noris network

DevOps for Modern IT Outsourcing:

An end-to-end agile software development concept

code
plan
Dev
build
release
deploy
test
Ops
operate
monitor

06/2017

DevOps for Modern IT Outsourcing

Contents

Executive Summary

DevOps represents the expansion of agile software development to operative IT operations. The goal: Resolving of any cooperation problems arising between agile software development and IT operations. Software for volatile markets or applications with highly fluctuating resource needs (e.g., webshops, cloud computing) are particularly well suited for agile development and operations matched to this development. Premium IT service providers such as noris network employ in-house teams with experienced DevOps specialists and offer the setup, maintenance and continued development of highly efficient DevOps workflows as a service. Ongoing quality monitoring and provision are supported by a high degree of automation for resource allocation. This enables goals to be achieved which could not be attained with classic processes (long-planned releases, change requests, etc.): A high number of releases from agile development (time-to-market), a high level of security and performance (stability, availability) at lower costs for operations (through synchronized tool chains and automation).

noris network AG

1 Introduction

The DevOps movement: An end-to-end agile software development concept

The triumphal march of new methods in software development began at around the turn of the millennium. Borrowing from lean theories applied in industrial production, software development processes were to be rendered leaner and more flexible. The methods developed at that time, such as Extreme Programming, Scrum or Test-driven Development, are today viewed as a whole as agile software development and are lived in a number of companies.

Agile methods are applied primarily for software development, centered around the interaction between users and the user experience – for example, cloud applications or webshops. The reason behind this: Ideas and requirements in these applications must be transformed as quickly as possible into product features and, hence, into product and competitive advantages.

Agile software development changes the way in which software is rolled out. Instead of deploying one or a few extensive releases each year, developers are now making new builds available several times a week, or even several times a day. Although this may have a positive resonance from a developer's perspective, there are consequences at other points: Agile methods with a higher frequency of build releases do not take the necessity of automated, standardized IT operations with binding availability and planable utilization of resources into account. The consequence: When imported releases create problems, a “blame game” scenario arises between the development and operations teams.

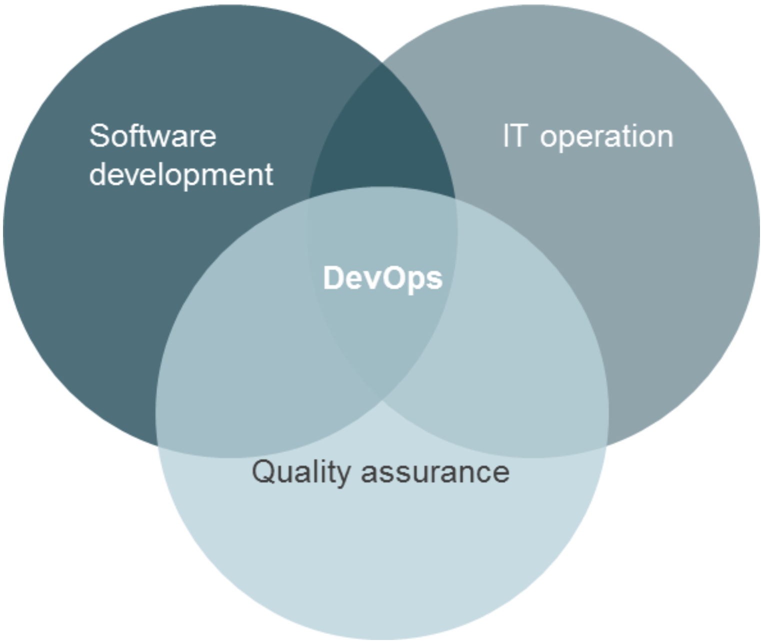
DevOps marks a cultural change

The developers and administrators who accepted the invitation of the Belgian IT consultant Patrick Debois to attend the DevOpsDay (https://www.devopsdays.org/) in Ghent were well aware of this problem.

DevOps, a combination of the terms Development and Operations not only stands for a catalog of technical measures, but is also described by its backers primarily as a culture for overcoming silo thinking. DevOps is about coordinating IT operations and rapidly ensuing software builds in a cooperative dialog between the development side and administrators. Central concepts here are Continuous Delivery and Continuous Deployment. The ultimate goal is to achieve a consistently harmonized process from development up to live operation of software, along with its further automation: from the release process, with the associated tests, up to provision of resources for ensuring stable operation of each new version.

Establishing of a DevOps culture will result in the development side giving more thought to downstream processes. If the test team, for example, works at a different rhythm, continuous delivery will be hampered. If the features of a new build require additional databases or more computing power, IT operations must be given the chance to provide these resources. At the same time, however, DevOps will mean the introduction of agile methods in the IT operations sector so as to make continuous deployment possible.

The DevOps culture not only includes the sectors of development and operations, from which the term gets its name, but also demands earlier testing, permanent monitoring and continuous assessment of customer feedback, along with planning of product development which arranges all of these aspects under economic factors. Even if the main focal points are corporate culture, communication and overcoming a silo mentality, DevOps is not just a concept that reduces frictional losses and unplanned additional work for correction of errors. DevOps represents the expansion of agility and lean to include the entire product life cycle and offers the competitive benefits of faster time-to-market and, at the same time, high availability and stability and reduced operating costs.



DevOps creates an environment of continuous dialog for different areas of responsibility and establishes a transparent continuous deployment workflow.

2 DevOps in practice: Organization and technology measures

2.1 New way of thinking and common responsibility is the key

Team building measures to overcome silo thinking, or group training of development and operations staff on lean and agile methods help drive the communication required for DevOps forward. And of course, a beneficial secondary effect of this is that the staff from both teams get to know and understand one another better. But measures for improving the work atmosphere alone are not enough: Management must create the organizational and technical prerequisites to ensure that cross-team communication and processes function well.

To put it simply: As long as stability is the sole criterion for success in IT operations, IT staff will always safeguard themselves to the greatest extent possible against disturbances – and the simplest way to do this is to reject any fast or non-meticulously prepared change. But on the other hand: If success is defined simply by rapid completion of features alone, the practicability of these features will continue to find little acceptance in operations.

One of the principle management duties in the introduction of DevOps is the elimination of assessment criteria which pit one team against the other and alienate teams from one another. Only when success has been defined as being achieved through cross-team collaboration can a sense of common responsibility be assumed.

2.2 Process optimization

An organization must nurture the basic lean principles and clear away all hindrances to efficiency independent of technical implementation of the delivery pipeline. Tell-tale signs that collaboration needs to be re-organized are unplanned additional work[1]or communications overhead. This holds true when tests reveal errors in software that was actually finished – errors which should have never occurred, or when this same information must be repeatedly communicated to other teams involved in the project.

In this context it must be emphasized that DevOps experts at IT service providers in no way should form an additional communications level or a type of administration level – as this would contradict the lean principle. Instead, the duty of such experts is to promote direct communication and permanently settle any conflicts involving process optimization. The best DevOps experts are those who make themselves superfluous over time.

2.3 Technical aids: The DevOps tool chain

From a technical viewpoint DevOps above all requires a tool chain for continuous integration, continuous delivery and continuous deployment based on an extensively virtualized IT infrastructure.

[1] Employees of a company who deploy a maximum of one software release per month reported that 38% of their time is spent for performing new work, 27% for unplanned additional work and 35% for other work activities such as meetings or routine maintenance. For employees in companies which deploy several software releases per day and which presumably have a distinct DevOps culture, these figures were 49 % for new work, 21% for unplanned and additional work and 30% for other duties. 2016 State of DevOps Report https://puppet.com/resources/white-paper/2016-state-of-devops-report

Agile software teams work in the DevOps environment with Jenkins for continuous integration of software components for example. Build automation tools such as Apache Ant or Gradle, and version management systems such as Git or Apache Sub and version and test tools such as Emma or Junit tie in well with this.

Tool chain and workflows for agile, continuous provision of new software versions are based primarily on a concept that enables further automation to be achieved: Infrastructure as code. All the processes and configuration required for setting up an infrastructure or for performing deployments can be represented as source code and handled and developed accordingly. This enables the standard solutions in software development for quality control to also be used for operations.

A further benefit of the infrastructure-as-code approach is the central administration of the source code while applying version monitoring. Moreover, testing of the configuration of servers and of virtual machines can be automated, thus ensuring the quality and repeatability of a deployment. A popular tool combination for this is the system management, provisioning and monitoring tool Foreman, together with configuration management tools such as Puppet, Chef or Salt.

If the IT infrastructure and environments for development, testing and operations are completely (or at least extensively) virtualized, any additional resources or reconfigurations required for a new build can be completely automated via APIs or corresponding consoles and provided in the infrastructure without any manual intervention.

3 DevOps for outsourcing of IT operations to IT service providers

3.1 Initial situation

The simple fact that a successful DevOps movement exists at all demonstrates that optimized collaboration between development and IT operations represents a cultural challenge even within a company. But when overall responsibility, development and operations are distributed among two (e.g., development provider and IT service provider) or even three companies (e.g., webshop operator, webshop developer and IT service provider) on the basis of an outsourcing partnership, the degree of complexity of the interfaces involved, as well of the organization, methods, tools and processes increases yet again.

Outsourcing, however, is for many operators of cloud applications and webshops an elementary component of their business model, allowing the providers to focus on their core business. The outsourcing partners ensure the scalability and reliability of the IT infrastructure on the basis of calculable costs – removing the burden posed by investment and operating risks from the company. Many companies have completely outsourced IT operations and do well both commercially and technically in placing the management of servers, storage, network infrastructure, connectivity, security and operating systems in the hands of operations specialists. Generally speaking however, outsourcing initially raises a wall between IT operations and software development.

How then is a company to exploit the benefits of an agile development with an outsourced IT infrastructure at an IT service provider?

3.2 DevOps as a service of premium IT service providers

Particularly innovative IT service providers are now beginning to react to the growing demands from the development departments of their customers. They are establishing their own DevOps experts in the sector of IT infrastructure who markedly intensify collaboration with customers‘ development teams and who automate and optimize the release process in the form of a deployment pipeline. noris network AG is a leading pioneer for this in Germany.[2]

In actual practice this requires, from a technical perspective, that the service provider set up pipelines which extend from the unit test, up through integration tests and on up to automated deployment in production. The interface between development and operations is both explicitly defined and bridged here by the deployment pipeline. The comprehensive degree of automation of the release process made possible by this not only renders this process repeatable at a high frequency, but also minimizes the risk of errors during operation at the same time.

In a service provider relationship, DevOps implies unusually close collaboration. Experienced DevOps specialists who form a team with the customer must be available at the outsourcing partner. These are employees who have an in-depth understanding of software and its development processes. Based on this understanding they can apply the requirements for stable operation to the application development. Important: Ideally, the work of DevOps begins at a very early stage – even when the scope of the planned hosting is typically not even known.

[2] Reports on this can be read in Funkschau 2015/01-08 and IT-DIRECTOR 2016/01-02 among others

How will the software developer design and implement software and how is the overall system to be managed after the project phase? Is the software specialist able to correctly assess the hardware and connectivity requirements? Do any special security requirements have to be taken into account? The DevOps specialists evaluate these topics and elaborate technical requirements together with the product owners, software developers and, where required, with specialists from the IT team.

The earlier this integration and discussion take place the better: All of the departments involved must work together from the very beginning in order for a DevOps culture to be established. One reason for this: Especially in the early design phase of software development decisions are taken which have a considerable impact on the test processes and later operation. A further reason: The earlier the tool chain is coordinated, the easier the transition to productive operation of the software.

Although the use of modern virtualization and container technologies with scalable cloud solutions often allows statements about necessary resources to be postponed “to a later date”, the foundations for subsequent scalability of the system must be set at an early stage. A minimum viable product can be used at the beginning of time-critical projects, for initial tests and user feedback. This product is limited to core functions, with other functions being subsequently delivered at the smallest possible increments and short iterations – with a continuous delivery strategy coordinated at an early stage.

4 Practical experience: Cost comparison between classic operations and DevOps

The previous statements and information make it clear: Anyone wishing to quickly and reliably incorporate the competitive edge provided by agile software development into operations and ultimately provide this to their customers will receive effective support for this by DevOps experts and services at their IT service provider. But what about the costs involved? What expenses are associated with teams in which the service provider's employees are involved, or with technical equipment for continuous delivery and continuous deployment?

To obtain valid statements about this, investment costs and management expenses for similar setups of two customers of noris network were juxtaposed. Both of these application systems consist of approximately 500 virtual servers and five to seven different interworking applications.

Classic operation of an application critical to business	Operation as DevOps cooperation
Implementation of the setup in physically separated environments for development, quality assurance and production. • Hardware and software equipment must be designed accordingly • Use of 50 to 60 hypervisors • Use of 20 dedicated security hardware systems • 8-10 dedicated database servers (use of dedicated databases on dedicated hardware systems for commercial/licensing reasons) • 40 to 60 core licenses from Oracle Enterprise • Not scalable by dynamic loading • Few dynamic scaling possibilities, rigid in the sense of technology and commercial possibilities	Implementation of the setup in physically separated environments for development, quality assurance and production. Hardware elements can, however, be connected more easily and more quickly to other environments thanks to the options provided by the automatic bare metal configuration. Loading of software can then be conducted at once. A uniform hypervisor structure can be set up through the use of non-extremely limited database software products. The number of hardware components is reduced through consolidation and virtualization of many of the hardware components. • Fewer hypervisors required • Use of 10 dedicated security hardware systems • Scaling options provided by the method of connecting the hypervisors • Database scaling provided using the same hypervisors and open license models of the database producers • Result is somewhat lower hardware expense
Initial equipment:	
• Stationary, simple setup, semi-automation easy to integrate • Relatively quick installation and migration, thus requiring 3-4 months lead time	Extra planning expenditure must be assumed for defining of the system architecture. Evaluation of the setup and its structure is considerably more complicated and must be accompanied by technically competent specialists. The setup must correspond to agile

	requirements in order to portray the value added for development and operations.
Operating expenditure:	
Rigid, stringently documented operating process. High demands placed on the customer and on the provider for change requests, in particular to avoid the creation of any incidents (outages) or security incidents. There must be a high level of technical and organizational competence on both sides. At the same time, reliability and a pedantic, continuous processing capability must also exist. Change requests must be examined and approved over several hierarchical levels. • Risk: Personnel is either overburdened or underburdened • Therefore, additional effort required according to the "double-check" principle • Greater control through the approval process • Greater expenditure on the Change Advisory Board (CAB) • On account of the relatively high risk involved, the CRs are, to a great extent, performed during less critical operating periods • As a result, the maintenance windows are completely booked, with a CR jam ensuing • CRs during operating periods must always be considered to be a risk • Full utilization of technical personnel from both sides • Greater attention to personnel from both sides so as to avoid hazardous overburdening of personnel	Standard changes are easy to perform due to the consistent setup of a continuous deployment pipeline in the past. Quality checks take place during development. A major portion of the changes can be performed in a standard administrator window. • Markedly lower change risk • Number of changes can be substantially increased • Marked reduction of implementation expenditure • Marked reduction of administrative monitoring
Expense analysis:	
Costs of initial hypervisors with software: EUR 1,250,000	Costs of initial hypervisors with software: EUR 500,000
Database license costs based on Oracle catalog price: EUR 2,000,000	None
Database maintenance costs based on the Oracle catalog price: EUR 400,000 per year	Database maintenance costs are pure operating costs and do not yield any license costs.
Initial personnel expenditure:	
Setup and migration: 2000 hours	Setup and implementation of the continuous deployment pipeline: 10,000 hours
Operation:	

200 hours per month	300 hours per month (additional 100 hours for maintenance of the continuous deployment pipeline)
Change Request expenditure:	
50 to 80 changes per week Expenditure per change 0.5 to 2 hours in ”double-check“ principle	Any number of changes can be integrated into continuous deployment
Provider expenditure per month: 600 to 800 hours	Provider expenditure per month: 100 hours, some due to unnecessary accompaniment of the deployment and automatic adaptation of the infrastructure
Expenditure quoted by the provider: 1000 hours per month	Expenditure quoted by the provider 400 hours per month
Based on this difference in personnel expenditure of 600 hours, the return on investment (ROI) for the markedly more personnel intensive DevOp solution is achieved in just over one year (from a provider cost perspective)	

As clearly illustrated by this comparison: The initial investment – time and costs – is markedly higher in the DevOps model, also due to the setup of the requisite continuous delivery/continuous deployment pipeline. These costs would be even higher if the customer would have had to implement all of these measures and infrastructures on its own, that is, if it had not profited from noris network’s many years of experience, along with its DevOps experts, its technical expertise in tool chains and the technical level of the virtualization platforms as in this setup.

A completely different picture is obtained for the ongoing costs for operations. When a high number of builds is put into operation using agile development methods, the setup of the pipeline quickly pays off, as the “manual“ application of these in the classic model results in substantially high personnel requirements at the outsourcing partner, which translates into higher costs.

For the 50 to 80 builds/patches assumed here, monthly expenses for operations will be more than cut in half. This savings effect increases even more with a higher number of builds, as the marginal costs for importing these additional builds drop to practically zero – whereas the manual effort required in the classic model practically remains high.

Conclusion: After about one year of operations, the DevOps installation will have "recovered" the initial additional costs through low operating costs. The maintenance effort cited here for the continuous delivery/continuous deployment pipeline, at 100 person-hours/month are even conservatively approximated.

5 Suitability for DevOps

Consequently, classic operations remains the more meaningful option when the associated application requirements can basically be fulfilled with a one-time setup and with few changes to the software and to resource requirements/IT infrastructure. In such cases, agile IT operations and the setup of a pipeline will pay off only over the long term, if at all. Examples of this would include established ERP systems or back-office systems such as Sharepoint and mail.

On the other hand, if a software has a high frequency of changes, for example if it is intended for use on volatile markets such as for webshops or cloud applications, agile development and operations scheduled in line with such development is almost a must. Other suitable applications include green-field developments and scaling requirements which are not apparent from the beginning. Continuous growth, which demands resource elasticity, is likewise better supported through DevOps and infrastructure-as-code than through classic, rigid forms of operation. Software startups in particular should look for an IT service provider with DevOps experience.

6 Conclusion

Today, the requirements placed on many companies for security, availability and scalability of their IT infrastructure have become so demanding that they can no longer do without outsourcing and support by IT service providers – both from a technical and from an economic standpoint. When IT service providers have DevOps specialists who form a team with developers from the customer (external or in-house), outsourcing is also possible, and even recommended for companies that develop agile software.[3]

The setup of a continuous delivery/continuous deployment pipeline tends to be faster and more reliably viable through the provision of experienced DevOps specialists by IT service providers. The initial investments required for establishing such a DevOps culture for process optimization and the technical setup of the pipeline remain markedly higher however than for classic installations with their Change Request processes. But these higher initial investments do pay off quickly – depending, of course, on the size of the installation and the number of software builds developed and put into operation using agile processes. ROI is typically reached after one to two years with staff experienced in DevOps available at the IT service provider who are able to provide their practical operating expertise to the customer's cross sectional developer team.

[3] High performance IT organizations which deploy releases 30 times more frequently than other organizations accomplish this in throughput times that are 200 times shorter. And the results of their work is even better: they experience 60-times fewer outages and recover from errors 168 times faster. 2015 State of DevOps Report https://puppet.com/resources/white-paper/2015-state-devops-report

noris network AG

Nuremberg-based noris network AG offers companies tailor-made ITC solutions for IT outsourcing, cloud services and network & security. The technological basis for these services is formed by a high-performance IT infrastructure with noris network's own high-performance backbone and high security data centers – including the NBG6 data center, recognized as one of the most modern and energy efficient data centers in Europe. In addition to individual, customized solutions, standardized premium data center products are also offered under the datacenter.de brand.

With its integrated management system, noris network AG is certified according to ISO / IEC 27001 (information security), ISO / IEC 20000-1 (service management) and ISO 9001 (quality management) for all business activities. In addition, the noris network's own data centers have received the ISO 27001 certificate based on the BSI IT Baseline Protection (Bundesamt für Sicherheit in der Informationsstechnik). The highly available data centers of noris network bear the maximum possible number of five stars, which the eco (association of the German internet economy) has to award in the eco Datacenter Star Audit.

Founded in 1993, noris network AG was among the German pioneers in the field of modern IT services and today collaborates with renowned companies such as adidas AG, Consorsbank, Flughafen Nürnberg GmbH, Firmengruppe Max Bögl, Küchen Quelle GmbH, Puma SE, Schmetterling Reisen GmbH & Co. KG, Teambank AG and many more.

noris network AG

Thomas-Mann-Strasse 16-20

90471 Nürnberg

Tel.: +49 911 9352-0

www.noris.de