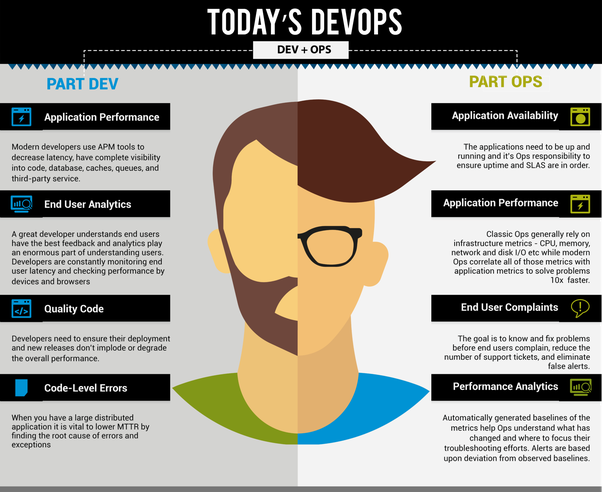
**he Rise of DevOps Engineers in the Current Market**



Today's DevOps

The current demand for DevOps Engineers in the market is rapidly increasing. This is primarily because these Engineers and their operations have resulted in great success for companies all around the world. Business organizations functioning with such Engineers are experiencing overwhelming returns compared to the firms that do not employ the services of these professionals. The engineers responsible can position codes as high as 30 times faster out of which, the success rate of each code positioned is doubled helping a company to gain that competitive edge.

However, the process of becoming an expert DevOps Engineer is highly complex. In order to be successful in this field, an aspiring individual must be well prepared and organized. Expertise only comes with extreme hard work and by having a good research background which serves as a platform for tech savvy freshers in particular.

**So how do you initiate your career as a DevOps Engineer?**

The most important factor which an aspiring DevOps Engineer should predominantly consider is to comprehend the basic fundamental principles of DevOps. People generally tend to disregard this factor and gravitate towards mastering the various tools of DevOps learning. Both are important concepts but without a strong foundation which can be only achieved by enhancing the fundamentals of DevOps, an individual can never be succeed in mastering this career option.

In my personal opinion, I have interacted with several candidates that only have interest in establishing the understanding of DevOps tools rather than concentrating on DevOps and its application itself. Another major misconception which people have is that they consider DevOps to be only related to automation. Tools such as Jenkins, Cron, and Hudson existed long before the popularity of DevOps which makes the prior argument redundant.

I also had the opportunity to interview various candidates for the position of the DevOps Engineer. The analysis that intrigued me the most was whether or not the applicant has a complete understanding of the DevOps concept. And thus, I always used to ask this particular question;

“*How would you explain the concept of DevOps to a person who has no technical background?”*

A difficult question indeed. But the people who have a true interest in this subject have always managed to come up with the correct answer. The usual answer is that DevOps in an automated framework that involves integration and strategic placement.

In technical terms, the answer stands true but it somehow never satisfied me. I always look to answer questions and solve queries in a practical manner. Mechanical definitions never fascinated me and should be the case in this field as well. The study will only make sense if a person has the ability to gather knowledge about the origin of the concept. Knowing the reason behind why DevOps came into existence and why there is a need for such functionality in the current economy will enable you to understand this postulate in a more efficient manner. It is also critical that an individual or an Engineer apprehends the challenges which a team faces while operating in this system. Only then will the theory and the fundamentals of this study begin to make sense.

Now to conclude the initial question of where and how do you begin your DevOps career, the most important thing is to emphasize on the principles and fundamentals before moving on to mastering the tools of the concept. The most effective way that I can suggest is to continuously research the topic. Gain valuable insight and experience. This can also be achieved by going through various novels and in particular, the book titled “Phoenix Project”. The content present in the book might be fictional but its application is definitely possible in the practical world. I would also recommend this book to business officials that are in the transition mode and are considering to incorporate DevOps in their current operational set up.

**Thinking from the point of view of a Developer**

An individual must be aware when the process of coding, re-engineering, testing, and scripting is being carried out. Gaining knowledge is one thing, and applying it on the field of work in another. Freshers reading this do not need to worry. You need not be an expert developer or programmer with complete knowledge about the DevOps tools. You just have to think smart, on your feet just as an experienced engineer would. Putting yourself in the shoes of a developer helps to solve queries in an efficient manner.

Personally, when ever faced with any challenging situation, I always used to count on my basic knowledge. Again, having a strong foundation helps. Learning the basic programming languages along with a complete command of the fundamentals will all help you down the line.

Knowing what a developer is most likely to do while developing a software or while framing and integrating the codes into an existing set up is the key to success. This aids in revolving vital issues which in turn, results in the positioning of the codes in a productive manner. Therefore, it is essential to know how the operation is to be carried out manually. Only then will the process seem easier when the tools of DevOps is incorporated in the long run.

**Gain absolute punditry in Operations before stepping into the field of DevOps**

I cannot emphasize this point enough. Gaining complete punditry in the art of operations and system administration before jumping into the field of DevOps is of absolute importance. A vital point for freshers and aspiring engineers to note. DevOps exists because of the advancement of the Information Technology sector. The survival of the IT was dependant on two factors, namely Sysadmins and Ops [engineers who were skilled in several coding and scripting languages]. With the help of the Sysadmins and Ops, various systems such as Linux and Windows were managed. Simultaneously it also helped to set up and manage various Web Servers along with the placement of strategic codes.

The sysadmins and ops engineers were proficient in Shell and used Script to reform and build automated function. All this was carried out way before DevOps was introduced in the Information Technology sector. The present generation engineers believe that such proficiency is no longer needed and therefore do not have complete control of these operational tools. This is purely a misconception that candidates should get out of their heads. Progress is made one step at a time. Similarly one cannot master DevOps if the concept and fundamentals of Ops are not clear. Thus, an individual must strive to become a specialist in administration/operations first, and then set sail on the DevOps journey. The following list will help you to gain more knowledge of DevOps engineering when processed correctly;

1. Learning how to operate Linux
2. Start to learn various scripting languages such as Ruby, Perl, Bash, Python, etc.
3. Study about Web Servers and the requirements needed to get it functioning
4. Practicing monitoring for the development of various infrastructures and softwares
5. Having a firm grip on the principles and fundamentals of networking
6. Development of servers manually without the utilization of tools
7. Practicing RDBMS, ext and NFS systems.

This may seem like a grueling task but it is bound to pay dividends in the long run. You will develop your skills from the roots up, so much so that tackling any challenges in the real time work atmosphere will become easier. Such is the importance of having operational and administrative experience before learning about DevOps.

**Learning how to effectively manage the codes**

Let us get one thing straight, in order to become a DevOps engineer, an individual must first have complete control of a Distributed Version Control System. DVCS umbrellas tools such as Mercurial which is of critical importance for engineers. DevOps initially became popular due to the presence of tools like Git and Mercurial, to begin with. The old school techniques of using FTP for the purpose of transferring codes are no longer followed.

The new wave and dynamic features of these tools helped to bring DevOps into the present market scenario. Thus, it is vital that you gain experience and learn how to use tools such as Git and Mercurial as it is more likely to be utilized and relied upon on a regular basis.

**Develop Jenkins as your long term server**

Let us begin this discussion by talking about Jenkins in particular. It can be simply coined as a solution to CI and CD. The sole purpose of CI is to collect all the codes originating from various developers and transfer it to a single system. This is carried out numerous times in order to avoid lag or downstream problems. Whereas CD, on the other hand, assembles all these collected codes and merges them together which then is utilized during production.

Jenkins has been existing in the IT sector for a very long time. It had captured a considerable portion of the market even before DevOps came into the picture. The incorporation of Jenkins as a tool with the Ops team proved to be very successful in the long run as the servers managed were more stable in nature which also featured automated functions. This eventually led to the domination of Jenkins in the CI and CD segments. There are other tools that are currently being used by DevOps engineers with regards to CI and CD segments. However, my personal preference would always be Jenkins as it is easy to operate and also is user-friendly.

**Efficient Configuration Management is the key**

The first rule of being a good DevOps engineer is never to be shy when it comes to getting your hands dirty. Therefore management of the identified tools present is extremely significant. This is where we link the factors we talked about before. With an unclear knowledge of operations, administration and the DevOps fundamentals, an engineer can never manage the configurations efficiently.

When I started to learn about DevOps, in the initial stage of my study I was very much intrigued by these configuration management tools such as Chef and SaltStack. I was absolutely fascinated with the fact that how configuration management and its tools can help an engineer to manage the infrastructure present as codes. Normally, OS installation was carried out manually which made it susceptible to errors. With the incorporation of the CM tool, possibilities of such errors are mitigated.

**Constant Monitoring in DevOps**

It is one of the factors which I personally feel should develop a habit. This aspect has been a part of the IT sector long before the existence of DevOps. Such tools help you to keep a track and monitor system records and its resources, in turn, increasing productivity, efficiency, and profits in the long run.

**Enter Virtualization**

This concept has been present in the IT industry for well over a decade. What DevOps did differently was that it provided engineers to configure, manage and develop machines on a virtual platform. All this and more is made possible by the existence of tools like Packer. Another sphere, which at present is high on the demand graph are Containers, which compared to the traditional virtualization are easier to operate and work with. Containers are popular simply because of Docker which I strongly recommend for the management of containers and virtualization.

**Upgrade to Cloud**

With the incorporation of the CM tools, the current market demanded a move into the cloud system. The management requirement of this cloud infrastructure initiated a higher demand of DevOps and its services. The engineers must consequently have a clear understanding about cloud providers and the services rendered by them. These cloud service providers also generate a quality certification which will surely add to your DevOps certification.

As already mentioned before, to become a DevOps Engineer requires a significant level of determination and hard work as it is no easy task to achieve it. Freshers and people with little or no experience in the technical field aspiring to be successful in this genre of study have an even more difficult task at hand. However, growth and success is only achieved with patience and perseverance. Therefore, do not be afraid to make mistakes as you can only learn from them.

