

# Advantages of DevOps in SaaS companies

Christina Sunnegårdh, Christine Rosquist

April 23, 2020

## Content

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Background</b>	<b>2</b>
2.1	DevOps . . . . .	2
2.2	Software as a Service (SaaS) . . . . .	4
<b>3</b>	<b>Case Study of Adobe</b>	<b>5</b>
3.1	Adobe Experience Cloud . . . . .	5
3.2	Transition into continuous updates . . . . .	5
3.3	Culture at Adobe . . . . .	6
<b>4</b>	<b>Discussion</b>	<b>6</b>
4.1	Continuous updates of software in SaaS . . . . .	6
4.2	The costs of SaaS . . . . .	7
4.3	The SaaS culture . . . . .	7
<b>5</b>	<b>Conclusion</b>	<b>8</b>

# 1 Introduction

Successful online companies, like Google and Slack, have increased customers expectations for great services at fast speed (Lwakatare et al. 2016). This is 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 technology, including cloud computing, have affected the ways in which software products are provided to the customers. In the cloud environment, providers of Software-as-a Service (SaaS) applications are expected to continuously update the software in a high pace (Wagner 2013). Today, an increasing number of companies are moving towards providing SaaS solutions (Columbus 2013, Robins 2017). As SaaS enables and thereby creates a demand for faster and more continuous updates, questions can be raised about whether the traditional ways of structuring teams are sufficient. 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).

As the demand and expectations for more continuous upgrades to services are increasing, one may wonder if and how DevOps practices could facilitate in various aspects for companies providing SaaS. This essay will therefore investigate advantages that DevOps practices can bring to SaaS companies, using a case study of Adobe in section 3 as an example. It will focus on the aspects of the continuous updates that are expected in SaaS, the cost of SaaS and the SaaS culture.

## 2 Background

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

### 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). The DevOps culture tends to extend the agile methods to develop the software faster, and to deliver it in an

automated matter so that the performance and quality assurance is improved. It also aims to enable organizations to deliver more features continuously while maintaining stability (Rajkumar et al. 2016). 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).

An example of a tool that is a part of the DevOps toolchain is the containerization tool Docker (Kang et al. 2016). Docker is an open platform for developers and system administrators to build, ship, and run distributed applications. Different applications can be run over Docker containers, which is an isolated environment where the software is packaged. Containers can also be used to cut down the costs of the infrastructure. Unlike the architecture of Virtual Machines (VM), where each VM has its own guest operating system, containers share the host operating system (Singh & Peddoju 2017). This leads to containers reducing the operating cost of having different guest OS for different container procedures, which results in a reduced infrastructure cost (Singh & Peddoju 2017).

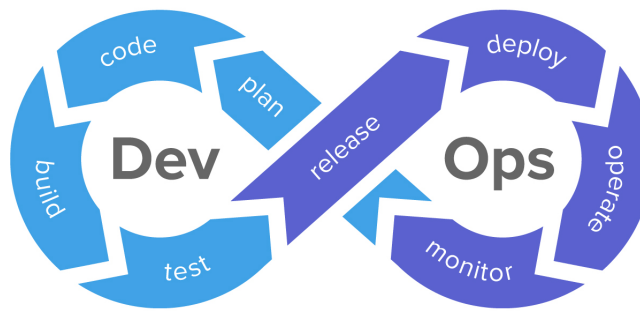


Figure 1: The DevOps delivery cycle

## 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. In contrast to traditional software, where buyers obtain an everlasting license, buyers of SaaS 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).

As mentioned in section 1, SaaS providers are expected to continuously update their software in high pace. This is opposed to the traditional software, that is installed on-premise. To exploit the continuous updates, and to release new features rapidly, a SaaS provider needs to adopt towards a more agile culture (Agarwal 2011). The agile approach has the goal to satisfy the customer based on early and continuous delivery of software (Iivari & Iivari 2011). One guiding principle of the agile approach is that individuals and interactions are more valuable in software development, than processes and tools. Another principle is that it is more important to respond to changing requirements, than to follow a plan. When it comes to the principles of the development process in the agile approach, one principle is that there should be a close collaboration between developers and customers.

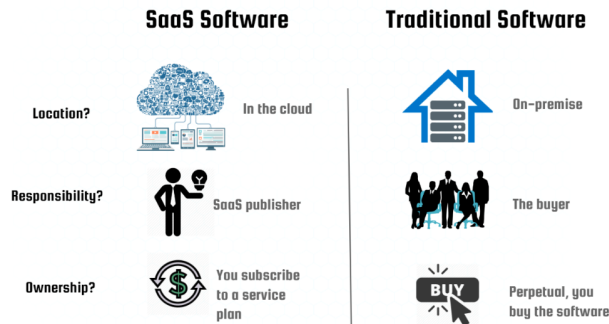


Figure 2: SaaS vs Traditional Software

### 3 Case Study of Adobe

Adobe Inc. is a multinational computer software company that was founded in 1982 (Adobe 2020a). 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).

#### 3.1 Adobe Experience Cloud

In 2012, another subscription-based product by name Adobe Marketing Cloud, was launched (Blattberg 2015). Adobe Marketing Cloud, now renamed to Adobe Experience Cloud, is a SaaS service that provides a collection of online marketing and web analysis products, hosted on Microsoft Azure (Adobe 2020b, Microsoft 2018). Microsoft Azure is a PaaS (Platform as a Service) which Adobe uses to free up resources, from working with data centers and infrastructure, to focus on the software development (Microsoft 2018). Similar to SaaS, PaaS is a subscription-based service which works as a complete development and deployment environment in the cloud (Microsoft 2020). Furthermore, PaaS is designed to support the complete web application life cycle, including providing infrastructure, middleware and database management (Microsoft 2020).

#### 3.2 Transition into continuous updates

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, including 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).

### 3.3 Culture at Adobe

According to Brandon Pulsipher, VP of cloud engineering at Adobe, culture was a big factor when Adobe started their transition into DevOps. The executives for the ops and developers teams created a DevOps framework and shared it with respective teams, traveled to go through plans with teams and empowered distributed leaders and set up specific procedures and accountability. He expresses how a common problem is employees viewing DevOps as a role instead of a methodology and that DevOps really is about autonomy, ownership, and agility. Furthermore, a culture that embrace this methodology should not be restricted by committee decree or through overly prescriptive rules. Rather, changes should originate from within the teams themselves (Doerrfeld 2019).

## 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 how DevOps can facilitate in the chosen aspects of SaaS. In section 3, 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 SaaS company.

### 4.1 Continuous updates of software in SaaS

As mentioned in section 2.2, SaaS companies need to continuously update their software in order to satisfy the expectations from customers. As explained by Cohen in section 3.2, 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 traditionally has 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 can assist in enabling this, as it acknowledges the demand for a continuous bridge between the development and its operational deployment.

It can be seen that Adobe has 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.

## **4.2 The costs of SaaS**

As mentioned in section 2.2, SaaS companies need to handle the maintenance of both hardware and software. Since the SaaS applications are fully hosted by the providers, one could argue that it is important for SaaS companies to decrease their costs. As hosting systems includes handling performance, reliability and infrastructure, there would need to be teams responsible for this. As mentioned in section 3, Adobe Experience Cloud is hosted on the Platform as a Service (PaaS) product Microsoft Azure. This means that Adobe are paying for the hosting as a service, instead of letting their own developers build and maintain it from scratch. It is claimed that this is to let their developers focus on software development rather than infrastructure. Hence, one can assume that this must be more beneficial for Adobe than hosting the application on their own. As one approach to DevOps is that all engineers work across the system development life cycle, PaaS can be seen as closely related to DevOps, as it allows teams to operate and deploy without having the knowledge of a senior operator. It could therefore be argued that the ideas of DevOps are to be credited for keeping down the costs of hosting the system. However, if a company wish to build and maintain the hosting service themselves, there are other ways DevOps can contribute to lowering the costs. Section 2.1 mentions Docker and its use of containers as an example of a DevOps tool that can reduce the infrastructure costs. This, as containers share the OS, which means that they are more lightweight and uses a smaller amount of resources compared to traditional Virtual Machines.

It can also be worth mention that DevOps could be said contribute to reducing software costs by the usage of automation tools in the DevOps toolchain. This entails less work and therefore less resources will thereby be needed. However, this applies to all companies, regardless of providing SaaS services or not.

## **4.3 The SaaS culture**

An example of a SaaS company practicing the DevOps culture is Adobe, as mentioned in section 3.3. As Pulsipher states that the culture was significant for the company when transitioning into DevOps, there arguably must be valuable

for the SaaS company to practice DevOps when it comes to the organizational culture. As mentioned in section 2.2, an agile culture is needed for the SaaS providers to fully exploit the continuous updates. Considering that the DevOps culture extends the agile methods to develop software at a higher pace, noted in section 2.1, this also indicates that the DevOps culture would be beneficial for SaaS companies. Since the main idea behind DevOps is to deliver software in a more automated matter, it follows that DevOps can contribute to the SaaS culture which idea is to satisfy the customer based on continuous delivery. Section 2.2 also mentions the importance of individuals and their interactions to achieve a beneficial culture within the SaaS organization. This relates closely to the belief that, in the DevOps culture at Adobe, changes should originate from within the teams themselves, as explained by Pulsipher in section 3.3. It can further be related to the definition of DevOps outlined in section 2.1, as DevOps contributes to the collaboration between the traditionally separated teams and thus, enabling more interactions between individuals.

## 5 Conclusion

In this essay, DevOps has shown to be beneficial for SaaS companies in all the three aspects that were investigated. Firstly, when it comes to the continuous updates that are expected in SaaS, DevOps contributes with 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. Further, concerning the second aspect that covers the cost of SaaS, DevOps can enable SaaS companies to keep down their costs by decreasing the costs of hosting the system. DevOps can also contribute to the final aspect, the SaaS culture, in a beneficial way. The DevOps culture strives for a faster delivery of software, and a close collaboration between teams, which is valuable for SaaS companies and their organizational culture. Hence, it can be concluded that DevOps facilitate for companies providing SaaS in the aspects of the continuous updates that are expected in SaaS, the cost of SaaS, and the SaaS culture.

## 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 cs 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 (2020a), ‘About adobe’, [Online] Available online:  
<https://www.adobe.com/about-adobe/fast-facts.html>. [Accessed on 2020-04-16].
- Adobe (2020b), ‘Why adobe’, [Online] Available online:  
<https://www.adobe.com/sea/experience-cloud/why-adobe.html>. [Accessed on 2020-04-16].
- Agarwal, P. (2011), Continuous scrum: agile management of saas products, *in* ‘Proceedings of the 4th India Software Engineering Conference’, pp. 51–60.
- 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.
- Columbus, L. (2013), ‘North bridge venture partners future of cloud computing survey: SaaS still the dominant cloud platform’, *Forbes* .
- Doerrfeld, B. (2019), ‘How CIOs can use DevOps to move customer experience forward’, *Adobe* .
- Iivari, J. & Iivari, N. (2011), ‘The relationship between organizational culture and the deployment of agile methods’, *Information and software technology* **53**(5), 509–520.
- Kang, H., Le, M. & Tao, S. (2016), Container and microservice driven design for cloud infrastructure DevOps, *in* ‘2016 IEEE International Conference on Cloud Engineering (IC2E)’, pp. 202–211.
- 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].
- Microsoft (2020), ‘What is PaaS?’, [Online] Available online: <https://azure.microsoft.com/en-us/overview/what-is-paas/>. [Accessed on 2020-04-20].
- Nub8 (2020), ‘What is DevOps?’, [Online] Available online: <https://nub8.net/what-is-devops/>. [Accessed on 2020-04-11].
- Rajkumar, M., Pole, A. K., Adige, V. S. & Mahanta, P. (2016), DevOps culture and its impact on cloud delivery and software development, *in* ‘2016 International Conference on Advances in Computing, Communication, & Automation (ICACCA)(Spring)’, IEEE, pp. 1–6.
- Robins, B. (2017), ‘How Adobe uses microservices and containers to bolster security’, [Online] Available online: <https://www.zendesk.com/blog/wild-west-days-saas-coming-end/>. [Accessed on 2020-04-17].

- Singh, V. & Peddoju, S. K. (2017), Container-based microservice architecture for cloud applications, *in* '2017 International Conference on Computing, Communication and Automation (ICCCA)', pp. 847–852.
- 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].
- Wagner, P. (2013), 'The benefits (and challenges) of saas updates', [Online] Available online: <https://www.accttwo.com/blog/benefits-and-challenges-saas-updates.html>. [Accessed on 2020-04-20].
- 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].