A Taxonomy for Workload Deployment Orchestration in the Edge-Cloud Continuum

T. Albers, M. Fogli, E. Harmsma, E. Lazovik and H. Bastiaansen

Service-Oriented and Cloud Computing (ESOCC 2023), 2023

Cite this as

T. Albers, M. Fogli, E. Harmsma, E. Lazovik and H. Bastiaansen, "A Taxonomy for Workload Deployment Orchestration in the Edge-Cloud Continuum," in *Service-Oriented and Cloud Computing (ESOCC 2023)*, LNCS vol. 14183, Cham, Switzerland: Springer, 2023, pp. 239-250, doi: 10.1007/978-3-031-46235-1_16.

A Taxonomy for Workload Deployment Orchestration in the Edge-Cloud Continuum

Toon Albers[1][0000−0002−1621−7402], Mattia Fogli[2][0000−0002−9875−4550], Edwin Harmsma[1], Elena Lazovik[1][0000−0001−9691−6723], and Harrie Bastiaansen[1]

[1] TNO, Groningen, The Netherlands
{toon.albers,edwin.harmsma,elena.lazovik,harrie.bastiaansen}@tno.nl
[2] University of Ferrara, Ferrara, Italy
mattia.fogli@unife.it

Abstract. As compute resources continue to proliferate from static large-scale enterprise-grade cloud environments to various types of more dynamic and resource-constrained edge environments, the need increases to orchestrate the deployment of workloads of data and processing applications across the emerging edge-cloud continuum. For many use cases in various application domains and contexts, similar workload deployment orchestration challenges arise. We present a taxonomy for workload deployment orchestration in the edge-cloud continuum that captures the various views and perspectives on workload deployment orchestration. We evaluate and valorise the proposed taxonomy by means of three illustrative and representative types of use cases from different domains and present opportunities for future research.

Keywords: Cloud Orchestration · Deployment Orchestration · Edge Computing · Edge-Cloud Continuum · Federated Cloud · Taxonomy

1 Introduction

Current digitisation of societal sectors and infrastructures can be characterised by: (1) ubiquitous sensing, processing, storage, and communication capabilities and (2) the emergence of (distributed and federated) edge-cloud data storage and processing infrastructures. As such, orchestration of data sharing and processing workloads over dynamic federated cloud-edge infrastructures poses a recurring challenge in different operational contexts. Workload deployment orchestration (also referred to as 'App Deployment Orchestration') is the process that enables cloud and application providers to define how to select, deploy, monitor and configure (multi-container) packaged applications in the cloud at run-time. It encompasses the deployment, execution and maintenance phases [4, 7]. A workload deployment orchestrator is responsible for resource limit control, scheduling, load balancing, health checking, fault tolerance management and autoscaling.

Exemplary for a broader context in which the importance of cloud technology and workload deployment orchestration are currently gaining major interest are the developments around the European Data and Cloud strategy [8]. Driven by

the importance attributed to the emerging data economy with extra attention to data sovereignty and security, legal [9] and technical frameworks are being created for a converged data and cloud environment. Reference architectures (such as the IDSA[3], Gaia-X[4] and DSBA[5] initiatives) and the DSSC[6] are being developed, coordinated under the umbrella of the EU Directorate-General Connect (DG CONNECT). These initiatives show the importance of the emerging 'converged' data plan and cloud infrastructure.

A large diversity of perspectives and challenges on workload deployment orchestration exists for an edge-cloud continuum. Therefore, a taxonomy describing these perspectives can be of major benefit for (1) efficiency in identifying its challenges and complexities in a specific context and, as such, (2) supporting the effective design and development of a workload deployment orchestrator for that context, or (3) evaluating existing orchestrators for a specific use. In this paper we refer to the edge-cloud continuum, whilst increasingly individual 'devices' are taken into account, giving rise to a 'device-edge-cloud continuum'. Although well-recognised by the authors, within this paper it will be referred to as edge-cloud continuum without impacting the contents of the taxonomy.

This paper addresses the research topic to "*design a taxonomy for workload deployment orchestration in the edge-cloud continuum that describes the broad variety of challenges in emerging and converged data sharing and federated cloud infrastructures, and supports architects in exploring the design complexities and developing solutions for such orchestration*". This topic has been addressed by means of the *Experience Report* methodology [17]. Based on experience in various use cases and projects such as ENERSHARE[7] and ECiDA[8], we established the relevance for an overarching taxonomy describing the various aspects of workload deployment orchestration in different contexts. In combination with an exploration of related work on such a taxonomy, the need for an extended taxonomy applicable to the edge-cloud continuum was identified and elaborated.

This paper has the following structure: Section 2 introduces and describes the proposed taxonomy for workload deployment orchestration in the edge-cloud continuum. An overview of related taxonomies is provided in section 3. The proposed taxonomy is evaluated and valorised by means of three representative types of use cases in section 4, after which the future direction and the overarching conclusions are presented in section 5 and section 6, respectively.

2 Taxonomy

We have identified a set of six high-level concepts, shown in Figure 1, constituting the general problem space of workload deployment orchestration in the

[3] International Data Spaces Association (IDSA), https://internationaldataspaces.org
[4] EU Gaia-X Initiative, https://www.gaia-x.eu
[5] Data Space Business Alliance (DSBA), https://data-spaces-business-alliance.eu
[6] Data Spaces Support Centre (DSSC), https://dssc.eu
[7] See enershare.eu
[8] See commit2data.nl/en/projects/ecida-evolutionary-changes-in-distributed-analysis

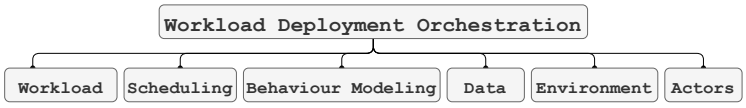

Fig. 1. Main concepts of the taxonomy

edge-cloud continuum: Workload, Scheduling, Behaviour Modeling, Data, Environment and Actors. When further broken down into sub-categories[9], these concepts allow specific problem instances to be identified and characterised.

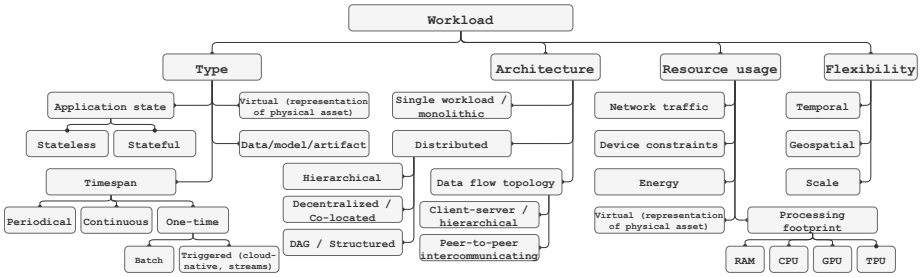

Fig. 2. Workload taxonomy

Workload (Figure 2) covers the asset that is being orchestrated. These are typically applications, but could also entail data gathering or movement, or virtual representations of physical assets. Specific challenges arise for applications depending on if they are stateful (storing data persistently to operate successfully) or stateless. We can also distinguish jobs in relation with their timespan: those running once until completion, periodic jobs and permanently running jobs require different strategical workload mechanisms to apply. There is also a difference in who triggers the job, and does it handle batch or streaming data. In the latter case, secondary control mechanisms ensure that effects decided by the orchestrator, are subsequently applied to the streams connected to real-world assets. Workload composition, e.g. monolithic versus distributed, also poses distinct challenges. Furthermore, we can identify different topologies in both dataflow and infrastructure, ranging from peer-to-peer data flows between multiple heterogeneous clusters, to hierarchical client-server communication. Workloads from applications often use resources - virtual or physical - and might have inherent characteristics that introduce flexibility in orchestration, for example when they can be postponed, scaled up, or moved to a different data centre.

Scheduling (Figure 3) covers the methods and means of the scheduling paradigm(s) employed by the orchestrator. Scheduling is done algorithmically

[9] A digital version of this taxonomy can be found at https://github.com/TNO/deployment-orchestration-taxonomy-supplimentary-materials

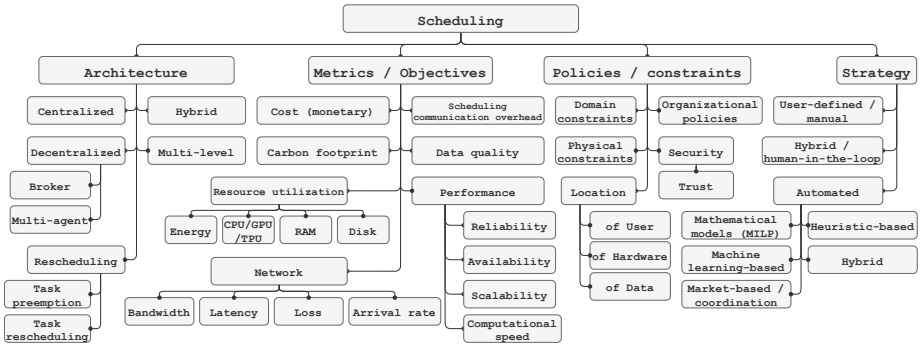

Fig. 3. Scheduling taxonomy

and is typically user-defined, though we also see hybrid methods where there is a human-in-the-loop to validate and approve, or give feedback to algorithmic approaches. Often, other problem characteristics dictate the specific algorithm. For example, in edge environments where bandwidth is limited and coordination is expensive, a locally deployed machine learning model working on incoming data may be more suitable. The scheduler operates with input metrics that are observed, a set of policies and constraints, and typically optimizes for one or more objectives. Depending on the actor model, as well as environmental constraints, the scheduling solution can be centralised (with a broker or similar middleware between nodes for more control) or decentralised (with nodes as independent actors with their own behaviour).

Behaviour Modeling (Figure 4) identifies the ways in which modeling approaches are used in deployment orchestration. Modeling can be applied to many types of challenges in deployment orchestration. Typical approaches include workload characterisation — where resource usage of workloads is monitored in order to construct a generalised model — and methods to achieve specific goals on consistency of running computations — for example, to use anomaly detection techniques as a method that aims to identify abnormal data or patterns, typically by comparing against previously established models. There are myriad methods to facilitate modeling, each with their strengths and weaknesses.

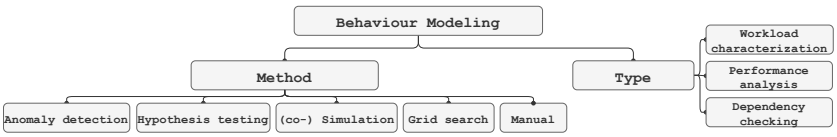

Fig. 4. Behaviour modeling taxonomy

Data (Figure 5) entails the different ways in which data is stored, accessed, used, monitored, and managed throughout deployment orchestration in the edge-

cloud continuum. We consider specific challenges in the security domain, for example when managing secrets, when handling privacy-sensitive data, and in facilitating access to restricted data. Additionally, data location can be altered for various purposes, such as for data gravity, caching, migrations or backups.

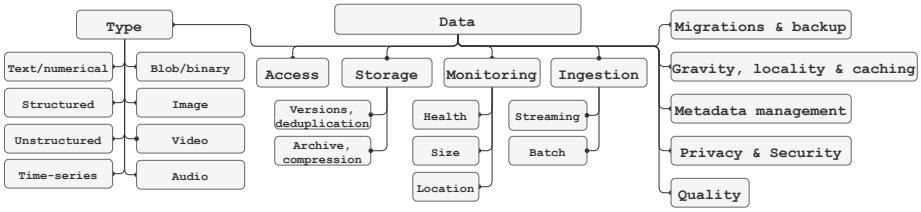

Fig. 5. Data taxonomy

Environment (Figure 6) characterizes the computational context and topology of the IT infrastructure hosting the workloads. This context can be static and unchanging, or dynamic. In the latter case, orchestration solutions should be aware of this changing environment, incurring additional monitoring overhead. Orchestration solutions might exist in, and manage, virtualised environments, or bare-metal infrastructure. Likewise, the compute and networking resources available to them may vary drastically, and even be unavailable for extended periods of time.

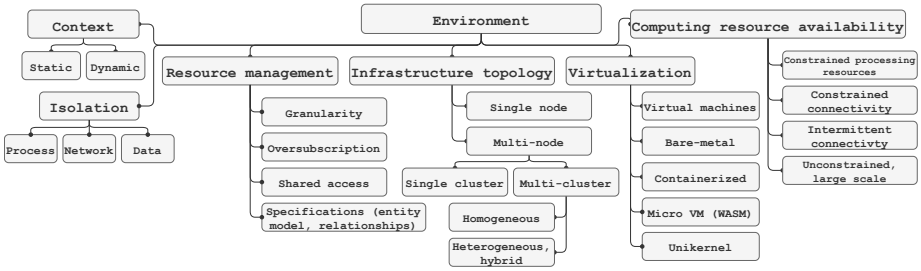

Fig. 6. Environment taxonomy

Actors (Figure 7) maps the ways in which stakeholders might interact in a given deployment orchestration problem. These actors could be end users, application providers, or even infrastructure providers. The way in which they interact, for example standalone, collaboratively, competitively, or even maliciously towards each other, introduces additional constraints to the deployment orchestration problem, and will be reflected in the scheduling solution's structure. Scheduling is easiest with a single stakeholder, and requires extra organisational and technical interoperability solutions if multiple stakeholders are involved.

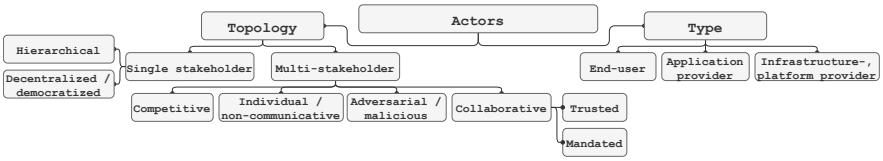

Fig. 7. Actors taxonomy

3 Related work

Other taxonomies often focus on specific deployment orchestration aspects. We have made a matching between these taxonomies and the taxonomy proposed in this paper to ensure full coverage of their concepts, aside from having added new concepts not present in related work.

Nguyen et al. [15] provide a taxonomy of IoT deployment orchestration, although they focus more on analyzing scientific literature compared to what is out in the field. Their taxonomy also discusses implementation details such as DSL, programming model and inter-workload communication mechanism, which we purposefully avoid.

Weerasiri et al. [18] created a taxonomy for cloud resource orchestration. They briefly cover workload deployment orchestration but do not go into details. They also cover expected types of users (DevOps, Application Developers, Domain experts), knowledge reuse (templates, resource snapshots, community support, etc.) and resource representation and access (CLIs, SDKs, etc) which we consider too implementation-specific for our taxonomy.

Bentaleb et al. [2] provide an overview of containerisation, but also include a taxonomy covering limited parts of orchestration and resource management. We cover the whole taxonomy of [2], often more elaborately because our taxonomy is not focused only on containerised workloads. This also results in some naming differences in the taxonomy. For example, they use application instead of workload, but workload is more generic and for orchestration we care about specific workloads and not the applications to which they belong. They do not elaborate the scheduling part of their taxonomy, instead delegating this task to tools such as Kubernetes, but for deployment orchestration we consider this an important aspect. Unlike [2], our taxonomy does not explicitly mention lifecycle management as we consider the whole orchestration to be based around it.

Mampage et al. [14] provide a taxonomy focusing on *serverless* workloads. However, many of the aspects of serverless workloads, such as workload management and QoS goals, are also applicable to (containerised) workloads in general. They also include 'billing model' in their taxonomy, which we consider to be part of the cloud provider perspective and therefore do not include.

Rodriguez and Buyya [16] provide an elaborate cluster orchestration taxonomy. Comparing these taxonomies, the workloads are defined in a similar manner but Rodriguez and Buyya expand upon *resource usage* in our taxonomy with *resource requests*, *limits* and *consumption estimation*. They also expand upon

network and performance isolation, but we consider these an implementation detail of the provider platform. The taxonomy presented in this paper more elaborately describes workload type, flexibility and scheduling strategy. The topics of deployment modeling, data and actors are also not covered by them.

Finally Carrión [3] describes a taxonomy specifically catered to the Kubernetes orchestrator, and referencing scientific literature aiming to improve this orchestrator. They separate performance from scheduling and application, however we consider performance metrics to be part of the scheduling process. They also separate infrastructure and cluster, but for us the cluster is part of the environment in which a workload runs. The taxonomy is also limited, for example only few scheduling metrics are mentioned.

None of these taxonomies completely describe all aspects related to orchestration in converged data sharing and federated clouds and edges. The majority concentrate on workload concepts, some of them also elaborate the scheduling more deeply. There is no taxonomy that includes data or (multi) actor aspects, or makes the bridge to the modeling of the application and infrastructure behaviour. As a result, no complete taxonomy exists which covers all relevant aspects of the orchestration, and that is why we propose our taxonomy. Some taxonomies cover the same aspects using different terms, depending on their research question and perspective. The taxonomy presented in this paper is created from the perspective of running in the cloud-edge continuum, therefore it uses terms from that domain. Some aspects of our taxonomy are also elaborated further compared to other taxonomies for the same reason.

4 Use cases

The applicability of the proposed taxonomy for workload deployment orchestration on federated and adaptive cloud infrastructures is demonstrated and assessed in this chapter by means of three types of use cases. As their scope varies strongly, they are considered to be illustrative and representative for a multitude of additional types of use cases for which the taxonomy can be used.

4.1 Local access to and processing of sensitive data

AI algorithms must be able to access data sources to be trained or deployed. However, the data cannot always simply be shared with an external organisation that provides the algorithm, e.g. when the amount of data to be transferred is too large or too sensitive to share (due to confidentiality, ethical, privacy or legal reasons). In such cases, the data must remain within the security domain of its provider and only controlled access to the data may be allowed for a (distributed) AI algorithm. This is also referred to as "*data visiting*". Data visiting can be implemented by means of Privacy Enhancing Technologies (PETs), such as secure Multi-Party Computation (MPC) [6] and Federated Learning (FL) [5]. PETs can be applied to train or deploy algorithms on data that is either horizontally or vertically partitioned. In the former, the same type of data is provided

by different organisations. In the latter, different types of data applying to the same "entity" are provided by different organisations.

For training over horizontally partitioned data, the use case of medical data distributed over various hospitals can be used. Data on a particular illness, but applying to different patients, is often spread over many hospitals. The dataset at a particular hospital could be too small to successfully train a model. Federated Learning allows a 'worker' to compute a version of the FL model on local data within the (security) domain of the hospital. Subsequently, the individually trained models are combined to compute an aggregate model. Repeating this process several times, the FL model is trained over the combined local datasets, without raw data having been shared.

For training over vertically partitioned data, we can use the use case of the emerging Smart Energy Systems. Due to an increase in renewable (distributed) energy sources, energy grids increasingly encounter congestion. Insight into the state of the energy grid is needed to improve the planning for congestion and mitigate the effects thereof. Data from diverse devices, owned by households and facilitated by third parties, can be used to get an insight into the state of the low-voltage (LV) grid. However, this is sensitive data as it reveals consumer behaviour patterns. Hence, data visiting could be used for LV grid state estimation. In this case, models should be provided access to local household data. The model weights are aggregated in the semi-global model of the FL framework to calculate the overarching grid state, allowing the physical LV system to be monitored and (re-)configured [5].

The taxonomy as proposed in this paper provides a good basis for identifying and defining the main deployment orchestration aspects for both the horizontally and vertically partitioned data. The usage of the taxonomy helped in this use case to identify the choice on scheduling mechanism, the KPIs for data management and requirements for the computational environment.

4.2 Varying trust and security levels of clouds

Data sharing and processing collaborations between organisations involving sensitive data require policies and agreements on the actual processing in cloud infrastructures that are owned and operated by external organisations. In this case, deployment of workloads on specific clouds will depend on whether their trust and security offerings comply with constraints as specified by the organisation that deploys the workload, and the applicable legislation.

Hence, the trust and security constraints introduce restrictions on the deployment on cloud and edge environments. Traditionally, legal agreements are made between the deploying organisation and the provider(s) of the cloud services in combination with an auditing approach on conformance with the agreements, rules, norms and standards. However, the more often that data collaboration occurs in which data(sets) are transferred between various cloud service providers, the more complex this (often manual) approach of contracting and auditing becomes. New efforts on automated compliance as for example in Gaia-X aim to solve the labor and overhead involved in this process [10]. Gaia-X introduces the

concept of labelling the trust and security levels of clouds, adhering to the European values of data sovereignty. As such, cloud service providers are expected to emerge with differentiation in trust and security offerings. In such a diverse cloud landscape, the deployment orchestrator has to ensure that the workload distribution over various clouds stays within the trust and security constraints as prescribed by business rules and applicable legislation. Additionally, the strategic multi-cloud perspective takes varying trust and security levels into account in the workload deployment orchestration. Users contract multiple cloud service providers with the ability to move workloads between them, to become less dependent on a single cloud provider.

This use case relates to the Scheduling concept of the taxonomy, in which it specifically addresses the security and trust aspects under policies. From a trust and security perspective other concepts in the taxonomy can apply as well. For instance, data classification may require that all data processing is performed only by mandated parties. This relates to the Actors concept in the taxonomy. Similarly, requirements on the availability of (processing) services applies to the Environment concept in the taxonomy.

4.3 Non-reliable availability of cloud and edge infrastructures

The (availability of) individual cloud processing infrastructures can not always be relied upon. A first use case applies to the emerging 5G network infrastructures with integrated edge and cloud computing service offerings. Applications that rely on (live) access to processing capabilities at the edge of a telecommunication network or in an interconnected remote cloud, face differences in availability of the network. In this case mobile edge computing (MEC) and cloud locations might be available, but from the device in the field that requires the processing capabilities it is (temporarily) not accessible. A workload might need to be stopped, moved or placed into a different operating mode to ensure the least impact of the signal-loss to the end-user. This type of self-adaptivity or resilience of the processing infrastructure influences the deployment orchestration as it introduces variability within the orchestrated environment. In previous research the conclusion was drawn that slicing within the 5G network cannot ensure end-to-end low latency and availability guarantees alone [12], and edge and redundancy planning of the deployment orchestrator is an important open challenge to be addressed. Moreover, in the particular case of the mobile edge, specific viewpoints are needed on offloading strategies.

A second use case in this category applies to military mission contexts, in which federations of tactical clouds aim at filling the gap between recent advancements in sensing, processing, and storage capabilities for acquiring and processing data and the still disadvantaged tactical networks interconnecting such capabilities on the battlefield [1]. Workload deployment orchestration in such an environment will fundamentally differ from those operating in civilian enterprise clouds: they are not provided with abundant cloud processing resources, they may be on the move (in aircraft, drones or battlefield vehicles), may not have guaranteed power supplies, may have very limited and unreliable connectivity

(in terms of bandwidth, latency and latency variation) and may not have trustworthy availability (due to military adversarial activity and resource mobility). Moreover, military mission partners must retain both autonomy and granular control over the policies that define what others can see and do on 'their' cloud environments. Various NATO Research Task Groups (RTGs) have explored the concept of a federation of tactical clouds in which mission partners mutually expose their cloud resources through well-defined Application Programming Interfaces (APIs) [1] and have quantified the performance of various common-of-the-shelve and open-source container orchestration distributions in military tactical networks [13]. This requires workload deployment orchestration across the edge-cloud continuum, for which the RTGs identified the following major challenges: (1) Some edge nodes provide specialised resources that might go offline quite frequently, although their workloads may be mission-critical. Typical examples are camera drones or free-floating ocean sensors; (2) Workloads with high priority may preempt resources previously allocated to other workloads, for which a one-shot ("fire-and-forget") deployment orchestration process is not acceptable; (3) Mission partners do not typically equally trust each other, and trust might change throughout the military operation. Hence, the deployment orchestrator must consider trust while scheduling workloads.

In these use cases multiple concepts from the taxonomy apply. Clearly, the (non-) reliability of cloud infrastructures relates to the Environment concept, and specifically to its availability branch. Moreover, the Actors concept applies from the perspective of the multi stakeholder environments in the context of these use cases. From Scheduling, the trust (and security) aspects are of major relevance.

5 Future directions

Based on the taxonomy as defined in Section 2 and the results of the deployment thereof in the use cases as presented in Section 4, we have identified two connected directions for future work. First of all, the execution of the use cases has triggered several areas in which the taxonomy itself should be extended:

1. Strategic decentralisation, e.g. for autonomous swarming concepts to ensure overall resilience [11].
2. Availability of secure enclaves in multi-cluster environments enabling confidential computing scenarios.
3. Dynamic utilisation of available programmable (or configurable) hardware elements, e.g., FPGA boards.

The three aspects above were identified either during the study or in the evaluation process of the use cases. Though, in the analysis the three aspects were not sufficiently mature or complete to be included in the taxonomy presented.

As a second angle of future work, this analysis demands a new design of an open and modular software architecture that enables implementations of a generic, re-usable, modular deployment orchestrator, where the various concerns

of the presented taxonomy can be implemented without creating a specific orchestrator for each individual use case as was done in this study. In such an architecture, openness is key to ensure that third parties can easily extend the orchestrator to address aspects of the taxonomy that are important to them.

6 Conclusions

The primary objective of this paper was to provide a taxonomy for deployment orchestration in a converged data sharing and federated cloud infrastructure, and assess its usability against a set of illustrative and representative use cases. A preliminary conclusion is that the proposed taxonomy for each of the use cases proved its added value in terms of (1) efficiency in defining the context and problem space in which an associated deployment orchestrator has to be developed, and (2) identifying the boundary condition and scope limitations in developing a generic deployment orchestrator being re-usable for multiple / various use cases. Future work includes further validation of this taxonomy from additional use cases and from development of a generic, modular and re-usable implementation based on the taxonomy.

Acknowledgments

This paper is the result of the joint effort of several research and innovation projects: the Dutch Centre-of-Excellence on Data Sharing and Cloud (CoE DSC, https://coe-dsc.nl/), the EU project Big Data for Energy (BD4NRG, https://www.bd4nrg.eu), and the NATO Research Task Group IST-193 'Edge Computing at the Tactical Edge'. We would like to thank the sponsors of these initiatives for providing us the opportunity to do this highly relevant and interesting research. Moreover, the authors would like to thank Mr. G. Pingen and Mrs. A. Kosek (both former TNO employees) for their valuable preliminary work forming the basis for this paper.

References

[1] Bastiaansen, H., Geest, J.v.d., Broek, C.v.d., Kudla, T., Isenor, A., Webb, S., Suri, N., Fogli, M., Canessa, B., Masini, A., Goniacz, R., Sliwa, J.: Federated control of distributed multi-partner cloud resources for adaptive C2 in disadvantaged networks (2020), https://doi.org/10.1109/MCOM.001.2000246

[2] Bentaleb, O., Belloum, A.S.Z., Sebaa, A., El-Maouhab, A.: Containerization technologies: Taxonomies, applications and challenges. J. Supercomput. **78**(1), 1144–1181 (2022), https://doi.org/10.1007/s11227-021-03914-1

[3] Carrión, C.: Kubernetes scheduling: Taxonomy, ongoing issues and challenges. ACM Computing Surveys **55**(7), 1–37 (2022), https://doi.org/10.1145/3539606

[4] Casalicchio, E., Iannucci, S.: The state-of-the-art in container technologies: Application, orchestration and security. Concurrency and Computation: Practice and Experience **32** (2020), https://doi.org/10.1002/cpe.5668
[5] Causevic, S., Sharma, S., ben Aziza, S., van der Veen, A., Lazovik, E.: LV grid state estimation using local flexible assets: A federated learning approach. In: 2023 International Conference and Exposition on Energy Distribution (CIRED), pp. 1–4 (2023)
[6] D'Acquisto, G., Domingo-Ferrer, J., Kikiras, P., Torra, V., de Montjoye, Y., Bourka, A.: Privacy by design in big data: An overview of privacy enhancing technologies in the era of big data analytics. CoRR (2015), URL http://arxiv.org/abs/1512.06000
[7] E. Casalicchio: Container Orchestration: A Survey, pp. 221–2035. Springer, Cham (2019), https://doi.org/10.1007/978-3-319-92378-9_14
[8] European Commission: A european strategy for data (2020), URL https://digital-strategy.ec.europa.eu/en/policies/strategy-data
[9] European Commission: European Data Governance Act (2022), URL https://digital-strategy.ec.europa.eu/en/policies/data-governance-act
[10] Gaia-X: Automated compliance - gaia-x institute position paper (2022), URL https://gaia-x.eu/publication/automated-compliance/
[11] Juan Ferrer, A.: Next Steps for Ad-hoc Edge Cloud and Swarm Computing Realization, pp. 189–195. Springer, Cham (2023), https://doi.org/10.1007/978-3-031-23344-9_12
[12] Ksentini, A., Frangoudis, P.A.: Toward slicing-enabled multi-access edge computing in 5g. IEEE Network **34**(2), 99–105 (2020), https://doi.org/10.1109/MNET.001.1900261
[13] Kudla, T., Fogli, M., Webb, S., Pingen, G., Suri, N., Bastiaansen, H.: Quantifying the performance of cloud-oriented container orchestrators on emulated tactical networks (2022), https://doi.org/10.1109/MCOM.003.00975
[14] Mampage, A., Karunasekera, S., Buyya, R.: A holistic view on resource management in serverless computing environments: Taxonomy and future directions. ACM Comput. Surv. **54**(11s) (2022), https://doi.org/10.1145/3510412
[15] Nguyen, P.H., Ferry, N., Erdogan, G., Song, H., Lavirotte, S., Tigli, J.Y., Solberg, A.: A systematic mapping study of deployment and orchestration approaches for IoT. IoTBDS pp. 69–82 (2019), https://doi.org/10.5220/0007675700690082
[16] Rodriguez, M.A., Buyya, R.: Container-based cluster orchestration systems: A taxonomy and future directions. CoRR (2018), URL http://arxiv.org/abs/1807.06193
[17] Tonella, Torchiano, D.B., Systa: Empirical studies in reverse engineering: state of the art and future trends. Empir Software Eng **12**, 551–571 (2007), https://doi.org/10.1007/s10664-007-9037-5
[18] Weerasiri, D., Barukh, M.C., Benatallah, B., Sheng, Q.Z., Ranjan, R.: A taxonomy and survey of cloud resource orchestration techniques. ACM Comput. Surv. **50**(2) (2017), https://doi.org/10.1145/3054177