The correlation between SaaS and DevOps

Christina Sunnegårdh, Christine Rosquist

April 16, 2020

Content

1 Introduction

Successful online companies, like Google and Slack, have increased customers expectations for great services at fast speed (Lwakatare et al. 2016). This, due to their almost immediate response times to the customers needs. Because of this, more companies from various fields are emulating their capabilities in order to cope with the competition. The development of the technology, including cloud computing, have affected the ways in which software products are developed and provided to the customers. In the cloud environment, providers of Software-as-a Service (SaaS) applications are expected to continuously update the software in faster release cycles.

Today, many companies are, or are moving towards becoming, SaaS companies. Overall, this means continuously updating their software, which is available to customers through the cloud by subscription. This results in many new possibilities that were not possible through the old model, where companies released annual updates that customers buy everlasting licences to. As SaaS enables faster and more continuous upgrades, questions can be raised about whether the traditional ways of structuring teams are still efficient. In traditional software development, the developer team and the operation teams are often separated, meaning that they have little or no shared knowledge (Ali 2016, p.55). Meanwhile, the idea of DevOps has quickly grown to become a major influence in software development and deployment. The idea of DevOps was born in 2008 and is a set of practices for developing and deploying software, which aims to shorten the system development life cycle (Bugwolf 2016).

1.1 Research question

As questions can be raised about whether the traditional ways of working are enough to support the demand for high quality service at fast speed in SaaS companies, one may wonder if DevOps practices could contribute to a more optimal way to work. This essay will therefore investigate what advantages that come with DevOps practices and especially how they correlate to, as well as how they can facilitate work within, SaaS companies.

2 Background

In this section we will outline the basic concepts of DevOps and SaaS. It is necessary to have a basic knowledge of the relevant concepts, which will be discussed in relation to the example about Adobe's transition to providing SaaS, outlined in section 3, in section 4.

2.1 DevOps

DevOps is a set of practices for trying to shorten the system development life cycle, while keeping high quality (Mala 2019, p.16). DevOps can be visually represented as in Figure 1 below, where the tasks of development (Dev) and operations (Ops) together form the DevOps delivery cycle (Nub8 2020). DevOps aims to reduce the efforts to commit a change to a system and have that change pushed into production (Bass et al. 2015). While there seems to be various opinions on how this should be done in practice, most sources agree on that it should be done by somehow eliminating the traditional obstacles between the developers and the operations teams. In the traditional structure, developers gave the code to testers and the operations team, who then had the responsibility to operate and monitor the system. This entails that the operations team, for example, had to wait for another team to do troubleshooting and that issues could arise due to that different environments were used (Chandu 2019). Eliminating these obstacles could be done by the two teams being merged into one, where the members work across the system development life cycle, demanding a wide range of skills of the engineers involved. Another approach is to make the teams more tightly integrated. In both approaches, automation tools, often known as part of the DevOps toolchain, are usually used to facilitate when developer and operation teams are merged, or create bridges between the teams (Amazon 2020).

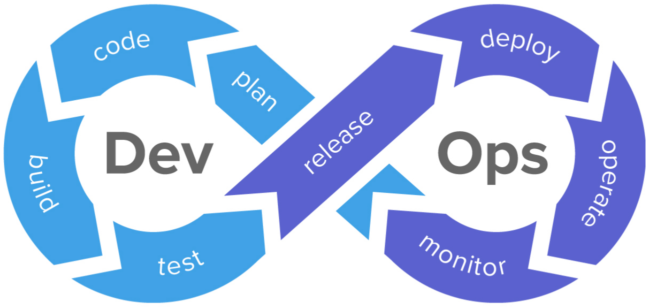

Figure 1: The DevOps delivery cycle

2.2 Software as a Service (SaaS)

SaaS, short for Software as a Service, is one of three main categories of cloud computing, along with Platform as a Service (PaaS) and Infrastructure as a Service (IaaS). Cloud computing is a term that refers to anything involving delivering hosted services through the Internet (Mahmood 2011). SaaS differs from the traditional software, where the software is located on-premises. That means that the traditional software runs on computers of the customer, instead of running remotely in the cloud, which is illustrated in Figure 2. Figure 2 illustrates the main differences between SaaS and traditional software, when it comes to location, responsibility and ownership of the software.

SaaS is a fast-growing model of software licensing and in contrast to traditional software, where buyers obtain an everlasting license, the SaaS buyers instead buy a subscription from the publisher. In traditional software, the buyer himself needs to install and maintain the hardware, software or other technical infrastructure needed. However, with SaaS, the software publisher handles the maintenance of the hardware and software, which can be seen in Figure 2. The buyers then gain access to it over the Internet (Choudhary 2007).

Using SaaS can be beneficial for the customer, as the cost of the infrastructure, the right to use the software and all the hosting and maintenance services are all bundled into one monthly charge. The market of SaaS is very wide, and the services can be anything from web-based email to online banking services, or database processing (Mahmood 2011).

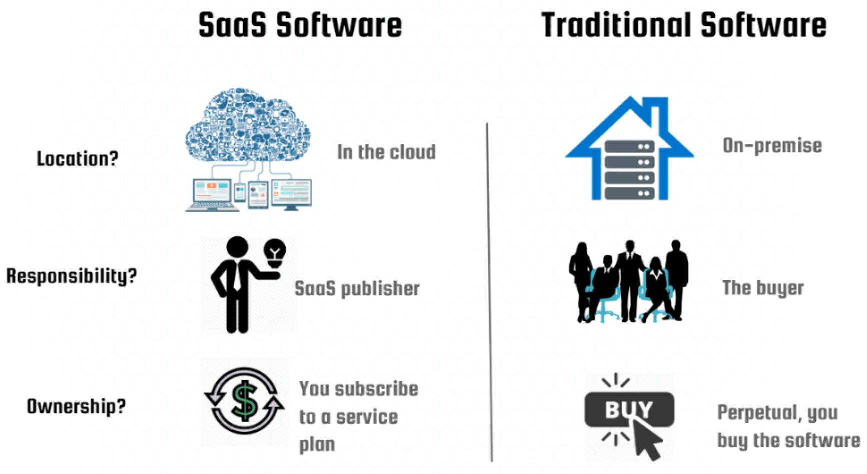

Figure 2: SaaS vs Traditional Software

3 Adobe: from annual releases to SaaS

Adobe Inc. is a multinational computer software company that was founded in 1982 (Adobe 2020*a*). Many know them for products like Photoshop, InDesign and Illustrator, that up until 2012 were released in a software suite by the name Adobe Creative Suite (Adobe 2012). Adobe Creative Suite was introduced in 2003 and included a variety of different programs that could be purchased through a one time payment, separately or as a package (Adobe 2003). The suite was continuously updated through large releases, featuring new functionality, with a few years gaps in between (Adobe 2005, 2007, 2008). At the release of Adobe Creative Suite 6 (CS6), Adobe announced that the suite would be available by a subscription-based membership in something called Adobe Creative Cloud (CC), as an alternative to one time purchase (Adobe 2012). In 2013, one year after the release of CS6, they announced that the Creative Suite would be discontinued and that all future updates instead would be available through CC. This, making Adobe a fully subscription-based company, where customers no longer could buy everlasting licences for their programs but instead download and update them through the cloud (Weber 2013). In 2012, another subscription-based product by name Adobe Marketing Cloud, was launched (Blattberg 2015). Adobe Marketing Cloud, now called Adobe Experience Cloud, is a SaaS service that provides a collection of online marketing and web analysis products, hosted on Microsoft Azure (Adobe 2020*b*, Microsoft 2018).

In an interview carried out by McKinsey in 2015, Dan Cohen, vice president of business operations and strategy of Adobe, discussed Adobe's transformation into offering web-based software and services (Sprague 2015). He points out that moving to the cloud affected both how they engineered the products and operations. Before, when thinking about adding new features to the next release, the scope was 18-24 months. Now they operate in an agile development model where the value proposition is about delivering high-quality service, not just new features. As a result, functional groups that used to work separately (product management, engineering, marketing, IT) are now closer integrated (Sprague 2015). Cohen further describes how, under the subscription-model, customers are essentially deciding every month whether to renew or not, putting pressure on the company to keep the products up to date. During a keynote at Container Security Summit 2019, the chief security officer of Adobe, Brad Arkin, emphasised the importance of automatising manual tasks for both development and operations. The keynote further discusses how Adobe is rapidly modernizing its software build and delivery process onto continuous integration (CI) and delivery (CD) platforms (Wang 2016).

4 Discussion

In this section we will use the background about DevOps and SaaS, given in section 2, as well as the example about Adobe, written about in section 3, to discuss the correlation between DevOps and SaaS, and answer the research question.

4.1 Correlation between SaaS and DevOps

With SaaS, unlike with traditional software, the publishers themselves need to handle the maintenance of both hardware and software. In section 3.1, it can be seen how Adobe has made a transformation from annual releases to a subscription-based cloud model. Even though the programs in the Creative Cloud are downloaded and installed on the customer's computer, the software type and high paced workflow to maintain it can be equated with SaaS. As they also provide SaaS products such as Adobe Experience Cloud, the company is arguably labeled a true SaaS company. SaaS companies need to continuously update their software in order to satisfy the expectations from customers. As mentioned by Cohen in section 3.1, customers are frequently making a choice whether to continue using and paying for the software or not, demanding companies to stay updated and ahead of others in a new pace then what has traditionally been seen. It is therefore essential to deliver products reliably, quickly, and with high quality for a SaaS-company to succeed. At Adobe, this resulted in a closer integration of the functional teams, using automated tools to facilitate manual work, and starting working with CI/CD. This is all part of the DevOps practice, which purpose, stated in section 2.1, is to shorten the system development life cycle while maintaining high quality. In order to continuously deploy new or updated software, the coordination of activities and the collaborative work among the developers, testers, and operations personnel needs to be effective. The DevOps practices enable this, as it acknowledges the demand for a continuous bridge between the development and its operational deployment. For many companies that operate on the web, it can also be a competitive advantage to be able to deploy new program features more frequently. Hence, even if it's not literally enunciated, Adobe seems to have made a transition into practicing DevOps when introducing SaaS products, to have the chance to keep their products as up to date as today's technology enables, both for them and other competing companies. This is done by aiming to continuously deploy to eliminate bugs and introduce new features while keeping high quality, which arguably requires this new, agile way of working.

Since the SaaS applications are fully hosted by the providers, it is essential for the companies to keep the cost of hosting as low as possible. DevOps does not only help the companies to deliver software more frequently, it can also contribute with reducing organizational costs involved in the software development, deployment and maintenance. For instance, DevOps can provide software teams with automated tools that can be re-triggered when needed after it is developed. This type of automation can be used to monitor and troubleshoot the application, which usually require a team of resources to accomplish. With DevOps, these types of automation can eliminate such costs for SaaS-companies.

In one way, the growth of SaaS products on the market, could be said to have increased the demand for the DevOps practice. When providing SaaS, it becomes difficult for functional teams to work in the traditional way, as iterations in the system development life cycle takes longer. This could have accelerated the ideas of agile methods and the DevOps practice, so that companies are able to handle the pressure of customers constantly evaluating the software. When instead doing annual releases of software, without continuously updating it, using DevOps might not be as important as for SaaS companies. This, as the importance of keeping a high pace to maintain the software is not as significant. However, even if it could be seen as less important, there is nothing that indicates that the practices of DevOps would not be beneficial for a non-SaaS company. As discussed in section 2.1, there are various opinions and ways that DevOps can be implemented in practice, non-SaaS companies might benefit from some of the practices within DevOps. For example, using tools from the DevOps toolchain to automate manual processes. Regardless of time pressure or not, it can be beneficial to have more time, as it can be spent on, for example, adding more features.

4.2 Conclusion

While SaaS and the cloud offer resources to the customers, DevOps intends to efficiently utilize these resources for better business development and user experience. Thus, DevOps facilitate the work within SaaS companies to a great extent, making it possible to deliver high quality services at a faster speed. This, in a way that is not achievable to the same extent when working in the traditional ways. There are many advantages of using DevOps within SaaS companies, and one could argue that DevOps is vital for maintaining a SaaS product on the market today. Yet, one could also argue that the growth of SaaS products enables companies to fully exploit the power of DevOps. An increased demand of DevOps surely promotes its development, hence, one could say that they both contribute to and enable each other.

References

Adobe (2003), 'All-new adobe creative suite delivers platform for the future of design and publishing', [Online] Available online: https://web.archive.org/web/20100509011539/http://www.adobe.com/aboutadobe/pressroom/pressreleases/200309/092903CREATIVESUITE.html. [Accessed on 2020-04-15].

Adobe (2005), 'Adobe pushes the boundaries of digital photography and imaging with debut of photoshop cs2', [Online] Available online: https://web.archive.org/web/20121113020418/http://www.adobe.com/aboutadobe/pressroom/pressreleases/200504/040405Photoshop.html. [Accessed on 2020-04-14].

Adobe (2007), 'Photoshop cs3 and photoshop cs3 extended push the boundaries of digital imaging', [Online] Available online: https://web.archive.org/web/20121113020524/http://www.adobe.com/aboutadobe/pressroom/pressreleases/200703/032707Photoshop.html. [Accessed on 2020-04-14].

Adobe (2008), 'Adobe creative suite 4 now shipping', [Online] Available online: https://web.archive.org/web/20081030170838/http://www.adobe.com/aboutadobe/pressroom/pressreleases/pdfs/200810/101508AdobeCS4avail.pdf. [Accessed on 2020-04-14].

Adobe (2012), 'Milestone adobe cs6 release delivers major innovations for design, web and video pros', [Online] Available online: https://www.adobe.com/aboutadobe/pressroom/pressreleases/201204/042312AdobeCreativeSuite6.html. [Accessed on 2020-04-15].

Adobe (2020*a*), 'About adobe', [Online] Available online: `https://www.adobe.com/about-adobe/fast-facts.html`. [Accessed on 2020-04-16].

Adobe (2020*b*), 'Why adobe', [Online] Available online: `https://www.adobe.com/sea/experience-cloud/why-adobe.html`. [Accessed on 2020-04-16].

Ali, J. (2016), *Mastering PHP Design Patterns*, Packt Publishing Ltd.

Amazon (2020), 'What is devops?', [Online] Available online: `https://aws.amazon.com/devops/what-is-devops/`. [Accessed on 2020-04-11].

Bass, L., Weber, I. & Zhu, L. (2015), *DevOps: A Software Architect's Perspective*, IGI Global.

Blattberg, E. (2015), 'Adobe goes high end in podcast push', [Online] Available online: `https://digiday.com/marketing/adobe-podcasts/`. [Accessed on 2020-04-15].

Bugwolf (2016), 'Devops best practices', [Online] Available online: `https://bugwolf.com/blog/the-origin-of-devops`. [Accessed on 2020-04-10].

Chandu, V. (2019), 'Top 5 reasons why devops is important', [Online] Available online: `https://dzone.com/articles/top-5-reasons-why-devops-is-important`. [Accessed on 2020-04-16].

Choudhary, V. (2007), Software as a service: Implications for investment in software development, *in* '2007 40th Annual Hawaii International Conference on System Sciences (HICSS'07)', IEEE, pp. 209a–209a.

Lwakatare, L. E., Kuvaja, P. & Oivo, M. (2016), 'An exploratory study of devops extending the dimensions of devops with practices', *ICSEA 2016* **104**.

Mahmood, Z. (2011), Cloud computing: Characteristics and deployment approaches, *in* '2011 IEEE 11th International Conference on Computer and Information Technology', IEEE, pp. 121–126.

Mala, D. J. (2019), *Integrating the Internet of Things Into Software Engineering Practices*, IGI Global.

Microsoft (2018), 'Why adobe', [Online] Available online: `https://customers.microsoft.com/en-us/story/adobe-partner-professional-services-azure`. [Accessed on 2020-04-16].

Nub8 (2020), 'What is devops?', [Online] Available online: `https://nub8.net/what-is-devops/`. [Accessed on 2020-04-11].

Sprague, K. (2015), ‘Reborn in the cloud’, [Online] Available online: `https://www.mckinsey.com/business-functions/mckinsey-digital/our-insights/reborn-in-the-cloud`. [Accessed on 2020-04-15].

Wang, C. (2016), ‘How adobe uses microservices and containers to bolster security’, [Online] Available online: `https://www.hpe.com/us/en/insights/articles/how-adobe-uses-microservices-and-containers-to-bolster-security-1612.html`. [Accessed on 2020-04-15].

Weber, H. (2013), ‘After nearly 10 years, adobe abandons its creative suite entirely to focus on creative cloud’, [Online] Available online: `https://thenextweb.com/insider/2013/05/06/after-nearly-10-years-adobe-abandons-its-creative-suite-entirely-to-focus-on-creative-cloud/`. [Accessed on 2020-04-15].