These problems were dealing with rapid spikes in demand, outages, and variable traffic patterns. Scryer was originally inspired by these three things but was ultimately triggered by how predictable the customer view patterns were (Jacobson, Yuan, Joshi, 2017). Scryer used a combination of outlier detections to get rid of spurious data points and then used Fast Fourier transform (FFT) and linear regression to smooth the data while keeping any traffic spikes that recur. With this, Netflix can accurately forecast traffic demand and improve the customer viewing experience, and service availability as well as reduce Amazon EC2 costs Kim, Debois, Willis, Humble, & Allspaw, 2017).

References

Thrope, S. (2019, August 23). The DevOps Equation. Retrieved October 8, 2019, from <https://caylent.com/devops-equation>.

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.

Jacobson, D., Yuan, D., & Joshi, N. (2017, April 18). Scryer: Netflix's Predictive Auto Scaling Engine. Retrieved October 8, 2019, from https://medium.com/netflix-techblog/scryer-netflixs-predictive-auto-scaling-engine-a3f8fc922270.