A Systematic Literature Review on Microservices

Conference Paper *in* Lecture Notes in Computer Science · July 2017

DOI: 10.1007/978-3-319-62407-5_14

3 authors:

Hulya Vural
Atilim University

Murat Koyuncu
Atilim University

Sinem Guney
Atilim University

A Systematic Literature Review on Microservices

Hulya Vural, Murat Koyuncu(✉), and Sinem Guney

Atilim University, Ankara, Turkey
hulya.vural.tr@gmail.com, sinemmguney@gmail.com, mkoyuncu@atilim.edu.tr

Abstract. The cloud is an emerging paradigm which leads the way for different approaches and standards. The architectural styles are evolving based on the requirements of the cloud as well. In recent years microservices is seen as the architecture style for scalable, fast evolving cloud applications. As part of this paper, a systematic mapping study was carried out around microservices. It is aiming to find out the current trends around microservices, the motivation behind microservices research, emerging standards and the possible research gaps. The obtained results can help researchers and practitioner in software engineering domain who want to be aware of new trends about SOA and cloud computing.

Keywords: Cloud · SOA · Web services · Microservices · Systematic mapping

1 Introduction

Service-oriented architecture (SOA) has emerged as a means of developing distributed systems where the components are stand-alone services [37]. Services are basic units which are developed independently and made accessible over the Internet. Standard internet protocols are used for service communication among different computers. SOA provides many advantages to develop easy and economic distributed software systems and, therefore, it is the leading technology for interoperability on today's internet world. Service-oriented software engineering defines evolution of existing software engineering approaches to develop dependable and reusable services considering the requirements and characteristics of this technology [37]. Service-oriented computing (SOC) is the paradigm that utilizes services as the fundamental elements for developing applications. Therefore, service-oriented software engineering aims at designing and developing service-based applications consonant with SOC paradigm and SOA principles using software engineering methodologies.

After the popularity of cloud computing in recent years, new trends in the software engineering have emerged, such as going to market with minimal viable product and making small development teams autonomous. The architectural styles have also evolved based on the cloud environment needs [36]. One of those new architectural styles is microservices. The aim of the microservices is to divide the business behavior into small services which can run independent of each other. As mentioned by Martin

O. Gervasi et al. (Eds.): ICCSA 2017, Part VI, LNCS 10409, pp. 203–217, 2017.
DOI: 10.1007/978-3-319-62407-5_14

Fowler, “While there is no precise definition of this architectural style, there are certain common characteristics around organization around business capability, automated deployment, intelligence in the endpoints, and decentralized control of languages and data” [1]. Another definition for microservices is “Microservices are small, autonomous services that work together” [2].

The characteristics of the microservices are listed as follows [1]:

- Componentization via Services
- Organized around Business Capabilities
- Products not Projects
- Smart endpoints and dumb pipes
- Decentralized Governance
- Decentralized Data Management
- Infrastructure Automation
- Design for failure
- Evolutionary Design

The microservices are developed, deployed and maintained separately. This allows the teams to be autonomous where they can decide on the technology to use which best addresses the current needs of the business behavior. The language and the database might be different from one microservice to another. They do not share data between each other, instead they use Representational State Transfer (REST) protocol to communicate to each other. The most important benefits of using microservices are agility, autonomy, scalability, resilience and easy continuous deployment.

Even though microservices were first mentioned at [60] in 2010, the definition of the microservice mentioned in that study does not totally map to the current microservice definition in literature. The study carried out in 2010 [60] defines microservices as light services using REST. It does not mention most of the characteristics listed at [1].

There has been another systematic mapping carried out on microservices in 2016 [40]. In that study, the research questions are around the architectural diagrams used for microservices’ representation, the quality attributes and the challenges. However, the emerging standards and de facto tools are not mentioned.

In this paper, the aim is to not only analyze the emerging standards but also the types of research conducted and the practical motivations around carrying out the microservices architecture.

2 Method

This study is conducted a systematic mapping as defined in [3] with one modification (see Fig. 1). The modification is that, we carry out the keywording according to the whole paper instead of keywording according to the abstract. The reason for the modification to the original process is to enhance the classification criteria through adding new areas.

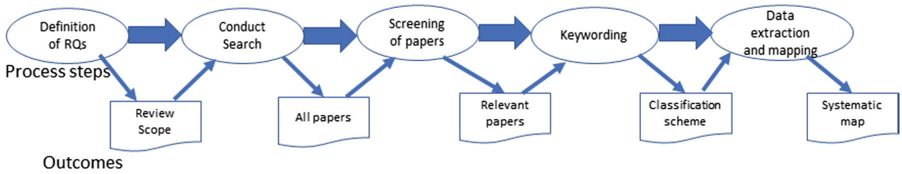

Fig. 1. Process steps and outcomes

2.1 Research Questions

Three research questions are determined as follows:

RQ1: What type of research is conducted on microservices?
RQ2: What are the main practical motivations behind microservices related research?
RQ3: What are the emerging standards and de facto tools on microservices solutions?

2.2 Search Sampling

The search is conducted using Web of Science (Thomson Reuters Web of Knowledge), which includes the following online databases:

- ACM (Association for Computing Machinery) Digital Library [4]
- CiteSeer [5]
- Computer Source [6]
- ebrary [7]
- Human-Computer Interaction Bibliography [8]
- IEEE Xplore [9]
- INSPEC [10]
- INSPEC Archive [11]
- Nature [12]
- Science [13]
- Science & Technology Collection [14]
- SciTech Connect [15]
- Springer LINK [16]

2.3 Search Iteration

The search is carried on with the following criteria:

- Keywords: microservice OR micro-service
- Research Area: Computer Science

The search iteration has returned 39 results [17–32, 38–60].

3 Screening the Papers

The papers are evaluated according to the inclusion and exclusion criteria. The ones which do not meet the criteria are excluded.

The inclusion criteria:

- All papers returned from the search criteria

Exclusion criteria:

- If the microservices is just mentioned in the research but the focus of the research is not directly on microservices.

Out of 39, 2 papers were excluded based on the exclusion criteria [30, 41]. As a result, 37 papers are included into the mapping process.

4 Keywording

As part of keywording four different categorization schemes are identified:

- Service models in cloud computing
- Operational areas
- Research types
- Emerging standards and tools.

4.1 Service Models in Cloud Computing

The service models in cloud computing are classified in three different types [33]:

- Infrastructure as a Service (IaaS): the infrastructure is supplied as a service (e.g. virtual machine, hard disk, load balancer etc.).
- Platform as a Service (PaaS): The platform is supplied as a service (e.g. Azure SQL, Tomcat etc.).
- Software as a Service (SaaS): The software itself is supplied as a service (e.g. Office 365, Gmail etc.).

Even though the microservice architecture style is shaped considering cloud needs, the research papers returned as part of the search criteria do not necessarily use cloud. As a result, on premise installations (OnPrem) are also included in the service models.

4.2 Operational Areas

In [34] several different operational areas are called out for cloud:

- Accounting and billing
- SLA management (Service Level Agreement)
- Service/resource provisioning

- Capacity planning
- Configuration management
- Security and privacy assurance
- Fault management

Some of the research papers included in the current study are focusing on cloud whereas some are not. As a result, the operational areas were modified to fit the needs as follows:

- Cost comparison
- Availability/Resiliency
- Performance
- Security
- Test technique
- Functionality/Design
- Analytics/Monitoring
- Scalability
- Deployment

The answer for the second research question (RQ2) will be based on the modified operational areas.

4.3 Research Types

In [35], 6 different research types are called out (See Table 1). The answer to the first research question (RQ1) will be based on these 6 research types.

4.4 Emerging Standards and Tools

The papers included in this systematic mapping study can be seen as a representation of the common tools used for microservices. Given that microservices is a new concept, the standards are not yet well formed. The current systematic study aims also to give an answer on emerging standards for microservices.

5 Data Extraction and Mapping

The results obtained from mapping are converted into different graphs and they are given below in a way to answer the defined research questions.

RQ1: What type of research is conducted on microservices?

The papers are mapped to the research types as seen in Fig. 2. The most widely used research type is Solution Proposal which is followed by Validation Research and Evaluation Research.

The papers are classified according to the service types as seen in Fig. 3. Almost half of the papers did not explicitly mention the service type they were targeting

Table 1. Research types

Class	Description
Validation research	Techniques investigated are novel and have not yet been implemented in practice. Techniques used are for example experiments, i.e., work done in the lab
Evaluation research	Techniques are implemented in practice and an evaluation of the technique is conducted. That means, it is shown how the technique is implemented in practice (solution implementation) and what are the consequences of the implementation in terms of benefits and drawbacks (implementation evaluation). This also includes identification of problems in industry
Solution proposal	A solution for a problem is proposed, the solution can be either novel or a significant extension of an existing technique. The potential benefits and the applicability of the solution is shown by a small example or a good line of argumentation
Philosophical papers	These papers sketch a new way of looking at existing things by structuring the field inform of a taxonomy or conceptual framework
Opinion papers	These papers express the personal opinion of somebody whether a certain technique is good or bad, or how things should have been done. They do not rely on related work and research methodologies
Experience papers	Experience papers explain what and how something has been done in practice. It has to be the personal experience of the author

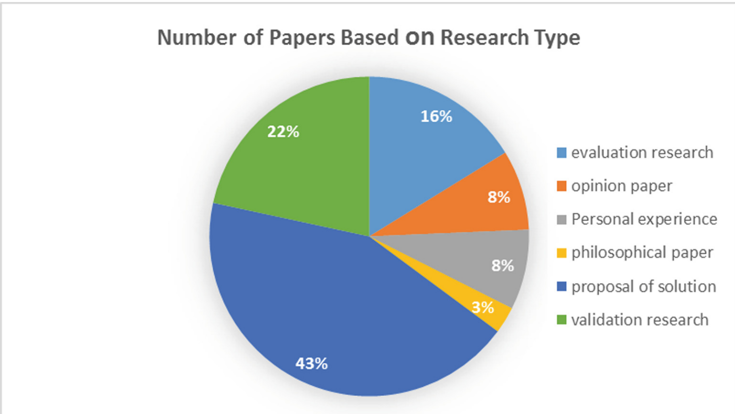

Fig. 2. Research types

(represented as NA in the figure). SaaS by far the most common service type being investigated. Also, some papers refer to more than one service type.

The bubble chart in Fig. 4 illustrates an analysis based on research types versus service types. The figure shows that there are only two studies on IaaS investigation regarding microservices. This is an expected outcome given that the microservices is a high level architectural style. On the other hand, there is only one official philosophical research papers on microservices. Most probably the reason is that the philosophical

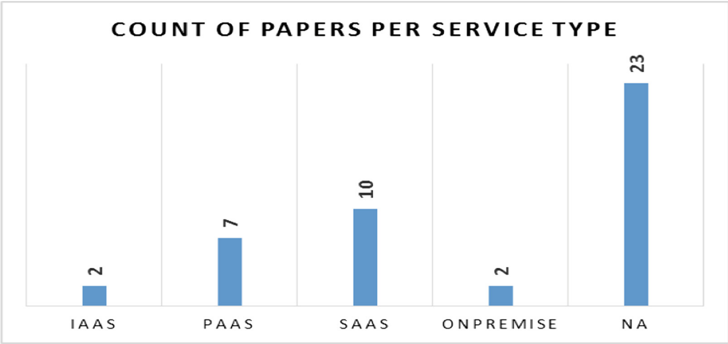

Fig. 3. Count of papers per service type

statement of microservices was laid out by Lewis and Fowler [1] on 2014. Mostly, the research is around Solution Proposal which do not explicitly call out the possible service types applicable for that solution.

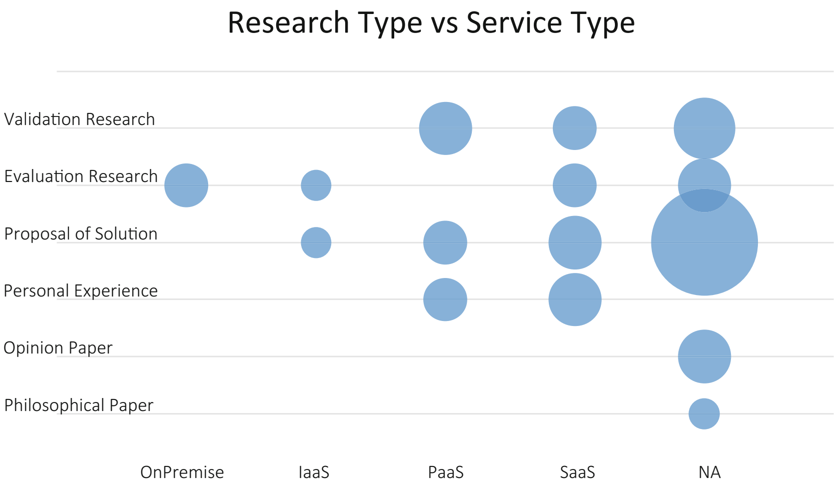

Fig. 4. Service types versus research types

RQ2: What are the main practical motivations behind microservices related research?

The papers are mapped to the operational areas and obtained results are shown in Fig. 5. The main motives are around functionality followed by performance and test techniques. Given that the microservices paradigm was first mentioned around 2014 and official research papers started to show up in 2015, it is natural to expect the functionality be main concerns of research.

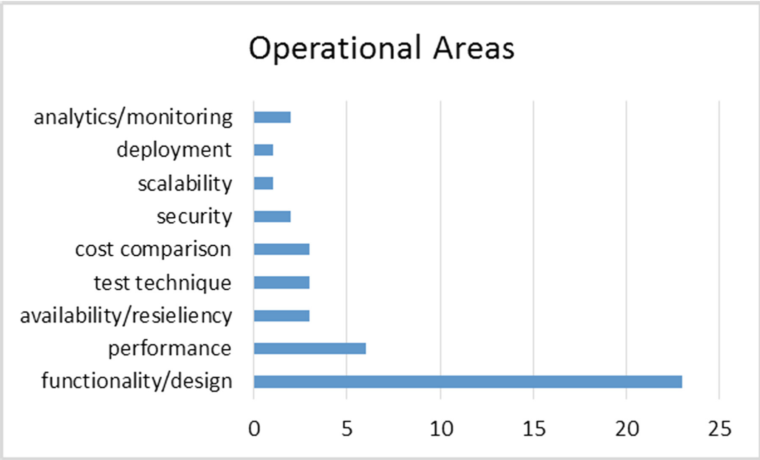

Fig. 5. Operational areas

Figure 6 aims to answer if the study has empirical results or not. Our analysis shows that the empirical studies are currently small in amount. Figure 7 illustrates an analysis based on operational areas versus service types. The most remarkable point is that most of the studies focus on the functionality/design issues. Figure 8 shows if there is a new solution proposed and/or implemented. As seen in the figure, most of the

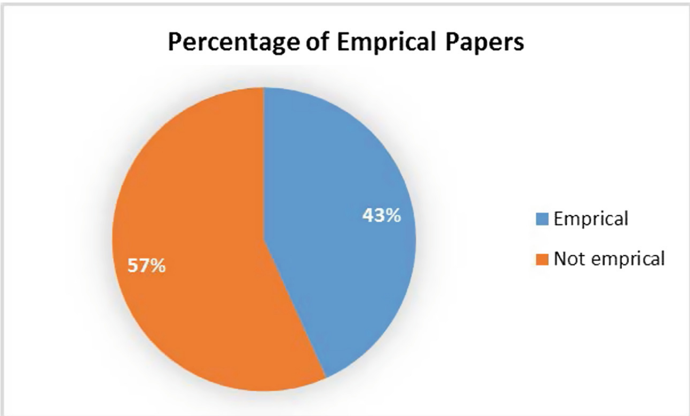

Fig. 6. Empirical results in research

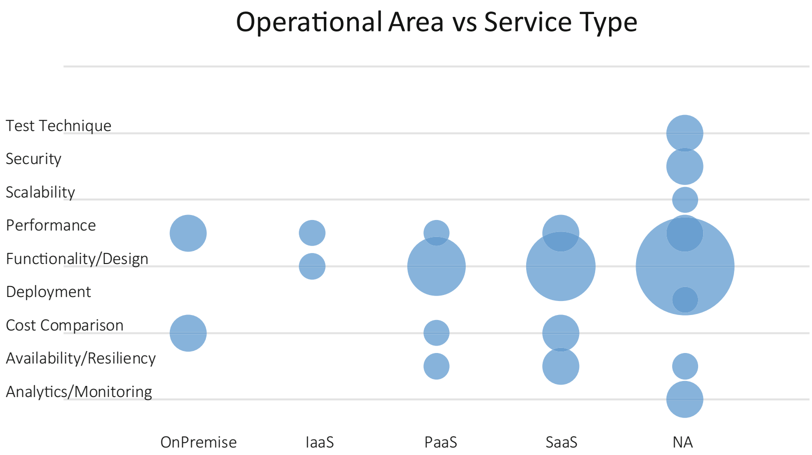

Fig. 7. Operational area vs service type.

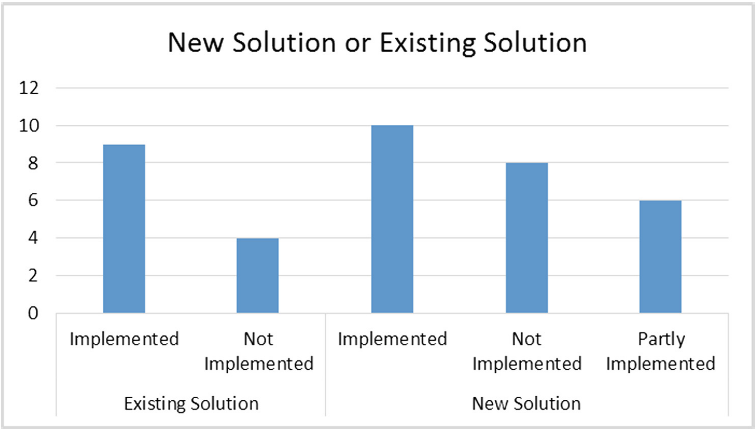

Fig. 8. Implementation of solutions.

research propose new solutions. Another noticeable point is that the implementation ratio of new solutions is higher than the implementation ratio of existing solutions.

RQ3: What are the emerging standards and de facto tools on microservices solutions?

The occurrence of standards proposed or implemented in the research papers included into the systematic mapping can be seen in Fig. 9. The figure includes all the standards either implemented or proposed in systematic mapping papers. As clearly seen in the figure, REST can be called out as the standard for Microservices, even though there is one outlier paper which used non-REST protocol in their study [28].

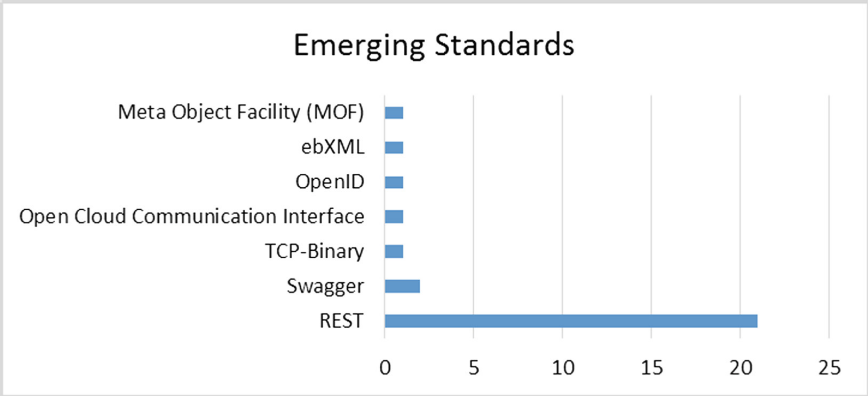

Fig. 9. Emerging standards

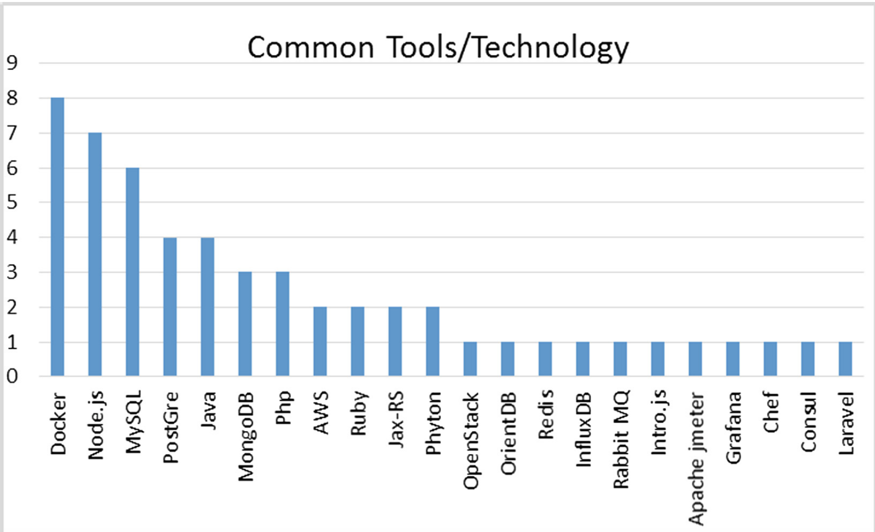

Fig. 10. Common tools used for implementing microservices

Only Swagger is used for microservice markup language. It is interesting to see that WADL or API Blueprint is not mentioned.

The occurrence of tools used in proposed or implemented solutions can be found in Fig. 10. Docker is seen as the most frequently used tool in studies.

The microservices topic is new and the official research started to show up in research papers in 2015. As a result, it is expected for the number of research on microservices to increase over time. Figure 11 shows publication numbers over time. The last search was carried out on the Web of Science in January 2017. On the figure,

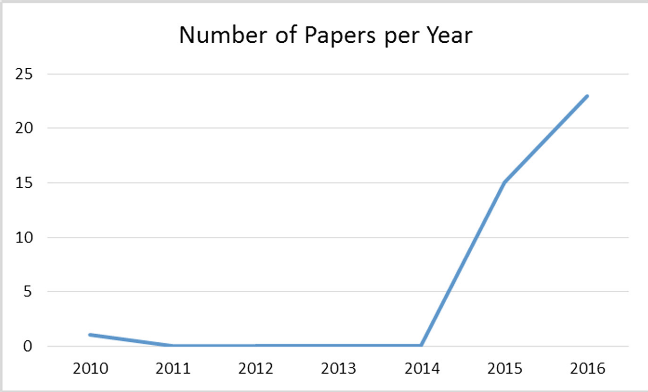

Fig. 11. Number of microservices papers over time without applying exclusion criteria (searched on January 20, 2017)

the line shows the trend. It is seen that the amount of papers increases radically and the trend line is going up.

6 Conclusions and Future Work

The term microservices was first appeared in 2014. All academic papers about microservices belong to 2015 and 2016. From that, we conclude that it is completely a new topic.

Considering the mapping results, we can conclude that microservices is a trending topic and our prediction is that we will see increasing trend in the near future.

Another important conclusion that we draw from the systematic mapping is that there are not enough empirical studies to clarify many issues under discussion related to microservices. Also, there is no research specifically targeting the fragile points of microservices such as distributed transactions.

References

1. Lewis, J., Fowler, M.: "Microservices" martinfowler.com. http://martinfowler.com/articles/microservices.html. Accessed 20 Dec 2016
2. Newman, S.: Building Microservices. O'Reilly Media, Inc., Sebastopol (2015)
3. Petersen, K., Feldt, R., Mujtaba, S., Mattsson, M.: Systematic mapping studies in software engineering. In: 12th International Conference on Evaluation and Assessment in Software Engineering, vol. 17, p. 1 (2008)
4. Dl.acm.org: ACM Digital Library (2016). http://dl.acm.org/dl.cfm. Accessed 05 Jan 2016
5. Citeseerx.ist.psu.edu: CiteSeerX (2016). http://citeseerx.ist.psu.edu/. Accessed 05 Jan 2016
6. Search.ebscohost.com: Computer Source (2016). http://search.ebscohost.com/login.aspx?authtype=ip,uid&profile=ehost&defaultdb=cph. Accessed 05 Jan 2016

7. Site.ebrary.com: ebrary: Server Message (2016). http://site.ebrary.com/lib/utexas. Accessed 05 Jan 2016
8. Hcibib.org: HCI Bibliography: Human-Computer Interaction Resources (2016). http://www.hcibib.org/. Accessed 05 Jan 2016
9. Ieeexplore.ieee.org: IEEE Xplore Digital Library (2016). http://ieeexplore.ieee.org/. Accessed 05 Jan 2016
10. Search.ebscohost.com: INSPEC (2016). http://search.ebscohost.com/login.aspx?authtype=ip,uid&profile=ehost&defaultdb=inh. Accessed 05 Jan 2016
11. Search.ebscohost.com: INSPEC Archive (2016). http://search.ebscohost.com/login.aspx?authtype=ip,uid&profile=ehost&defaultdb=ieh. Accessed 05 Jan 2016
12. Nature.com: Journal home: Nature (2015). http://www.nature.com/nature. Accessed 05 Jan 2016
13. Sciencemag.org: Science (2016). http://www.sciencemag.org/. Accessed 05 Jan 2016
14. Search.ebscohost.com: Science and Technology Collection (2016). http://search.ebscohost.com/login.aspx?authtype=ip,uid&profile=ehost&defaultdb=syh. Accessed 05 Jan 2016
15. Osti.gov: SciTech Connect: Your connection to science, technology, and engineering research information from the U.S. Department of Energy (2016). http://www.osti.gov/scitech/. Accessed 05 Jan 2016
16. Springerlink.com: Home - Springer (2016). http://www.springerlink.com. Accessed 05 Jan 2016
17. Lysne, O., Hole, K., Otterstad, C., Ytrehus, O., Aarseth, R., Tellnes, J.: Vendor malware: detection limits and mitigation. Computer **49**(8), 62–69 (2016)
18. Heorhiadi, V., Rajagopalan, S., Jamjoom, H., Reiter, M., Sekar, V.: Gremlin: systematic resilience testing of microservices. In: 2016 IEEE 36th International Conference on Distributed Computing Systems (ICDCS) (2016)
19. Villamizar, M., Garces, O., Ochoa, L., Castro, H., Salamanca, L., Verano, M., Casallas, R., Gil, S., Valencia, C., Zambrano, A., Lang, M.: Infrastructure cost comparison of running web applications in the cloud using AWS lambda and monolithic and microservice architectures. In: 2016 16th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid) (2016)
20. Villamizar, M., Garces, O., Castro, H., Verano, M., Salamanca, L., Casallas, R., Gil, S.: Evaluating the monolithic and the microservice architecture pattern to deploy web applications in the cloud. In: 2015 10th Computing Colombian Conference (10CCC) (2015)
21. Sun, Y., Nanda, S., Jaeger, T.: Security-as-a-service for microservices-based cloud applications. In: 2015 IEEE 7th International Conference on Cloud Computing Technology and Science (CloudCom) (2015)
22. Rahman, M., Gao, J.: A reusable automated acceptance testing architecture for microservices in behavior-driven development. In: 2015 IEEE Symposium on Service-Oriented System Engineering (2015)
23. Le, V., Neff, M., Stewart, R., Kelley, R., Fritzinger, E., Dascalu, S., Harris, F.: Microservice-based architecture for the NRDC. In: 2015 IEEE 13th International Conference on Industrial Informatics (INDIN) (2015)
24. Alpers, S., Becker, C., Oberweis, A., Schuster, T.: Microservice based tool support for business process modelling. In: 2015 IEEE 19th International Enterprise Distributed Object Computing Workshop (2015)
25. Bak, P., Melamed, R., Moshkovich, D., Nardi, Y., Ship, H., Yaeli, A.: Location and context-based microservices for mobile and internet of things workloads. In: 2015 IEEE International Conference on Mobile Services (2015)

26. Malavalli, D., Sathappan, S.: Scalable microservice based architecture for enabling DMTF profiles. In: 2015 11th International Conference on Network and Service Management (CNSM) (2015)
27. Krylovskiy, A., Jahn, M., Patti, E.: Designing a smart city internet of things platform with microservice architecture. In: 2015 3rd International Conference on Future Internet of Things and Cloud (2015)
28. Ciuffoletti, A.: Automated deployment of a microservice-based monitoring infrastructure. Procedia Comput. Sci. **68**, 163–172 (2015)
29. Meinke, K., Nycander, P.: Learning-based testing of distributed microservice architectures: correctness and fault injection. In: Bianculli, D., Calinescu, R., Rumpe, B. (eds.) SEFM 2015. LNCS, vol. 9509, pp. 3–10. Springer, Heidelberg (2015). doi:10.1007/978-3-662-49224-6_1
30. Pahl, C., Jamshidi, P.: Software architecture for the cloud – a roadmap towards control-theoretic, model-based cloud architecture. In: Weyns, D., Mirandola, R., Crnkovic, I. (eds.) ECSA 2015. LNCS, vol. 9278, pp. 212–220. Springer, Cham (2015). doi:10.1007/978-3-319-23727-5_17
31. Nicolaescu, P., Klamma, R.: A methodology and tool support for widget-based web application development. In: Cimiano, P., Frasincar, F., Houben, G.-J., Schwabe, D. (eds.) ICWE 2015. LNCS, vol. 9114, pp. 515–532. Springer, Cham (2015). doi:10.1007/978-3-319-19890-3_33
32. Koren, I., Nicolaescu, P., Klamma, R.: Collaborative drawing annotations on web videos. In: Cimiano, P., Frasincar, F., Houben, G.-J., Schwabe, D. (eds.) ICWE 2015. LNCS, vol. 9114, pp. 671–674. Springer, Cham (2015). doi:10.1007/978-3-319-19890-3_54
33. Wikipedia: Cloud computing (2016). https://en.wikipedia.org/wiki/Cloud_computing. Accessed 05 Jan 2016
34. Fatema, K., Emeakaroha, V., Healy, P., Morrison, J., Lynn, T.: A survey of cloud monitoring tools: taxonomy, capabilities and objectives. J. Parallel Distrib. Comput. **74**(10), 2918–2933 (2014)
35. Wieringa, R., Maiden, N., Mead, N., Rolland, C.: Requirements engineering paper classification and evaluation criteria: a proposal and a discussion. Requirements Eng. **11**(1), 102–107 (2005)
36. Are Gartner's Predictions on Track - Gartner's Top 10 Strategic Technology Trends for 2016: At a Glance from October 6, 2015 (2016). https://www.linkedin.com/pulse/how-well-did-gartner-do-prediction-gartners-top-10-strategic?trk=pulse-det-nav_art. Accessed 20 Dec 2016
37. Sommerville, I.: Software Engineering, 10th edn. Pearson, London (2016). (Chap. 18)
38. Braun, E., Düpmeier, C., Kimmig, D., Schillinger, W., Weissenbach, K.: Generic web framework for environmental data visualization. In: Wohlgemuth, V., Fuchs-Kittowski, F., Wittmann, J. (eds.) Advances and New Trends in Environmental Informatics. PI, pp. 289–299. Springer, Cham (2017). doi:10.1007/978-3-319-44711-7_23
39. Linthicum, D.: Practical use of microservices in moving workloads to the cloud. IEEE Cloud Comput. **3**(5), 6–9 (2016)
40. Alshuqayran, N., Ali, N., Evans, R.: A systematic mapping study in microservice architecture. In: 2016 IEEE 9th International Conference on Service-Oriented Computing and Applications (SOCA) (2016)
41. Inagaki, T., Ueda, Y., Ohara, M.: Container management as emerging workload for operating systems. In: 2016 IEEE International Symposium on Workload Characterization (IISWC) (2016)
42. Ueda, T., Nakaike, T., Ohara, M.: Workload characterization for microservices. In: 2016 IEEE International Symposium on Workload Characterization (IISWC) (2016)

43. Florio, L., Nitto, E.: Gru: an approach to introduce decentralized autonomic behavior in microservices architectures. In: 2016 IEEE International Conference on Autonomic Computing (ICAC) (2016)
44. Gadea, C., Trifan, M., Ionescu, D., Ionescu, B.: A reference architecture for real-time microservice API consumption. In: Proceedings of the 3rd Workshop on CrossCloud Infrastructures & Platforms - CrossCloud 2016 (2016)
45. Renz, J., Hoffmann, D., Staubitz, T., Meinel, C.: Using A/B testing in MOOC environments. In: Proceedings of the Sixth International Conference on Learning Analytics & Knowledge - LAK 2016 (2016)
46. Hasselbring, W.: Microservices for scalability. In: Proceedings of the 7th ACM/SPEC on International Conference on Performance Engineering - ICPE 2016 (2016)
47. Scarborough, W., Arnold, C., Dahan, M.: Case study. In: Proceedings of the XSEDE16 on Diversity, Big Data, and Science at Scale – XSEDE 2016 (2016)
48. Kecskemeti, G., Marosi, A., Kertesz, A.: The ENTICE approach to decompose monolithic services into microservices. In: 2016 International Conference on High Performance Computing and Simulation (HPCS) (2016)
49. Barais, O., Bourcier, J., Bromberg, Y., Dion, C.: Towards microservices architecture to transcode videos in the large at low costs. In: 2016 International Conference on Telecommunications and Multimedia (TEMU) (2016)
50. Kang, H., Le, M., Tao, S.: Container and microservice driven design for cloud infrastructure DevOps. In: 2016 IEEE International Conference on Cloud Engineering (IC2E) (2016)
51. Messina, A., Rizzo, R., Storniolo, P., Tripiciano, M., Urso, A.: The database-is-the-service pattern for microservice architectures. In: Renda, M.E., Bursa, M., Holzinger, A., Khuri, S. (eds.) ITBAM 2016. LNCS, vol. 9832, pp. 223–233. Springer, Cham (2016). doi:10.1007/978-3-319-43949-5_18
52. Hassan, S., Bahsoon, R.: Microservices and their design trade-offs: a self-adaptive roadmap. In: 2016 IEEE International Conference on Services Computing (SCC) (2016)
53. Bogner, J., Zimmermann, A.: Towards integrating microservices with adaptable enterprise architecture. In: 2016 IEEE 20th International Enterprise Distributed Object Computing Workshop (EDOCW) (2016)
54. Kratzke, N., Peinl, R.: ClouNS - a cloud-native application reference model for enterprise architects. In: 2016 IEEE 20th International Enterprise Distributed Object Computing Workshop (EDOCW) (2016)
55. Thiele, T., Sommer, T., Stiehm, S., Jeschke, S., Richert, A.: Exploring research networks with data science: a data-driven microservice architecture for synergy detection. In: 2016 IEEE 4th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW) (2016)
56. Qanbari, S., Pezeshki, S., Raisi, R., Mahdizadeh, S., Rahimzadeh, R., Behinaein, N., Mahmoudi, F., Ayoubzadeh, S., Fazlali, P., Roshani, K., Yaghini, A., Amiri, M., Farivarmoheb, A., Zamani, A., Dustdar, S.: IoT design patterns: computational constructs to design, build and engineer edge applications. In: 2016 IEEE First International Conference on Internet-of-Things Design and Implementation (IoTDI) (2016)
57. Guo, D., Wang, W., Zeng, G., Wei, Z.: Microservices architecture based cloudware deployment platform for service computing. In: 2016 IEEE Symposium on Service-Oriented System Engineering (SOSE) (2016)
58. Safina, L., Mazzara, M., Montesi, F., Rivera, V.: Data-driven workflows for microservices: genericity in jolie. In: 2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA) (2016)

59. Kratzke, N.: About microservices, containers and their underestimated impact on network performance. In: 6th International Conference on Cloud Computing, GRIDs, and Virtualization (CLOUD COMPUTING) (2015)
60. Fernandez-Villamor, J.I., Iglesias, C., Garijo, M.: MICROSERVICES lightweight service descriptions for rest architectural style. In: 2nd International Conference on Agents and Artificial Intelligence (ICAART 2010) (2010)