KTH ROYAL INSTITUTE OF TECHNOLOGY

AUTOMATED SOFTWARE TESTING AND DEVOPS

STOCKHOLM, SWEDEN 2021

ITOPS VS. DEVOPS

What is the difference and how they can be used together?

Carl Leijonberg(carllei@kth.se) and
Ioana Cotutiu(cotutiu@kth.se

1 Introduction

Society has become notably dependent on computers and software mainly due to the global trend of increased digitalization. Nowadays, most modern companies are massively relying on their IT infrastructure to be able to perform most of their business operations. Hence, a well-established and functioning IT infrastructure is essential for traditional companies, that sell non-tech related products and services, as well as for technology-focused companies. Due to this dependency, appropriate tools and best practices were needed to build and manage the organization's IT infrastructure. The need for DevOps arose when technology companies required a more agile and quicker approach than ITOps to software development. However, when ITOps is used in conjunction with DevOps, it can generate a set of issues ranging from conflicting goals and practices to a potential need for responsibility shift between teams.

2 ITOps

Information Technology Operations (ITOps) is a broad term that can hold slightly different implications depending on how a company implements it. Generally speaking, ITOps broadly refers to all IT-related operations and involves any IT roles that require delivering and maintaining services, technologies, and applications needed for the business to operate. ITOps does not include software development roles which are instead a fundamental part of DevOps. Therefore, the development of applications, systems, and software is not incorporated in ITOps[1].

ITOps can be a particularly useful tool for IT departments and make their job of maintaining delivery of services, technologies, and applications significantly easier. Since the vast majority of companies heavily rely on computers to operate, making the IT department's job easier is important to guarantee that the business operates smoothly[9].

The main functions of ITOps can be classified into network infrastructure, computer operations and help desk, and management of service and device. For network infrastructure, ITOps is used to install and manage both internal

and external network functions. This can include providing users with remote access using VPN or two-factor authentication. ITOps also helps in ensuring that the company's network complies with regulatory and security standards. For computer operations and help desk, ITOps is used to manage the physical data center. This includes responsibilities such as managing electricity, battery backups, cooling, and overseeing backups of data in the event of data loss. It also includes the managing of the internal IT help desk. The IT help desk has the responsibility of responding to any IT issues that can occur within the organization. For the management of service and devices, ITOps handles the configuration and maintenance of network applications and IT infrastructure. It also handles the upgrading and patching of the IT infrastructure in the organization[4].

Due to the definition of ITOps being so broad, the responsibilities within ITOps are often divided into several groups. This allows ITOps teams to better define their team's domain and responsibilities. A common division of responsibilities is having separate teams focused on administration and maintenance, network management, systems management, and technical support[7].

3 Differences between ITOps and DevOps

Since the definitions for ITOps and DevOps are very broad, the lines between ITOps and DevOps can at times seem blurry, leaving room for potential overlaps. There is some overlap between ITOps and DevOps as seen in Figure 3.1. The principles behind ITOps and DevOps can help distinguish between them. Principles such as rigidity, flexibility, and delivery times are prioritized differently between ITOps and DevOps. ITOps' approach to installing and maintaining systems tends to be precise, rigid, and linear. This approach tends to reduce the amount of risk that can occur however, it lacks flexibility since it is difficult to respond to and implement changes promptly. This approach is often used for IT tasks with commercial-off-the-shelf software. DevOps prioritizes a more flexible and dynamic approach to completing tasks instead. In DevOps, there is a greater willingness to experiment with new approaches to find the fastest and most suitable solution. This is in contrast with ITOps where stability is prioritized

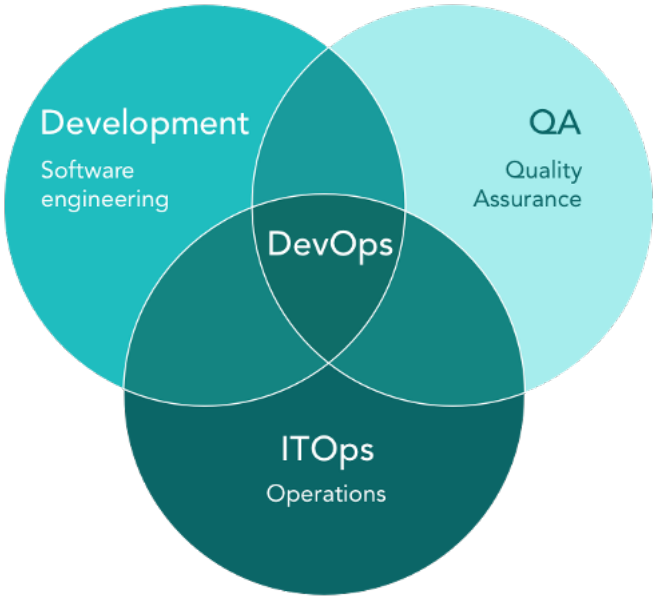

Figure 3.1: Relationship between DevOps and ITOps[7]

which reduces the willingness to experiment with different or novel approaches. DevOps also uses an agile approach to software development, which embraces changes. This approach breaks down the software development process into smaller "sprints" which allows the developers to apply new and smaller changes more frequently. Since ITOps prioritizes stability and security over dynamics when working with IT infrastructure, the ITOps process usually requires more time. In DevOps, short delivery times are of utmost importance. Their main responsibility is to reduce the software development lifecycle and to get the software to the end-users as soon as possible[4].

ITOps, unlike DevOps, needs to have a wider perspective over the organization's whole IT infrastructure to ensure it functions properly. Meanwhile, in DevOps, a broad view of the whole IT infrastructure doesn't have to necessarily be considered. Instead, DevOps places a greater focus on individual tasks. DevOps teams might therefore not always be aware of the downstream effects they might have unintentionally created on the technology stack at an organization [3]. Since ITOps also places great value in stability and long-time reliability as seen in Figure 3.2, they tend to use highly tested commercial software for the construction of the IT infrastructure. Updating software and hardware is a more difficult task

	Dimensions	Traditional IT	DevOps
Planning & Organization	Batch Size	Big	Micro
	Organization	Skill Centric Silos	Dedicated Cells
	Scheduling	Centralized	Decentralized & Continuous
Performance & Culture	Release	High Risk Event	Non Event
	Information	Disseminated	Actionable
	Culture	Do Not Fail	Fail Early
Measure	Metric	Cost & Capacity	Cost, Capacity, and Flow (Time)
	Define "Done"	*"I did my job"*	*"Its ready to deploy"*

Figure 3.2: A comparison of DevOps and ITOps[2]

in ITOps than DevOps since this can cause stability issues. Quick changes to the IT infrastructure are therefore not always possible in ITOps. We can see in Figure 3.2 that ITOps makes changes in bigger batches but more seldom unlike in DevOps [9].

4 The Ways ITOps and DevOps Work Together

Regardless of the previously discussed differences between ITOps and DevOps, their work is frequently intertwined, thus creating multiple interdependencies. If not taken into account and managed accordingly, these can quickly become problematic. For example, let's consider two teams that work together in an organization or company: one with a focus on DevOps and one on ITOps. The problem that we would like to highlight is related to an inevitable and frequently occurring phenomenon in software development, namely, change. Such a change can entail anything from new code pushed by the DevOps team to enterprise-level integration of new technologies, such as virtual storage, cloud, and microservices. Therefore, if DevOps fails to notify ITOps of occurring changes, then the system's stability and security are put at risk. Meanwhile, if DevOps waits on feedback from ITOps in every iteration, then the process will become too slow[1].

At first sight, this might seem like a circular and complex problem that is difficult to solve: if changes are not notified then the system is at risk, meanwhile, waiting for feedback in every iteration creates delays. Thus, the solution might come as

a surprise: communication. Since at the core of the development process lays the collaboration between multiple teams, their coordination, and their ability to efficiently cooperate and communicate is essential. Aligning and coordinating the DevOps and ITOps teams comes as helpful in preventing potential arising misunderstandings and conflicts. Moreover, from a broader perspective, this optimizes the software development process[1, 3].

From another perspective, data acts as a unifier between DevOps and ITOps teams. More specifically, when both teams are provided with real-time, and accurate data, a foundation for decision-making is created. This data is extracted, under a standard model, from a data pool. Hence, based on this common data model, the DevOps team can ground its code on reliable data sources. When the code is pushed into the enterprise, both teams know when this happens and how the introduced changes will impact every associated downstream function[3, 6].

In theory, there are some claims that ITOps should abandon certain areas of responsibility to ensure that DevOps possesses all the required tools for a rapid and innovative development process. However, in practice, both DevOps and ITOps must efficiently communicate and cooperate to ensure that the product under development satisfies the core features provided by both concepts: having frequent and timely releases while preserving its reliability and security features[3, 10].

5 The Process of Selecting Tools in ITOps

As previously mentioned, ITOps is primarily focused on help desk and computer operations, device and server management, and network infrastructure. Therefore, the tools that are used in connection with it range from various monitoring tools, such as application performance, network, and IT infrastructure monitoring, to management tools, like configuration, and log management. In addition to these, enterprises often use IT operations analytics and AIOps (Algorithmic IT operations) tools. For each category, there is a plethora of available tools on the market, and ultimately the exact choice and combination

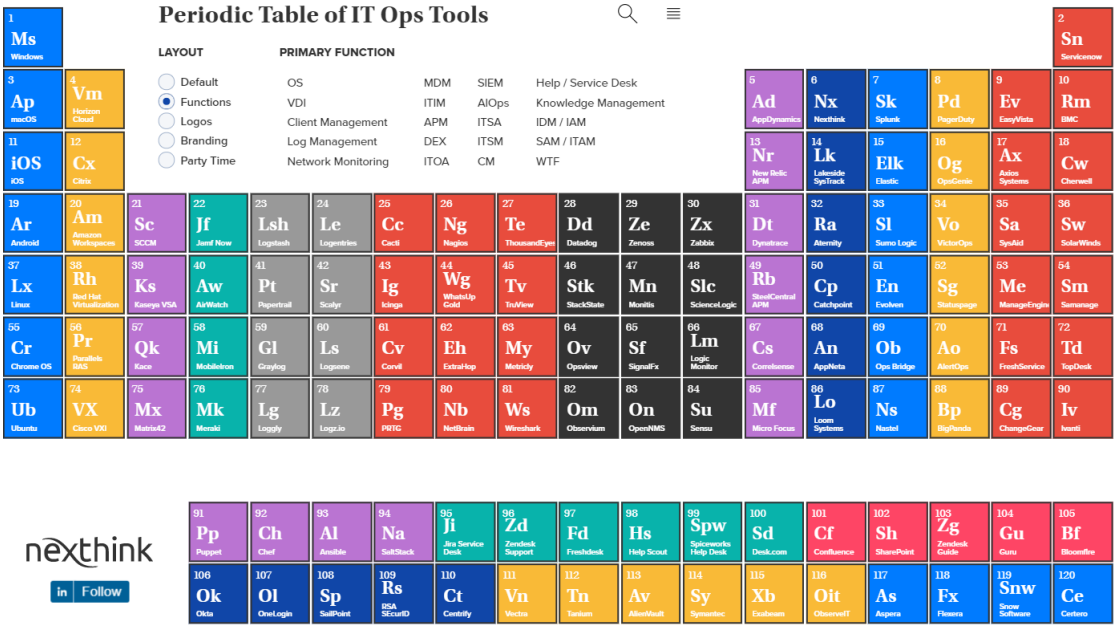

Figure 5.1: ITOps tools classified by their main role[5]

of tools varies greatly between companies. The grounds on which these tools are selected comprise a multitude of factors, such as previously used tools, existing infrastructure, current, and future scalability needs, and market dynamics. A quite comprehensive ”Periodic Table” of the main categories of tools used in ITOps, together with their associated most frequently used tools can be seen in figure Fig. 5.1.

As mentioned in the previous section, ITOps and DevOps teams need to nurture an efficient collaboration and align their ends. Thus, working with the same tools and data as frequently as possible aids this collaboration. Mainly, centralized log management is essential for collecting, manipulating, and consistently managing data across teams. Moreover, to ensure a cohesive tech stack, log management should ideally be integrated with change management automation[1]. Hence, especially in this log management area, DevOps and ITOps share a set of tools, and also ground their tool choice in other areas, such as automation, based on one another.

6 Conclusion

Today's CS industry requires IT teams to ensure security and stability while operating efficiently in increasingly interconnected and energetic environments. While DevOps gained considerable momentum in comparison to other OPS, its dynamics push the limits of the more traditional ITOps, generating backslashes like unsatisfied customers and sometimes even compromised services. Exactly this speed employed by DevOps, combined with its rather task-focused policy, is at the core of the contradiction with ITOps, which in turn, embraces a broader view of the technological landscape of an enterprise. So far, striking a balance between speed and reliability, between timely and frequent releases and system security proved to be challenging. Reliable shared data sources, structured based on a common model, act as the core element of the DevOps-ITOps cooperation and collaboration. As enterprises grow, they favor a shift from traditional ITOps towards IT operations management fueled by artificial intelligence. This shift will require extensive reorganization. However, the benefits of ensuring flexibility and speed while enhancing security and reliability by harnessing the powers of artificial intelligence might be the missing piece in solving the ITOps-DevOps disagreement[1, 10, 8].

References

[1] Carstensen, Nick. *ITOPS VS. DEVOPS: WHAT IS THE DIFFERENCE?* 2019. URL: https://www.graylog.org/post/itops-vs-devops-what-is-the-difference. (accessed: 15.04.2021).

[2] *DevOps Versus Other 'Ops'*. 2015. URL: https://devops.com/comparing-devops-traditional-eight-key-differences/. (accessed: 18.04.2021).

[3] *ITOps vs. DevOps: Overcoming the Conflict*. 2019. URL: https://sciencelogic.com/blog/itops-vs-devops#:~:text=Information%5C%20Technologies%5C%20Operations%5C%20(ITOps)%5C%20is,that%5C%20comprise%5C%20your%5C%20IT%5C%20infrastructure. (accessed: 17.04.2021).

[4] Melnichuk, Alex. *ITOps, DevOps, and NoOps: Concept and Basics Explained*. 2020. URL: https://ncube.com/blog/itops-devops-and-noops. (accessed: 18.04.2021).

[5] *Periodic Table of IT Ops Tools*. URL: https://www.nexthink.com/periodic-table/?utm_source=blog&utm_medium=Nexthink&utm_content=Periodic%20Table%20of%20IT%20Ops%20Tools. (accessed: 25.04.2021).

[6] Sarangi, Tridivesh. *Why ITOps Needs A New Approach to Automation, Too*. URL: https://www.workato.com/the-connector/itops-new-approach-automation/. (accessed: 25.04.2021).

[7] Stempniak, Adam and Cybulski, Zbigniew. *What Is AIOps, BizDevOps, CloudOps, DevOps, ITOps, NoOps? A Gentle Introduction to Digital Business Transformation*. URL: https://www.stxnext.com/blog/aiops-bizdevops-cloudops-devops-itops-noops-introduction-digital-business-transformation/. (accessed: 16.04.2021).

[8] Thurai, Andy. *How To Rise Above The ITOps Chaos Using AI*. URL: https://www.forbes.com/sites/cognitiveworld/2020/02/08/how-to-rise-above-the-itops-chaos-using-ai/?sh=52fba11731c9. (accessed: 25.04.2021).

[9] Watts, Stephen and Kidd, Chrissy. *ITOps, DevOps, and NoOps Explained*. 2017. URL: https://www.bmc.com/blogs/itops-devops-and-noops-oh-my/. (accessed: 15.04.2021).

[10] *What Is ITOps?* URL: https://www.splunk.com/en_us/data-insider/what-is-itops.html. (accessed: 25.04.2021).