# A Survey on DevOps Techniques Used in Cloud-Based IOT Mashups

**Article** *in* Advances in Intelligent Systems and Computing · December 2020

DOI: 10.1007/978-981-15-8289-9_37



**2 authors**, including:

Vigneshwaran Pandi
SRM Institute of Science and Technology

# A Survey on DevOps Techniques Used in Cloud-Based IOT Mashups

**M. Ganeshan and P. Vigneshwaran**

**Abstract** Development life cycle involves writing scripts, testing, bug finding, recoding, documentation, and deployment, and this activity consumes more time. DevOps looks at automating the entire process. DevOps is a total software and template delivery system that gives importance to the interaction between operations and development fellows. Choosing the right technologies and crafting an IoT infrastructure need a sophisticated comprehensive methodology which can be difficult. DevOps consists of a set of people-oriented, technology-oriented, organizational methods, and tools that face challenges one after the other so that a smooth, easy, and flexible system is obtained. There are many STS practices involved in the development and crafting cloud-based IOT Mashups. We provide few theoretical points of the major IoT threats mentioned in the latest papers and a brief of research activity undertaken in DevOps world to overcome these demerits. This paper briefs the correlation between DevOps technical practices and the tools and engineering that enable uninterrupted integration and transmission, test mechanization, and quick implemen-

M. Ganeshan (✉)
Department of Computer Science and Engineering, Jain Deemed to be University, Bengaluru, India
e-mail: link2ganeshan@gmail.com

P. Vigneshwaran
Department of Computer Science and Engineering, Faculty of Engineering and Technology, Jain University, Bengaluru, India
e-mail: vigneshwaran05@gmail.com



M. Tuba et al. (eds.), *ICT Systems and Sustainability*, Advances in Intelligent Systems and Computing 1270, https://doi.org/10.1007/978-981-15-8289-9_37

tation. This paper also gives insight on the tools that unite to give a "tool chain" of technology and services to excel the modular tasks of the template deployment and schedule of activities.

**Keywords** IOT · DevOps · Mashups · Cloud · Technical automation practices

# 1 Basic Concepts

We present a theoretical structure of basic principles of DevOps framework. The goal of our structured plan of study of the papers is to gain an understanding of principles of DevOps, based on the literature, to aid the analysis and understanding of DevOps challenges.

## 1.1 Scholarly Reviewed Literature

The survey questions mentioned in Table 1 are answered and helpful for completion of the survey paper. After selection of papers, the papers are divided into seven groups, such as methodologies, result report, influence, fundamentals, challenges, knowledge, and implementation as shown in Fig. 1. The divisions are explained in the following list:

- Methodologies: Gives tools, methods, templates and plans to improve DevOps. These give ground rules for choosing implementation automation [8], metrics values plays important role in deciding the nature of deployment [9], how to aid communication among devices across the world using sophisticated network for cloud based technology [10], and how to reconstruct the old infrastructure to latest supportive platform reducing faults at initial stage and make the system reliable for IoT devices connected on cloud network [11].

**Table 1** Survey research questions

| Survey question | Source |
|---|---|
| SQ1: What does DevOps culture in IoT Mashups mean? | References [1, 2] |
| SQ2: What are the problems that lead to the need for implementation of DevOps technology? | References [3–6] |
| SQ3: What are the major advantages of utilizing DevOps culture in IoT mashups? | References [1, 2] |
| SQ4: What are the probable major problems/limitations of using DevOps technology/tools? | References [3, 4, 6] |
| SQ5 What is continuous integration/continuous deployment? | References [3, 7] |
| SQ6 What problems are encountered when implementing DevOps? | References [5, 7] |

**Fig. 1** Review literature categories

Quantity ↑ Category →

| Methodoligies | Result. report | Influence | Fundamentals | Challenges | Knowledge | Implementaion |
|---|---|---|---|---|---|---|
| Qty:15 | Qty:8 | Qty:7 | Qty:7 | Qty:6 | Qty:4 | Qty:3 |
| 3 | 2 | 25 | 4 | 10 | 1 | 13 |
| 5 | 6 | 39 | 14 | 15 | 8 | 34 |
| 11 | 7 | 40 | 16 | 28 | 9 | 45 |
| 17 | 12 | 41 | 21 | 29 | 23 | |
| 19 | 18 | 42 | 22 | 31 | | |
| 24 | 20 | 43 | 36 | 35 | | |
| 26 | 33 | 49 | 37 | | | |
| 27 | 34 | | | | | |
| 30 | | | | | | |
| 32 | | | | | | |
| 38 | | | | | | |
| 46 | | | | | | |
| 47 | | | | | | |
| 48 | | | | | | |
| 50 | | | | | | |

- Result report: Points toward the end result of companies implementing DevOps culture in the deployment of APIs between private and public cloud components.
- Influence: Deals with how DevOps culture improves the efficiency in the area of APIs development between the private architecture and cloud-based network architecture [12], there is importance of designing security criteria before the implementation of crafted template [13], or the impact of DevOps culture in the APIs development in specific cases of real-time computation of data research [14].
- Fundamentals: Covers the basic theory needed for cultivating DevOps culture in continuous delivery and data integration using the concept of the deployment pipeline [15].
- Challenges: Explains the threats present in implementation of DevOps culture. Few sources deal with specific threats like DevOps used in old domains and new domains supporting the new technologies [16], most of the devices connected in IoT are embedded systems in nature with multiple computation nature [17], or communication APIs between different IoT mashup devices with heterogeneous characteristics [18].
- Knowledge: Deals with the problems faced while training people in DevOps. Papers in this sub-division introduce teaching styles [19], as wel l as explain the key attributes, abilities and exposure needed for DevOps technical engineers, leads and experts [20, 21].
- Implementation: This sub-division deals with the different prototypes for DevOps culture tools for deployment and scripting the templates [3] and shows how a

company identifies and shortlists a set of standards tools to implement and gauge respective tools in DevOps activity [22].

## 2 Role of DevOps in IOT Mashup

DevOps, with its focus on reducing delivery cycle times, can increase the speed at which IoT heterogenous devices are available in the technology market throughout the globe. Google states that the number of IoT heterogenous devices will exceed non-IoT by 2025. The technologies and principles of IoT will have a major impact on organizations with regard to planning business, management of risk, and design of architecture and network. Any error occurring in any module regarding security and data integration of IoT Mashup architecture can affect the whole DevOps process.

## 3 Implementing DevOps Architecture Style

The course in the needs of DevOps architecture is the utilization of connected portable microservices with suitable APIs for the micro services engineering along with DevOps methodologies [23]. An appropriate tool to orchestrate virtualization mentioned in Table 2 can be decided by the DevOps team to achieve a continuous integration and a recovery system.

**Table 2** DevOps tools to orchestrate virtualization technology

| Tools | Description |
|---|---|
| CloudSlang | CloudSlang is an open-source tool, we can use or customize ready-made YAML-based workflow |
| Dockers | Docker, and its open-source its being widely adopted for real-world data backs |
| OpenVZ | Open-source container-based virtualization for Linux. |
| Kubernetes | Kubernetes is a free package for container-composition system used in automation of APIs and monitoring of code |
| Containership | Containership simplify the build/test/deploy pipelines in DevOps |
| Packer | Packer is user friendly used in automation of machine image creation of different types |
| Linux Containers | Image building tool for LXC/LXD, Support for a lot of distributions and architectures |
| Nomads | Nomad is an open-source clustered scheduling engine. It can run different applications of micro-service, batch, containerized and non-containerized applications |

**Table 3** Survey on IoT Mashups security

| Year | Reference | Contributions |
|---|---|---|
| 2016 | [24] | A brief discussion of vulnerabilities faced by the edge side layer of IoT |
| 2017 | [25] | Survey on using end points for IoT devices on cloud computing planform |
| 2017 | [26] | Discussion on similarities and differences between IoT orchestration using cloud computing and fog computing |
| 2017 | [27] | A brief discussion on most relevant limitations of IoT devices |
| 2017 | [28] | Discussion on security key factors to be improved on targeted endpoints in network-based IoT services for heterogeneous devices |
| 2017 | [29] | Security issues related to the IoT middle ware |
| 2018 | [30] | Security mechanism for IoT security like SDN and NFB |
| 2019 | [31] | Trust management techniques for Internet of Things |

### *3.1 DevOps Tools for Template/Scripts/Code Management*

Script/code management DevOps system is in need of tools mostly intended to improve relationship among developers of different domains. DevOps tools are most required for today's world of bigdata to deploy continuous integration/delivery system. While using traditional approach in deployment and code management, the delivery concepts cannot match the data handling for present world data/code processing requirement. Even developer and operational team cannot manually conduct source code/data management. Therefore, automation scripts are needed to obtain workflow information that can affect the activities carried out in the work frame. For any IoT Mashup instance, a code error can create a bug which may result workflow disturbance and data leak that affects system security [8] and Table 3 lists some of the IoT Mashups Security issues faced in the years mentioned against them.

### *3.2 Thingworx Versus AWS in IoT Platform*

While comparing IoT platforms, choosing a better platform is not an issue since you do not have to worry about choosing among options. You can pick Thingworx [32] for the industry expertise and then use AWS for your cloud platform as an integral solution. Thingworx is a third-party development tool for implementing mechanical drives in industry platform for high-level IoT architecture.

AWS IoT Core can automate devices in report generation and limitless logs maintained using S3 services [33]. Management and maintenance are cost effective, with less waiting time. The communication can be upscaled to as many heterogenous devices as required. AWS IoT Core is compatible with protocols like HTTP, MQTT and WebSocket's [34]. Information security is provided by a protocol TLS [35].

## 4 Conclusion

DevOps culture in deployment and orchestration automation tools for IoT should be sophisticated to provide information exchange through well-designed APIs to confirm lesser downtime in information transfer between IoT mashup heterogenous devices. Edge location in cloud platform must be well defined by appropriate protocols for specific DevOps tools. In this survey, we have explored the different technologies on DevOps culture to automate the deployment, orchestration, monitoring and recovery approaches for IoT Mashup devices on cloud platform by studying a systematic mapping approach. The references were categorized as shown in Table 4 for conceptual study. Table 5 gives the list of cloud IoT features.

Many challenges were realized in building a data mashup service between IoT smart gadgets. But, the number of electronic gadgets and mechanical drives are increasing rapidly and are inter-dependent. Wearable technology in coming days is ready with big data lake for analysis. Every human being has increased his count of personal electronic smart devices that are connected to network for computation. All the smart devices require data computation and the process for analyzing the data needs DevOps culture. This rapid growth of technology and IoT devices needs DevOps culture in deployment and Ochestration to ensure friendly Web services. Technically, well-defined research is required in DevOps culture to provide the services with reliable performance. Cloud-based services and tools for IoT Mashups is helpful for conducting high-end workflow. AWS IoT Core provides end-to-end services and embeds DevOps culture in deployment, monitoring, and recovery to increase workflow reliability and efficiency by applying more focus on the DevOps tool-based automated APIs and devices configuration from cross platform.

**Table 4** Study of references

| Reference | Category | Concepts | Goal |
|---|---|---|---|
| References [36, 37] | Source code management | Automate the versioning on DevOps tools | Human collaboration |
| Reference [38] | | Improve the DevOps culture and methods in scripts/code management | Continuous delivery and recovery using tool for code management |
| References [18] | | Overcoming the barriers between different | Decision support system (DSS) is implemented for specific technologies to obtain desired results |
| | | Tool platform | |
| References [1, 20, 36, 39] | Continuous integration | Develop APIs with can handle continuous information flow between domains | Continuous delivery; |
| References [1, 12, 19, 40, 41] | | DevOps can be used in testing environment | Optimize the method to secure devices in IoT domain, using cross-layered network |
| Reference [42] | Deployment automation | Real time monitoring/recovery | Continuous delivery |
| References [37] | | Implement continuous delivery and integration of system | Reliability |
| References [43–45] | | Infrastructure as code Virtualization, containerization cloud services, automation | To improve APIs connectivity for smooth information flow |
| References [1, 40] | Monitoring and logging | DevOps tools can improve the scalability of instances. Provide reliable work platform with automatic metric value and alerting | Reliability |
| References [3, 10, 26, 27, 35] | | | To migrate existing infrastructure to new domain on cloud platform |
| Reference [24] | IoT as a network of networks | Three layers in technological innovation—cloud-pipe-device | Bridge between cloud and IoT |

**Table 5** Survey on cloud IoT features

| IoT cloud provider | Network protocols | Data store | Push not | Geo | BLOB | Trigger |
|---|---|---|---|---|---|---|
| Amazon | REST/MQTT | ✓ | ✓ | ✓ | ✓ | ✓ |
| Azure | MQTT/AMQP/REST | ✓ | ✓ | ✓ | ✓ | ✓ |
| Google | REST | ✓ | ✓ | ✓ | ✓ | ✓ |
| Cloud-foundry | REST | ✓ | ✖ | n/a | ✓ | ✖ |
| Kinvey | REST | ✓ | ✓ | n/a | ✓ | ✓ |
| Heroku | MQTT | ✓ | ✖ | n/a | ✓ | ✖ |
| Parse | REST | ✓ | ✓ | ✓ | ✓ | ✓ |
| Bluemix | MQTT | ✓ | ✓ | ✓ | ✖ | ✓ |
| Dream-factory | REST | ✓ | ✓ | n/a | ✓ | ✓ |

## References


1. D. Cukier, DevOps patterns to scale web applications using cloud services, in *Proceedings of the 2013 Companion Publication for Conference on Systems, Programming, & Applications: Software for Humanity (SPLASH '13)* (ACM, 2013), pp. 143–152. (Code: A35)
2. A. Khattab, H.S. Hamza, S.M. Khattab, Enhancing user experience in IoT mashup using semantic technology, in *2018 IEEE Global Conference on Internet of Things (GCIoT)*, Alexandria, Egypt (2018), pp. 1–6. https://doi.org/10.1109/gciot.2018.8620164
3. K. Nybom, J. Smeds, I. Porres, On the impact of mixing responsibilities between Devs and Ops, in *International Conference on Agile Software Development (XP 2016)* (Springer International Publishing, 2016), pp. 131–143. (Code: S18)
4. J. Roche, Adopting DevOps practices in quality assurance. Commun. ACM **56**(11), 38–43 (2013). (Code: A74)
5. F.M.A. Erich, C. Amrit, M. Daneva, A qualitative study of DevOps usage in practice. J. Softw. Evol. Proc. **29**(6), e1885 (2017). (Published online in Wiley InterScience, www.interscience.wiley.com, https://doi.org/10.1002/smr)
6. K. Beck, C. Andres, *Extreme Programming Explained: Embrace Change* (Addison-Wesley Professional)
7. A. Rahman, Characteristics of defective infrastructure as code scripts in DevOps, in *Proceedings of the 40th International Conference on Software Engineering (ICSE '18)* (ACM, 2018), pp. 476–479. (Code: A806)
8. J. Wettinger, V. Andrikopoulos, F. Leymann, Automated capturing and systematic usage of DevOps knowledge for cloud applications, in *2015 IEEE International Conference on Cloud Engineering* (IEEE, 2015), pp. 60–65. (Code: A61)
9. N. Forsgren, M. Kersten, DevOps metrics. Commun. ACM **61**(4), 44–48 (2018). (Code: B21)
10. K. Magoutis, C. Papoulas, A. Papaioannou, F. Karniavoura, D.-G. Akestoridis, N. Parotsidis, M. Korozi, A. Leonidis, S. Ntoa, C. Stephanidis, Design and implementation of a social networking platform for cloud deployment specialists. J. Internet Serv. Appl. **6**(1) (2015). (Code: S3)
11. N. Basiri, N. Behnam, R. de Rooij, L. Hochstein, L. Kosewski, J. Reynolds, C. Rosenthal, Chaos engineering. IEEE Softw. **33**(3), 35–41 (2016). (Code: A76); J. Clerk Maxwell, *A Treatise on Electricity and Magnetism*, 3rd edn., vol. 2 (Clarendon, Oxford, 1892), pp. 68–73
12. M. Shahin, M.A. Babar, L. Zhu, The intersection of continuous deployment and architecting process: practitioners' perspectives, in *Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM '16)* (ACM, 2016), pp. 44:1–44:10. (Code: A64)
13. L. Zhu, D. Xu, A. B. Tran, X. Xu, L. Bass, I. Weber, S. Dwarakanathan, Achieving reliable high-frequency releases in cloud environments. IEEE Softw. **32**(2), 73–80 (2015). (Code: B12)

14. M. de Bayser, L.G. Azevedo, R. Cerqueira, ResearchOps: the case for DevOps in scientific applications, in *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)* (2015), pp. 1398–1404. (Code: I40)
15. J. Humble, J. Molesky, Why enterprises must adopt DevOps to enable continuous delivery. Cutter IT J. **24**(8), 6 (2011). (Code: B5)
16. T. Laukkarinen, K. Kuusinen, T. Mikkonen, DevOps in regulated software development: case medical devices, in *Proceedings of the 39th International Conference on Software Engineering: New Ideas and Emerging Results Track (ICSE-NIER '17)* (IEEE Press, 2017), pp. 15–18. (Code: A26)
17. L.E. Lwakatare, T. Karvonen, T. Sauvola, P. Kuvaja, H.H. Olsson, J. Bosch, M. Oivo, Towards DevOps in the embedded systems domain: why is it so hard? in *49th Hawaii International Conference on System Sciences (HICSS)* (IEEE, 2016), pp. 5437–5446. (Code: A42)
18. E. Diel, S. Marczak, D.S. Cruzes, Communication challenges and strategies in distributed DevOps, in *11th IEEE International Conference on Global Software Engineering (ICGSE)* (2016), pp. 24–28. (Code: I19)
19. H.B. Christensen, Teaching DevOps and cloud computing using a cognitive apprenticeship and story-telling approach, in *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '16)* (ACM, 2016), pp. 174–179. (Code: A47)
20. S. Chung, S. Bang, Identifying knowledge, skills, and abilities (KSA) for DevOps-aware server-side web application with the grounded theory. J. Comput. Sci. Coll. **32**(1), 110–116 (2016). (Code: A16)
21. W. Hussain, T. Clear, S. MacDonell, Emerging trends for global DevOps: A New Zealand Perspective, in *Proceedings of the 12th International Conference on Global Software Engineering (ICGSE '17)* (IEEE Press, 2017), pp. 21–30. (Code: A25)
22. B. Snyder, B. Curtis, Using analytics to guide improvement during an agile/DevOps transformation. IEEE Softw. **35**(1), 78–83 (2018). (Code: I7)
23. A. Sill, Cloud standards and the spectrum of development. IEEE Cloud Comput. **1**(3), 15–19 (2014). (Code: I67)
24. A. Mosenia, N.K. Jha, A comprehensive study of security of internet-of-things. IEEE Trans. Emerg. Topics Comput. **5**(4), 586–602 (2017)
25. W. Yu, F. Liang, X. He, W.G. Hatcher, C. Lu, J. Lin, X. Yang, A survey on the edge computing for the internet of things. IEEE Access **6**, 6900–6919 (2018)
26. J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, W. Zhao, A survey on internet of things: architecture, enabling technologies, security and privacy, and applications. IEEE Internet Things J. **4**(5), 1125–1142 (2017)
27. Y. Yang, L. Wu, G. Yin, L. Li, H. Zhao, A survey on security and privacy issues in internet-of-things. IEEE Internet Things J. **4**(5), 1250–1258 (2017)
28. L. Chen, S. Thombre, K. Järvinen, E.S. Lohan, A. Alén-Savikko, H. Leppäkoski, M.Z.H. Bhuiyan, S. Bu-Pasha, G.N. Ferrara, S. Honkala, J. Lindqvist, L. Ruotsalainen, P. Korpisaari, H. Kuusniemi, Robustness, security and privacy in location-based services for future IoT: a survey. IEEE Access **5**, 8956–8977 (2017)
29. A.H. Ngu, M. Gutierrez, V. Metsis, S. Nepal, Q.Z. Sheng, IoT middleware: a survey on issues and enabling technologies. IEEE Internet Things J. **4**(1), 1–20 (2017)
30. I. Farris, T. Taleb, Y. Khettab, J. Song, A survey on emerging SDN and NFV security mechanisms for IoT systems. IEEE Commun. Surv. Tutorials **21**(1), 812–837 (2018)
31. U. Din, M. Guizani, B.-S. Kim, S. Hassan, M.K. Khan, Trust management techniques for the internet of things: a survey. IEEE Access **7**, 29763–29787 (2019)
32. L. Bass, I. Weber, L. Zhu, *DevOps: A Software Architect's Perspective* (Addison-Wesley Professional, 2015)
33. B. Beyer, C. Jones, J. Petoff, N.R. Murphy, *Site Reliability Engineering: How Google Runs Production Systems* (O'Reilly Media, 2016)
34. J. Bonér, D. Farley, R. Kuhn, M. Thompson, The Reactive Manifesto (2014). https://www.reactivemanifesto.org/. Accessed Aug 2018

35. S. Neely, S. Stolt, Continuous delivery? Easy! Just change everything (well, maybe it is not that easy), in *2013 Agile Conference* (2013), pp. 121–128. (Code: B23)
36. X. Bai, M. Li, D. Pei, S. Li, D. Ye, Continuous delivery of personalized assessment and feedback in agile software engineering projects, in *Proceedings of the 40th International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET '18)* (2018), pp. 58–67. (Code: I818)
37. L. Bass, The software architect and DevOps. IEEE Softw. **35**(1), 8–10 (2018). (Code: I33)
38. K. Brown, B. Woolf, Implementation patterns for microservices architectures, in *Proceedings of the 23rd Conference on Pattern Languages of Programs (PLoP '16)*, Article 7 (The Hillside Group, 2016), pp. 7:1–7:35 (2016). (Code: A104)
39. L. Chen, Continuous delivery: huge benefits, but challenges too. IEEE Softw. **32**(2), 50–54 (2015). (Code: B15)
40. P. Debois, Devops: a software revolution in the making. Cutter IT J. **24**(8), 3–5 (2011). (Code: B4)
41. C. Ebert, G. Gallardo, J. Hernantes, N. Serrano, DevOps. IEEE Softw. **33**(3), 94–100 (2016). (Code: A54)
42. A. Balalaie, A. Heydarnoori, P. Jamshidi, Microservices architecture enables DevOps: migration to a cloud-native architecture. IEEE Softw. **33**(3), 42–52 (2016). (Code: A2)
43. M. Kersten, A cambrian explosion of DevOps tools. IEEE Softw. **35**(2), 14–17 (2018). (Code: I808)
44. N. Forsgren, V.A. Brown, G. Kim, N. Kersten, J. Humble, 2016 State of DevOps Report (2016). https://puppet.com/resources/whitepaper/2016-state-of-devops-report
45. H. Barua, The role of configuration management in a containerized world (2015). https://www.infoq.com/news/2015/12/containers-vs-config-mgmt
46. M. Callanan, A. Spillane, DevOps: making it easy to do the right thing. IEEE Softw. **33**(3), 53–59 (2016). (Code: A67)
47. G.G. Claps, R.B. Svensson, A. Aurum, On the journey to continuous deployment: technical and social challenges along the way. Inf. Softw. Technol. **57**, 21–31 (2015). (Code: B13)
48. R. de Feijter, S. Overbeek, R. van Vliet, E. Jagroep, S. Brinkkemper, DevOps competences and maturity for software producing organizations, in *Enterprise, Business-Process and Information Systems Modeling* (Springer, 2018), pp. 244–259. (Code: S805)
49. A. Dyck, R. Penners, H. Lichter, Towards definitions for release engineering and DevOps, in *2015 IEEE/ACM 3rd International Workshop on Release Engineering* (2015), p. 3. (Code: B26)
50. D.G. Feitelson, E. Frachtenberg, K.L. Beck, Development and deployment at Facebook. IEEE Internet Comput. **17**(4), 8–17 (2013). (Code: B7)
51. J. Gray, A conversation with Werner Vogels. ACM Queue **4**(4), 14–22 (2006). (Code: B3)
52. J. Humble, Continuous delivery sounds great, but will it work here? Queue **15**(6), 57–76 (2017). (Code: B22)
53. M.G. Jaatun, Software security activities that support incident management in Secure DevOps, in *Proceedings of the 13th International Conference on Availability, Reliability and Security (ARES 2018)* (ACM, 2018), pp. 8:1–8:6. (Code: A803)
54. M.G. Jaatun, D.S. Cruzes, J. Luna, DevOps for better software security in the cloud, in *Proceedings of the 12th International Conference on Availability, Reliability and Security (ARES '17)* (ACM, 2017), Article 69, pp. 69:1–69:6. (Code: A85)
55. H. Kang, M. Le, S. Tao, Container and microservice driven design for cloud infrastructure DevOps, in *2016 IEEE International Conference on Cloud Engineering (IC2E)* (2016), pp. 202–211. (Code: I58)
56. M. Leppanen, S. Makinen, M. Pagels, V. Eloranta, J. Itkonen, M.V. Mantyla, T. Mannisto, The highways and country roads to continuous deployment. IEEE Softw. **32**(2), 64–72 (2015). (Code: B14)
57. Z. Li, Q. Lu, L. Zhu, X. Xu, Y. Liu, W. Zhang, An empirical study of cloud API issues. IEEE Cloud Comput. **5**(2), 58–72 (2018). (Code: I802)

58. H.H. Olsson, H. Alahyari, J. Bosch, Climbing the "Stairway to Heaven"—a multiple-case study exploring barriers in the transition from agile development towards continuous deployment of software, in *38th Euromicro Conference on Software Engineering and Advanced Applications* (2012), pp. 392–399. (Code: B17)
59. C. Pang, A. Hindle, Continuous maintenance, in *2016 IEEE International Conference on Software Maintenance and Evolution (ICSME)* (2016), pp. 458–462. (Code: I55)
60. R. Punjabi, R. Bajaj, User stories to user reality: a DevOps approach for the cloud, in *2016 IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT)* (2016), pp. 658–662. (Code: I17)
61. M. Rajkumar, A.K. Pole, V.S. Adige, P. Mahanta, DevOps culture and its impact on cloud delivery and software development, in *2016 International Conference on Advances in Computing, Communication, Automation (ICACCA)* (2016), pp. 1–6. (Code: I48)
62. S. Van Rossem, W. Tavernier, D. Colle, M. Pickavet, P. Demeester, Introducing development features for virtualized network services. IEEE Commun. Mag. **56**(8), 184–192 (2018). (Code: I77)
63. R. Siqueira, D. Camarinha, M. Wen, P. Meirelles, F. Kon, Continuous delivery: building trust in a large-scale, complex government organization. IEEE Softw. **35**(2), 38–43 (2018). (Code: B29)
64. E. Woods, Operational: the forgotten architectural view. IEEE Softw. **33**(3), 20–23 (2016). (Code: A82)
65. H. Yasar, K. Kontostathis, Where to integrate security practices on DevOps platform. Int. J. Secure Softw. Eng. (IJSSE) **7**(4), 39–50 (2016). (Code: A58)
66. H. Yasar, K. Kontostathis, Secure DevOps process and implementation, in *2016 IEEE Cybersecurity Development (SecDev)*, Boston, MA (2016), p. 166. https://doi.org/10.1109/secdev.2016.048
67. xMatters Atlassian DevOps Maturity Survey Report 2017 (2017). https://www.xmatters.com/press-release/xmatters-atlassian-2017-devops-maturity-survey-report/
68. How Netflix Thinks of DevOps (2018). https://www.youtube.com/watch?v=UTKIT6STSVM
69. H. Beal, Where are you on the DevOps Maturity Scale Webcast (2015). https://www.youtube.com/watch?v=a50ArHzVRqk
70. R. Brigham, DevOps at Amazon: A Look at Our Tools and Processes (2015). At AWS re: Invent 2015, https://www.youtube.com/watch?v=esEFaY0FDKc. Accessed June 2018
71. D. Brown, Our DevOps journey—Microsoft's internal transformation story, in *DevOneConf 2018* (2018). https://www.youtube.com/watch?v=cbFzojQOjyA. Accessed July 2018
72. D. Budgen, P. Brereton, Performing systematic literature reviews in software engineering, in *Proceedings of the 28th International Conference on Software Engineering (ICSE '06)* (ACM, 2006), pp. 1051–1052
73. N. Ceresani, The Periodic Table of DevOps Tools v.2 Is Here (2016). https://blog.xebialabs.com/2016/06/14/periodic-table-devops-tools-v-2/. Accessed Apr 2018