Continous Integration and the Cloud
"from a beginners perspective"

Kasper Liu, Gunnar Applelid

May 2020

Contents

1 Introduction

In the field of DevOps there exists a myriad of different terms that could be highly confusing for a novice student, or a beginner that has only heard of Testing or Continuous Integration. If these terms are unfamiliar, it could be even more confusing how they do relate to the "Cloud".

One description of the Cloud, is that *"The cloud is often seen as a very amorphous, even nebulous, concept, one often applied to anything to do with computing infrastructure that you can't either directly or physically touch and control. As a result, it's become a somewhat derided term that marketing departments seem to use with increasing abandon."* [1]

According to Amazon, *"Continuous integration is a DevOps software development practice where developers regularly merge their code changes into a central repository, after which automated builds and tests are run"* [2]

To many people, there is no connection between the term "Cloud" and Continuous Integration. Instead, they do associate the term only with providing storage or computing resources. However, there do exist tools for DevOps and especially for Continuous Integration in the "Cloud". This essay will therefore try to provide an orientation for the beginner, in the area of "Continuous Integration and The Cloud" and try to answer the questions; What is Continuous Integration? What common tools do exist? What are some of the considerations of companies, when choosing Continuous Integration tools? What are the differences between On-premise CI and Hosted CI? What services do some of the "big" Cloud Providers offer? Hopefully, after reading this essay the questions will be answered and the drawings in figure 1 will be more clear. It would be difficult to draw strict lines between all of these subjects, because they are closely related.

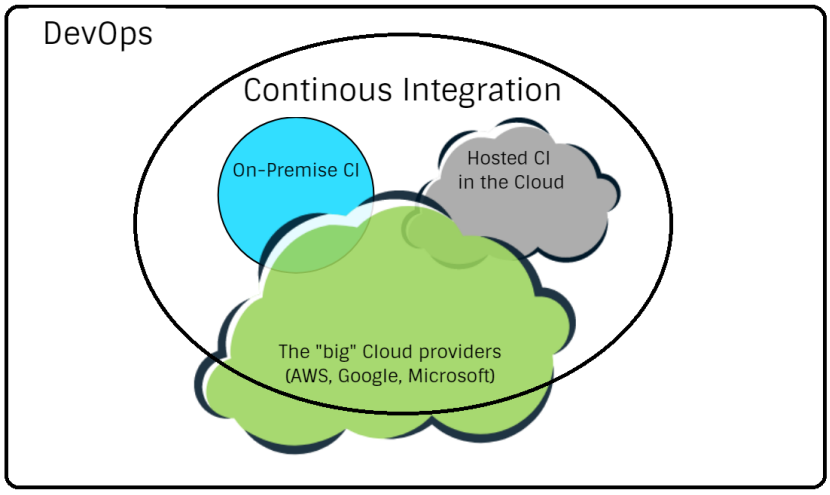

Figure 1: Illustration of Continuous Integration and the "Cloud"

2 Continuous Integration

This section will describe the background of Continuous Integration and explain what it means. Also, some of the common tools will be presented and the differences between On premise CI and Hosted CI will be explained followed by the "big" Cloud providers and a conclusion.

2.1 Background of Continuous Integration

The Background of Continuous Integration (CI) is that a small software project with one developer could be an "easy" non-complex process. But what if the number of developers increases and the scope of the software project gets larger? It would then be dangerous to test all the different parts from the developers, just before the product should be delivered to the customer. The risk of problems in the final integration is very high and this could lead to delays in delivering. Therefore, the process of Continuous Integration requires testing of the different components of a software project, many times during the development cycle. This could be done by each developer having their own build which they work on. When they are done with their work, they commit their code to a version control repository. Then an Integration build is conducted together with tests that must pass for the build to be accepted. This process could be done automatically and therefore increase both effectiveness and correctness[3]. It is important to understand that the process itself is what is called Continuous Integration. A very simple example of Continuous Integration are scripts that run in the following order: launches a testing tool, builds the software after successful testing and sends a notification to developers with success or failure. But commonly, there are more advanced features with databases, web servers and more depending on the needs of the organization or the user.[1]

2.2 On-premise and Hosted CI tools

Today there are many tools to aid with Continuous Integration for all scales of projects. Universities and companies push their employees to develop with CI in mind. There are mainly two types of CI, On-premise CI and hosted CI. Jenkins is a good example of a typical on-premise CI and Codeship offer a service for hosted CI in the cloud. With the advent of cloud and cloud computing a lot of CI tools have been adopted by the "big" cloud providers as well.

Many companies use their tools for Continuous Integration on local servers, maintained by their own personnel. This could lead to large effort and costs involved in maintaining the infrastructure. Therefore, many companies have started to use tools for Continuous Integration in the cloud. In this way there can be more focus on the development itself, while the cloud provider handles the infrastructure, security, support and more[4]. There will still exist a need of configuration, but not at the same level and it will mostly be involved with the pipeline itself[1].

"Simply put, cloud computing is the delivery of computing services—including servers, storage, databases, networking, software, analytics, and intelligence—over the Internet ("the cloud") to offer faster innovation, flexible resources, and economies of scale." [5]

If cloud computing frees us from the pain of installing and maintaining physical servers. CI frees us from the pain of building, testing and deploying of the code. Both of the "concepts" aim to remove some of the workload from developers, then it sounds like the two matched together is an excellent choice. Not only does the combination save time, it can also save money.

2.2.1 Jenkins

"Jenkins is a free and open source automation server. Jenkins helps to automate the non-human part of the software development process, with continuous integration and facilitating technical aspects of continuous delivery." [6].

It is an easy solution for creating an environment of continuous integration and delivery for many different languages. Jenkins supports different applications for version control such as Git. One of the main reasons for its popularity is its easy installation, configuration and appealing web interface. There is also a large amount of plugins that can be used for Jenkins to work with different tools. Works can also be distributed between different machines, eg. for faster deployment[7].

Jenkins do support both simple jobs and something called pipelines. The latter can configure the whole life-cycle of an application, from commit to delivery, in a very complex way. The pipeline is often described as different blocks that are executed sequentially. These blocks could be named Build, Test and Deploy and do contain further steps and tasks. Some of the pipeline functionality provided by Jenkins are that pipelines are restored after a system crash,they can be paused, multiple pipelines can work in parallel, checkpoints can be used and more. [8][9][10].

2.2.2 Codeship

Codeship offers a service for hosted CI and basically handles everything related to CI for the customers[11]. Codeship was acquired by the Cloudbees company in 2018. Before this time there were many different providers of Continuous Integration tools, but in the later years many of them have merged. With a user friendly interface, and provided dependencies it helps to "automate the testing and deployment of the code". It supports fast building by parallelism and caching. Also, the organisation has a strong support organisation. The overall goal is that the customer should focus on their product and code, not on the configuration of Continuous Integration. [12]

2.2.3 Travis

Travis is also a hosted CI service. If we compare it with Jenkins, then the differences is that the enterprise version isn't free and if more support is needed the price increases. However, the community of Jenkins is larger. It is easier to configure Travis and it has a shorter time to get started. The use of YAML files for configuration makes it possible for advanced configurations. It also has support for Git and parallel execution. So if you need a lot of in depth configuration probably Jenkins is the choice for you. But if you want an easy startup process you should consider Travis. Today Jenkins is the most popular of the two based on questions on Stack Overflow[13].

2.2.4 Bamboo

Bamboo is a Continuous Integration service created by Atlassian. It was provided both in the cloud and as a local service, but they stopped with the cloud version in 2017. As the other tools it does support automatic build, test and deploy of software as well as parallel jobs. It also does have support for version control like Git and testing frameworks. One of the big differences from Jenkins is that there are more built in support and plugins so there is less need of configuration. It isn't open source as Jenkins and some people could consider it more simple. The pricing is depending on how many build agents you use and this could lead to a better cost for small companies[14].

2.3 On-Premise CI vs Hosted CI

The two different types of CI have their own advantages and disadvantages. With an on-premise CI such as Jenkins, you have complete control of it. The CI can be configured to one's own needs and are not in the mercy of a third-part service. There is a huge supply of plugins to allow all manner of customisation. There is no need to share resources with other customers of a host based service and from a security point of view, you are in control of it yourself for better or worse. One example, from one of the "big" cloud providers, is the recent Amazon AWS S3 outage[15]. For all the customization and control one has to divert resources for it. There probably needs to exist someone who knows the system well, staff to be assigned to setup, maintain and configure the on-premise CI. On the contrary, a Hosted CI outsources all the work that comes with on-premise CI to a third party who handles it all. In exchange for control, customization and money. Furthermore, one has to share the resources of the third party and are at mercy of their service. Just as the incident given above, if it were to happen, how are the consequences handled. There are many factors one gives up by choosing a Hosted CI.

3 The "big" Cloud providers Microsoft, Amazon and Google

This section will describe some of the "big" cloud providers and give examples of their relationship to Continuous Integration. With the advent of the cloud, new services and options have opened up to developers. There are many advantages with utilising the cloud and more developers are relying on the "big" cloud providers, a trend that is likely to continue. Even though cloud has been around for over a decade, it isn't until recent years it has seen a massive growth[16]. With the market now seeing the potential with cloud computing.

The global cloud computing market size is expected to grow from $272 billion in 2018 to $623 billion by 2023[17]. The Three of the biggest public actors are Amazon Web Services (AWS), Google Cloud and Microsoft Azure with a combined market share of roughly 60%[18].

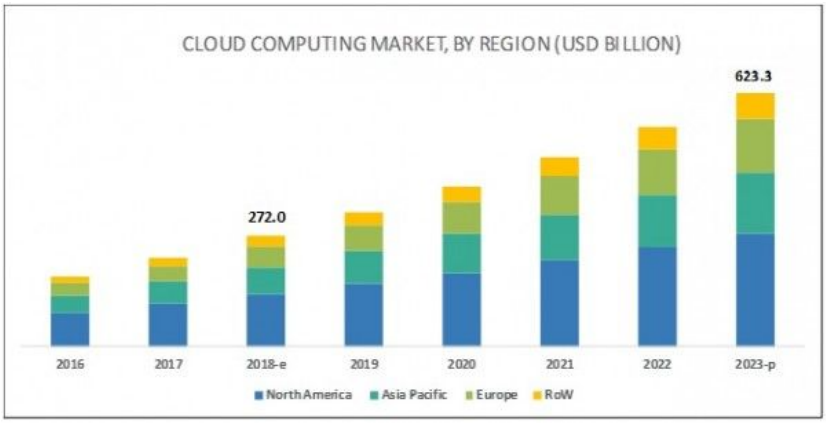

Figure 2: Projection of the cloud market[17]

3.1 AWS

AWS is by far the biggest player right now, and alone has roughly one third of the market share in cloud computing [18]. They offer services such as online storage, virtual cluster of computers to emulate CPU,GPU, RAM etc. databases, webserver, block chain, Internet of Things and much more. Furthermore, they offer DevOps tools such as Continuous Integration in the form of AWS CodePipeline. But also the option to set up other CI-tools such as Jenkins with their services. Currently AWS have support for Cloudbees, Jenkins, Teamcity[19].

AWS CodePipeline is Amazon's Continuous Delivery service, which is a practice that expands upon CI by developing all code changes to a testing environment after the build stage[20]. AWS CodePipeline runs on their services on the web and has many similarities to a hosted CI, therefore it comes with the advantages associated with hosted CI. However, as of writing it only integrates with one outside source provider, Github[21].

3.2 Google Cloud

Google Cloud Platform offers different services such as Storage, Machine Learning, GPU Computing, Data Analytics, Security, Networking and more. But also services within DevOps and Continuous Integration. Some of the tools that are offered are the Developer Tools which include Cloud Build, Jenkins and Tekton. Cloud build is Google's own tool which offers support for all languages and the designing of custom workflows for build, test and deployment between Virtual machines, Kubernetes, Firebase and more. Some of the differences from Jenkins are that the official support is more extensive and that it has container support for the build environment. Tekton is a open source tool for Continuous Integration and Continuous Delivery, that tries to modernize the field and create an industry standard. In Tekton the different tasks in the pipeline are executed in one or multiple containers, with the help of Kubernetes. This could be done both sequential or parallel and you can increase resources dynamically at runtime. It is even possible to build, test and deploy on different cloud providers. This leads to an increasing flexibility and is one of the reasons for its popularity[22][23][24].

3.3 Microsoft Azure

Azure is a cloud computing service from Microsoft that provides many different services. The product is similar to Google Cloud and Amazon Web Services. Some examples are services for Computing, Storage, Machine Learning, Blockchain, Internet of the things and more. Also the Azure Devops service is provided which was originally released in 2013. It supported the Visual Studio service online together with version control, testing and more. Azure Devops provides Continuous Integration pipelines together with version handling such as Git and support for any language. There is also support for deploying with

Google and Amazon. Parallel execution is supported in Azure as well as a new project called Pandas. Pandas tries to automatically test a pull request at different environments such as Windows, Linux and Mac. The basic pricing is free for open source projects and one free job a month for other projects if you don't want to pay [25][26].

3.4 Comparison

AWS, Google Cloud and Azure offers support for third party tools while at the same time offering their own tools as well. Although there are some limits to using third-party tools such as Jenkins on Google Cloud it should still cover the needs of the majority of users. Overall, the different services are very similar when it comes to handling CI and the decision on what tool to use is more a matter of company culture and taste. CI being such an important part of DevOps ensures that the companies will continue to support these services and support more in the future. Since the three cloud providers offer different services and infrastructure outside of the CI spectrum, it is important to look at other services related to ones project. They each have their own strength and weaknesses to take into consideration.

4 Conclusion

Continuous Integration is a well established practice today, with cloud providers offering different solutions. Although there exist some limits, there are many benefits. Given the current trend of cloud computing with more services moving to the cloud. The trends point towards increased options of CI integration with the cloud. It is impossible to say that one option is better than the other. One has to adapt the choices to one's own needs and weigh the pros and cons. A large company with a high need of privacy will maybe think twice before running in the cloud. However, a small company could make use of the benefits if the pricing is advantageous. It is also possible to start out with CI alone and later migrate to the cloud depending on the tool. And future development of cloud might add support for other CI tools, but keep in mind that not all CI tools are supported by the three cloud providers covered here. On the other hand, the companies might find it in their interest to not support more CI tools and rather promote their own tools.

The take-away message(s) of this essay is that Continuous Integration provides automation and correctness in large software projects. And as more of the traditional work as a software engineer is moving to the cloud, it is important to get to know theses new environment. There exist many different tools, both in the cloud or as an on premise service. A single answer does not exist, to companies that need to choose a CI service and each company should carefully consider their own needs. The big cloud providers do offer their own CI tools as well as support for others, on premise and hosted CI services. Finally, it is hard to draw a solid line between the different areas of Continuous Integration

and the Cloud because they are closely related.

References

[1] M. R. B. Setter, *The Benefits of Continuous Integration in the Cloud - via @codeship*. Available at https://rollout.io/blog/the-benefits-of-continuous-integration-in-the-cloud/,
(2017-08).

[2] Amazon, *AWS Continous Integration*. Available at https://aws.amazon.com/devops/continuous-integration/,
(2019).

[3] P. Duvall, S. Matyas, P. Duvall, and A. Glover, *Continuous Integration: Improving Software Quality and Reducing Risk*. A Martin Fowler signature book, Addison-Wesley, 2007.

[4] K. Yap and H. B. Singh, *10 things to look for when adopting a cloud CI/CD tool*. Available at https://bitbucket.org/blog/10-things-to-look-for-when-adopting-a-cloud-ci-cd-tool,
(2019-10).

[5] A. beginner's guide — Microsoft Azure, *What is cloud computing*. Available at https://azure.microsoft.com/en-in/overview/what-is-cloud-computing/,
(2020-03-30).

[6] Wikipedia, *Jenkins*. Available at https://en.wikipedia.org/wiki/Jenkins$_{(}software)$,
$(2020 - 03 - 20)$.

[7] Jenkins, *Jenkins*. Available at https://jenkins.io/,
(2020-03-30).

[8] O. Son, Bryant, *Building CI/CD pipelines with Jenkins*. Available at https://opensource.com/article/19/9/intro-building-cicd-pipelines-jenkins,
(2020-03-30).

[9] I. Heller, Martin, *What is Jenkins? The CI server explained*. Available at https://www.infoworld.com/article/3239666/what-is-jenkins-the-ci-server-explained.html,
(2020-03-30).

[10] Jenkins, *Getting started with pipelines*. Available at https://jenkins.io/pipeline/getting-started-pipelines/pipeline-defined,
(2020-03-30).

[11] C. Inc, *Simple CI CD Service Costs: Codeship Basic Pricing*. Available at https://codeship.com/pricing/basic,
(2020-03-30).

[12] CloudBees, *Wikipedia CloudBees*. Available at https://en.wikipedia.org/wiki/CloudBees,
(2020-03-30).

[13] Travis, *Travis vs Jenkins*. Available at https://www.guru99.com/jenkins-vs-travis.html,
(2020-03-30).

[14] Bamboo, *Bamboo*. Available at https://www.automation-consultants.com/bamboo-vs-jenkinsl,
(2020-03-30).

[15] D. T. Etherington, *Amazon AWS S3 outage is breaking things for a lot of websites and apps*. Available at https://techcrunch.com/2017/02/28/amazon-aws-s3-outage-is-breaking-things-for-a-lot-of-websites-and-apps/,
(2017-02).

[16] G2, *Exploring the Future of Cloud Computing in 2020 and Beyond*. Available at https://learn.g2.com/future-of-cloud-computing.

[17] M. R. Firm, *Cloud Computing Market*. Available at https://www.marketsandmarkets.com/Market-Reports/cloud-computing-market-234.html/,
(2020-03-30).

[18] F. Richter, Statista Infographics, *Infographic: Amazon Leads 100 Billion Cloud Market*. Available at https://www.statista.com/chart/18819/worldwide-market-share-of-leading-cloud-infrastructure-service-providers/,
(2020-02).

[19] AWS, *AWS CodePipeline product integrations*. Available at https://knapsackpro.com/https://aws.amazon.com/codepipeline/product-integrations/,
(2020-03-30).

[20] I. A. Freeman, Emily John Wiley Sons, *DevOps*. Available at https://aws.amazon.com/devops/continuous-delivery/,
(2019).

[21] S. G. P. Amazon Engdahl, *Blogs*. Available at https://aws.amazon.com/blogs/devops/integrating-git-with-aws-codepipeline/,
(2008).

[22] IBM, *Tekton-A-Modern-Approach-to-Continuous-Delivery*. Available at https://www.ibm.com/cloud/blog/tekton-a-modern-approach-to-continuous-delivery.

[23] J. vs Google Cloud Build comparison of Continuous Integration servers, *Jenkins vs Google Cloud Build comparison of Continuous Integration servers What are the differences between Jenkins and Google Cloud Build?* Available at

https://knapsackpro.com/ci$_c$*omparisons/jenkins/vs/google − cloud − build/*,
$(2020 - 03 - 30)$.

[24] Google, *Google Cloud Build*. Avalible at https://console.cloud.google.com/marketplace/details/google/cloudbuild.googleapis.com, (2020-03-30).

[25] Microsoft, *Azure*. Available at https://azure.microsoft.com/en-us/services/devops/pipelines/, (2020-03-30).

[26] Wikipedia, *Wikipedia Azure*. Available at https://en.wikipedia.org/wiki/Microsoft$_A$*zure*, $(2020 - 03 - 30)$.