DevOps: Misconceptions and Best Practices Whitepaper

**DBmaestro**
DevOps for Database

What is DevOps?

DevOps is a natural evolution of the software industry, enabling organizations to do more with fewer resources while differentiating leading and successful companies from their competitors. Delivering products faster with better quality has led to agile practices that are dramatically impacting how organizations deliver software. As agile delivers shorter and smoother development cycles - known as sprints - organizations are enjoying less risk by deploying smaller development packages with shorter delivery cycles. There is also a greater involvement of the part of development with understanding business needs and requirements.

Operations teams however, are finding themselves struggling to deal with releases on a much more accelerated basis. This has created many software release challenges as Knight Capital Group discovered when the organization suffered a $440 million dollar loss due to old software that was inadvertently reactivated when a new program was installed.

While agile has helped improve development productivity and quality, the next natural step of the software delivery evolution was to take agile to production - linking development with operations - which gave birth to DevOps. DevOps is a set of practices and principles intended to help development and operations work more effectively together. The challenge is how to effectively implement DevOps in a demanding business setting.

DevOps aim is aligning development and operations roles and processes in the context of shared business objectives, and organizing them into principles and practices. DevOps is about the culture and about creating better collaboration between development and operations. It is about building well-defined processes and about being agile.

DevOps and Continuous Delivery have been tightly connected and go hand in hand with efficiency, cost-reduction, better organizational performance, faster time to market and risk mitigation. However, many are still confused as to what exactly DevOps is and what differentiates it from Continuous Delivery. This white paper will provide an overview of these topics, and suggests best practices for implementing them.
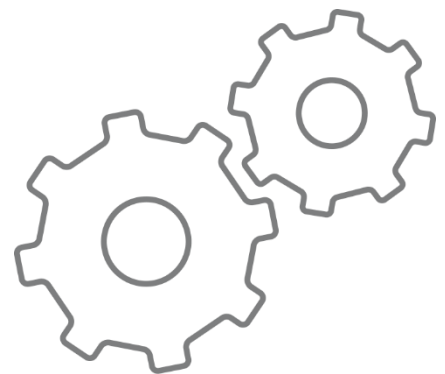
80% of Global 1000 organizations expected to have adopted DevOps by 2019
Source: "IDC MaturityScape: DevOps," December 2014

Key factors for successful DevOps

The main goal of operations is to maintain stable and healthy applications, while the main goal of development is to rapidly produce new features to meet business and customer needs. Change is implicitly the greatest enemy of stability, and understanding the conflict between development and operations is one of main goals of DevOps.

Bridging the DevOps conflict, or taking down the wall that has grown between these silos, is partly a cultural change and partly a technological change. Better communication, better collaboration, and more productive relationships between development and operations will surely contribute to a healthier and more productive organization, but just communicating alone is not enough to enable organizations to succeed in today's challenging business environment.

Technology and work-related concepts must support this transformation, or we will be left with a lot of good will but not much in terms of successful process improvements. We need to increase our overall efficiency and deal with risk resulting from business end users demands for frequent changes, robust features and new product releases.

Our experience has been that the key factors for successful DevOps are:

- **Adopting agile concepts over waterfall concepts**, thus enabling smaller, more focused and iterative blocks of development, that result in quicker time to market, as well as smaller changes which reduce the potential risk to operations, and ultimately, the customer.

- **Better collaboration and coordination** is either supported by (1) employees who drives change adoption and who's goals are to gather information and make sure everyone is in sync or (2) by collaboration tools, chats, portals, and conferences to ensure wider understanding by everyone on both "Dev" and "Ops" of the changes about to be implemented and what needs to be considered by both parties.

- **Automation** based on accessible information. This applies specifically to frequent changes. We cannot rely on people's ability to remember every action they take, as well as what the action might influence, what they need to take into account in case of scope of work changes, etc.

DevOps misconceptions

The fact that many people are still looking to properly explain what DevOps is, demonstrates how unclear DevOps remains to the general public. DevOps does not have a consistent definition within the industry as it changes with different impacts, so it is important to dispel some of the myths:

- **"DevOps is Continuous Delivery."**
  DevOps does not just deal with automation, but works to affect cultural change, create a better collaboration between development and operations, build well-defined processes, and be Agile.

- **"Continuous Delivery is automation."**
  While automation is an important part of Continuous Delivery, there is also a lot more to it. Continuous Delivery also works to build repeatable processes, create feedback loops, define the scope of changes, and determine what should actually be promoted or defined as "done".

- **"DevOps implementation requires DevOps teams."**
  This is actually the exact opposite of what DevOps advocates! DevOps works to break silos, not create a need for new teams and new silos! It is important to make sure that the business team talks to the development team, and that the development team talks to the operations team, etc. Everyone should talk! You can't achieve that by creating a new team that needs to talk to the Dev team, Ops team and everyone else.

- **"DevOps is Agile in production."**
  This one is not too far from the truth, but not exactly true either. Yes, the purpose of DevOps is to help the organization become more agile, but that does not mean it is exactly like agile development methods and implementation – just in in production. DevOps is about the cultural change, about creating better communication between Dev and Ops, and about the development team thinking about the operations team and vice versa. A better definition would probably be, "Extending agile and lean thinking from development to organization."

- **"Everyone needs to incorporate DevOps."**
  While DevOps does bring a heap of benefits, if nothing is flawed with what you have, why change anything? If you are quick to respond to market needs, are outmaneuvering your competition, and are well-managing your risks without DevOps, then you obviously don't need it. But if you are trying to make your system quicker, safer, and better, DevOps may make a lot of sense.

DevOps best practices

The potential of DevOps is clear, according to advocates who say it helps dedicated teams of development and operations staff quickly deliver innovative software with a minimum of errors. But few experts say implementing DevOps is easy, especially for large and established enterprises that must manage fundamental cultural changes and reformulate legacy programming practices.

Here are five common practices that can smooth the way to DevOps success.
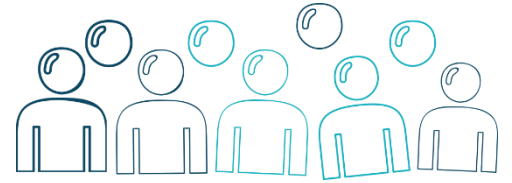
- **Breaking Down Silos**
  - Breaking down silos contributes to a reduction of duplication of efforts; the hope is that you get a better output that incorporates the best ideas from all employees throughout the company.
  - Breaking down silos is the opening up of information so that it's accessible to everyone. This enables people to see what's been tried in the past, the people who were involved, and maybe even insight into why it didn't work. Breaking down silos enables expertise to be leveraged across the entire enterprise.

- **Cultivating Trust Among Key Stakeholders**

  - Organizations can overcome mistrust and other sources of skepticism by highlighting DevOps wins as they materialize and making sure staff members inside and outside the IT department see the results. Promoting successes prevents people from proclaiming DevOps as "just a bunch of buzzwords."

- **Finding the Right Reasons to Move to DevOps**
  - Organizations shouldn't implement DevOps just because it is trendy or because of how it can improve IT operations. The catalyst for change should be fueled by business drivers. You should only go through this turmoil in the organization if there's something fundamental about your development process that isn't working for your business.

- **Filling Skills Gaps with Existing Talent**
  - Organizations new to DevOps may be tempted to jumpstart the endeavor by hiring experienced outsiders to staff the new teams. But experts discourage fast-tracking in this way. Instead, organizations need to develop their own people first by enabling them to achieve some early successes.

- **Implementing Continuous Delivery**
  - You can probably embrace and practice the DevOps philosophy without implementing Continuous Delivery. But that's inefficient, can be hard to maintain, and will be very hard to improve on. Automating everything from code, to infrastructure,

to database, and constantly improving these automations, will ensure your CD processes will evolve, mature and stabilize over time to contribute dramatically to productivity, stability, agility and safety.

DevOps and Continuous Delivery share a background in agile methods and lean thinking: small and quick changes with focused value to the end customer. They are well communicated and collaborated internally, thus helping achieve quick time to market, with reduced risk.

The combination of 'softer' or flexible DevOps philosophical concepts that go hand in hand with a very practical set of continuous delivery best practices, means that the two are evangelizing about similar, but not identical things.

DevOps is hot despite the fact that the term is broadly misunderstood. DevOps is not teams, it is not a bunch of systems administrators writing automation scripts, and is not just another software methodology. DevOps is an approach to delivering better, high-quality software more reliably that relies on a culture of teamwork and collaboration across all technical domains. But most importantly, a DevOps mindset is a key enabler for achieving scale.