Data Science Workflows for the Cloud/Edge Computing Continuum

Valerio Grossi
ISTI-CNR
Pisa, Italy
valerio.grossi@isti.cnr.it

Roberto Trasarti
ISTI-CNR
Pisa, Italy
roberto.trasarti@isti.cnr.it

Patrizio Dazzi
ISTI-CNR
Pisa, Italy
patrizio.dazzi@isti.cnr.it

ABSTRACT

Research infrastructures play a crucial role in the development of data science. In fact, the conjunction of data, infrastructures and analytical methods enable multidisciplinary scientists and innovators to extract knowledge and to make the knowledge and experiments reusable by the scientific community, innovators providing an impact on science and society. Resources such as data and methods, help domain and data scientists to transform research in an innovation question into a responsible data-driven analytical process. On the other hand, Edge computing is a new computing paradigm that is spreading and developing at an incredible pace. Edge computing is based on the assumption that for certain applications is beneficial to bring the computation as closer as possible to data or end-users. This paper introduces an approach for writing data science workflows targeting research infrastructures that encompass resources located at the edge of the network.

CCS CONCEPTS

• **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

KEYWORDS

Workflow languages, edge computing, research infrastructure

ACM Reference Format:
Valerio Grossi, Roberto Trasarti, and Patrizio Dazzi. 2021. Data Science Workflows for the Cloud/Edge Computing Continuum. In *Proceedings of the 1st Workshop on Flexible Resource and Application Management on the Edge (FRAME '21), June 25, 2021, Virtual Event, Sweden.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3452369.3463820

1 INTRODUCTION

The conjunction of data, infrastructures, and analytical methods enable multidisciplinary scientists and innovators to extract knowledge and to make the knowledge and experiments reusable by the scientific community, innovators providing an impact on science and society. Data science represents an opportunity for improving our society and boosting social progress. It can support policy-making, it offers novel ways to produce high-quality and high-precision statistical information, can help to promote ethical uses of big data. At this point, it is clear why Research infrastructures (RIs) play a crucial role in the advent and development of data science. Resources such as data and methods help domain and data scientists to transform research or an innovation question into a responsible data-driven analytical process. This process is executed onto the platform, supporting experiments that yield scientific output, policy recommendations, or innovative proofs-of-concept. An infrastructure offers means to define complex analytical processes and *workflows*, thus bridging the gap between experts and analytical technology. As a collateral effect, experiments generate new relevant data, methods, and workflows that can be integrated into the platform by scientists, contributing to the expansion of the RI. As a drawback, the availability of data creates opportunities but also new risks. The use of data science techniques could expose sensitive traits of individual persons and invade their privacy.

SoBigData RI is a platform for the design and execution of large-scale social mining experiments, open to users with diverse backgrounds, accessible on cloud, and also exploiting super-computing facilities. All the SoBigData components are introduced for implementing data science: from raw data management to knowledge extraction, with particular attention to legal and ethical aspects. SoBigData serves a cross-disciplinary community of scientists studying all the elements of societal complexity from a data- and model-driven perspective. Moreover, pushing the FAIR (Findable, Accessible, Interoperable, Reusable) and FACT (Fair, Accountable, Confidential and Transparent) principles, furthermore SoBigData++ RI renders social mining experiments more easily designed, adjusted and repeatable by experts that are not data scientists. SoBigData++ RI moves forward from the simple awareness of ethical and legal challenges in social mining to the development of concrete tools that operationalize ethics with value-sensitive design, incorporating values and norms for privacy protection, fairness, transparency and pluralism.

From the perspective of the actual deployment on physical resources, a relevant challenge for these tools consists in providing a way to exploit those resources that can not be directly involved in the administrative domains of the research infrastructure; even if their exploitation could be beneficial for supporting the execution of the application. A notable example of such kind of resources are the Edge devices, i.e., those devices that can be exploited by using a pay-per-use approach, typical of utility computing paradigm [2, 41], but are not part of a large centralized installation (like a Cloud) and are instead distributed on a large, dispersed, area. The aim of this approach is in fact to provide a pervasive computing infrastructure with the objective of bringing the computation as much close as possible to the data producers (e.g., sensors, cameras, etc.)

FRAME '21, June 25, 2021, Virtual Event, Sweden.

ACM ISBN 978-1-4503-8384-4/21/06…$15.00
https://doi.org/10.1145/3452369.3463820

and/or data consumers (e.g., users, applications, etc.). The resulting research infrastructure extended to the edge, will encompass resources of different kinds; this complex set of heterogeneous resources – having different capacities and means to access – has the potentiality to enable quite more interesting scenario at a cost of an increased complexity in its actual management. This complexity is not limited to the actual setup and operation of the platform but also impacts on the approach to adopt for developing applications that should run on top of this heterogeneous and distributed resource infrastructure.

The aim of this paper is to briefly highlight how workflow-based solutions can be properly instrumented an attempt to address the aforementioned challenges. The remaining of this paper is structured as follows. Section 2 contextualize this work by placing it in the scientific literature by presenting a few related works. Section 3 introduces the workflows as a solution for developing solutions targeting traditional research infrastructures. Section 4 presents a receipt for emending existing workflows approaches to match the peculiar needs of the aforementioned extension of the infrastructure. Finally, Section 5 draws our conclusive remarks and introduces some works that we plan to undertake in the near future.

2 RELATED WORK

Liew et al. [27] have analyzed selected Workflow Management Systems (WMSs) that are widely used by the scientific community, namely: Airavata [31], Kepler [8], KNIME [6], Meandre [28], Pegasus [18], Taverna [40], and Swift [42]. Such systems have been reviewed according to the following aspects: *(i)* processing elements, i.e., the building blocks of workflows envisaged to be either web services or executable programs; *(ii)* coordination method, i.e., the mechanism controlling the execution of the workflow elements envisaged to be either orchestration or choreography; *(iii)* workflow representation, i.e., the specification of a workflow that can meet two goals human representation and/or computer communication; *(iv)*data processing model, i.e., the mechanism through which the processing elements process the data that can be bulk data or stream data; *(v)* optimization stage, i.e., when optimization of the workflow (if any) is expected to take place that can either be build time or run-time (e.g., data workflow processing optimization [5, 7, 25, 29, 30]. The aforementioned approaches are defined based on the assumption that workflows are composed of machine-executable actions, i.e., performed by agents that can be programmatically invoked. They do not address the needs, motivated by several scientific contexts, e.g., Big Data and Social Mining [22], Biodiversity and Cheminformatics domains [20, 35], of defining workflows that include "manual actions", e.g., data manipulation and adaptation using editors or shell commands. Attempts in this direction exist but embrace a fully manually-oriented approach, e.g., protocols.io [37], enabling the digital representation, publishing, and sharing of digital fully manual workflows.

The main contribution of this paper is an extension of workflow language and execution platform, whose intuition was earlier presented in [11]. HyWare was designed to enable the description of "hybrid" workflows, obtained as sequences of machine-executable and manually-executable actions. As such, the language can serve the mission of Open Science by addressing the reproducibility of digital science beyond traditional approaches in contexts where workflow actions are not entirely performed by machines. In recent years there have been several efforts in studying the most appropriate solution for structuring applications for Cloud/Edge environment. TOSCA is one of the most successful standards. As Binz [9] states, the goals of TOSCA include the automation of application deployment and the representation of the application in a cloud agnostic way. This standard has been leveraged by several products and research initiatives, e.g., BASMATI [1, 38], and Tosker [10]. Tosker works with an extended TOSCA YAML and generates a deployment plan for Docker. TOSCA has also been used with Kubernetes [24] to define application components along with their deployment and run-time adaptation on Kubernetes clusters across different countries. All these solutions are general purpose and not focus on a specific class of the application; that is instead the approach that we follow in this paper.

3 NEED FOR AD-HOC INSTRUMENTS

In the era of Open Science, where aspects such as reproducibility and transparency of science and FAIRness of research data (Findable, Accessible, Interoperable, Reusable research data) are becoming central to the whole research life-cycle, workflow languages play a special role in the diffusion of data science. Workflows are tools for the representation of the scientific process and the steps the researchers had to perform to execute an experiment using the e-infrastructure tools. Workflows can inherently contribute to the implementation of two data principles which are at the base of the modern data processing and analysis: *(i)* the FAIR data principles by accurately collecting, processing, and managing data and metadata on behalf of the researchers, while tracking provenance according to standards [23]; *(ii)* the FACT data principles stating that the data processing should be fair, accurate, confidential and transparent. A workflow language [26]. Moreover, workflows are digital objects in their own right, they can be published, discovered, shared, and cited for reproducibility and for scientific attribution of science like research articles, research data, and research software. Known approaches include: *workflows as digital artefacts*: workflow files are published in a repository with bibliographic metadata (e.g. Zenodo[1], [33]) and can be possibly related to their inputs and outputs [21, 36]; *workflow-as-a-Service*: workflows are shared via a platform gateway that enables discovery and execution [4, 14, 16, 19].

Today, SoBigData scientists can integrate their tools for VRE-integrated reuse but cannot represent a sequence of actions in order to share it and reproduce it. We are working on equipping SoBigData VREs with a workflow allowing scientists to attach to a specific result the entire process used to obtain it. This makes the environment evolve into a living laboratory, which contains not only the methods and the results but also the experience of the researcher using the methods, and composing an analytical process with it. On the other side, our workflow has to manage the computational component needed to execute an action both not only considering federated ones [13]. In this paper, we consider only machine action, i.e., actions executable by a computational node and characterized by a description, expressed by the respective properties, but also by a standard way to invoke a third-party service and get back the

[1] https://zenodo.org/

results. To this aim, for each machine action class will integrate a mediator capable of invoking the external service with given action input parameters and collect the parameters to return them in accordance with the action output type. Each machine action of our workflow language (Fig. 1) includes three main aspects: *(i)* Configuration parameters: this information allows the system to instantiate a generic action class of the method to a specific instance ready to be executed. *(ii)* Execution annotation: represents a form of syntactic metadata that are directives to the workflow execution environment and the Edge computing orchestrator (i.e. memory usage, multi-core-executable, execution placement, latency constraints); for example, these annotations can be used to reduce the latency of an execution of an action under a certain value, or constraints on data transmission. [15, 17]. *(i)* Returning result: packaging the results in order for them to be available to the subsequent action instance execution.

Driving Profile Example: computing driving profiles and monitoring driving behaviors of users [32] will be done at different levels of the cloud/edge continuum: the single action class will be instantiated as: *(i)* an instance of model computation at the device level to compute the user profile using the personal data produced during its daily activities and an assessment module to check if the model holds; *(ii)* a global modeling at the cloud level which combines local models and updates it if something changes in the nodes.

4 A RECEIPT FOR WORKFLOWS TARGETING EDGE COMPUTING

Workflows represent an effective approach to structure applications describing scientific processes that enable the development of many data science solutions. As such, it is a very good candidate to focus on, as its empowerment will edge-enable a large set of data-centric applications. To achieve this goal we envision the extension of our previously proposed workflow engine [11], with an edge computing orchestrator able to properly manage the execution of workflow actions on top of edge resources.

Such an extension needs to revolve around the following aspects: *(i)* interoperability: the first step to undertake in order to allow the exploitation of Edge resources is to enable the actual deployment of workflow actions at the Edge. In spite of the many solutions proposed in the literature, an off-the-shelf solution targeting extended research infrastructures is not existing. We plan to work on an extension of TOSCA standard tailoring it to our specific needs, also in consideration of the high degree of interoperability that TOSCA guarantees, as we highlighted in Sec. 2; *(ii)* resource Indexing and Discovery and representation: the workflow engine has to be provided with the ability of enlisting and indexing the resources available, on which to map the workflow actions; To this end we envision the exploitation of solutions borrowed from the peer-to-peer field that we investigated in the past [3, 12] demonstrated to be quite effective solutions to this end; *(iii)* application Monitoring and Orchestration: a fundamental element for achieving an efficient exploitation of edge resources is an efficient monitoring subsystem that feeds an orchestration subsystem aimed at conducting a match-making process to provide applications with the best resources possible, among the one available; In the literature have been presented several solutions to this end, both from the domain of Clouds (e.g., Wen [39]) and Networks (e.g., Sahu [34]). We plan to build upon an existing orchestration solution for edge to tailor with the specific needs of the workflows.

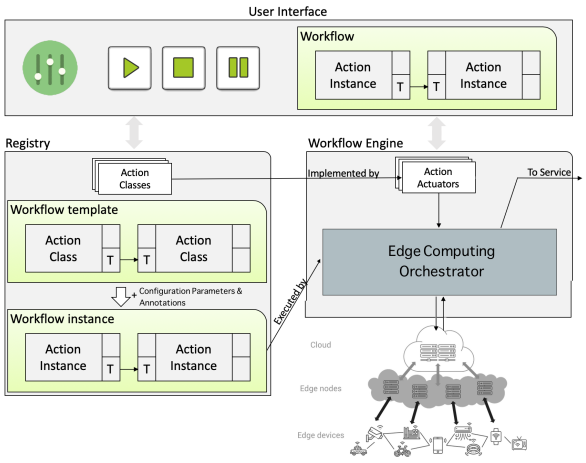

Figure 1: Workflow engine architecture. The classes of actions can be instantiated according to the underlying e-infrastructure and then combined into workflows.

5 CONCLUSION

This paper discusses a potential solution for enabling the extension of research infrastructure to the edge, without the need for actually federating edge resources into the infrastructure. The proposed approach is based on the identification of a few key features represented by a set of annotations that need to be integrated into a workflow engine. This work represents the first step toward a definition of a workflow language enabling the use of extended computational resources represented by Edge computing. The advantage of this approach resides in both enhance the computational power of RI and enabling the execution of actions where data are generated.

ACKNOWLEDGMENTS

This work is supported by the European Community's H2020 Program under the scheme "INFRAIA-01-2018-2019", Grant Agreement n.871042, "SoBigData++: European Integrated Infrastructure for Social Mining and Big Data Analytics" (http://www.sobigdata.eu).

REFERENCES

[1] Jörn Altmann, Baseem Al-Athwari, Emanuele Carlini, Massimo Coppola, Patrizio Dazzi, Ana Juan Ferrer, Netsanet Haile, Young-Woo Jung, Jamie Marshall, Enric Pages, Evangelos Psomakelis, Ganis Zulfa Santoso, Konstantinos Tserpes, and John Violos. 2017. BASMATI: An Architecture for Managing Cloud and Edge Resources for Mobile Users. In *Economics of Grids, Clouds, Systems, and Services*, Congduc Pham, Jörn Altmann, and José Ángel Bañares (Eds.). Springer International Publishing, Cham, 56–66.

[2] Gaetano F Anastasi, Emanuele Carlini, and Patrizio Dazzi. 2013. Smart cloud federation simulations with cloudsim. In *Proceedings of the first ACM workshop on Optimization techniques for resources management in clouds*. 9–16.

[3] Ranieri Baraglia, Patrizio Dazzi, Barbara Guidi, and Laura Ricci. 2012. GoDel: Delaunay Overlays in P2P Networks via Gossip. In *IEEE 12th International Conference on Peer-to-Peer Computing (P2P)*. IEEE, 1–12.

[4] Ranieri Baraglia, Patrizio Dazzi, Matteo Mordacchini, Laura Ricci, and Luca Alessi. 2011. Group: A gossip based building community protocol. In *Smart spaces and next generation wired/wireless networking*. Springer, Berlin, Heidelberg, 496–507.

[5] Marcello M. Bersani, Salvatore Distefano, Luca Ferrucci, and Manuel Mazzara. 2015. A Timed Semantics of Workflows. In *Software Technologies*, Andreas Holzinger, Jorge Cardoso, José Cordeiro, Therese Libourel, Leszek A. Maciaszek, and Marten van Sinderen (Eds.). Springer International Publishing, Cham, 365–383.

[6] Michael R. Berthold, Nicolas Cebron, Fabian Dill, Thomas R. Gabriel, Tobias Kötter, Thorsten Meinl, Peter Ohl, Kilian Thiel, and Bernd Wiswedel. 2009. KNIME - the Konstanz Information Miner: Version 2.0 and Beyond. *SIGKDD Explor. Newsl.* 11, 1 (Nov. 2009), 26–31. https://doi.org/10.1145/1656274.1656280

[7] Massimiliano Bertolucci, Emanuele Carlini, Patrizio Dazzi, Alessandro Lulli, and Laura Ricci. 2015. Static and dynamic big data partitioning on apache spark.. In *PARCO*. 489–498.

[8] Ludäscher Bertram, Altintas Ilkay, Berkley Chad, Higgins Dan, Jaeger Efrat, Jones Matthew, Lee Edward A., Tao Jing, and Zhao Yang. [n.d.]. Scientific workflow management and the Kepler system. *Concurrency and Computation: Practice and Experience* 18, 10 ([n. d.]), 1039–1065. https://doi.org/10.1002/cpe.994

[9] Tobias Binz, Uwe Breitenbücher, Oliver Kopp, and Frank Leymann. 2014. TOSCA: portable automated deployment and management of cloud applications. In *Advanced Web Services*. Springer, 527–549.

[10] Antonio Brogi, Luca Rinaldi, and Jacopo Soldani. 2018. TosKer: a synergy between TOSCA and Docker for orchestrating multicomponent applications. *Software: Practice and Experience* 48, 11 (2018), 2061–2079.

[11] Leonardo Candela, Valerio Grossi, Paolo Manghi, and Roberto Trasarti. 2021. A workflow language for research e-infrastructures. *International Journal of Data Science and Analytics* (2021). https://doi.org/10.1007/s41060-020-00237-x

[12] Emanuele Carlini, Massimo Coppola, Patrizio Dazzi, Domenico Laforenza, Susanna Martinelli, and Laura Ricci. 2009. Service and resource discovery supports over p2p overlays. In *2009 International Conference on Ultra Modern Telecommunications & Workshops*. IEEE, 1–8.

[13] Massimo Coppola, Patrizio Dazzi, Aliaksandr Lazouski, Fabio Martinelli, Paolo Mori, Jens Jensen, Ian Johnson, and Philip Kershaw. 2012. The Contrail approach to cloud federations. In *Proceedings of the International Symposium on Grids and Clouds (ISGC'12)*, Vol. 2. 1.

[14] Gianpaolo Coro, Giancarlo Panichi, Paolo Scarponi, and Pasquale Pagano. 2017. Cloud computing in a distributed e-infrastructure using the web processing service standard. *Concurrency and Computation: Practice and Experience* 29, 18 (2017), e4219. https://doi.org/10.1002/cpe.4219 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/cpe.4219 e4219 cpe.4219.

[15] Marco Danelutto and Patrizio Dazzi. 2008. Workflows on top of a macro data flow interpreter exploiting aspects. In *Making Grids Work*. Springer, Boston, MA, 213–224.

[16] Marco Danelutto, Patrizio Dazzi, et al. 2005. A Java/Jini Framework Supporting Stream Parallel Computations.. In *PARCO*. 681–688.

[17] Marco Danelutto, P Dazzi, D Laforenza, M Pasin, L Presti, and M Vanneschi. 2006. PAL: High level parallel programming with Java annotations. In *Proceedings of CoreGRID Integration Workshop (CIW 2006) Krakow, Poland, Academic Computer Centre CYFRONET AGH*. 189–200.

[18] Ewa Deelman, Karan Vahi, Gideon Juve, Mats Rynge, Scott Callaghan, Philip J. Maechling, Rajiv Mayani, Weiwei Chen, Rafael Ferreira da Silva, Miron Livny, and Kent Wenger. 2015. Pegasus, a workflow management system for science automation. *Future Generation Computer Systems* 46 (2015), 17 – 35. https://doi.org/10.1016/j.future.2014.10.008

[19] R. Filgueira, M. Atkinson, A. Bell, I. Main, S. Boon, C. Kilburn, and P. Meredith. 2014. EScience gateway stimulating collaboration in rock physics and volcanology. *Proceedings - IEEE 10th International Conference on eScience, eScience 2014* 1 (2014), 187–195. https://doi.org/10.1109/eScience.2014.22

[20] Daniel Garijo, Pinar Alper, Khalid Belhajjame, Oscar Corcho, Yolanda Gil, and Carole Goble. 2014. Common motifs in scientific workflows: An empirical analysis. *Future Generation Computer Systems* 36 (2014), 338 – 351. https://doi.org/10.1016/j.future.2013.09.018 Special Section: Intelligent Big Data Processing Special Section: Behavior Data Security Issues in Network Information Propagation Special Section: Energy-efficiency in Large Distributed Computing Architectures Special Section: eScience Infrastructure and Applications.

[21] Daniel Garijo, Yolanda Gil, and Oscar Corcho. 2017. Abstract, link, publish, exploit: An end to end framework for workflow sharing. *Future Generation Computer Systems* 75 (2017), 271 – 283. https://doi.org/10.1016/j.future.2017.01.008

[22] Fosca Giannotti, Roberto Trasarti, Kalina Bontcheva, and Valerio Grossi. 2018. SoBigData: Social Mining & Big Data Ecosystem. In *Companion of the The Web Conference 2018 on The Web Conference 2018*. International World Wide Web Conferences Steering Committee, 437–438.

[23] Carole Goble, Sarah Cohen-Boulakia, Stian Soiland-Reyes, Daniel Garijo, Yolanda Gil, Michael R. Crusoe, Kristian Peters, and Daniel Schober. 2020. FAIR Computational Workflows. *Data Intelligence* 2, 1-2 (2020), 108–121. https://doi.org/10.1162/dint_a_00033 arXiv:https://doi.org/10.1162/dint_a_00033

[24] Dongmin Kim, Hanif Muhammad, Eunsam Kim, Sumi Helal, and Choonhwa Lee. 2019. TOSCA-based and federation-aware cloud orchestration for Kubernetes container platform. *Applied Sciences* 9, 1 (2019), 191.

[25] Georgia Kougka, Anastasios Gounaris, and Alkis Simitsis. 2018. The many faces of data-centric workflow optimization: a survey. *International Journal of Data Science and Analytics* 6, 2 (2018), 81–107. https://doi.org/10.1007/s41060-018-0107-0

[26] Bruno Lepri, Nuria Oliver, Emmanuel Letouzé, Alex Pentland, and Patrick Vinck. 2018. Fair, Transparent, and Accountable Algorithmic Decision-making Processes: The Premise, the Proposed Solutions, and the Open Challenges. *Philosophy & Technology* 31 (12 2018). https://doi.org/10.1007/s13347-017-0279-x

[27] Chee Sun Liew, Malcolm P. Atkinson, Michelle Galea, Tan Fong Ang, Paul Martin, and Jano I. Van Hemert. 2016. Scientific Workflows: Moving Across Paradigms. *Comput. Surveys* 49, 4 (2016). https://doi.org/10.1145/3012429

[28] X. Llorà, B. Ács, L. S. Auvil, B. Capitanu, M. E. Welge, and D. E. Goldberg. 2008. Meandre: Semantic-Driven Data-Intensive Flows in the Clouds. In *2008 IEEE Fourth International Conference on eScience*. 238–245. https://doi.org/10.1109/eScience.2008.172

[29] Alessandro Lulli Lucchese, Laura Ricci, Emanuele Carlini, Patrizio Dazzi, and Claudio. 2015. Cracker: Crumbling Large Graphs Into Connected Components. In *20th IEEE Symposium on Computers and Communication (ISCC) (ISCC2015)*. IEEE.

[30] Alessandro Lulli, Emanuele Carlini, Patrizio Dazzi, Claudio Lucchese, and Laura Ricci. 2016. Fast connected components computation in large graphs by vertex pruning. *IEEE Transactions on Parallel and Distributed systems* 28, 3 (2016), 760–773.

[31] Suresh Marru, Lahiru Gunathilake, Chathura Herath, Patanachai Tangchaisin, Marlon Pierce, Chris Mattmann, Raminder Singh, Thilina Gunarathne, Eran Chinthaka, Ross Gardler, Aleksander Slominski, Ate Douma, Srinath Perera, and Sanjiva Weerawarana. 2011. Apache Airavata: A Framework for Distributed Applications and Computational Workflows. In *Proceedings of the 2011 ACM Workshop on Gateway Computing Environments* (Seattle, Washington, USA) *(GCE '11)*. ACM, New York, NY, USA, 21–28. https://doi.org/10.1145/2110486.2110490

[32] Mirco Nanni, Roberto Trasarti, Anna Monreale, Valerio Grossi, and Dino Pedreschi. 2016. Driving Profiles Computation and Monitoring for Car Insurance CRM. *ACM Trans. Intell. Syst. Technol.* 8, 1, Article 14 (Aug. 2016), 26 pages. https://doi.org/10.1145/2912148

[33] David De Roure, Carole Goble, and Robert Stevens. 2009. The design and realisation of the Experimentmy Virtual Research Environment for social sharing of workflows. *Future Generation Computer Systems* 25, 5 (2009), 561 – 567. https://doi.org/10.1016/j.future.2008.06.010

[34] Anit Kumar Sahu, Tian Li, Maziar Sanjabi, Manzil Zaheer, Ameet Talwalkar, and Virginia Smith. 2018. On the convergence of federated optimization in heterogeneous networks. *arXiv preprint arXiv:1812.06127* 3 (2018).

[35] Nalini Schaduangrat, Samuel Lampa, Saw Simeon, Matthew Paul Gleeson, Ola Spjuth, and Chanin Nantasenamat. 2020. Towards reproducible computational drug discovery. *Journal of Cheminformatics* 12, 1 (2020), 9. https://doi.org/10.1186/s13321-020-0408-x

[36] A. Shaon, S. Callaghan, B. Lawrence, B. Matthews, A. Woolf, T. Osborn, and C. Harpham. 2011. A Linked Data Approach to Publishing Complex Scientific Workflows. In *2011 IEEE Seventh International Conference on eScience*. 303–310. https://doi.org/10.1109/eScience.2011.49

[37] Leonid Teytelman, Alexei Stoliartchouk, Lori Kindler, and Bonnie L. Hurwitz. 2016. Protocols.io: Virtual Communities for Protocol Development and Discussion. *PLOS Biology* 14, 8 (08 2016), 1–6. https://doi.org/10.1371/journal.pbio.1002538

[38] John Violos, Vinicius Monteiro de Lira, Patrizio Dazzi, Jörn Altmann, Baseem Al-Athwari, Antonia Schwichtenberg, Young-Woo Jung, Theodora Varvarigou, and Konstantinos Tserpes. 2017. User behavior and application modeling in decentralized edge cloud infrastructures. In *International Conference on the Economics of Grids, Clouds, Systems, and Services*. Springer, Cham, 193–203.

[39] Zhenyu Wen, Renyu Yang, Peter Garraghan, Tao Lin, Jie Xu, and Michael Rovatsos. 2017. Fog orchestration for internet of things services. *IEEE Internet Computing* 21, 2 (2017), 16–24.

[40] Katherine Wolstencroft, Robert Haines, Donal Fellows, Alan Williams, David Withers, Stuart Owen, Stian Soiland-Reyes, Ian Dunlop, Aleksandra Nenadic, Paul Fisher, Jiten Bhagat, Khalid Belhajjame, Finn Bacall, Alex Hardisty, Abraham Nieva de la Hidalga, Maria P. Balcazar Vargas, Shoaib Sufi, and Carole Goble. 2013. The Taverna workflow suite: designing and executing workflows of Web Services on the desktop, web or in the cloud. *Nucleic Acids Research* 41, W1 (2013), W557–W561. https://doi.org/10.1093/nar/gkt328

[41] Chan-Hyun Youn, Min Chen, and Patrizio Dazzi. 2017. *Cloud Broker and Cloudlet for Workflow Scheduling*. Springer.

[42] Y. Zhao, M. Hategan, B. Clifford, I. Foster, G. von Laszewski, V. Nefedova, I. Raicu, T. Stef-Praun, and M. Wilde. 2007. Swift: Fast, Reliable, Loosely Coupled Parallel Computation. In *IEEE Congress on Services (Services 2007)*. 199–206. https://doi.org/10.1109/SERVICES.2007.63