



**SOCIETY OF  
ACTUARIES®**

Article from

**CompAct**

October 2018

Issue 58

# DevOps: A Journey to an Efficient Regulatory Compliance Practice

By Soumava Dey

Regulatory compliance became more prominent for a variety of big organizations within the last decade and a few occurrences of audit compliance violations were experienced which lead to legal punishment in the form of huge federal fines. Being a part of an insurance company, I did find myself dealing with a lot of audit compliance and various IT regulations on several occasions.

It is obvious that when we manage customer's financial data we do have to comply with all regulations aligned with either business or IT functionalities. The Insurance sector has always been strictly regulated and it is more reluctant to change when it comes to its core functionalities like underwriting, risk assessment, risk projection, etc.; and the Actuarial department is not an exception to that. During my last few years with the Actuarial department I came to the realization that actuaries analyze financial risk and uncertainty using mathematics, statistics and financial theory. Their work is essential to the insurance industry as the actuarial models help businesses to mitigate the cost of a potential risk. So, all IT processes that support actuarial functionalities are subject to financial risk because any drawbacks in the process could cause a substantial impact on a company's actuarial reserve value. Sometimes it reminds me of a famous quote: "With great power comes great responsibility." Hence, it is deemed necessary to implement a continuous scrutiny and compliance review process to ensure the effectiveness of actuarial IT functionalities.

But how do we showcase our credibility to regulate the actuarial IT function so that it can adhere to the different audit compliances, for example, Sarbanes-Oxley Act (SOX), Health Insurance Portability and Accountability Act of 1996 (HIPAA), Financial Control Unit (FCU) and guidelines specific to its business? Fortunately, there is a software engineering practice popularly known as DevOps (combination of development and operation) which can be our best answer to the regulatory compliance issues.

## WHAT IS DEVOPS?

The term DevOps became quite popular in the IT industry in 2008. DevOps brings a culture containing a set of practices that could help companies to comply with different IT and business regulations. This technology not only brings automation to the development world, but it also makes sure that any IT functionalities like development and deployment practices should be reliable, traceable and repeatable.

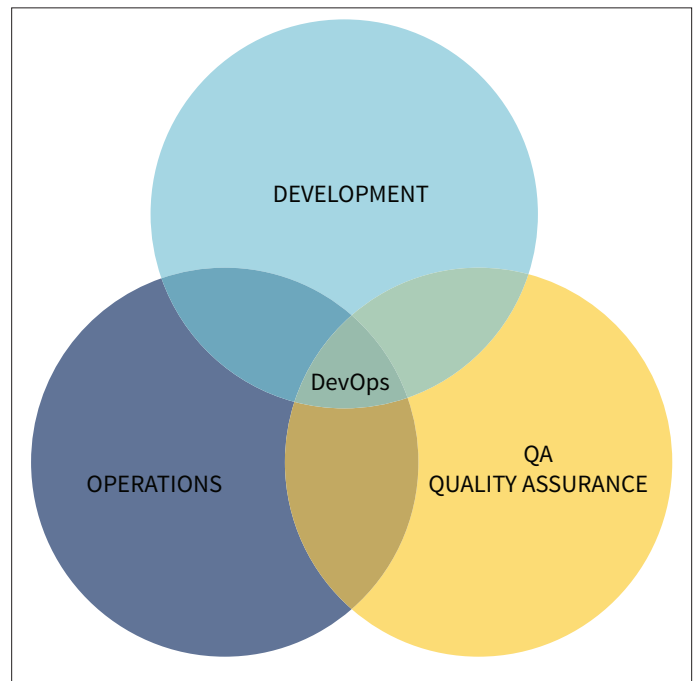
The term "reliable" means bringing transparency into the workplace by getting all the stakeholders (developers, testers, operation staffs, etc.) on the same page in terms of software development progress.

The term "traceable" introduces us to the practice of keeping all the project artifacts in a single repository where everyone has access to it. Eventually this practice leads us to implementing a secure version control solution in a software development world.

The term "repeatable" depicts the significance of a reusable process which supports the continuous integration of a software development practice in an autonomous environment.

Within an organization it helps employees to collaborate using the scientifically automated approach that combines software integration, continuous development and quality testing along with a proper monitoring activity. (Fig. 1)

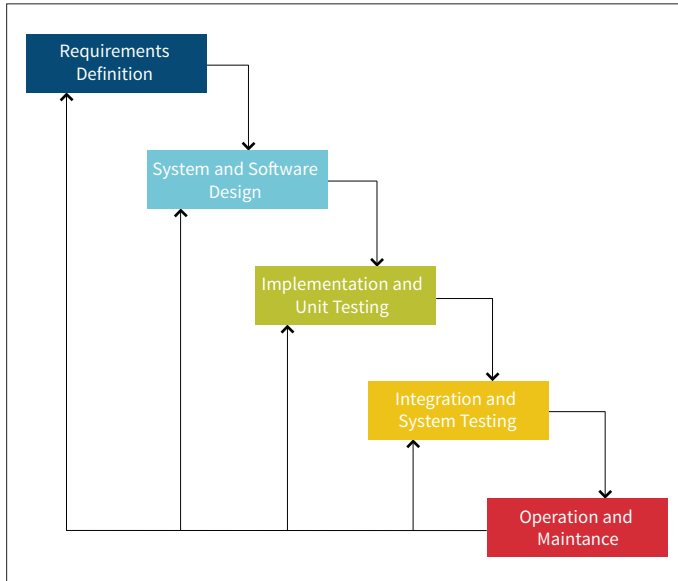
Figure 1  
The Elements of DevOps



## HOW IS IT DIFFERENT THAN THE TRADITIONAL WATERFALL MODEL AND AGILE?

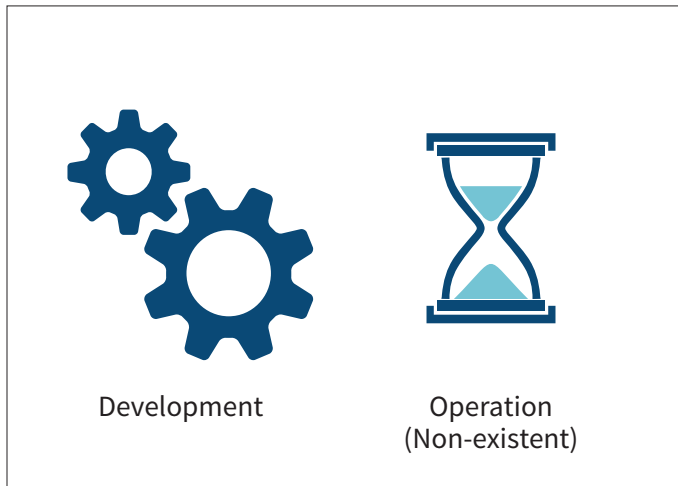
The traditional software development life cycle (SDLC) has mostly depended upon the Waterfall model (Fig. 2) since its inception.

Figure 2  
Waterfall Model



The Waterfall model is considered to be the linear approach in software development where results of each phase need to be signed-off before moving to the next one. That means all types of requirements need to be documented in detail before a project starts. Therefore, the scope of radical changes to the

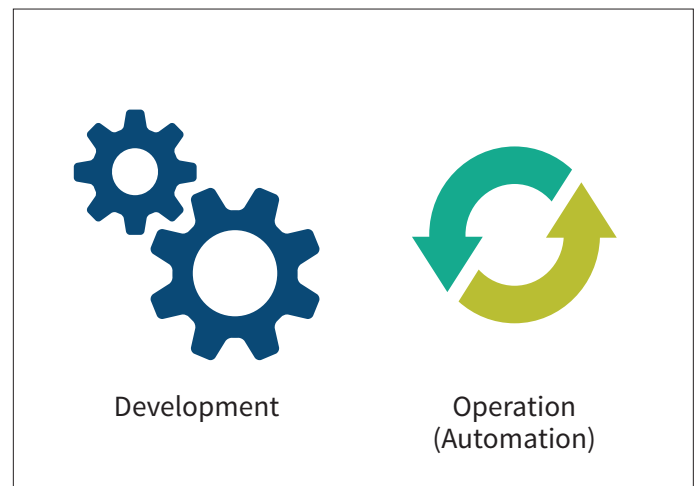
Figure 3  
Agile vs. DevOps



requirements during development time should be minimal. So, eventually the success of this development approach totally relies on well-defined plans and detailed documentation. But what if the customer will be dissatisfied with the end-product after the completion of the development project? It is most likely that rollback is not an option at that moment because it is very costly to repeat the development process. This is recognized as one of the biggest drawbacks of the Waterfall model.

But technology found a response to the waterfall approach which could make the development process more transparent, iterative and agile. The Agile process was the best approach that software developers found out at the beginning of the last decade in order to eradicate the limitations of the traditional model. There is no concept like “end product” in Agile because this approach eliminates the stereotypical concept of development and makes the software development process more incremental and flexible. The customer has frequent and early opportunities to see the work being delivered, and to make decisions and changes throughout the development project.

But although Agile came as a savior to the software development world, sometimes it crumbled due to the lack of communication between separate departments within the IT organization. This issue led to a movement called DevOps which eradicated the communication problem between developers and IT operations. It builds on a concept based on continuous communication between the developer, operation and configuration management team. Usually DevOps introduces an operations person who can help build communication between cross-functional teams leading toward a smooth transition from software development to deployment. (Fig. 3)





## WHY DEVOPS IS AN ESSENTIAL PROCESS FOR THE ORGANIZATION NOW?

DevOps is a mindset to break the barrier between the Dev and Ops. Building a culture of collaboration, transparency and faster communication is the basic foundation of any DevOps team. Often, it is difficult to communicate between various IT departments within the same organization. And that lack of communication could lead to a blame game when project deliverables go wrong. The introduction of DevOps builds a more trusting relationship and a more sustainable model of development process, run by the development and operation team.

As an example, DevOps tools like TFS, Atlassian JIRA, Basecamp, etc., are helpful in letting the teams collaborate and work together during each phase of a software development life cycle.

This is not just a concept used in the Silicon Valley Tech companies; it is a process being implemented across major organizations around the world. From financial services to health care, retail to manufacture, you can see the digital footprint of DevOps everywhere now.

Let's talk about some of the key aspects of DevOps that made it superior to any other development process models being used by the IT departments in past few decades.

### Culture of Collaboration

DevOps builds a culture which stands on "cross-functional collaboration" in every organization. All the development projects are meaningless unless the software developers, IT/Operation professionals and infrastructure put up a cumulative effort to deliver a quality product. In my experience, it is the best culture which can put the Agile process to better use in any circumstance. Most successful companies are on board with the DevOps culture based of a series of communications on a

periodic basis, keep focusing on customer's requirements and adjust project timeline if it is deemed necessary.

### Continuous Automation

Automation brings flexibility to DevOps. It brings more agility to the operations along with other important features, which are beneficial for audit compliance, as mentioned below:

- **Version control:** DevOps adopts source code tracking feature using the latest versioning tool available in the market. It means that developers can keep source code in a secure repository that is being offered by various version control tools like git, svn, tfs, etc.
- **Audit trail:** Implement an automation process to keep logging build, deploy and test results.
- **Security:** DevOps brings the concept of repository manager that can only allow certain people to make changes in the code repository and no one has permission to make changes in production unless they are from the IT Operations department.
- **Failover and recovery:** Automatically rolling back a deployment gone badly is way less error-prone than doing it by hand. And it's traceable, and it's faster, which helps you meet your service level agreements (SLAs).

When the IT department follows these four key aspects of automation logics in a repetitive manner, it not only prevents failure in the development project, it also keeps the auditor away.

### Documentation

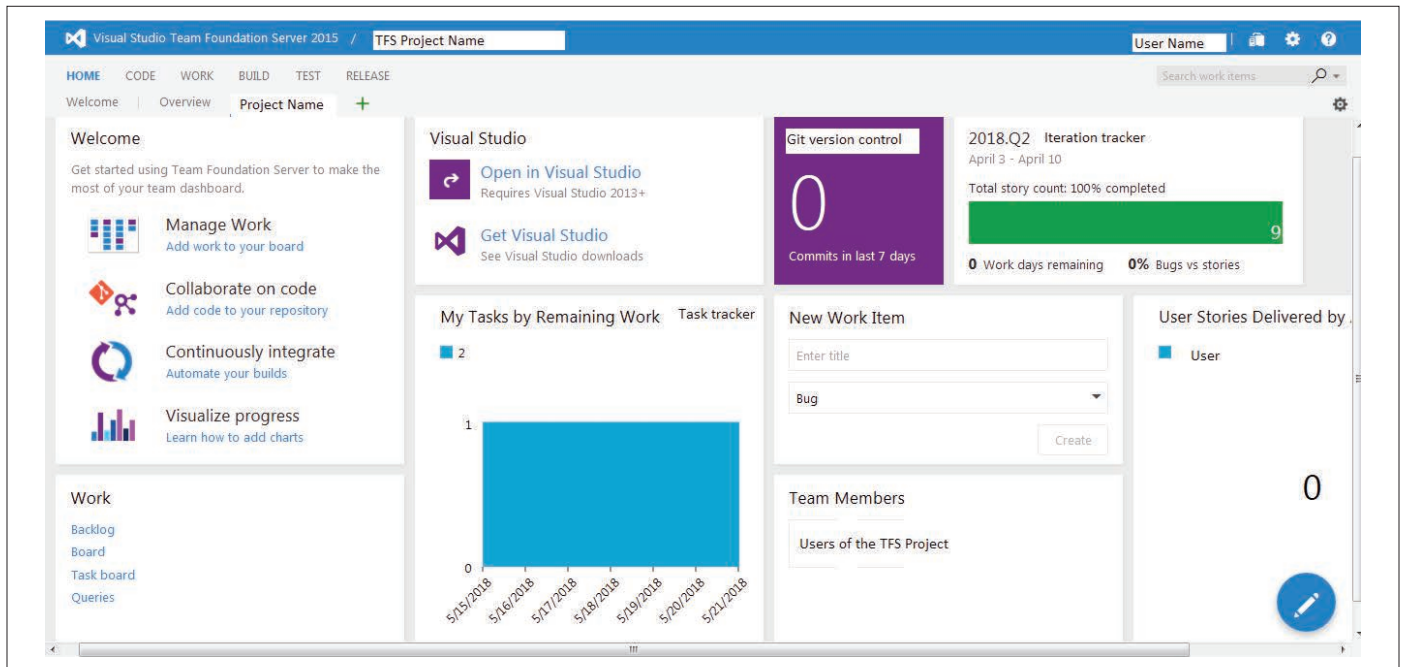
Unlike the Waterfall model, this process allows a customer to review documentation at each phase/iteration of the project. This approach always reduces the chances of delivering an end-product which is not aligned with the project requirement. So, by all means, documentation is required to make customers, stakeholders and team members aware of the project status, requirement change, and final deliverables at each phase, etc. Most DevOps enabling tools like Team Foundation Server (TFS) (Fig. 4), JIRA also introduced dashboard, team portal concept which provides a graphical display of project management work. A TFS dashboard looks like Figure 4.

### Measurement

When it comes to following a path of continuous improvement you need to set a benchmark at every starting point. There are different parameters to measure the improvement of a DevOps process. When it comes to process improvement we often asked these questions:

How long does it take to go from development to deployment phase?

Figure 4  
Sample TFS Dashboard



How many iterations do we need to complete a development project?

How long does it take for team members to complete a certain task?

DevOps often creates a good foundation that could easily capture the performance metrics and help us find answers to the above mentioned questions. That information comes in handy to help make team decisions, to create the future road map and to go for the next big move.

### IT'S A JOURNEY, SO MAKE IT A HABIT

I hope you have a notion about the DevOps concept by now. DevOps technique is more like a mindset than a process or a model, so any actuarial IT function needs to be more adaptive to include the DevOps practices to its development model. DevOps enables IT departments to achieve speed without risking stability and governance.

Any data project, data analysis or continuous development work could easily be monitored by the DevOps in the actuarial world. Main DevOps features like documentation, continuous integration and automation technique are becoming less intimidating to the auditors. In reality, we all know that the work priority changes frequently in a big insurance organization and it can be a cumbersome process for the managers to reallocate resources based on certain prioritization. DevOps is super-efficient to

help the actuarial managers who sometimes find it difficult to transition and prioritize the unplanned work.

Based on my own experience, I would like to emphasize the fact that DevOps plays a key role in transforming some key actuarial IT functionalities from a clumsy state to an agile state within a very short time span. Most of those IT functionalities include activities like managing vendor software, automating the production deployment process, handling SOX and FCU audit control practices, etc. The bottom line is DevOps brings the automation that helps organizations stay in compliance and reduces the overhead cost of extensive audit control process in every year. We all could foresee a future when DevOps will introduce a fully automated audit compliance process and eliminate most of the audit oversight processes that actuarial IT departments go through every year. Eventually that will give internal auditors more opportunities to engage in other compliance activities that happen every year in an insurance company.

In my humble opinion, DevOps is a journey which we all should embark on; trust me, in the future, you won't regret doing that! ■



Soumava Dey is an actuarial systems analyst at AIG. He can be contacted at [soumava.dey@aig.com](mailto:soumava.dey@aig.com).