

# **Role Of DevOps in minimizing downtime on Applications using Continuous Deployment**

**Renugadevi Anbarasan, Yuxin Meng**

## **1. Introduction of Downtime**

As we all know, time is money. And for the computer industry, time is even more precious, however, the occasional network failure leading to downtime can result in considerable financial losses for the company. Downtime is a computer industry term that refers to the time when a computer or IT system is unavailable, offline, or not working. The Business Dictionary defines downtime as: “Period during which equipment or machine is not functional or cannot work. It may be due to technical failure, machine adjustment, maintenance, or non-availability of inputs such as materials, labor, power[1].” In this article, we focus on downtime in a computer environment.

There are many reasons for downtime, including maintenance downtime (called scheduled downtime), human error, and software or hardware failure. It can be divided into planned downtime and unplanned downtime depending on the causes.

Planned downtime is the time when the IT department needs to do the necessary maintenance and upgrades to the system to shut down the network. The planned downtime is inevitable, it can make the network function to maintain a better continuity. There are several reasons for the planned downtime: hardware replacements, system diagnostics, configuration updates, and so on. Taking advantage of planned downtime can minimize losses and enable employees to perform other tasks efficiently even when network conditions are affected.

Unplanned downtime is what people don't want to happen, and it does more harm to the business. Due to an unexpected system failure, an unexpected network outage can occur at any time. As for the causes of network downtime, there are many reasons that can lead to unexpected network failures. For example, 97% of IT personnel stated that human error is the cause or a contributing factor in at least some network outages[2]. Besides, hardware failures, disaster or crisis situations, and other situations could also cause unplanned downtime.

## **2. Impacts of downtime on deployment**

When the system fails, there will be downtime, which will bring great loss and impact. While companies can schedule planned network downtime to minimize these costs, unplanned downtime can lead to a number of unexpected costs that can have an impact on deployment.

### **2.1 Data Security**

The deployment process requires data protection. Loss of data during downtime can have an impact on the data center. It increases the vulnerability of the company[4]. In addition, customer data loss and corruption can lead to other costs.

### **2.2 Loss of productivity**

For global companies, the need to keep their systems in good working order at all times means that downtime can be inconvenient for users whenever it occurs. And the probability of error increases with each deployment operation, so you need as little downtime as possible. Sometimes employees and third parties need some kind of web service to get the job done. When these services are unavailable, affected parties will be unable to perform certain tasks, which reduces overall productivity and makes deployment difficult.

### **2.3 SLA penalty**

As reliance on data, IT, and network services increases, more and more companies require formal assurances from service providers, and as service-level agreements (SLA) become more likely to deliver services at their desired level. SLAs often use monthly downtime or availability percentages for billing calculation[3]. Failure to ensure uptime will result in financial penalties. Besides, planned downtime due to system maintenance, etc. is usually not included in the availability percentage of the SLA. Long downtime can affect the availability of their systems, which in turn can affect deployment. If the agreement can not be fulfilled as a result, then there will be financial penalties.

## **3. Use of DevOps to minimize the downtime**

To reduce IT downtime and its negative impact, companies need to find solutions that address the needs of their customers. At the same time, the solution needs to have a certain degree of security and convenience. As a result, many people opt for a DevOps strategy to meet demand and avoid downtime[5]. Simply put, the main benefits that DevOps can bring to an organization are increased communication, collaboration, integration, and automation. DevOps accelerates deployment by focusing on speed and automating as much as possible. DevOps also minimizes downtime by reducing the time required to solve application problems and changes the software architecture so that updates can be seamlessly integrated and implemented quickly.

### **3.1 Minimize planned downtime**

For planned downtime, the deployment strategy needs to be improved if downtime is to be reduced. For example, the more widely used zero downtime deployment[6]. Zero-down deployment (ZDD) allows you to deploy new versions of your system without disrupting existing services. From the user's and the company's point of view, this should be the best way to deploy, as new features can be introduced and bugs fixed without any disruption. The following are some of the more widely used deployment methods.

#### **3.1.1 Blue/Green Deployment**

Blue/green deployment is about building another complete production environment for the next release. The development and operations teams can safely build the next version of the product in this separate production environment. When the next release is complete, the user is automatically redirected to the next release by modifying the configuration.

#### **3.1.2 Rolling deployment**

Rolling deployment is the process of selecting one or more service units from a service cluster, performing version updates after the service is stopped, putting them back into service, and repeating them until all service instances in the cluster are updated to the new version, instead of updating all the service instances at once.

### **3.2 Minimize unplanned downtime**

In the case of unplanned downtime, things we can do with DevOps are monitor metrics and implement effective maintenance strategies[6]. With proper monitoring, it is possible to monitor and set up problems before they become larger and cause downtime, thus avoiding some of the downtime costs. So, based on the statistics that have been gathered, we can have alerts. Alerts will enable the team to keep an eye out for problems as they arise. In addition to monitoring metrics to prevent further problems, it is important that the system has an efficient maintenance strategy when problems occur. You can try to solve problems that arise by periodically inspecting your software systems according to appropriate maintenance guidelines.

## **4. Pros & Cons in CI/CD pipeline**

CI/CD (Continuous Integration/Continuous Delivery) product feature velocity is high. The high velocity improves the time spent investigating and patching defects. Continuous deployment, like Continuous integration, is associated with quantifiable threats. Once they are in operation, the pipelines become deployable unit routes.

CI/CD is a major pathbreaker in web services & SaaS (Software as a service) applications where the new features are being deployed constantly without downtimes. The development pipelines could well be in line with the latest deployment thus pushing a new feature without outage or suspension of the services rendered. Integrating the current technologies with legacy applications often maintain the believability of the service also providing new additional features [8]. This Continuous delivery model is often

appreciated hands down because of its nature in incorporating technologies & applications by making them usable to the vast majority of audience.

The Continuous deployment reduces the MTTR (Mean Time To Resolution) and makes it shorter and faster. It also isolates the fault so that the rest of the application/service works without being in a Stalemate. Increased and more streamlined release processes enable faster handling of issues towards resolution. It also improves the test reliability due to more minor & precise updates. Because of the said nature, it decreases the amount of errors in CI/CD pipeline thus making happy customers.

The major challenge in the CI/CD pipeline is choosing the right automation procedure. There are no defined rules to pick the right delivery model, but it's important that the chosen model stands for the organization's values without compromising the quality, speed and efficiency. Adopting a framework and an established way of working among the teams involved is always a challenge in large organizations. The developers of a dashboard are usually not the ones that will be using the dashboard. So, there are high chances we fall victim to logical fallacy. Therefore, getting the requirements right is as important as the development itself [9]. The DevOps plays a humongous role in bridging the gap between the IT & business customers.

Involving more teams in the CI/CD pipeline could be crucial because keeping them in sync and also autonomous requires human intervention. Some of the CI/CD between these teams could be connected and can only be deployed together where the said human element plays a big role.

## **5. Development Trend**

DevOps aims to provide a collaborative environment where development and operations teams can design, validate, and deliver applications rapidly, efficiently, and regularly. The organizations meet their goals quicker and deploy new features, bug fixes, and security updates with high altitude, resulting in seamless implementations. Beyond simply adopting this strategy, it is essential to embrace the recent technology innovations to obtain the maximum return on investment [11].

To adapt to the innovative technologies/solutions, a lot of organizations are open to use the following tools & software to reduce the downtimes.

### **5.1 Infrastructure Automation (IA) tools**

For monitoring IT infrastructure, data collecting tools are essential. They should, in particular, give a lot of information to assist you in determining the reasons for downtime, such as logs, specific issues, and the period when systems were down. Some of the most often used monitoring tools include Prometheus and Graphite. These technologies are extremely scalable and simple to incorporate into your existing systems[1].

IaaS tools can assist you in automating and managing all infrastructure components, such as networks, servers, applications, firewalls, cloud and on-premise managed services, and so on. You may use the infrastructure code to establish an environment whenever you want without much manual effort once you have it ready. Only the parameters would change, but the code would stay the same[10].

## Conclusion

IT teams often struggle with turning big business goals into projects and processes, like the development and deployment of applications. Reducing downtime while cutting costs is key for any business, therefore driving innovation by bringing on the DevOps team needs to be a core focus for any organisation. DevOps teams are designed to enact cultural and technological changes within organisations, and are typically built around the sole aim of reacting faster and more effectively to changing business demands.

## References

1. Alex. (2019, June 21). DevOps: How to Have Less Downtime. Alibaba Cloud Community. [https://www.alibabacloud.com/blog/devops-how-to-have-less-downtime\\_594973](https://www.alibabacloud.com/blog/devops-how-to-have-less-downtime_594973).
2. Collins, E. (2020). The Cost of Downtime. Nutanix. <https://www.nutanix.com/blog/the-cost-of-downtime>
3. K, B. (2017, September 6). DevOps is key to reduce IT downtime. <https://www.bestdevops.com/devops-is-the-key-to-reducing-it-downtime/>
4. UK Cyber Security Association. (2020, November 19). Common Security Threats Of Downtime. Cyber Security Association. <https://cybersecurityassociation.co.uk/common-security-threats-of-downtime/>
5. Suse. (2018, February 20). What is Downtime? Answer from SUSE Defines. SUSE Defines. <https://susedefines.suse.com/definition/downtime/>
6. What is Network Downtime? Unplanned; Planned Downtime. Worldwide Services. (2020, September 18). <https://worldwideservices.net/what-is-network-downtime/>
7. Rudrabhatla, C. K. (2020, October). Comparison of zero downtime based deployment techniques in public cloud infrastructure. In 2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) pp. 1082-1086.
8. Resource Center - Genesys Cloud [https://help.mypurecloud.com/articles/benefits-continuous-integration-continuous-deployment-ci-cd/?fbclid=IwAR1RxB3w11YQ-0ekmXKYJilRqelyQWj8AK4YFD-OdpuSj\\_M\\_OfpFjz\\_1Gzc](https://help.mypurecloud.com/articles/benefits-continuous-integration-continuous-deployment-ci-cd/?fbclid=IwAR1RxB3w11YQ-0ekmXKYJilRqelyQWj8AK4YFD-OdpuSj_M_OfpFjz_1Gzc)
9. C. Singh, N. S. Gaba, M. Kaur and B. Kaur, "Comparison of Different CI/CD Tools Integrated with Cloud Platform", 2019 9th International Conference on Cloud Computing, Data Science & Engineering, 2019, pp. 7-12.
10. J. Sandobalín, E. Insfran and S. Abrahão, "On the Effectiveness of Tools to Support Infrastructure as Code: Model-Driven Versus Code-Centric," in *IEEE Access*, vol. 8, pp. 17734-17761, 2020.
11. W. Hussain, T. Clear and S. MacDonell, "Emerging Trends for Global DevOps: A New Zealand Perspective", 2017 IEEE 12th International Conference on Global Software Engineering, 2017, pp. 21-30.