

Towards a Definition of Computing Continuum

Pablo Josue Rojas Yepes^{1*}, Carlos Jaime Barrios Hernández¹, Oscar Carrillo², Frédéric Le Mouë¹

¹Supercomputación y Cálculo Científico (SC3UIS), Grupo de Investigación en Cómputo Avanzado y de Gran Escala (CAGE), Universidad Industrial de Santander (UIS), Bucaramanga, Colombia

²Univ Lyon, CPE, INSA Lyon, Inria, CITI, EA3720, F-69621 Villeurbanne, France

*Corresponding author, email: sthongurd@hotmail.com

Hacia una definición del Computing Continuum

Abstract

The evolution of the Computing Continuum, coupled with DevOps practices, marks a significant transformation in modern computing. This paper examines the integration of cloud, fog, edge, and Internet of Things (IoT) technologies to enhance resource utilization, scalability, and collaboration. The synergy between DevOps and orchestration systems automates essential processes, optimizing both performance and security. Despite challenges such as coordination complexities and talent shortages, these advancements hold the potential for increased flexibility and efficiency in Computing Continuum environments. The paper concludes by proposing a definition of the Computing Continuum, informed by state-of-the-art concepts and the interplay between multi-architecture orchestration and DevOps culture.

Keywords: Cloud-to-Edge continuum, Cloud-to-Things continuum, Fog computing, Edge computing, IoT application, DevOps culture, Orchestration and Resource Management.

Resumen

La evolución del Computing Continuum, junto con las prácticas de DevOps, marca una transformación significativa en la computación moderna. Este documento examina la integración de las tecnologías Cloud, Fog, Edge e Internet of Things (IoT) para mejorar la utilización de recursos, la escalabilidad y la colaboración. La sinergia entre DevOps y los sistemas de orquestación automatiza procesos esenciales, optimizando tanto el rendimiento como la seguridad. A pesar de los desafíos como las complejidades de coordinación y la escasez de talento, estos avances tienen el potencial de aumentar la flexibilidad y la eficiencia en los entornos del Computing Continuum. El documento concluye proponiendo una definición del Computing Continuum, basada en conceptos de vanguardia y la interacción entre la orquestación multiarquitectura y la cultura DevOps.

Palabras clave: Continuidad de la Nube al Borde, Continuidad de la Nube al IoT, Computación en la Niebla, Computación en el Borde, Aplicaciones IoT, Cultura DevOps, Orquestación y Administración de Recursos.

INTRODUCTION

Since the early 21st century, computational capabilities have addressed significant human needs, leading to the development of methods and tools to ensure continuity across



Licencia Creative Commons
Atribución-NoComercial 4.0



Editado por /
Edited by:
Dennis Cazar

Recibido /
Received:
19/11/2024

Aceptado /
Accepted:
24/12/2024

Publicado en línea /
Published online:
25/07/2025



various domains such as data, infrastructure, operations, and development. A Computing Continuum system is designed to manage continuous data streams and enable real-time processing for ultra-scale computing applications. Different definitions of Computing Continuum exist, such as the one in [1], which focuses on integrating resources from the edge to the cloud to support dynamic, data-driven applications, or the approach in [2], which introduces a “Markov Blanket”-based methodology to manage distributed systems, overcoming current limitations and establishing a more flexible framework.

The contemporary computing landscape is dynamic, heterogeneous, and vast, necessitating a paradigm shift for effective management and orchestration of applications across this complex ecosystem. The Computing Continuum extends beyond the traditional cloud model, incorporating entities like edge devices, fog nodes, and the IoT, all contributing to the continuum by providing analysis, processing, storage, and data generation capabilities.

Architecturally, a Computing Continuum system encompasses large-scale, complex, and distributed systems, combining parallel computing, heterogeneity, scalability, and large data volumes. Applications within this continuum often scale by orders of magnitude, similar to ultra-scale computing systems [3]. However, while ultra-scale computing focuses on extending individual systems, the Computing Continuum seeks to unify computing resources into a seamless continuum. The shared architectural features suggest that ultra-scale computing systems can be components of a Computing Continuum.

The primary objective in application deployment within the continuum is to leverage its extensive capabilities. This includes the low latency and proximity benefits of edge computing and the scalability of cloud data centers, as well as the integration of resources across edge, core, and data paths to support dynamic, flexible, data-driven workflows. Effective orchestration is crucial in this context.

Orchestration systems automate the seamless delivery of applications across diverse cloud environments, ensuring that Quality of Service (QoS) objectives are met. These systems manage tasks like resource selection, deployment, monitoring, and runtime control. The rise of DevOps has significantly transformed software development and delivery, emphasizing collaboration and agility. The synergy between orchestration and DevOps strengthens the Computing Continuum by fostering efficiency, flexibility, and resilience in managing applications across diverse architectures.

This work is divided into four sections. The first one will address topics related to the concept of the Computing Continuum and related works (the bibliography is extensive, so individual articles may be omitted). The second will present the contributions of orchestration and DevOps CI/CD to the Computing Continuum. The third will discuss how the contributions of Synergy into Apps, Orchestrations, Devops and infrastructure section relate to those outlined in the concept of Computing Continuum and related works section. Finally, the fourth section will present the conclusions of this work. It should be noted that this study is an exploration of the extensive theory surrounding the concept of the Computing Continuum, the characterization of the challenges to be faced, and proposals for potential solutions based on well-defined topics (Orchestration and DevOps). This is why we do not include use cases in this work.

CONCEPT OF COMPUTING CONTINUUM AND RELATED WORKS

The Computing Continuum has evolved over nearly a decade, propelled by advancements such as diverse cloud services, the surge in computing [4] and data manipulation [5] capabilities in fog and edge devices, the democratization of IoT through open hardware initiatives, and the adoption of DevOps culture. DevOps integrates repositories, development, deployment, and operations with orchestrators, facilitating the efficient management of large-scale data across multiple architectures. Conceptually, the Computing Continuum unifies various computing resources with distinct capabilities and characteristics.

Fog and cloud computing address challenges like quality of service [6] and security [7] in SDN and NFV environments, enabling applications in sectors such as agriculture 4.0 [8]. The digital transformation and IoT require new strategies, highlighting the importance of time-aware data spaces for scalability [9]. Fog computing extends cloud services to the network edge, supporting IoT, 5G, and AI applications [10]. The cloud-to-things continuum, integrated with technologies like iCloudFog [11, 12], manages latency-sensitive IoT data and addresses resource management and security challenges, facilitating real-time applications like tsunami warning systems [13].

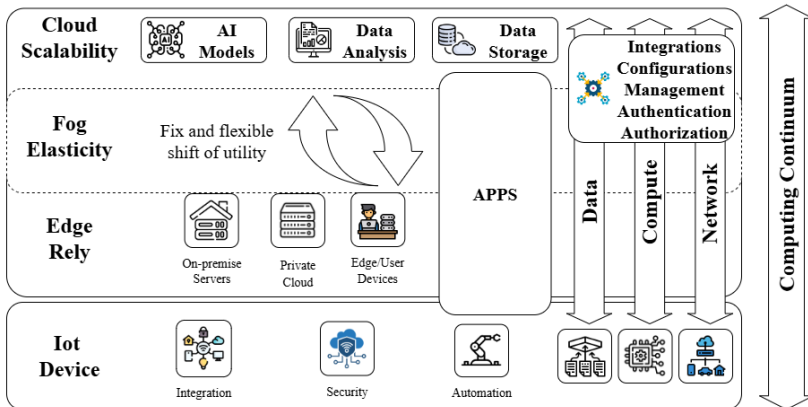


Figure 1. Representation of a Computing Continuum Concept.

Optimizing resources in IoT-fog-cloud environments is crucial [14]. Rule-based resource matching [15] and emulation approaches [16, 17] enhance service placement, maximizing performance in computing continua. Serverless computing [18] and dynamic function distribution offer scalable solutions for complex applications [19]. Tools like Continuum Deployer [20] and models like CODA [21] optimize application and service allocation, considering performance, carbon footprint [22], and quality of service. Innovative approaches like EPOS Fog [23] and the Markov Blanket concept [24] improve efficiency in distributed environments [25].



Key features of the Computing Continuum include minimizing latency [21] and optimizing data processing [15] by distributing computational tasks across edge, fog, and cloud levels [14], enhancing overall system performance [25, 26, 27].

A conceptual architecture of a Computing Continuum system, as depicted in Fig. 1, identifies different elements and levels, illustrating their relationships across varying scales, while considering application behavior and data characteristics. This representation extends beyond the conventional cloud model, incorporating edge, fog, and IoT entities that contribute to the continuum by offering analysis, processing, storage, and data generation capabilities [6].

This evolution, underscored by integrating DevOps with these entities, highlights the dynamic nature of modern computing paradigms [11]. Central to this concept is the ACDC (Application Choose Device Carefully) proposal [28], which emphasizes the flexibility and scalability of deploying general-purpose systems across diverse devices. Orchestrators play a crucial role in automating application delivery across environments, though orchestrating this continuum requires solutions that ensure efficient execution and robust security, which will be addressed in the next section.

SYNERGY INTO APPS, ORCHESTRATIONS, DEVOPS, AND INFRASTRUCTURE

Orchestration Across Multiple Architectures

Orchestration systems play a pivotal role in automating the seamless delivery of applications across diverse cloud environments, ensuring various Quality of Service (QoS) goals are met. These systems manage complex tasks such as resource selection, deployment, monitoring, and runtime control [29]. Across this spectrum, a plethora of solutions exist, ranging from vendor-specific offerings like Amazon's AWS CloudFormation [30], Openstack orchestration [31], Microsoft Azure's Resource Manager (ARM) templates [32], Google cloud deployment manager [33], to open-source cloud-agnostic initiatives like Kubernetes [34], Docker Swarm [35], Apache Brooklyn [36], Cloudify orchestration platform [37], Cloudiator [38], Alien4Cloud [39] and MiCADO-Edge [40].

Traditional cloud architectures often fall short when confronted with the demands of Fig. 1, paving the way for emerging paradigms like fog computing, edge computing, and compute continuum. Fog computing serves as an intermediary layer between the cloud and edge, spanning across multiple network topology layers [41]. Conversely, edge computing involves nodes close to IoT devices, often embedded within them [42]. Fig. 1 extends the cloud by integrating energy-efficient and low-latency devices closer to data sources at the network edge [22]. This amalgamation of fog, edge, and IoT entities endows diverse capabilities such as analysis, processing, storage, and data generation [43].

However, extending orchestration requirements to encompass this Fig. 1 poses a new challenge, as existing solutions struggle to address its diverse and heterogeneous nature [44]. Challenges include federated coordination across administrative domains, volatility and mobility management, and efficient monitoring mechanisms [45]. Orchestration



solutions must adapt to these challenges by offering efficient runtime mechanisms, enforcing policy-based deployment, and reconfiguring to meet SLA goals [46]. With the scale of Fig. 1 being massive, orchestration systems must ensure security against various attack scenarios while minimizing user configuration needs [47].

Recent studies have delved into the challenges and opportunities of orchestration in heterogeneous computing environments, with a particular focus on fog computing [44]. They analyze advanced orchestration architectures and systems to effectively manage the Internet of Things [48], presenting a framework to evaluate the business value of emerging technologies such as fog, edge, and 5G [49]. The orchestration of IoT services in fog environments presents challenges due to system heterogeneity and dynamics. This is explored with solutions based on genetic algorithms and adapting cloud frameworks [50], in addition to analyzing the integration of edge and fog resources in 5G [51, 52] and 6G networks, using artificial intelligence to improve orchestration [53].

The challenges and opportunities of orchestration in IoT systems are explored, analyzing existing approaches and tools [54], as well as cloud-based and edge-based architectures [55]. Research gaps are identified, and innovative solutions are proposed [56], such as artificial intelligence and Kubernetes technologies, to address the requirements of real-time processing, fault tolerance, and other key features [57]. Furthermore, the challenges of orchestration in cloud-edge and fog environments are explored, with a focus on smart city and IoT applications [58]. Architectures such as Kubernetes are analyzed, as well as security and privacy techniques to ensure data protection in these environments [59, 60]. Additionally, frameworks based on microservices and containers are proposed to provide secure and intelligent solutions at the network edge [61].

Culture DevOps

The rise of DevOps culture has transformed software development and delivery, promoting a collaborative and agile approach that departs from traditional waterfall methodologies. DevOps focuses on bridging the gap between development and operations teams by fostering collaboration and automating processes throughout the software development lifecycle. In Cyber-Physical Production Systems (CPPS), DevOps faces unique challenges due to long-term investments, limited flexibility in asset-heavy environments, and the physical nature of hardware [62]. Addressing these challenges requires tailored approaches to ensure successful implementation.

The proliferation of DevOps tools mirrors the Cambrian explosion, with specialized tools emerging to address specific aspects of the software value stream [63]. This diversification is driven by the need for faster lead times, smaller batch sizes, and increased automation. The democratization of the toolchain, fueled by open-source technologies like Git, empowers practitioners to choose tools that best suit their needs, rather than following a one-size-fits-all model [64].

As the software development landscape evolves, adopting DevOps practices becomes crucial for organizations aiming to stay competitive in a fast-paced market [65]. However, the path to DevOps implementation is fraught with challenges. A significant barrier is the shortage of skilled professionals proficient in DevOps methodologies [66]. Recruiting and retaining talent, along with the steep learning curve associated with new tools and



processes, complicates adoption and necessitates comprehensive training programs [67]. Additionally, resistance to change and uncertainty within organizations impede DevOps adoption, requiring effective change management strategies [68].

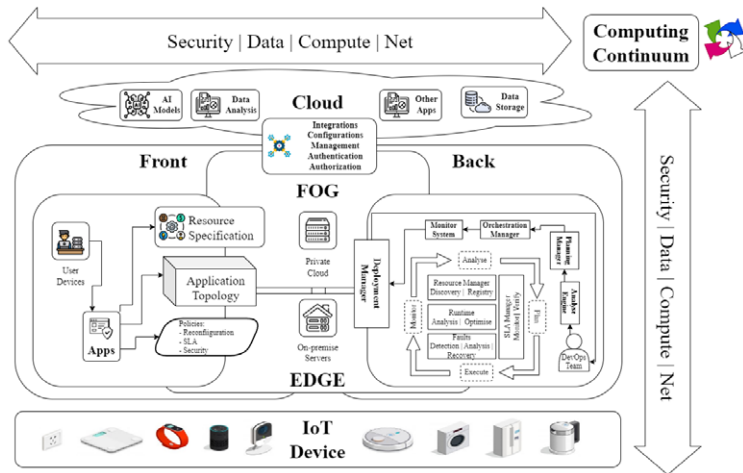


Figure 2. Synergy into apps, orchestrations, and infrastructure.

Infrastructure challenges, as illustrated in Fig. 2, also hinder the smooth integration of DevOps practices [69]. Organizations must develop a robust infrastructure capable of supporting DevOps, addressing issues related to design, provisioning, and hardware/software problem-solving [70]. The size and maturity of an organization further impact its ability to successfully implement DevOps, particularly in larger corporations with complex legacy systems [71]. Transitioning from outdated platforms demands significant resource investment and careful planning.

Siloed communication channels obstruct team collaboration, as teams accustomed to working separately may struggle with the cross-functional collaboration essential to DevOps [72]. Overcoming these barriers requires cultivating a culture of transparency, open communication, and shared responsibility. Additionally, maintaining strong security measures in the context of accelerated release timelines is a significant challenge [73]. Rapid deployments and continuous delivery can compromise security evaluations, making it crucial to bridge the communication gap between development and security teams.

The constant evolution of DevOps processes necessitates adaptability and a commitment to continuous improvement and learning [70]. The growing demand for user experiences tailored to specific roles has led vendors to specialize their offerings, fueling the expansion of the DevOps toolchain. This specialization reflects the evolutionary nature of the market, where vendors cater to the diverse needs of organizations developing increasingly complex software.

The synergy between DevOps and orchestration systems enhances resource utilization and scalability, emphasizing automation and collaboration. This alignment is vital for navigating the complexities of the Computing Continuum, ensuring seamless



deployment across diverse architectures and fostering innovation and security, as depicted in Fig. 2. This transition represents a transformative journey towards greater efficiency and resilience in modern software development.

DISCUSSION

The integration of orchestration systems and DevOps practices within the Computing Continuum as shown in Fig. 2 represents a powerful synergy that addresses many of the challenges associated with managing heterogeneous and distributed computing environments. This synergy is highlighted in Table 1, which outlines the relationship between various computing architectures and their performance, offering a clear view of how these technologies interact to create a more efficient and resilient computing infrastructure.

TABLE 1. Relationship between Computing Continuum References (Architecture and Performance)

Architecture / Performance	Reference
Cloud	[6, 7, 12, 13, 16, 18, 20–22, 25, 29, 30, 32, 43, 51, 54, 55, 57]
Fog	[6, 10–22, 25–27, 41, 44, 47–51, 53, 54, 57–61]
Edge	[6, 9, 13–16, 18–27, 36–42, 44, 46–61]
IoT	[6–8, 12–14, 18–27, 41, 43–61]
Anywhere	[14, 16–27, 44–61]

Orchestration systems such as Kubernetes and Docker Swarm play a pivotal role in managing the complex landscape of the Computing Continuum by automating the deployment, scaling, and management of applications across diverse environments, from the cloud to the edge. These systems are essential for ensuring that (QoS) goals are met, particularly in dynamic and distributed computing environments where resource heterogeneity and geographical distribution present significant challenges [29]. The ability of orchestration systems to handle resource selection, deployment, and runtime control is crucial for optimizing resource utilization and minimizing latency across the continuum [30, 34]. This is particularly important in scenarios where real-time processing and fault tolerance are critical, as these systems can dynamically adjust to changing conditions and maintain service continuity.

The integration of DevOps culture within the orchestration framework further enhances the effectiveness of these systems. DevOps practices emphasize automation, collaboration, and continuous improvement, breaking down traditional silos between development and operations teams and fostering a culture of shared responsibility. This alignment between orchestration and DevOps enables organizations to achieve greater efficiency, flexibility, and resilience in their computing infrastructure [64]. By automating the integration, testing, and delivery of software artifacts, DevOps ensures that development and operations teams work seamlessly together to deliver and maintain applications across the Computing Continuum [44, 66, 67].



The convergence of orchestration and DevOps within the Computing Continuum also enhances security by automating security measures and enforcing policy-based deployment and configuration. This is critical in environments where applications and data span multiple architectures, including cloud, edge, and IoT devices. The security mechanisms provided by orchestration systems help mitigate the challenges of volatility, mobility management, and security, ensuring that the Computing Continuum remains secure and resilient against various threats [44, 69].

Despite the clear benefits, implementing an integrated Computing Continuum with DevOps practices is not without challenges. The complexity of coordinating across multiple administrative domains, managing resource volatility, and ensuring efficient and secure monitoring are significant obstacles. Additionally, the shortage of skilled DevOps professionals and organizational resistance to change can hinder the adoption and success of these practices. Overcoming these challenges requires effective change management strategies and robust training programs to equip teams with the necessary skills [42, 6, 69, 72].

The synergy between orchestration systems and DevOps practices is essential for navigating the complexities of the Computing Continuum. As illustrated in Table 1, this synergy enhances the efficiency, scalability, and security of computing environments, enabling organizations to adapt quickly to evolving demands and technological advancements. Future research should focus on developing more adaptive and secure orchestration solutions, as well as exploring strategies for training and organizational change that facilitate the successful integration of DevOps within the Computing Continuum.

CONCLUSION

In conclusion, the Computing Continuum represents a transformative and integrated ecosystem that encompasses the entire spectrum of computing resources, from the network edge to the cloud. This paradigm integrates a variety of computing architectures—cloud, fog, edge, and IoT—into a unified system designed to deliver scalable, efficient, and resilient solutions. The evolution of this concept is driven by advances in cloud services, the increasing adoption of edge and fog computing, the rise of open hardware initiatives, and the incorporation of DevOps practices.

At its core, the Computing Continuum aims to reduce latency and optimize data processing by distributing computational tasks across multiple layers. This strategic approach enhances resource utilization and system performance by enabling the simultaneous execution of workloads on edge, fog, and cloud tiers. The seamless integration of these architectures, supported by containerization technologies and orchestration, provides the flexibility and agility necessary to meet diverse application requirements.

Furthermore, the integration of DevOps within the Computing Continuum strengthens its capabilities by fostering collaboration and automation. DevOps practices dissolve traditional barriers between development and operations teams, enabling more efficient and scalable application deployment across heterogeneous environments. This synergy addresses critical challenges such as federated coordination, volatility management, and security, thereby ensuring robust and adaptable computing infrastructures.



The Computing Continuum signifies a pivotal shift in computing paradigms, expanding beyond traditional cloud models to incorporate edge, fog, and IoT devices. By clearly defining the elements, contexts, and relationships within this ecosystem, the Computing Continuum emerges as a scalable and distributed computing system, capable of supporting a wide range of applications with varying demands. However, the orchestration of such a complex and heterogeneous system presents challenges, including managing resource capabilities, data diversity, resilience, mobility, and the scalability of applications. Overcoming these challenges is crucial to fully unlocking the potential of the Computing Continuum in the modern computing landscape.

In essence, the Computing Continuum is an abstract that offers a unifying approach to computing resources. By integrating diverse architectures and leveraging innovative technologies, it can deliver scalable, efficient, and resilient solutions. What truly distinguishes the Computing Continuum is its focus on minimizing latency and optimizing data processing through the simultaneous execution of tasks across multiple computing layers, a capability made possible by containerization technologies and DevOps practices.

ACKNOWLEDGEMENTS

It is with great pleasure that we acknowledge the invaluable contribution of Collaboration France-Colombie en Technologies Avancées de l'Information et la Communication (CATAI) in facilitating the collaboration between the Universidad Industrial de Santander (UIS) and Inria. Your support has not only enabled us to achieve our research goals but has also strengthened the ties between our two institutions.

AUTHOR CONTRIBUTIONS

Pablo Josue Rojas Yepes contributed to the roles of Investigation, Methodology, and Writing – Original Draft. He led the methodological design, executed the experiments, and prepared the initial manuscript.

Carlos Jaime Barrios Hernández participated in Validation, Project Administration, Supervision, and Funding Acquisition. He ensured the quality of the results, managed project activities, and secured the necessary financial resources.

Oscar Carrillo performed roles in Validation, Formal Analysis, Supervision, and Writing – Review & Editing. His contributions included detailed data analysis, academic supervision of the work, and critical review of the manuscript.

Frédéric Le Mouël participated in Validation, Conceptualization, Project Administration, and Supervision. His contributions involved defining the conceptual approach, securing funding, managing the project, and providing strategic guidance.

CONFLICT OF INTEREST

Authors declare they have no conflicts of interest in this research.

REFERENCES

- [1] Balouek, D., Renart, E. G., Zamani, A. R., Simonet-Boulogne, A., & Parashar, M. (2019). Towards a computing continuum: Enabling edge-to-cloud integration for data-driven workflows. *The International Journal of High Performance Computing Applications*, 33(6), 1159–1174. <http://dx.doi.org/10.1177/1094342019877383>
- [2] Dustdar, S., Casamayor Pujol, V., & Donta, P. K. (2022). On distributed computing continuum systems. *IEEE Transactions on Knowledge and Data Engineering*, 35(4), 4092–4105. https://dsg.tuwien.ac.at/team/sd/papers/Zeitschriftenartikel_2022_SD_On_Distributed.pdf
- [3] Carretero, J., García-Blas, J., & Ciegis, R. (2016). Introduction to sustainable ultrascale computing systems and applications. *Journal of Supercomputing*, 72(10), 4043–4046. <https://doi.org/10.1007/s11227-016-1822-8>
- [4] Matsuoka, S. (2018). Cambrian Explosion of Computing and Big Data in the Post-Moore Era. In *Proceedings of the 27th International Symposium on High-Performance Parallel and Distributed Computing (HPDC '18)* (pp. 1–2). ACM. <https://doi.org/10.1145/3208040.3225055>
- [5] Matsuoka, S., Hanawa, T., Endo, T., Amano, H., Nakajima, K., Inoue, K., Kudoh, T., Maruyama, N., Taura, K., Iwashita, T., Katagiri, T. (2016). From FLOPS to BYTES: Disruptive change in high-performance computing towards the post-Moore era. In *CF '16: Proceedings of the ACM International Conference on Computing Frontiers* (pp. 274–281). ACM. <http://dx.doi.org/10.1145/2903150.2906830>
- [6] Gupta, H., Nath, S. B., Chakraborty, S., & Ghosh, S. K. (2016). SDFog: A software defined computing architecture for QoS aware service orchestration over edge devices. *arXiv preprint arXiv:1609.01190*. <http://dx.doi.org/10.48550/arXiv.1609.01190>
- [7] Kahvazadeh, S., Masip, X., Marin-Tordera, E., & Gómez-Cárdenas, A. (2019). Securing combined fog-to-cloud systems: Challenges and directions. In L. A. D. Al-Sakran, B. Ko, S. C. Mishra, & M. S. Obaidat (Eds.), *FTC 2019: Proceedings of the Future Technologies Conference (FTC) 2019* (pp. 877–892). http://dx.doi.org/10.1007/978-3-030-32520-6_63
- [8] Xhafa, F., & Krause, P. (2021). IoT-based computational modeling for next generation agro-ecosystems: Research issues, emerging trends and challenges. In F. Xhafa & A. T. H. Yachia (Eds.), *IoT-Based Intelligent Modelling for Environmental and Ecological Engineering* (pp. 1–21). <http://dx.doi.org/10.1007/978-3-030-71172-6>
- [9] Zeiner, H., & Unterberger, R. (2021). Time-aware data spaces—A key computing unit in the edge-to-cloud continuum. In *2021 FiCloud: Proceedings of the 9th IEEE International Conference on Future Internet of Things and Cloud* (pp. 250–255). IEEE. <https://doi.org/10.1109/FiCloud49777.2021.00043>
- [10] Chiang, M., & Zhang, T. (2016). Fog and IoT: An overview of research opportunities. *IEEE Internet of Things Journal*, 3(6), 854–864. <https://doi.org/10.1109/JIOT.2016.2584538>
- [11] Coughlin, T. (2017). Convergence through the cloud-to-thing consortium [future directions]. *IEEE Consumer Electronics Magazine*, 6(3), 14–17. <http://dx.doi.org/10.1109/MCE.2017.2684914>
- [12] Peng, L., Dhaini, A. R., & Ho, P. (2018). Toward integrated cloud-fog networks for efficient IoT provisioning: Key challenges and solutions. *Future Generation Computer Systems*, 88(8), 606–613. <http://dx.doi.org/10.1016/j.future.2018.05.015>
- [13] Balouek, D., Renart, E. G., Zamani, A. R., Simonet-Boulogne, A., & Parashar, M. (2019). Towards a computing continuum: Enabling edge-to-cloud integration for data-driven workflows. *The International Journal of High Performance Computing Applications*, 33(6), 1159–1174. <http://dx.doi.org/10.1177/1094342019877383>
- [14] Spillner, J., Gkikopoulos, P., Buzachis, A., & Villari, M. (2020). Rule-based resource matchmaking for composite application deployments across IoT-fog-cloud continuums. In *2020 IEEE/ACM International Conference on Utility and Cloud Computing (UCC)* (pp. 336–341). <http://dx.doi.org/10.1109/UCC48980.2020.00053>
- [15] Kassir, S., de Veciana, G., Wang, N., Wang, X., & Palacharla, P. (2020). Service placement for real-time applications: Rate-adaptation and load-balancing at the network edge. In *2020 IEEE International Conference on Cloud and Big Data Computing (CSCloud) and 2020 IEEE International Conference on Edge Computing (EdgeCom)* (pp. 207–215). <https://doi.org/10.1109/CSCloud-EdgeCom49738.2020.00044>
- [16] Beckman, P., Dongarra, J., Ferrier, N., Fox, G., Moore, T., Reed, D., & Beck, M. (2020). Harnessing the computing continuum for programming our world. In R. Buyya & S. Dastjerdi (Eds.), *Fog Computing: Theory and Practice* (pp. 215–230). <https://doi.org/10.1002/9781119551713.ch7>



- [17] Luckow, A., Rattan, K., & Jha, S. (2021). Exploring task placement for edge-to-cloud applications using emulation. In *2021 IEEE International Conference on Fog and Edge Computing (ICFEC)* (pp. 79–83). <https://doi.org/10.1109/ICFEC51620.2021.00019>
- [18] Risco, S., Moltó, G., Naranjo, D. M., & Blanquer, I. (2021). Serverless workflows for containerised applications in the cloud continuum. *Journal of Grid Computing, 19*(30). <https://doi.org/10.1007/s10723-021-09570-2>
- [19] Spillner, J. (2021). Self-balancing architectures based on liquid functions across computing continuums. In *2021 IEEE/ACM International Conference on Utility and Cloud Computing (UCC)* (pp. 1–6). IEEE. <https://doi.org/10.1145/3492323.3495589>
- [20] Hass, D., & Spillner, J. (2021). Interactive application deployment planning for heterogeneous computing continuums. In L. T. Yang, M. M. S. Khan, R. S. N. Ochi, & S. N. T. K. Tseng (Eds.), *2021 IEEE International Conference on Advanced Information Networking and Applications (AINA)* (pp. 551–560). http://dx.doi.org/10.1007/978-3-030-75078-7_55
- [21] Mehran, N., Kimovski, D., & Prodan, R. (2021). A two-sided matching model for data stream processing in the cloud-fog continuum. In *2021 IEEE International Conference on Cloud Computing (CCGrid)* (pp. 514–524). IEEE. <https://doi.org/10.1109/CCGrid51090.2021.00061>
- [22] Kimovski, D., Mathá, R., Hammer, J., Mehran, N., Hellwagner, H., & Prodan, R. (2021). Cloud, fog, or edge: Where to compute? *IEEE Internet Computing, 25*(4), 30–36. <https://arxiv.org/pdf/2101.10417>
- [23] Nezami, Z., Zamanifar, K., Djemame, K., & Pourmaras, E. (2021). Decentralized edge-to-cloud load balancing: Service placement for the Internet of Things. *IEEE Access, 9*, 64983–65000. <https://doi.org/10.1109/ACCESS.2021.3074962>
- [24] Dustdar, S., Casamayor Pujol, V., & Donta, P. K. (2022). On distributed computing continuum systems. *IEEE Transactions on Knowledge and Data Engineering, 35*(4), 4092–4105. <https://doi.org/10.1109/TKDE.2022.3142856>
- [25] Spillner, J., Borin, J. F., & Bittencourt, L. F. (2022). Intent-based placement of microservices in computing continuums. In M. Klymash, M. Beshley, & A. Luntovskyy (Eds.), *Future Intent-Based Networking* (pp. 38–50). https://doi.org/10.1007/978-3-030-92435-5_3
- [26] Milojević, D. (2020). The edge-to-cloud continuum. *Computer, 53*(11), 16–25. <https://doi.org/10.1109/MC.2020.3007297>
- [27] Rosendo, D., Silva, P., Simonin, M., Costan, A., & Antoniu, G. (2020). E2clab: Exploring the computing continuum through repeatable, replicable and reproducible edge-to-cloud experiments. In *2020 IEEE International Conference on Cluster Computing (CLUSTER)* (pp. 176–186). IEEE. <https://doi.org/10.1109/CLUSTER49012.2020.00028>
- [28] Rojas Yepes, P. J., Barrios Hernandez, C. J., & Steffanel, L. A. (2022). A Methodology for Evaluating the Energy Efficiency of Post-Moore Architectures. In *High Performance Computing. CARLA 2021. Communications in Computer and Information Science* (Vol. 1540). https://doi.org/10.1007/978-3-031-04209-6_4
- [29] Tomarchio, O., Calcaterra, D., & Di Modica, G. (2020). Cloud resource orchestration in the multi-cloud landscape: A systematic review of existing frameworks. *Journal of Cloud Computing, 9*(1), 1–25. <https://doi.org/10.1186/s13677-020-00194-7>
- [30] Amazon. (2024). *Aws cloudformation: Speed up cloud provisioning with infrastructure as code*. Retrieved May 4, 2024, from <https://aws.amazon.com/cloudformation/>
- [31] OpenStack. (2024). *Openstack orchestration*. OpenStack Wiki. <https://wiki.openstack.org/wiki/Heat>
- [32] Azure. (2024). *Azure Resource Manager (ARM) templates*. Microsoft Learn. <https://docs.microsoft.com/en-us/azure/azure-resource-manager/templates/overview>
- [33] Google. (2024). *Google Cloud Deployment Manager*. Google Cloud. <https://cloud.google.com/deployment-manager>
- [34] Kubernetes. (2024). *Kubernetes: Production-grade container orchestration*. <https://kubernetes.io/>
- [35] Docker. (2024). *Docker Swarm*. Docker Documentation. <https://docs.docker.com/engine/swarm/>
- [36] Apache Brooklyn. (2024). *Apache Brooklyn: Software for managing cloud applications*. <https://brooklyn.apache.org/>
- [37] Cloudify. (2024). *Cloudify orchestration platform - Multi cloud, cloud native & edge*. <https://cloudify.co/>
- [38] Cloudiator. (2024). *Cloudiator: A multi-tenant, cross-cloud orchestration framework*. GitHub. <https://github.com/cloudiator>
- [39] Alien4Cloud. (2024). *Alien 4 Cloud*. <https://alien4cloud.github.io/>



- [40] Ullah, A., Dagdeviren, H., Ariyattu, R. C., Casale, G., & Pllana, S. (2021). MiCADO-Edge: Towards an application-level orchestrator for the cloud-to-edge computing continuum. *Journal of Grid Computing*, 19(4), 47. <https://doi.org/10.1007/s10723-021-09589-5>
- [41] IEEE. (2018). IEEE standard for adoption of openfog reference architecture for fog computing. *IEEE Std 2018*, 1–176. <https://doi.org/10.1109/IEEESTD.2018.8423800>
- [42] Kimovski, D., Mathá, R., Hammer, J., Mehran, N., Hellwagner, H., & Prodan, R. (2021). Cloud, fog, or edge: Where to compute? *IEEE Internet Computing*, 25(4), 30–36. <https://arxiv.org/pdf/2101.10417>
- [43] Moreschini, S., Pecorelli, F., Li, X., Naz, S., Hästbacka, D., & Taibi, D. (2022). Cloud continuum: The definition. *IEEE Access*, 10, 131876–131886. <https://doi.org/10.1109/ACCESS.2022.3229185>
- [44] Svorobej, S., Bendeache, M., Griesinger, F., & Domaschka, J. (2020). Orchestration from the Cloud to the Edge. In T. Lynn, J. G. Mooney, B. Lee, & P. T. Endo (Eds.), *The Cloud-to-Thing Continuum: Opportunities and Challenges in Cloud, Fog and Edge Computing* (pp. 61–77). Springer International Publishing, Cham. https://doi.org/10.1007/978-3-030-41110-7_4
- [45] Bittencourt, L., Immich, R., Sakellariou, R., Fonseca, N., Madeira, E., Curado, M., Villas, L., DaSilva, L., Lee, C., & Rana, O. (2018). The internet of things, fog and cloud continuum: Integration and challenges. *Internet of Things*, 3-4, 134–155. <https://doi.org/10.1016/j.iot.2018.09.005>
- [46] DesLauriers, J., Kiss, T., Ariyattu, R. C., Dang, H.-V., Ullah, A., Bowden, J., Krefting, D., Pierantoni, G., & Terstysnszky, G. (2021). Cloud apps to-go: Cloud portability with TOSCA and MiCADO. *Concurrency and Computation: Practice and Experience*, 33(19), e6093. <https://doi.org/10.1002/cpe.6093>
- [47] Ullah, A., Dagdeviren, H., Ariyattu, R. C., DesLauriers, J., Kiss, T., & Bowden, J. (2021). MiCADO-Edge: Towards an application-level orchestrator for the cloud-to-edge computing continuum. *Journal of Grid Computing*, 19(4), 1–28. <https://doi.org/10.1007/s10723-021-09589-5>
- [48] Velasquez, K., Abreu, D. P., Assis, M. R., Senna, C., Aranha, D. F., Bittencourt, L. F., Laranjeiro, N., Curado, M., Vieira, M., Monteiro, E., & Madeira, E. (2018). Fog orchestration for the Internet of Everything: State-of-the-art and research challenges. *Journal of Internet Services and Applications*. <https://doi.org/10.1186/s13174-018-0086-3>
- [49] Lynn, T., Mooney, J. G., Lee, B., & Endo, P. T. (2020). *The cloud-to-thing continuum: Opportunities and challenges in cloud, fog and edge computing*. Springer. <https://doi.org/10.1007/978-3-030-41110-7>
- [50] Wen, Z., Yang, R., Garraghan, P., Lin, T., Xu, J., & Rovatsos, M. (2017). Fog orchestration for internet of things services. *IEEE Internet Computing*, 21(2), 16–25. <https://doi.org/10.1109/MIC.2017.36>
- [51] Jiang, Y., Huang, Z., & Tsang, D. H. (2018). Challenges and solutions in fog computing orchestration. *IEEE Network*, 32(1), 12–19. <https://doi.org/10.1109/MNET.2017.1700271>
- [52] Comma-Di, L., Abdullaziz, O. I., Antevski, K., Chundrigar, S. B., Gdowski, R., Kuo, P. H., Mourad, A., Yen, L. H., & Zabala, A. (2018). Opportunities and challenges of joint edge and Fog orchestration. In *2018 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)* (pp. 53–58). IEEE. <https://doi.org/10.1109/WCNCW.2018.8369006>
- [53] Velasquez, K., Abreu, D. P., Curado, M., & Monteiro, E. (2022). Resource orchestration in 5G and beyond: Challenges and opportunities. *Computer Communications*, 192, 311–315. <https://doi.org/10.1016/j.comcom.2022.06.019>
- [54] Nguyen, P. H., Ferry, N., Erdogan, G., Song, H., Lavirotte, S., Tigli, J. Y., & Solberg, A. (2019). Advances in deployment and orchestration approaches for IoT—A systematic review. In *Proceedings - 2019 IEEE International Congress on Internet of Things, ICIOT 2019 - Part of the 2019 IEEE World Congress on Services* (pp. 129–136). IEEE. <https://doi.org/10.1109/ICIOT.2019.00021>
- [55] Wu, Y. (2020). Cloud-edge orchestration for the Internet of Things: Architecture and AI-powered data processing. *IEEE Internet of Things Journal*, 8(16), 12792–12805. <https://doi.org/10.1109/JIOT.2020.3014845>
- [56] Vaquero, L. M., Cuadrado, F., Elkhatib, Y., Bernal-Bernabe, J., Srirama, S. N., & Zhani, M. F. (2019). Research challenges in nextgen service orchestration. *Future Generation Computer Systems*, 94, 780–792. <https://doi.org/10.1016/j.future.2018.07.039>
- [57] Böhm, S., & Wirtz, G. (2022a). Towards orchestration of cloud-edge architectures with Kubernetes. In *Science and Technologies for Smart Cities: 7th EAI International Conference, SmartCity360, Virtual Event, December 2-4, 2021, Proceedings* (Vol. 423, pp. 207–230). http://dx.doi.org/10.1007/978-3-031-06371-8_14



- [58] Böhm, S., & Wirtz, G. (2022). Cloud-edge orchestration for smart cities: A review of Kubernetes-based orchestration architectures. *EAI Endorsed Transactions on Smart Cities*, 6(18), e2–e2. <https://doi.org/10.4108/eetsc.v6i18.1197>
- [59] Fakude, N. C., Tarwireyi, P., Adigun, M. O., & Abu-Mahfouz, A. M. (2019). Fog orchestrator as an enabler for security in fog computing: A review. In *Proceedings - 2019 International Multidisciplinary Information Technology and Engineering Conference (IMITEC)* (pp. 1–6). IEEE. <https://doi.org/10.1109/IMITEC45504.2019.9015896>
- [60] Šatkauskas, N., Venčkauskas, A., Morkevičius, N., & Liutkevičius, A. (2020). Orchestration security challenges in the fog computing. In A. Venčkauskas, A. M. S. Elgohary, & M. M. R. M. N. (Eds.), *Information and Software Technologies: 26th International Conference, ICIST 2020, Kaunas, Lithuania, October 8–9, 2020, Proceedings* (pp. 196–207). http://dx.doi.org/10.1007/978-3-030-59506-7_17
- [61] Al-Doghman, F., Moustafa, N., Khalil, I., Sohrabi, N., Tari, Z., & Zomaya, A. Y. (2023). AI-enabled secure microservices in edge computing: Opportunities and challenges. *IEEE Transactions on Services Computing*, 16(2), 1485–1504. <https://doi.org/10.1109/TSC.2022.3155447>
- [62] Koren, I., Rinker, F., Meixner, K., Matevska, J., & Walter, J. (2023). Challenges and opportunities of DevOps in cyber-physical production systems engineering. In *2023 IEEE 6th International Conference on Industrial Cyber-Physical Systems (ICPS)* (pp. 1–6). IEEE. <https://doi.org/10.1109/ICPS58381.2023.10128073>
- [63] Kersten, M. (2018). A Cambrian Explosion of DevOps Tools. *IEEE Software*, 35(2), 14–17. <https://doi.org/10.1109/MS.2018.1661330>
- [64] Kim, G., Humble, J., Debois, P., & Willis, J. (2016). *The DevOps handbook: How to create world-class agility, reliability, and security in technology organizations*. IT Revolution.
- [65] Smeds, J., Nybom, K., & Porres, I. (2015). DevOps: A definition and perceived adoption impediments. In *International Conference on Agile Software Development* (pp. 166–177). https://doi.org/10.1007/978-3-319-18612-2_14
- [66] Chen, L. (2018). Continuous delivery at scale: Challenges and opportunities. In *2018 IEEE/ACM 4th International Workshop on Rapid Continuous Software Engineering (RCoSE)* (p. 42). IEEE. <https://doi.org/10.1145/3194760.3194764>
- [67] Senapathi, M., Buchan, J., & Osman, H. (2018). DevOps capabilities, practices, and challenges: Insights from a case study. In *Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018* (pp. 1–10). ACM. <https://doi.org/10.1145/3210459.3210465>
- [68] Riungu-Kalliosaari, L., Mäkinen, S., Lwakatare, L. E., Tiihonen, J., & Männistö, T. (2016). DevOps adoption benefits and challenges in practice: A case study. In *Product-Focused Software Process Improvement: 17th International Conference, PROFES 2016, Trondheim, Norway, November 22–24, 2016, Proceedings 17* (pp. 590–597). https://doi.org/10.1007/978-3-319-49094-6_44
- [69] Azad, N., & Hyrnsalmi, S. (2023). DevOps critical success factors — A systematic literature review. *Information and Software Technology*, 157, 107150. <https://doi.org/10.1016/j.infsof.2023.107150>
- [70] Trigo, A., Varajão, J., & Sousa, L. (2022). DevOps adoption: Insights from a large European Telco. *Cogent Engineering*, 9(1), 2083474. <https://doi.org/10.1080/23311916.2022.2083474>
- [71] Leite, L., Rocha, C., Kon, F., Milojić, D., & Meirelles, P. (2019). A survey of DevOps concepts and challenges. *ACM Computing Surveys (CSUR)*, 52(6), 1–35. <https://doi.org/10.1145/3359981>
- [72] Khan, M. S., Khan, A. W., Khan, F., Khan, M. A., & Whangbo, T. K. (2022). Critical challenges to adopt DevOps culture in software organizations: A systematic review. *IEEE Access*, 10, 14339–14349. <http://dx.doi.org/10.1109/ACCESS.2022.3145970>
- [73] Zarour, M., Alhammad, N., Alenezi, M., & Alsarayah, K. (2020). DevOps process model adoption in Saudi Arabia: An empirical study. *Jordanian Journal of Computers and Information Technology*, 6(3), 209–224. <http://dx.doi.org/10.5455/jjcit.71-1580581874>