DASA DevOps Fundamentals

Course Book

Copyright and Disclaimer

DASA DevOps Fundamentals | r1.4.1

Copyright

Trade Marks

DevOps Fundamentals™ is a registered trademark of DASA Limited.

Disclaimer

Information provided about the course, modules, topics and any services for courses including simulations or handouts, are an expression of intent only and are not to be taken as a firm offer or undertaking. The Publisher reserves the right to discontinue or vary or maintain such course, modules, topics, or services at any time without notice and to impose limitations on enrolment in any course.

The course materials provided may have hypertext links to a number of other web sites as a reference to users. This service does not mean that the publisher endorses those sites or material on them in any way. The publisher is not responsible for the use of a hypertext link for which a commercial charge applies. Individual users are responsible for any charges that their use may incur.

The publisher is also not responsible for the discontinued service if any, for example, the Web links in the course might not work in future.

The information in this course is written using a blend of British and American English. Although every effort has been made regarding the usage of correct spelling, punctuation, vocabulary, and grammar with regard to the Standard English, the publisher accepts no responsibility for any loss or inconvenience caused due to the regional differences in the usage of the English language.

Contents

Acknowledgements

We would like to sincerely thank the experts who have contributed to the development of the DevOps Fundamentals.

Lead Author

Rik Farenhorst, Business Unit Manager at Xebia

Rik has a PhD in software engineering and more than 10 years of experience in the business-IT domain. He has worked as Enterprise/IT Architect, Consultant, Trainer and Coach, and has later grown into sr. management positions. He guides organizations and individuals in raising their bar by making IT simpler, better, and of higher quality. Rik works as Business Unit Manager at Xebia, a specialized international IT consultancy, solutions and training company specialized in high-performance IT, digital transformation, and thought leader in DevOps. Rik is a member of the editorial board of the DevOps Agile Skills Association (DASA).

Michiel Sens, Solution Architect at Xebia

Michiel Sens has worked in the IT Industry for over 18 years, is co-writer of the book 'Continuous Delivery for Managers' and works as a Principal Consultant/ Solution Architect at Xebia in The Netherlands. Michiel specializes in Continuous Delivery and full lifecycle software development programs and as such focuses on coaching Agile Teams, setting up central build environments, automating tests, automating deployment over multiple environments and implementing fully automated PaaS platforms. He advocates the use of Continuous Delivery at seminars and meetups and remains in touch with details by performing actual implementations as well.

Frederik Schukken, Managing Consultant at Quint Wellington Redwood

Frederik is a Managing Consultant at Quint Wellington Redwood where he and his team help businesses to adopt disruptive thinking and create high-performance IT. He is passionate about helping others to adopt Lean and DevOps methodologies and break out of rigid, traditional ways of working.

Thomas Kruitbosch, DevOps Consultant at Xebia

Thomas is specialized in Continuous Delivery and optimizing IT for fast adoption of new technologies. Thomas has extensive experience and in-depth knowledge of both development and operations. As Consultant, Trainer, and Conference Speaker, Thomas advocates the use of Agile and Automation Delivery principles. Also, he performs technical implementations of Continuous Delivery Automation and (cloud) platforms.

Niels Loader, Managing Consultant at Quint Wellington Redwood

As an advisor to 10 of IT organizations, Niels has extensive knowledge and experience in implementing IT Service Management, IT Performance Management, Lean IT, and DevOps within IT organizations. In 2010 and 2011, he was one of the initiators of the Lean IT Foundation certification and spent four years as the Chief Examiner for the APMG Lean IT certification. He is the Lead of the Content team of the Lean IT Association.

1

COURSE INTRODUCTION

Let's Get to Know Each Other

Introduce yourself in the following format:

- Name
- Company
- Role and background
- Familiarity with DevOps concepts and their practice
- Experience in application development, infrastructure development, and/or operations
- Expectations from this course

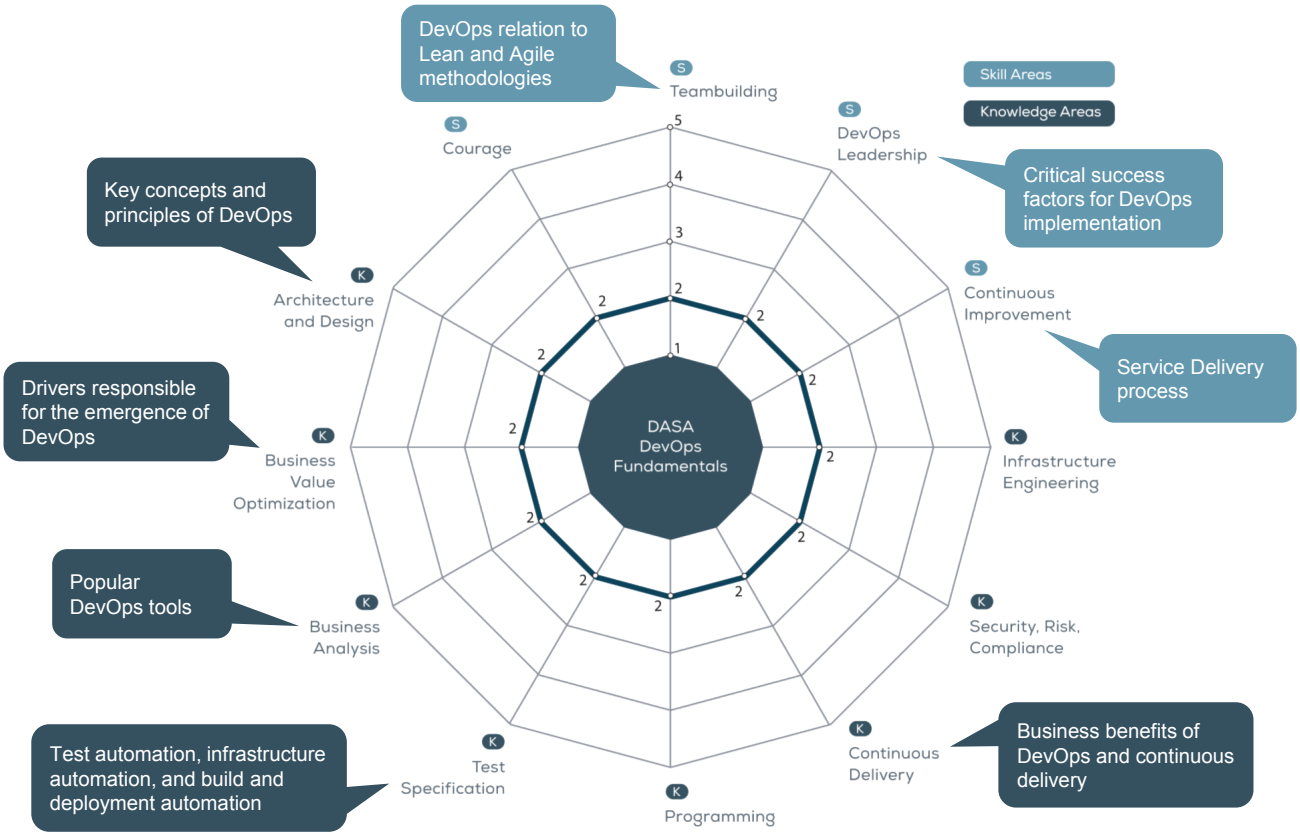
Overview

This 3-day course provides learners an extensive introduction to the core Agile DevOps principles. It covers all 12 key knowledge and skill competences that have been defined by the DevOps Agile Skills Association (DASA). With the help of key DevOps concepts and terminology, cases or scenarios, group discussions, and examples included in the course, you will acquire fundamental knowledge of DevOps.

DevOps Fundamentals is the starting point for anyone involved in an Agile and/or DevOps team. Improved workflows and faster deployment start with a core understanding of DevOps fundamentals by all team members. This course is designed to provide the core education necessary to build your DevOps vocabulary and to understand its principles and practices. The course will inspire you to serve as a

change champion by sharing and using what you have learned, and continue to learn, about DevOps to lead and mentor others.

Course Objectives



At the end of this course, you will be able to:

- Explain the drivers responsible for the emergence of DevOps.
- Define and discuss the key concepts and principles of DevOps.
- List and explain the business benefits of DevOps and continuous delivery.
- Describe the Service Delivery process.
- Explain the concepts of test automation, infrastructure automation, and build and deployment automation.
- Describe how DevOps relates to Lean and Agile methodologies.
- List the most common and popular DevOps tools.
- Discuss the critical success factors for DevOps implementation.

Course Agenda

DAY 1	
Module	**Subject**
01	Course Introduction
02	DevOps Introduction
03	Culture
04	Organization

DAY 2	
Module	**Subject**
04	Organization (Contd.)
05	Processes
06	Automation

DAY 3	
Module	**Subject**
06	Automation (Contd.)
07	Measure and Improvement
	Recap
	Mock Exam
	Certification Exam (Optional)

Though the Mock Exam is provided in the Course Book, you need to visit the DASA website, *http://www.devopsagileskills.org/*, for the latest version. Therefore, it is possible for the Mock Exam provided in the Course Book to be different from the one provided on the website.

Note: It is not necessary to appear for the Certification Exam on Day 3. You can attempt the exam later as well.

Type of Activities

Group Discussions

The course contains group discussions, spread out in all modules, with the intent of enhancing participants' understanding, adding context to the content, broadening participants perspective, reinforcing knowledge, and building confidence.

By interacting among themselves and responding to the varying viewpoints, participants tend to learn continually. These discussion allow the participants to come across the thoughts of their peers, which help them know about each other's past experience, perspectives, and opinions in the context of the topic in discussion.

Caselets

A case (or scenario) with related exercises and activities is used in the course. These exercises can include:

- Brainstorming
- Discussion Forum
- Group Discussion

Exam

At the end of the course, an exam will be conducted. The exam details are:

- **Bloom Level:** 1 and 2
- **Question Type:** Multiple Choice Questions (MCQs)
- **Question Number and Passing Mark:** 40 questions with a minimum passing rate of 65% (26 correct out of 40)
- **Time:** 60 minutes (15 minutes extra for non-native English speaker)
- **Exam Type:** Closed book
- **Suggestion:** Recommended that participants take the exam after completion of the course

Activity: ***Group Discussion***

Activity Time: 10 mins

Write down your expectations from this training on a sticky note and attach it to a wall.

MODULE SUMMARY

In this module, you learned that:

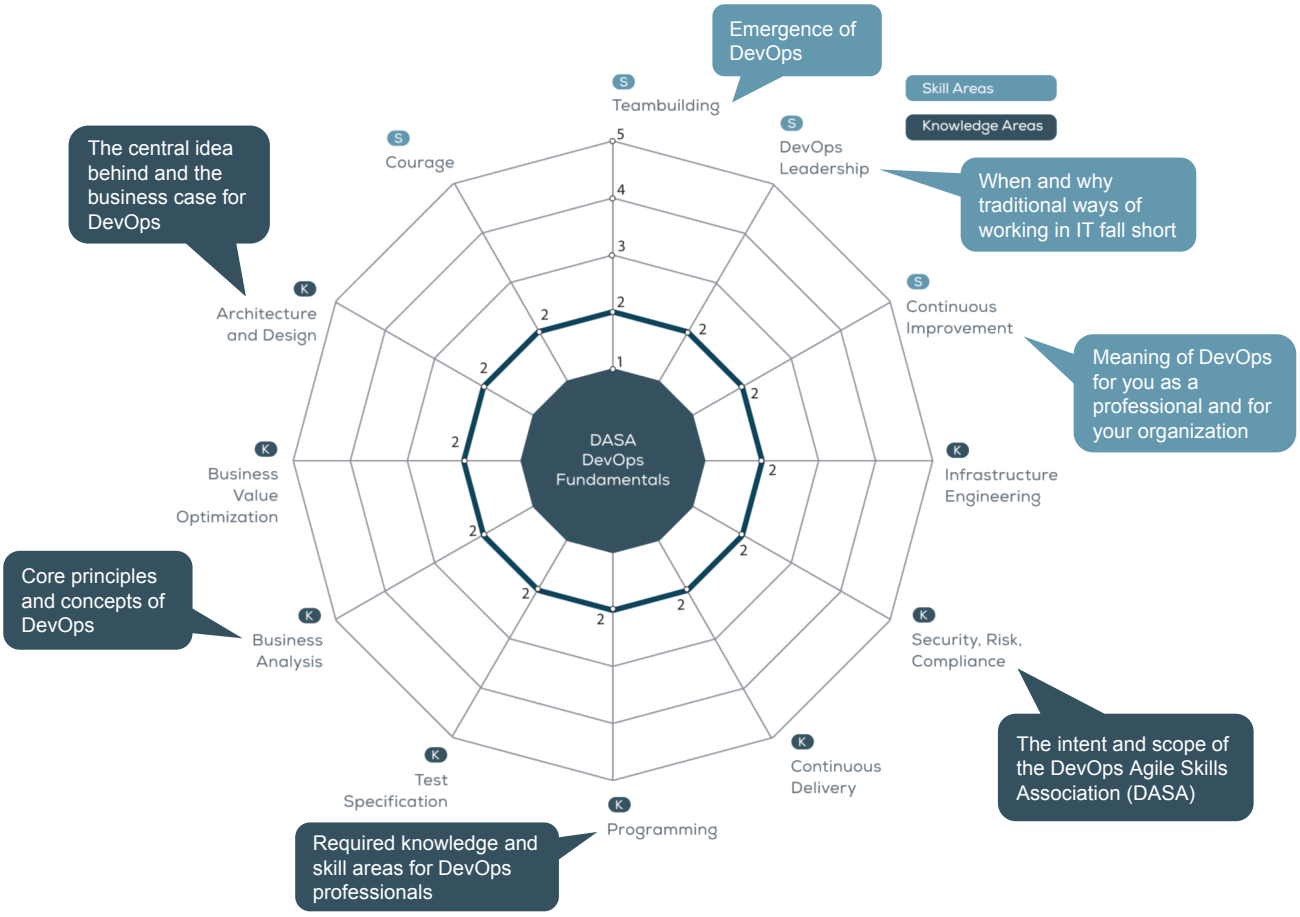
- DevOps Fundamentals is a 3-day course that is designed to provide the basic education required to build your DevOps vocabulary and understand its principles and practices.
- What are the various objectives that this course will help you accomplish?
- What is the 3-day schedule of the training?
- The course contains group discussions and activities for better understanding of the concepts.
- The exam of this course will have 40 MCQs, and its duration will be 60 minutes.

My Notes

2

DEVOPS INTRODUCTION

Module Objectives



At the end of this module, you will be able to:

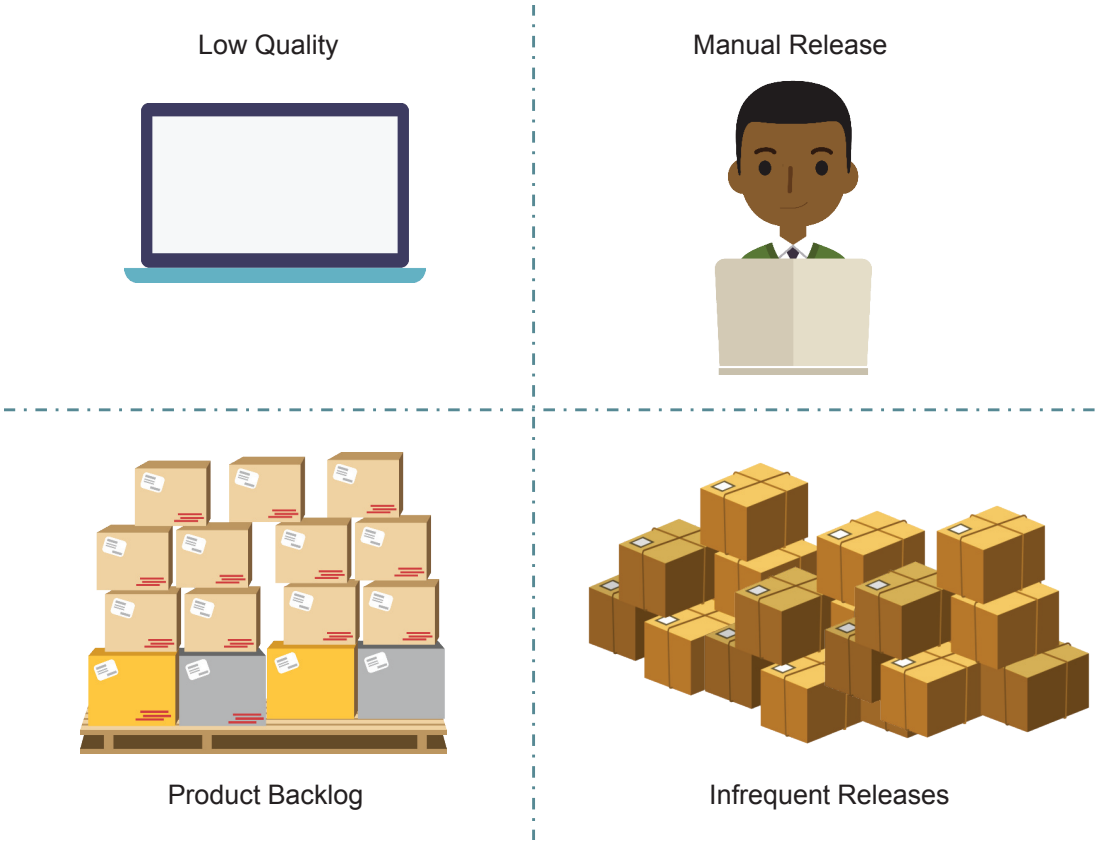
- Describe the emergence of DevOps.
- Explain when and why traditional ways of working in IT fall short.
- Define the central idea behind and the business case for DevOps.
- Explain the core concepts and principles of DevOps.
- Define the required knowledge and skill areas for DevOps professionals.
- Explain the intent and scope of the DevOps Agile Skills Association (DASA).
- Discuss what DevOps means for you as a professional and for your organization.

Module Topics

- Emergence of DevOps
- Core Principles of DevOps
- DevOps Agile Skills Association

EMERGENCE OF DEVOPS

Typical Challenges Traditional IT Organizations Face



In many organizations, IT is more a bottleneck than a strategic enabler or differentiator. In times in which "software is eating the world", it is a necessity to improve the overall quality of the IT capability. In many organizations nowadays, software is still delivered late, with many errors, released only a few times per year, and costs and processes are far from being Lean and mean. There is no option to sit still and do nothing about this. If organizations want to survive the ever fasting pace competitors enter their market, customer requires new services/ products as existing products become obsolete. Therefore, they need to deal with these typical challenges.

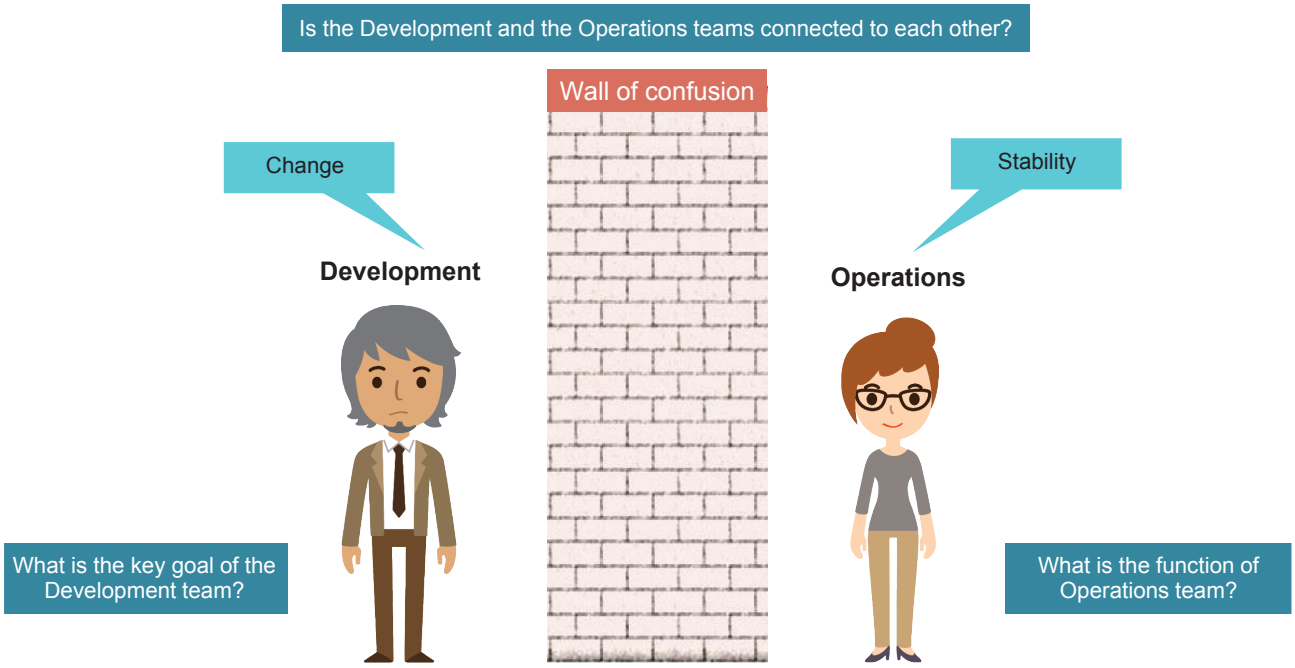
The business demands faster and continuous delivery. However, it is not easy due to various challenges that occur due to contracting goals of the various teams, especially the Development and the Operations teams, involved in the software development and delivery. The job of the Development team is to build software, apply changes to incorporate new features, and fulfill the internal as well as the external requirements. On the other hand, the Operations team focuses on stability, reliability, and performance of the systems maintained by them. The two competing contradicting goals of the two teams result in a wall of confusion.

Wall of Confusion

Think of the wall of confusion as a solid brick wall where no communication is possible between the people standing on either side. The traditional way of developing software is negatively impacted by this wall of confusion. If the Development team is working on an application, they hand it over to the Operations team only when it is complete. This approach has many loopholes and ultimately results in severe problems in production that causes blast like situation.

Need for Change and Stability Causes Tension

The typical challenge that traditional IT organizations face is to deal with the wall of confusion between Development and Operations.



The **wall of confusion** is a psychological and procedural barrier that obstructs the flow of communication between the Development and the Operations teams and can result in severe problems in production, such as:

- No methodical hand-over to the Operations team is done. Consequently, the Operations team faces problems in production

that they are unable to solve and look to the Development team for resolving the problem. Such a feedback loop delays problem resolution.

- In the absence of required discussions between the Development and the Operations teams during earlier phases of development, a lot of useful information is not shared between the two teams. Such information is crucial for the Operations team to get ready for the upcoming changes to the applications under development.
- The Operations team can share valuable information from their experience of managing the Production environment. This information can help the Development team design and develop more robust applications. However, due to lack of communication between the two teams, this information sharing is missed.
- A critical part of the transition between the Development and the Operations teams is knowledge articles. These articles help the Operations team to solve known problems. In the presence of the wall of confusion, these knowledge articles are missed. As a result, the Operations team spends valuable time solving trivial problems for which the solution is already known.

Activity: ***Group Discussion***

Wall of Confusion

Activity Time: 10 mins

What can be the possible problems that can arise due to the wall of confusion between Development and Operations?

Problems

Although the wall of confusion plays a significant role in the problems IT organizations face, it should be clear that dissolving the wall of confusion requires tackling a variety of underlying problems, such as:

- **Organizational Silos:** In many IT organizations, the Development and Operations teams tend to work in isolation from one another. This ensures that they are not confronted with each other until the critical moments of the delivery of IT products and services. The stress of these moments tends to lead to irritation rather than understanding.
- **Different Mindsets:** The Development team aims to incorporate new techniques or features to do their work efficiently. On the other hand, these changes not only to the IT service but also to the underlying components make the work of the Operations team difficult as such changes result in instability. Therefore,

the Operations team sees these changes as a challenge in maintaining the integrity of the Production environment.

- **Different Implementations:** The different implementations to perform the same work by the two teams result in incompatibility and lead to various bugs in the QA and the Production environments.
- **Different Tools:** The different tools used by the two teams lead to various errors and bugs in the Production environment. As an example, Development might deploy to a Test environment using a dependency management tool, while Operations might use a self-made script for the process.
- **Lack of Interest in Learning Other Tools:** Each team considers its tool or style of working to be the best and does not want to learn a new tool.
- **Different Environments:** The different environments, such as Development, Test, and Production, are one of the largest sources of errors and bugs raised by the different teams.
- **Loss of Work:** The various errors and bugs result in loss of valuable efforts.
- **Blame Game:** For each error, bug, or resulting incident, each team tries to ensure they are not identified as the cause of the issue. Therefore, they tend to pass on the blame of delayed delivery or errors to each other, leading to further irritation, lack of understanding, and intensifying of the wall of confusion.
- **Build Rollback:** Many times the build rollbacks are required as a result of a variety of causes, such as incorrect client requirements, incorrect database in the QA or Production environment, incompatible tools and others.
- **Disintegrated Processes:** The traditional structure of the various processes of the two teams are generally based on different frameworks, such as ITIL, ASL, COBIT, and Scrum. Therefore, development processes do not integrate well with operations processes leading to disjointed processes and different vocabulary, again intensifying the wall of confusion.
- **No Feedback Loop:** In many IT organizations, there is in fact a feedback loop. However, it is generally negative and strongly affected by the blame game. Lack of a feedback loop that is both positive and continuous in nature between development and operational processes causes the lack of understanding to increase.

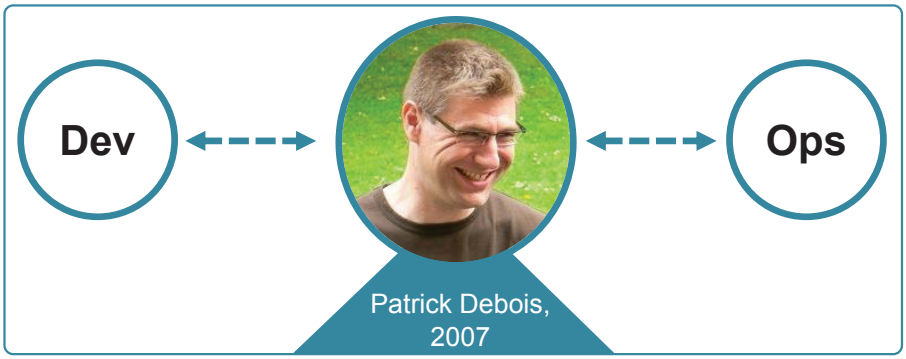
There are many symptoms and causes that enhance the wall of confusion to a point that a truly concerted effort is required to bring it down. DevOps encompasses a series of ideas, concepts, and concrete actions that focus on removing these symptoms and causes.

FOOD FOR THOUGHT

Learn about the different frameworks, such as ITIL and COBIT, and try to understand the relevance of processes in these frameworks.

A Brief History of DevOps

The problems due to the wall of confusion have been encountered by many IT organizations. DevOps took birth due to the wall of confusion between Development and Operations.



According to Damon Edwards, co-founder of DTO Solutions, the DevOps movement was germinated in Belgium back in 2007. Patrick Debois, an IT consultant, was frustrated by struggle, lack of communication, and disconnection between Development and Operations departments. He found himself straddling between the two teams, while working on a huge data center migration project for the Belgium Government Ministry. Patrick was performing the dual role of firefighting in the unpredictable world of IT operations and working on the Agile development. He was confident that there was a better way of working which would allow bridging the substantial gap between the two teams.

At the Agile 2008 Conference in Toronto, Canada, Andrew Shafer (a partner at Reductive Labs) proposed a discussion topic for an ad hoc session entitled 'Agile Infrastructure'. However, the session got cancelled due to lack of interest and feedback. Patrick Debois was the only to follow the discussion and eventually tracked down Andrew at the conference, and then they had an in-depth discussion about their mutual frustrations. The discussion gave rise to the Agile System Administration group on Google Groups by Andrew Shafer and Patrick Debois. Although the group was not overly popular, but it leads to some fascinating discussion. Moving forward to Velocity Conference, San Jose, the famous talk was given by John Allspaw and Paul Hammond of Flickr in 2009.

DevOpsDays was born, and the first event was conducted from October 30th to 31st , 2009, Ghent, Belgium. The event attracted administrators, developers, and managers from all around the world. The event's success inspired other DevOpsDays events in different countries. These events acted like a catalyst for the conversation and a grassroots movement. Although vendors, analysts, and traditional enterprise IT shops mostly ignored the movement, but it ignited the passions of those who were concerned. As a result, the movement began to flow and spawned tools, such as Vagrant, Puppet, Chef,

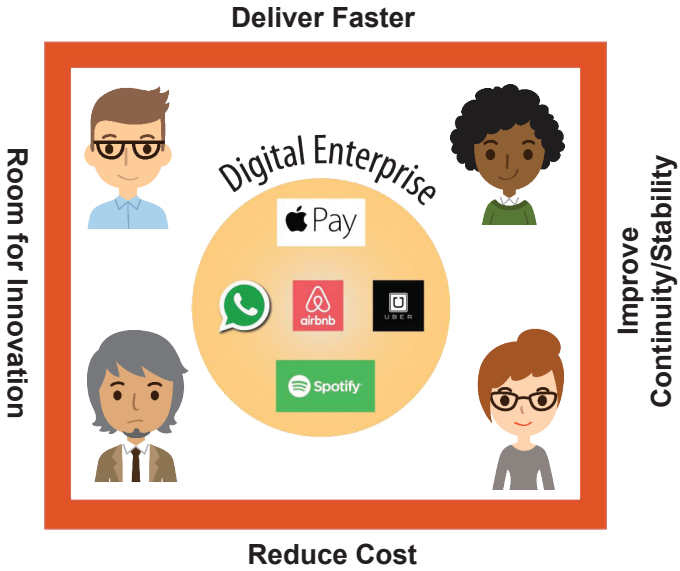
and FPM. The movement also started running circles around legacy enterprise IT systems.

Reference Reading:

https://blog.newrelic.com/2014/05/16/devops-name/

Benefits of DevOps

Fast movers displace traditional companies in all industry domains. To survive, companies need to radically rethink their IT strategy. Where will your company be in five years?



What is commonality?

- IT is the strategic differentiator.
- Fast movers use Automation, Continuous Improvement, and simplified operating models.
- Operations and Development are in sync.

Software lends itself very well for fast and dynamic delivery!

Many organizations have started to tear down the walls between business and IT, and the even thicker walls between technical departments within IT. They have replaced their technical departments with organizational forms that ensure quick feedback loops and short iterations. The leaders of such organizations are now starting to realize that IT is a strategic differentiator. Market conditions demand higher responsiveness and moving slow is not an option as the competition is eager to grow their market share.

CEOs read almost daily in the media, the stories about organizations that have been dramatically transforming themselves by adopting an engineering culture and moving towards a new world of IT. The success of extremely fast concept-to-cash or low time-to-market, and much lower operating and capital expenditures are enticing stories. Another

type of story, which offers equally interesting lessons learned, focuses on organizations that have either gone bankrupt or lost huge parts of their market share as they have been replaced by a startup or an App. In all industries, this digital disruption is happening. The examples, such as AirBNB, Spotify, Uber, WhatsApp, and Apple Pay, give a good impression of how industries are being shaken up or becoming redundant because of fast movers, who are able to develop and release new or enhanced services to customers on a frequent basis.

The common factor among all these fast movers is that they have triggered many of the traditional organizations to look at the world differently. Embracing a digital transformation is now key to survival and the four key goals listed on the preceding figure are now placed high on the agenda of top management.

TIPS

Marc Andreesen said: "Software is eating the world". The pressure to deliver faster, better and cheaper software products has considerably increased the focus on DevOps principles.

As Marc Andreesen said: "Software is eating the world". Therefore, making the right software in the right way matters a lot. The pressure to deliver faster, better and cheaper software products has also increased the focus on DevOps principles. As a result, considering the community-driven vision on collaboration and culture, DevOps has become a necessity for staying in business.

Cycle Time Reduction

Jack Welch, former CEO of General Electric and the writer of the book 'Winning', touches upon the essence why companies need to reduce cycle time reduction.

> "If the rate of change on the outside exceeds the change on the inside, the end is near."
>
> Jack Welch

Antifragility

In order to stay in business, digital enterprises need to be antifragile organizations.

> Antifragility is the ability of systems (or organizations) to get better as a result of shock, disruptions or disorder.
>
> Antifragility is about being the opposite of fragile. It means thriving on stressors.
>
> *Source: Design to Disrupt Whitepaper, Mastering Digital Disruption with DevOps, Sogeti VINT, March 2016*

Antifragile systems love randomness and uncertainty, going beyond resilience or robustness. Such systems get stronger with stress and volatility. Startups tend to be antifragile. However, large, successful organizations tend to be fragile.

IT organizations need to take on more antifragile characteristics as the world of IT is changing rapidly. Businesses need their IT organizations to be responsive and need their IT architectures to be adaptable to shocks or disruptions. The whole concept of DevOps is aimed at creating this kind of environment; creating organizational units that can autonomously act on the software and hardware required to deliver the IT service. DevOps requires short delivery times to ensure disruptions (in the form of different choices resulting from market changes) can readily be accommodated and used to improve the IT service. In other words, creating highly automated environments that can be easily reprogrammed if necessary.

Building Antifragile Organizations Using DevOps

Core ingredients of antifragile organizations are Management Innovation, Lean Startup, and DevOps.

FOOD FOR THOUGHT

Are "antifragile organizations" same as "robust organizations"? Think about it!

THE ANTIFRAGILE ORGANIZATION

MANAGEMENT INNOVATION	LEAN STARTUP	DEVOPS
Multiply funding for new initiatives	Implement experimentation systems	Continuous iterative innovation
Learn from the fringe	Experiment is a product	Embrace a culture of 'Fail Fast'
Community over hierarchy	Instill entrepreneurship	Make DevOps teams responsible
Employees first	Customer first	Empower employees to put customers first
Ensure transparency in decision making	Validated learning	Short feedback loops & measure end-to-end
Kill bureaucracy	Minimum Viable Bureaucracy (MVB)/lean	System thinking, crossing silos
Think competencies and platforms	Think digital innovation	Innovate digitally and use cloud platforms
Honor Web-inspired value	Honor end customer value	Honor end customer value
Reinvent management	Self-management	Self-management
Speed	Pivot	Flow
Human component back into focus	**Stop wasting people's time**	**Accelerate innovation**

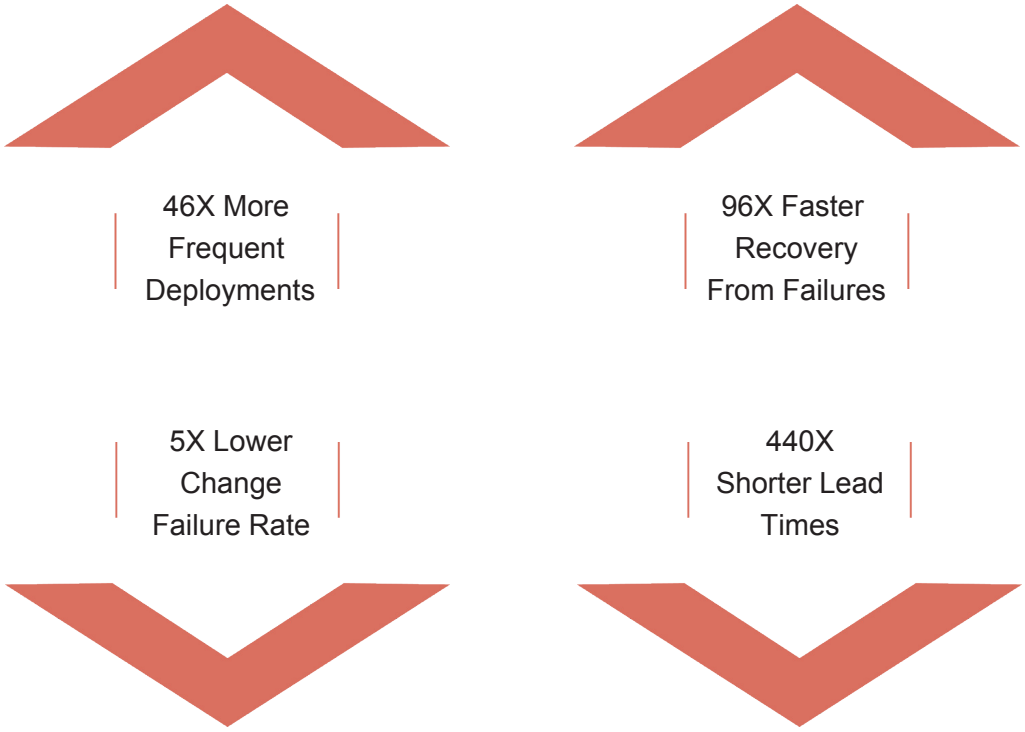
Source: Design to Disrupt Whitepaper, Mastering Digital Disruption with DevOps, Sogeti VINT, March 2016

DevOps is one of the central pillars on which many of the new breed of IT organizations realize a new modus operandi for delivering IT services. Using DevOps across the entire organization, sometimes dubbed "enterprise DevOps" or "BusDevOps", organizations redesign their business and IT departments using a new operating model that replaces traditional demand-supply models, centralized IT operations, and complex value streams with an excess of handovers, waste, and error-prone manual activities.

High-performing IT organizations have significant advantages over their competitors as a 2014 State of DevOps Survey by PuppetLabs indicated: high performing IT organizations deploy 30 times more frequently, have 200 times shorter lead times, have 60 times fewer failures, and recover 16 times faster!

Management Innovation, Lean Startup, and DevOps are three manifestations of the same phenomenon, a different way of working that is in line with modern practices. There are, however, minor differences. DevOps is the most holistic and more likely to take cultural aspects and the existing operation into consideration. Lean Startup tends to focus more on a method for product development. Both of these explicitly work on Management Innovation and put up a vigorous fight against bureaucracy, make teams and staff responsible, and urge the customer to get involved when it comes to digital innovations. This is how speed, staff engagement, and customer obsession come within reach of any organization.

Business Case: The High-performing IT Organization



Source: State of Devops report 2017

The preceding numbers are taken from the State of DevOps 2017 report. Over the past 5 years, they have surveyed more than 25,000 professionals worldwide to better understand how DevOps practices impact IT and organizational performance.

There is a tangible difference between high-performing organizations and their low-performing counterpart. One of the reasons that enable high-performing organizations to perform better is their style of working according to a set of patterns. This set of patterns is known as DevOps.

TIPS

The current State of DevOps report can be downloaded from *https://puppet.com/resources/whitepaper/state-of-devops-report.*

Business Case: Seven Reasons for DevOps

1. **Improved speed to market:** Increased speed to market allow an organization to gain a competitive edge in an industry. However, software and tools get outmoded almost as quickly as these are released. Introducing a DevOps approach enable an organization to go from an initial concept to a viable product in a shorter timescale than was traditionally acceptable with a Waterfall approach.
2. **Continuous Integration and delivery:** Continuous Integration is a development practice that involves deploying code to a shared repository several times per day. Using an automated build process combined with automated testing helps verify each

check-in, which produces more stable software. The updated 2015 State of DevOps report confirmed the finding of the 2014 report that a DevOps approach and culture allows organizations to deploy code more frequently, with shorter lead times.

3. **Higher quality, fewer failures, and higher stability:**

   a) **Higher quality:** The systems not only entertains the **functional requirements** (what the system will do considering what the customer wants) but also meets the **nonfunctional requirements** (unsaid customer requirements but expected from the system; how the system will do a particular task?), such as robustness, reliability, maintainability, and security.

   b) **Fewer failures:** The 2014 State of DevOps report showed that high-performing organizations had 50 percent fewer failures. The 2015 State of DevOps report showed a continuation in trend by revealing that the organizations who adopt a DevOps mindset and culture have 60 times fewer failures than those not implementing a DevOps approach.

   c) **Higher stability:** DevOps allows a single team to handle both, new functionality and the stability of the system. Each team member takes ownership of the business goals. Deploying often and in smaller, indivisible groups allows engineers to troubleshoot and resolve issues faster. The combination of tools and best practices, along with automation allows a DevOps team to increase overall stability.

4. **Innovation and creativity:** Continuous Integration, standardized production environments, and automated deployments allow practitioners to focus on the more inventive and creative side of their role. The environment and culture of DevOps encourage a deeper understanding and implementation of best practices in an organization.

5. **Increased employee engagement and job satisfaction:** DevOps provides a collaborative and multi-skilled environment, which contributes heavily to job satisfaction. DevOps practices and culture increase employee satisfaction, which leads to better business outcomes.

6. **Breaking down silos and eliminating waste; It is all about collaboration!:**

   a) **Breaking down silos and eliminating waste:** Combining multiple teams and disciplines into a single DevOps team that has a cross-functional skill set and communicates efficiently helps removing the organizational silos. In addition, applying the Lean focus on removing waste ensures that DevOps allows teams to complete their tasks quickly and efficiently while maintaining stability and quality.

Functional and Nonfunctional Requirements

Functional requirements are associated with the tasks a product/service should do. Nonfunctional requirements refer to the attributes of a product/service, such as autonomy, resiliency, maintainability, testability, scalability, and reliability.

b) **It is all about collaboration!:** Every individual and every single team, whether it is Development, Operations, QA, or Support, all come across challenges on a daily basis. These challenges impact the business and its operations, and an incident or impediment in one department can resonate in another. Regardless of who is responsible, the problem needs to be solved. It is unnecessary placing blame and pointing fingers. The key is to use the available time and resources to solve the issue.

   Without collaboration, this process takes longer and can create further problems that might not be immediately apparent. Working together and communicating efficiently allow teams to implement solutions that can help prevent similar incidents in the future. In complex enterprises with many fast-paced IT innovations, investing in collaboration helps to leverage all knowledge and skills available. It helps to work faster and smarter than ever before. Moreover, it raises awareness of the challenges that each department faces on a day-to-day basis and helps to thoroughly understand the business needs.

7. **Resource and cost reduction:** By implementing a DevOps approach, an organization is able to significantly reduce the costs and demand for resources associated with traditional IT implementations. When the organizations use continuous delivery and Lean Management practices, higher quality results and shorter cycle times are achieved, which further reduce costs. Other factors that help to reduce cost and resource requirements include minimal project startup and ongoing operational costs, increased collaboration, increased data availability and accessibility, and improved security.

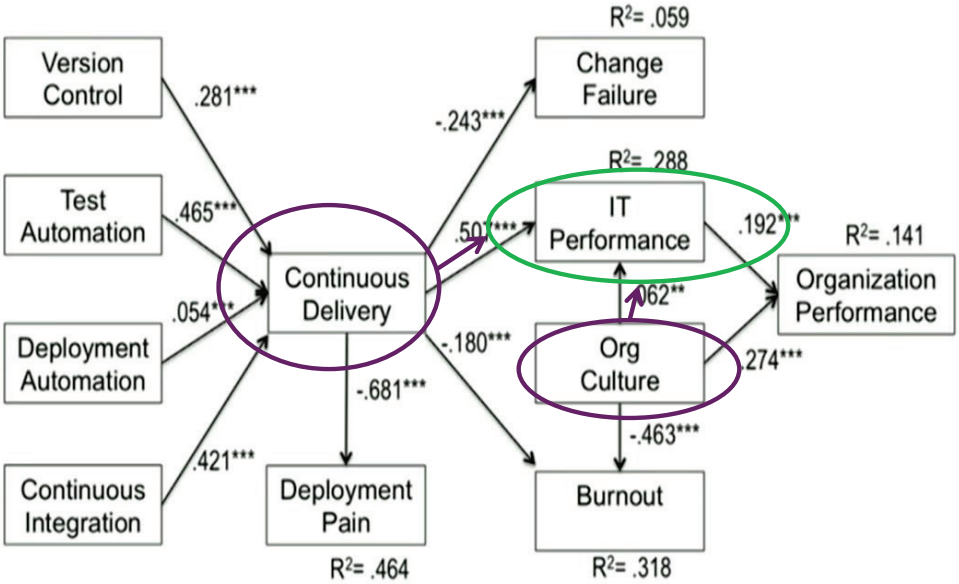
   Note: Cost reduction is not measured as part of the State of DevOps Survey. It is not a goal on its own and seen as a byproduct.

RESULT: Increased Performance

Standardized production environments and automation tools help make deployments predictable. These processes free people from routine tasks, allowing them to concentrate on the more creative aspects of their role. Consequently, it leads to increased performance of the people.

Research Confirms the Benefits of DevOps

The following Prediction model is based on the State of DevOps report 2016. It shows the relationship between continuous delivery, culture, and IT performance.



Forsgren, N., J. Humble (2016). "The Role of Continuous Delivery in IT and Organizational Performance." In the Proceedings of the Western Decision Sciences Institute (WDSI) 2016, Las Vegas, NV. Available at SSRN: http://ssrn.com/abstract=2681909

Source: State of DevOps Report 2016

This model is a result of research done on the data from the state of DevOps reports. The model highlights the interesting relationships between continuous delivery and burnout (engineers getting frustrated with the development and the release processes). The left part of the model contributes to continuous delivery and positively or negatively impacts the right part.

R2 is the percentage that the influencer impacts / has influence on the variance of the topic. Continuous delivery explains 5.9% of the variance in the change failure rate.

As the research has shown, based on data gathered from a large amount of organizations, continuous delivery and organizational culture have the most effect on IT performance!

Reference Reading:

https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2681909

CORE PRINCIPLES OF DEVOPS

Activity: ***Group Discussion***

Activity Time: 10 mins

What do you think are the core principles when "going DevOps"?

Some DevOps Definitions

> "DevOps isn't a thing. It's not a product, standard, specification, framework or job title. DevOps is about experiences, ideas and culture. It's about the close communication and collaboration between IT operations and development, and how they can improve the products and services that they produce by thinking differently about how they work together, using a new mentality."
>
> Gareth Daine, Devops Evangelist
>
> "Fundamentally, DevOps is the activity of optimizing the development-to-operations value stream by creating an increasingly smooth, fast flow of application changes from development into operations, with little waste. Optimization of the value stream takes place continuously using various continuous improvement techniques like the Toyota Kata."
>
> Dave Roberts, Executive Advisor at BMC Software

FOOD FOR THOUGHT

Find out some other definitions of DevOps and ponder over the similarities and difference between these definitions.

Hundreds of definitions for DevOps exist. Some essential elements of the preceding definitions are highlighted. These elements stand for something larger than a 'thing', such as a product, standard, or framework. This makes DevOps intangible but also applicable to enterprise-wide IT improvement and continuous innovation of the IT capability.

Considering the preceding three definitions, the four key points are:

- DevOps is about communication, helping tightly integrated disciplines to understand each other's roles and challenges in a better manner.
- DevOps is about working together toward a common goal. It enables businesses to succeed by helping them release quality and stable products and services, faster, and with fewer failures and bottlenecks.
- There is no overall authority on DevOps. It is a movement, inspired by practitioners for practitioners.