An Osmotic Ecosystem for Data Streaming Applications in Smart Cities

Emanuele Carlini*
CNR-ISTI, National Research Council
Pisa, Italy
emanuele.carlini@isti.cnr.it

Lorenzo Carnevale
University of Messina
Messina, Italy
lcarnevale@unime.it

Massimo Coppola
CNR-ISTI, National Research Council
Pisa, Italy
massimo.coppola@isti.cnr.it

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

Gabriele Mencagli
University of Pisa
Pisa, Italy
mencagli@di.unipi.it

Domenico Talia
University of Calabria
Rende, Italy
talia@dimes.unical.it

Massimo Villari
University of Messina
Messina, Italy
massimo.villari@unime.it

ABSTRACT

Modern multi-tier Cloud-Edge-IoT computational platforms seamlessly map with the distributed and hierarchical nature of smart cities infrastructure. However, classical tools and methodologies to organise data as well as computational and network resources are poorly equipped to tackle the dynamic and heterogeneous environments of smart cities. In this paper we propose a reference architecture that aims to establish a unified approach for the orchestration of modern Cloud-Edge-IoT infrastructures and resources specifically tailored for data streaming applications in smart-cities. Stemming from the proposed reference architecture, we also discuss a series of open challenges, which we believe represent relevant research directions in the nearest future.

CCS CONCEPTS

• **Computer systems organization → Cloud computing**; *Peer-to-peer architectures*; *Self-organizing autonomic computing*; **Sensor networks**; **Architectures**; • **Information systems → Data stream mining**; **Data analytics**; *Sensor networks*; • **Security and privacy** → *Data anonymization and sanitization.*

KEYWORDS

Cloud Computing, Edge Computing, Internet of Things, Smart Cities, Data Streaming

*Authors are in alphabetical order. All contributed equally to the realisation of this paper.

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

ACM ISBN 978-1-4503-8384-4/21/06…$15.00
https://doi.org/10.1145/3452369.3463822

ACM Reference Format:
Emanuele Carlini, Lorenzo Carnevale, Massimo Coppola, Patrizio Dazzi, Gabriele Mencagli, Domenico Talia, and Massimo Villari. 2021. An Osmotic Ecosystem for Data Streaming Applications in Smart Cities. 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, 5 pages. https://doi.org/10.1145/3452369.3463822

1 INTRODUCTION

The pace of technological changes is growing rapidly and so are the demands for new functions and systems able to improve the quality of life and generate value for our society. Several technological areas are the protagonists underlying this change: Internet of Things (IoT), Edge and Cloud Computing, Artificial Intelligence (AI) are revolutionising our lives, especially in large urban centers. As a matter of fact, today's urban centers are growing rapidly: the percentages of people living in cities went from the 29% of 1950 to the 50% of 2008, and it is expected to reach 65% in 2040 [26]. This massive population density makes urban environments valuable platforms (i.e. Smart Cities) to build instantaneous, life-enhancing services demanded by digital citizens: transportation, decision making, and real-time alerting services are few examples of this.

Edge computing is a key technology to enrich smart cities with distributed and widely available computing capabilities, suitable for those services that are interactive and require a fast computation to enable some form of decision making, such as video surveillance [19] and parking organization [6]. We argue that, in the smart cities of the future, the classical definition of Edge resources can be stretched to include even the more fine-grained IoT resources, such as CCTV cameras, traffic lights, sensors and many others. Such devices are often located in the key elements of the urban area (e.g. main squares or streets, prominent city buildings, and underground metro stations) and are often equipped with small computational capabilities (such as single-board computers) and can also make use of different hardware accelerators available as small low-power System-on-Chip devices.

With such computational capabilities available, there is the need to explore solutions that advance the state of the art when performing distributed data-intensive tasks. Such tasks present complex requirements: to handle multiple streams of data coming from various sensors, to perform distributed data-analysis tasks and to organize computational resources [8]. Further, the computational infrastructure available in a smart city is often composed by a rather volatile and "sparse" pool of low-latency, highly heterogeneous edge and IoT devices eventually paired with Cloud Computing resources in a so-called *Cloud-Edge-IoT continuum*. A key characteristic of such infrastructure is the frequent fluctuations in computational capacity as well as the network performance both in time and space terms. This characteristic will inevitably lead to challenges regarding the support of data streaming applications, and guarantee their availability, reliability, and performance.

In this paper we propose a *reference* high-level architecture to tackle the scenarios defined above. The architecture is based on the concept of the *Osmotic Computing* [31] and frames the technological context and the necessary computational environment targeting data-intensive applications. By starting from the reference architecture, we describe a series of open research challenges and directions that need to be addressed toward an actual realisation of the proposed systems.

2 THE OSMOTIC ECOSYSTEM

We define an integrated ecosystem (called the *Osmotic Ecosystem*, see Figure 1), of services and methodologies aimed at the definition of Cloud-IoT-Edge workflows and the orchestration of computational resources and data, specifically designed for smart-city infrastructures and data streaming applications. . The core idea of the osmotic ecosystem is to support the translation of a monolithic application [24] towards the decomposition of separated non-monolithic micro elements (defined here as MELs) that need to be logically orchestrated, distributed, configured, deployed, and monitored via a comprehensive set of policies [31, 32]. The osmotic ecosystem goes beyond simple elastic management of deployed services, since deployment and migration strategies are related to requirements of both infrastructure (e.g. load balancing, reliability, availability) and applications (e.g. sensing/actuation capabilities, context awareness, proximity, performance and user experience), and can change over time.

The proposed osmotic ecosystem is composed of three main macro building blocks built on top of the Cloud-Edge-IoT continuum. These main blocks are described in the following sections.

2.1 Osmotic Computing Workflow

The first block is dedicated to the definition of the *Osmotic Computing Workflow* that abstracts resources and applications, also defining a programming model. This block provides programming constructs and high-level interfaces to define a collection of MELs capable of encapsulating microservices, microdata, microcomputations or microactuators. It also supports specific abstractions to connect MELs in complex Cloud-Edge-IoT workflows, enriched with meta information to drive the orchestration and deployment in the computing environment.

2.2 Osmotic Computing Environment

The *Osmotic Computing Environment* provides a set of services and methodologies that abstract the underlying heterogeneous computational infrastructures (Edge, IoT, Cloud) to the above computing workflow. These services can be logically divided into two layers.

The *Application Facilities* layer encompasses those services tailored toward the applications, and are mostly related to the management of the data streams. These include the indexing and querying of streams, for fast retrieval and search of data streaming sources. Also, it contains methods for streaming data analysis based on composable parallel building blocks, which are to be selected in order to exploit at best the computational capabilities offered at the edge, with optimization techniques to balance efficiency and performance with memory and energy constraints. Low-overhead, memory-constrained data provenance methods can be employed to automatically enrich streaming analysis outcomes with trustworthiness and privacy-related information [25].

The *Computing Infrastructure* layer tackles those challenges related to the logical organization, elastic control and configuration of MELs, as well as their monitoring, workload contention, interference evaluation and re-deployment actions. This layer also includes the abstractions and implementations of cloud-edge network overlays [13] that connect a distributed set of resources or services that have common semantic features. Overlays are built by considering both static or dynamic properties, as well as individual properties (e.g. available memory) or collective ones (e.g. mutual interconnection properties).

2.3 Privacy and Security

A key concern that expands across applications, edge- and cloud-based services and down to data processing is related to *Privacy* and *Security*. The growth of aggregated data magnifies privacy challenges to limit access to certain types of data and prevent unauthorised access (confidentiality) by protecting data from being modified or corrupted without detection. The following methodologies and techniques toward the security and privacy are considered in the osmotic ecosystem: *(i)* adherence to the General Data Protection Regulation (GDPR) in the various phases of data management; *(ii)* privacy-preserving data processing, with a focus on distributed machine learning based on homomorphic cryptography, which allows to infer conclusions on encrypted data without having access to the original input data; *(iii)* secure and privacy-aware resource orchestration, by borrowing the methodologies from the concept of Software Defined Membranes [32].

3 OPEN CHALLENGES AND RESEARCH DIRECTIONS

The implementation of the osmotic ecosystem vision into a realistic setting raises a number of challenges, which in turn open several research directions. Without any claim of completion, in the next sections we present those directions we argue are the most relevant in the current landscape of research.

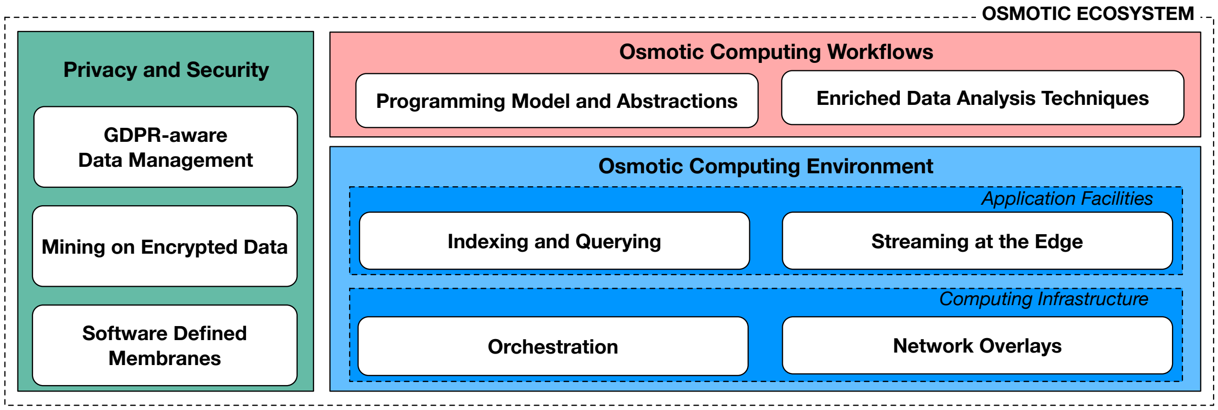

Figure 1: The Osmotic Ecosystem

3.1 Novel Programming Models for IoT workflows

The benefits of integrating different computing paradigms for supporting novel IoT applications have already been acknowledged by academic and industry-based initiatives, including Cisco, Amazon AWS, and the OpenFog Consortium [33]. Examples of edge applications applied in smart cities are discussed [2], where an intelligent offloading method for collaborative services is proposed. However, the full convergence of several computing systems aimed at making the best of the IoT has not been investigated yet. The programming model we are looking for should guarantee openness, interoperability, and programmability, possibly leveraging and extending existing workflows descriptors like the TOSCA standard [5].

To this end, a programming model for Cloud-Edge-IoT workflows should allow the development of data transformation applications as composition of existing components seamlessly integrated to leverage distributed infrastructures like smart cities. This poses several challenges for developers that need to be aware of the heterogeneity at different levels: resources (IoT, edge or cloud), virtualization technologies (container or hypervisor), runtime implementations of data analysis tasks (streamed or batched), geographic distribution, and network uncertainties.

As in any workflow language, an application must be developed as a graph of interconnected activities independently developed using different tools. With respect to our architecture, each activity is encapsulated in a separate MEL with uniform interfaces helping their interconnection. However, besides the generality of the content of each MEL, the programming model should allow the developer to provide the necessary meta information required to drive the deployment and orchestration support of the MELs workflow.

3.2 Services Organization and Orchestration

Production systems of smart cities will be often reconfigured in the future as part of the engineering processes. This aspect needs to achieve the adequate orchestration and security levels in an automated way, reducing the current static procedures and manual efforts [15]. Indeed, even though several automated deployment of applications have been developed, the management of deployed applications in a multi-cloud and/or IoT-Edge environment is only partially covered by existing approaches [17].

The adoption of the Osmotic Computing methodology in our proposed architecture aims to harmonize the resource provisioning and services deployment over both cloud and edge, proposing a new de-facto standard based on cost functions driven by AI-models [30–32]. The MEL abstractions enable the support of a virtual environment that can be adapted on the basis of the available hardware equipment, where each MEL is autonomous from a development and deployment standpoint. Scaling and managing these types of systems in smart-city environments, given the resource heterogeneity and the privacy and security constraints, is complex, so a novel orchestrator is necessary to leverage such dynamic and heterogeneous computing infrastructures.

Therefore one should extend the traditional "cloud-only" notion of runtime control and reconfiguration to resources that are deployed and available at the Cloud-Edge-IoT continuum. This possibly requires the study of machine learning techniques for developing predictive models to forecast workload inputs and performance metrics across multiple, co-located MELs on Cloud-Edge-IoT resources, in order to understand the nature of their composition and decide which MELs can coexist and can be deployed together.

3.3 Distributed Data Analysis in the Cloud-Edge-IoT Continuum

The exploitation of different distributed computing paradigms and systems can effectively enable new ways to implement scalable data analysis in IoT/Edge environments [20]. Decentralized scenarios may offer useful solutions in many application fields. In order to reach this goal, we need innovative decentralized models, integrated architectures, scalable software frameworks and distributed data mining algorithms for managing and analyzing data between edge and cloud systems [4, 11, 12, 21–23, 29]. In particular, there is the need for new data mining and machine learning strategies like federated learning, collective data mining and ensemble learning [28].

Furthermore, data management and analysis systems must be designed to process data and produce knowledge by leveraging edge nodes and IoT resources not only for gathering raw data, but also directly using their computational capabilities in a way to satisfy the requirements given the nature of the data analysis tasks and the capabilities of the utilized resources. This requires distributed data processing tasks able to extract useful patterns and learn models. They must be suitably designed to be decomposed and profitably executed on the Cloud-Edge-IoT continuum.

Therefore, our vision is based on the definition of new methodologies for designing data analysis and machine learning algorithms by specializing the XaaS (Everything as a Service) stack into a Data Analysis as a Service (DAaaS) model. According to this approach, every data analysis element will be provided as a service and it can be composed/orchestrated in a distributed workflow of services and microservices running on the different hardware nodes. This approach will make possible the implementation of distributed data analysis applications based on federated and ensemble learning services running in the Cloud-Edge-IoT continuum.

3.4 Cutting-edge Methodologies for Online Streaming Analysis on Edge/IoT

Smart-city environments produce a huge volume of data available in the form of information streams. A central requirement of these applications is to be able to process and analyse these streams in a timely fashion. However, although data stream processing is a paradigm with a long tradition [1], traditional systems like Apache Storm and Flink, which have a wide popularity and support continuous streaming, target homogeneous clusters and clouds and are not designed for the edge [14].

As a matter of fact, the shift to the Edge/IoT advocates new software engineering techniques to develop efficient streaming runtime systems, which should exhibit a high-degree of reconfigurability of the underlying implementation to leverage different kinds of resource-constrained hardware components in an efficient way and in face of dynamic workload, networking and energy conditions. Recent attempts [16] enhance traditional systems to fit the constraints of edge resources, by re-implementing parts of their runtime system introducing explicit scheduling of streamed data analysis tasks using custom scheduling policies. However, they represent custom prototypes which require to be maintained together with the standard code base of the traditional systems.

Therefore, our vision aims to identify parallel/concurrent building blocks that can be composed to build complex streaming applications, and whose internal implementation can leverage different kinds of resources transparently to the end user. This idea percolates the consolidated approach of Parallel Patterns and Algorithmic Skeletons [27] at the implementation level of the runtime system design of a framework, where each block describes a recurrent computation or communication pattern, which can be implemented with efficient mechanisms and with special focus on the constraints of embedded devices.

3.5 Overlays-as-a-Service for Cloud-Edge-IoT Continuum

The osmotic ecosystem vision is built on top of network abstractions that spawn from cloud to edge and vice versa, and which has the aim of improving the performance of the communication among the various resources of the platform. The combined activities of these abstractions shall enable resources characterisation, indexing, querying and even their organisation. In addition, they should offer useful solutions for supporting application-specific processing of data (between edge and datacenter) and providing network management abstraction (i.e. network overlays) independent of the underlying technology. Several recent works in this direction propose either general purpose distributed approaches, such as for service allocation [10] information aggregation, [13], and service discovery [3, 7, 9] or targeting specific smart city applications, such as video surveillance [18] and parking [6]. At the best of our knowledge, those and many other research works assume that overlay are statically designed and created.

Here we envision the creation of an *overlay-as-a-service* methodology, in which different overlay networks can be created on-demand according to specific contexts, applications requirements, and status of available computational resources. The goal is to support the osmotic ecosystem in a twofold fashion. First, with the creation of overlays to effectively support a scalable indexing and retrieval of computational resources and data streams. Second, by providing application and support services with the creation of on-demand ephemeral or permanent overlay networks among all resources in the Cloud-Edge-IoT continuum, to be used both for gathering data as well as for disseminating alerts and notifications to end users.

4 CONCLUSIONS

This position paper presents the architecture of the Osmotic Ecosystem, an integrated set of services and methodologies aimed at the orchestration of computational resources and data streaming applications in the smart city context. The Osmotic Ecosystem design considers the deployment and migration strategies strictly related to the dynamic and heterogeneous requirements of the network and computational smart city infrastructure as well as the one of the applications. The reference architecture is used as a starting point to discuss several relevant open research directions that would shape the work toward the realisation of the Osmotic Ecosystem vision in the future.

REFERENCES

[1] Henrique CM Andrade, Buğra Gedik, and Deepak S Turaga. 2014. *Fundamentals of stream processing: application design, systems, and analytics.* Cambridge University Press.
[2] Nelson Baloian, Yanling Chen, and Ashot Harutyunyan. 2019. Collaborative Technologies and Data Science in Smart City Applications J. UCS Special Issue. *Journal of Universal Computer Science* 25, 8 (2019), 865–867.
[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] 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.
[5] 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.
[6] Harshitha Bura, Nathan Lin, Naveen Kumar, Sangram Malekar, Sushma Nagaraj, and Kaikai Liu. 2018. An edge based smart parking solution using camera networks and deep learning. In *2018 IEEE International Conference on Cognitive Computing (ICCC).* IEEE, 17–24.
[7] Zhiming Cai, Ivan Lee, Shu-Chuan Chu, and Xuehong Huang. 2019. SimSim: a service discovery method preserving content similarity and spatial similarity in P2P mobile cloud. *Journal of Grid Computing* 17, 1 (2019), 79–95.

[8] Hung Cao, Monica Wachowicz, Chiara Renso, and Emanuele Carlini. 2019. Analytics everywhere: generating insights from the internet of things. *IEEE Access* 7 (2019), 71749–71769.

[9] 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.

[10] Emanuele Carlini, Massimo Coppola, Patrizio Dazzi, Matteo Mordacchini, and Andrea Passarella. 2016. Self-optimising decentralised service placement in heterogeneous cloud federation. In *2016 IEEE 10th International Conference on Self-adaptive and Self-organizing Systems (SASO)*. IEEE, 110–119.

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

[12] Patrizio Dazzi, Pascal Felber, Lorenzo Leonini, Matteo Mordacchini, Raffaele Perego, Martin Rajman, Étienne Rivière, et al. 2009. Peer-to-peer clustering of web-browsing users. In *LSDS-IR@ SIGIR*.

[13] Patrizio Dazzi and Matteo Mordacchini. 2020. Scalable Decentralized Indexing and Querying of Multi-Streams in the Fog. *Journal of Grid Computing* 18, 3 (2020), 395–418.

[14] Marcos Dias de Assuncao, Alexandre da Silva Veith, and Rajkumar Buyya. 2018. Distributed data stream processing and edge computing: A survey on resource elasticity and future directions. *Journal of Network and Computer Applications* 103 (2018), 1–17.

[15] Marco Ehrlich, Henning Trsek, Martin Gergeleit, Julius Paffrath, Kostyantyn Simkin, and Jürgen Jasperneite. 2019. Secure and Flexible Deployment of Industrial Applications inside Cloud-Based Environments. In *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, 1269–1272.

[16] Xinwei Fu, Talha Ghaffar, James C Davis, and Dongyoon Lee. 2019. Edgewise: a better stream processing engine for the edge. In *2019 {USENIX} Annual Technical Conference ({USENIX}{ATC} 19)*. 929–946.

[17] Lukas Harzenetter, Uwe Breitenbücher, Frank Leymann, Karoline Saatkamp, Benjamin Weder, and Michael Wurster. 2019. Automated generation of management workflows for applications based on deployment models. In *2019 IEEE 23rd International Enterprise Distributed Object Computing Conference (EDOC)*. IEEE, 216–225.

[18] Hanna Kavalionak, Emanuele Carlini, Alessandro Lulli, Claudio Gennaro, Giuseppe Amato, Carlo Meghini, and Laura Ricci. 2017. A prediction-based distributed tracking protocol for video surveillance. In *2017 IEEE 14th International Conference on Networking, Sensing and Control (ICNSC)*. IEEE, 140–145.

[19] Hanna Kavalionak, Claudio Gennaro, Giuseppe Amato, Claudio Vairo, Costantino Perciante, Carlo Meghini, Fabrizio Falchi, and Fausto Rabitti. 2020. Edge-Based Video Surveillance with Embedded Devices. In *Proceedings of the 28th Italian Symposium on Advanced Database Systems, Villasimius, Sud Sardegna, Italy (virtual due to Covid-19 pandemic), June 21-24, 2020 (CEUR Workshop Proceedings, Vol. 2646)*, Maristella Agosti, Maurizio Atzori, Paolo Ciaccia, and Letizia Tanca (Eds.). CEUR-WS.org, 278–285. http://ceur-ws.org/Vol-2646/25-paper.pdf

[20] Neeraj Kumar and Aaisha Makkar. 2020. *Machine learning in cognitive IoT*. CRC Press.

[21] 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.

[22] Alessandro Lulli, Laura Ricci, Emanuele Carlini, and Patrizio Dazzi. 2015. Distributed current flow betweenness centrality. In *2015 IEEE 9th International Conference on Self-Adaptive and Self-Organizing Systems*. IEEE, 71–80.

[23] Alessandro Lulli, Laura Ricci, Emanuele Carlini, Patrizio Dazzi, and Claudio Lucchese. 2015. Cracker: Crumbling large graphs into connected components. In *2015 IEEE symposium on computers and communication (ISCC)*. IEEE, 574–581.

[24] Matteo Nardelli, Stefan Nastic, Schahram Dustdar, Massimo Villari, and Rajiv Ranjan. 2017. Osmotic flow: Osmotic computing+ iot workflow. *IEEE Cloud Computing* 4, 2 (2017), 68–75.

[25] Dimitris Palyvos-Giannas, Bastian Havers, Marina Papatriantafilou, and Vincenzo Gulisano. 2020. Ananke: A Streaming Framework for Live Forward Provenance. *Proc. VLDB Endow.* 14, 3 (Nov. 2020), 391–403.

[26] Postscapes. 2019. *Smart city infographic*. Retrieved April 2021 from https://www.postscapes.com/anatomy-of-a-smart-city

[27] Fethi Rabhi. 2003. *Patterns and skeletons for parallel and distributed computing*. Springer Science & Business Media.

[28] Domenico Talia. 2019. A view of programming scalable data analysis: from clouds to exascale. *Journal of Cloud Computing* 8, 1 (2019), 1–16.

[29] Domenico Talia, Paolo Trunfio, and Fabrizio Marozzo. 2015. *Data analysis in the cloud: models, techniques and applications*. Elsevier.

[30] Massimo Villari, Antonio Celesti, and Maria Fazio. 2017. Towards osmotic computing: Looking at basic principles and technologies. In *Conference on Complex, Intelligent, and Software Intensive Systems*. Springer, 906–915.

[31] Massimo Villari, Maria Fazio, Schahram Dustdar, Omer Rana, and Rajiv Ranjan. 2016. Osmotic computing: A new paradigm for edge/cloud integration. *IEEE Cloud Computing* 3, 6 (2016), 76–83.

[32] Massimo Villari, Antonino Galletta, Antonino Celesti, Lorenzo Carnevale, and Maria Fazio. 2018. Osmotic computing: software defined membranes meet private/federated blockchains. In *2018 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 01292–01297.

[33] Yang Yang, Jianwei Huang, Tao Zhang, and Joe Weinman. 2020. *Fog and Fogonomics: Challenges and Practices of Fog Computing, Communication, Networking, Strategy, and Economics*. John Wiley & Sons.