DevOps 2.0 for Digital Transformation

More from contributor

June 8, 2017

**By: contributor on 0 Comments**

We already know that DevOps is a culture, a movement, a philosophy of software development and delivery. It focuses on the communication, collaboration and continuous integration of the entire organization—a new conception of software development whose adoption is not easy. It aims to streamline software delivery by removing barriers between developers and IT operations teams. It also aims to reduce the inefficiencies of processes, adopting new tools, and even proposing changes in business architecture.

Agile practices allow development teams to deliver iteratively more frequently, among other things. In DevOps, the entire process cycle is automated to achieve continuous integration and continuous deployments

(CI/CD). The test and safety teams are involved from start to finish following an end-to-end strategy. At the same time, with the principles of Lean, we have removed from the whole process what does not add value. Thus, it is getting more productivity, maximum value and improvement in the time to market.
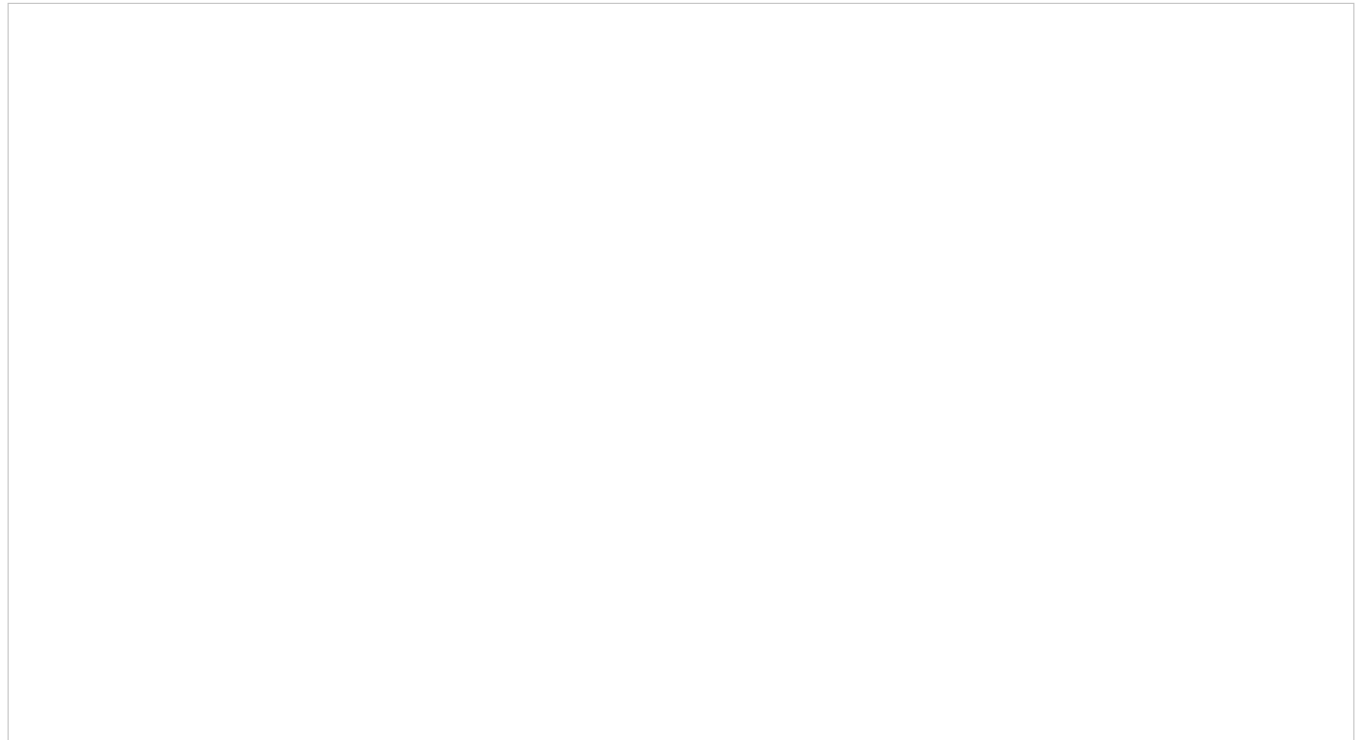
Talking about DevOps is thinking about techniques, patterns and good practices. It will help us to provide our organization with an efficient software architecture—an architecture based on microservices, with everything-as-a-service mentality (XaaS). We are talking about container, automation, elasticity, self-scaling, infrastructure such as code, monitoring, configuration management, self-sustaining and adaptable systems and ecosystem, all of which allow us to deliver value with fast, reliable and frequent developments, not to mention zero downtime and, of course, the ability to undo the changes.

How Do We Prepare to Meet User Expectations?

DevOps aims to help an organization to produce software products and services quickly. But end users are becoming more and more demanding.

We must use techniques that allow us to make faster, functional deliveries to the production environment, mitigating the risk involved.

At present, we face users with greater technological knowledge and greater expectations in the dynamic and interactive experience than we used to some years ago. This has evolved and given rise new one called DevOps 2.0 or BizDevOps (business, development and operations).

DevOps 1.0, as the origin of the movement, was focused on operations teams giving continuous feedback to development teams as well as to SQA teams. However, DevOps 2.0 is now focused on extending the benefits of feedback to the entire organization (marketing, sales, product, etc.). This is what will allow a greater adaptation to the IT boom and the digital transformation (IoT, intelligent platforms, big data, etc.). Additional benefit of DevOps 2.0 is faster continuous software deliveries, without compromising the stability of systems in production.

Toggling Features or Feature Flags

At this point, the key is to use techniques that helps to decouple the software delivery of new functionalities. In other words, it is about making these new functionalities available to the end user based on business rules instead of the operation teams. The idea would be that, for instance, marketing and sales department can decide, using "flags," when to release or enable a new functionality to the end user. It will allow us to deploy new functionalities to the production environment more frequently without enabling them to the end user. This removes the "fear" generated by the daily deployment processes in production environments. This is what is known, in DevOps 2.0, as the rollout.

Using this simple best practice for development, called Flag Driven Development, we do not make available the new functionality to 100 percent of users. It will be delivery in stages, starting with 1 percent of users, then with 10 percent, then with 30 percent, etc. with the ability to quickly enable or disable the functionality when something does not work as expected. In addition, it allows us to make available different versions of a functionality for different types of users (customer segmentation). There are several SaaS models and open-source solutions we can use to implement this type of techniques like Feature flags, toggles or controls.

What are the Benefits of DevOps 2.0 Techniques?

The use of this type of techniques during the development process provides these benefits:

- Quick scaling
- Stability of the system
- Integrity of the applications is not compromised.

This way we succeed in reducing the risk associated to each new delivery of functionality, in any productive environment. Business stakeholders can control when new features are available to end users, who continuously

demand new changes and improvements.

For organizations, DevOps 2.0 will bring DevOps power to non-technical teams such as marketing, design, sales and business. Now all teams will have the skills and responsibilities that are required in a DevOps culture. It enhances the confidence, communication and feedback that is needed from all teams in a DevOps strategy.

Therefore, one of the major pillars of the DevOps 2.0 approach is the ability to control, through a control panel interface, the launch of new features of applications in production environments. This process would be launched in a controlled way by both technical and non-technical people. Also, the process will be separate from the development and continuous deployment.

At the same time, this type of coding techniques will allow to perform real-

time analytics, making changes to the functionalities of a system that may also impact in the application performance monitoring (APM) tools. This makes it possible to determine when a particular functionality is degrading the global system performance and needs to be disabled quickly, as well as identify which functionalities are improving the end user experience (UI/UX).

There are a lot of articles about these "Fast Feedback" practices. I hope this article help you better understand the potential impact of DevOps on your organization.

About the Author / Jesús Pérez Franco
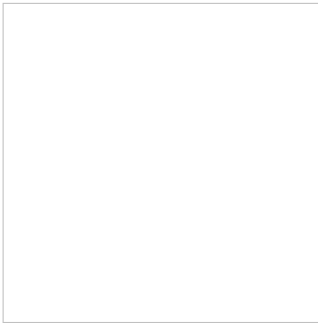
Jesús Pérez Franco is an Operations Manager at Gfi Spain. His big passions are Agile Development and DevOps Continuos Delivery (Continuous Integration, Deployment, QA Automation and Infrastructure as Code, etc.). Connect with him on Twitter and LinkedIn.

TwitterLinkedInFacebookRedditEmailShare