Multi-Access Edge Computing in Action

Multi-Access Edge Computing in Action

Dario Sabella
Alex Reznik
Rui Frazao

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

Printed on acid-free paper

International Standard Book Number-13: 978-0-367-17394-4 (Hardback)

Visit the Taylor & Francis Web site at
http://www.taylorandfrancis.com

and the CRC Press Web site at
http://www.crcpress.com

Contents

Part 2 MEC and the Market Scenarios

Foreword

I have contributed to the telecommunications industry standards for over 16 years, largely in 3GPP and ETSI, and to a lesser extent or indirectly in IETF, BBF, and in industry bodies such as NGMN and GSMA.

The authors and I first collaborated over 5 years ago at the very inception of the ETSI Industry Standardisation Group for Mobile Edge Computing, the first standards development body to produce interoperable standards and APIs for what later became multi-access edge computing (MEC) and which is still the leading body in that field. Most of us have held office in that organization. We have all contributed to its technical, marketing, or outreach publications in all that time, and continue to do so.

In its founding white paper, the ISG recognized that a new ecosystem was needed in order to serve perceived new markets that required very low communication latency or provision of content very close to its end consumer. That meant decentralizing cloud-based compute workloads and MEC was born. Communication service providers, infrastructure vendors, platform vendors, application developers, etc., all came together to produce interoperable solutions for a number of different deployment options and to raise awareness of the valuable new consumer and enterprise markets that only MEC could serve.

Five years on, much of that vision has become or is nearing reality. MEC trials are commonplace and commercial offerings exist. MEC is included in proposed solutions for 5G, factory automation, smart cities, and intelligent transport system use cases, to name just a few. Other standardization bodies and forums such as 3GPP, OpenFog Consortium, and the Open Edge Computing initiative are broadening the MEC footprint to such an extent that it has become mainstream.

This book provides the readers with a unique opportunity to learn about and assimilate the technical, deployment, market, ecosystem, and industry aspects of MEC, all in one place. Its authors are leaders in the field, and they have made key contributions in all of those areas through further specific ETSI White Papers and presentations at numerous industry-wide events.

Dr Adrian Neal
Senior Manager, Industry Standards
Vodafone Group Services Ltd.

Acknowledgments

We would like to warmly thank all colleagues and collaborators who are working with us in the area of edge computing and multi-access edge computing (MEC) standards, as in some cases the ideas expressed in this book are also a result of a collaborative effort with them. Our knowledge and expertise on this technology took advantage from the huge networking and tremendous interest of companies in this area, starting from operators, technology providers, system integrators, but also smaller companies and start-ups. The interest of the industry in edge computing is continuously increasing, so our perception of the ecosystem is taking benefit from the numerous feedback received from the various stakeholders.

On the private sphere, we would also like to thank our wives and families for their patience and support, since the writing of this book subtracted time to our family duties. So, we hope our efforts were worthwhile, and we would like to thank in advance the readers for their interest in this book.

Authors

Dario Sabella is working with INTEL as Senior Manager Standards and Research, acting also as company delegate of the 5GAA (5G Automotive Association). In his role within the Next Generation Standards division, Dario is driving new technologies and edge cloud innovation for the new communication systems, involved in ecosystem engagement and coordinating internal alignment on edge computing across SDOs and industry groups, in support of internal and external stakeholders/customers. In 2019, he has been appointed as ETSI MEC Vice-Chairman. Previously, he was serving as multi-access edge computing (MEC) Secretary and Lead of Industry Groups, and from 2015 as Vice-Chairman of ETSI MEC (Mobile Edge Computing) IEG. Prior to February 2017, he worked in TIM (Telecom Italia group), in the Wireless Access Innovation division, as responsible in various TIM research, experimental and operational activities on OFDMA technologies (WiMAX, LTE, 5G), cloud technologies (MEC), and energy efficiency (for energy saving in TIM's mobile network). From 2006, he was involved in many international projects and technological trials with TIM's subsidiary companies (ETECSA Cuba, TIM Brasil, Telecom Argentina). Since joining TIM in 2001, he has been involved in a wide range of internal and external projects (including FP7 and H2020 EU funded projects), often with leadership roles. Author of several publications (40+) and patents (20+) in the field of wireless communications, radio resource management, energy efficiency, and edge computing, Dario has also organized several international workshops and conferences.

Alex Reznik is a Hewlett Packard Enterprise (HPE) Distinguished Technologist, currently driving technical customer engagement on HPE's Telco strategic account team. In this role, he is involved in various aspects of helping a Tier 1 Telco evolve towards a flexible infrastructure capable of delivering on the full promises of 5G. Since March 2017, Alex also serves as Chair of ETSI's multi-access edge computing (MEC) ISG – the leading international standards group focused on enabling edge computing in access networks.

Prior to May 2016, Alex was a Senior Principal Engineer/Senior Director at InterDigital leading the company's research and development activities in the area of wireless internet evolution. Since joining InterDigital in 1999, he has been involved in a wide range of projects, including leadership of 3G modem ASIC architecture, design of advanced wireless security systems, coordination of standards strategy in the cognitive networks space, development of advanced IP mobility and heterogeneous access technologies, and development of new content management techniques for the mobile edge.

Alex earned his B.S.E.E. Summa Cum Laude from The Cooper Union, S.M. in Electrical Engineering and Computer Science from the Massachusetts Institute of Technology, and Ph.D. in Electrical Engineering from Princeton University. He held a Visiting Faculty appointment at WINLAB, Rutgers University, where he collaborated on research in cognitive radio, wireless security, and future mobile Internet. He served as the Vice-Chair of the Services Working Group at the Small Cells Forum. Alex is an inventor of over 150 granted U.S. patents and has been awarded numerous awards for Innovation at InterDigital.

Rui Frazao is the CTO and EVP of EMEA Operations at B-Yond. Prior to B-Yond, Rui was CTO of Vasona Networks, an innovative provider of multi-access edge computing (MEC) solutions, acquired by ZephyrTel (2018). He also previously held various group technology positions during his 15 years at Vodafone including serving as the

Director of Network Engineering overseeing network activities across Germany, the Netherlands, Hungary, and the Czech Republic. His work with Vodafone included implementing the industry's earliest VoLTE deployments and launched the first virtualized network core platforms in Europe. Rui previously held roles with Cisco, payment network SIBS, and the Lisbon Stock Exchange. He has completed studies spanning business strategies, computer systems, electrical engineering, and telecommunications.

Introduction

Edge presence is viewed as absolutely necessary to enable the key use cases driving the need for the 5th Generation of Mobile Technology ("5G"). Among these are Tactile Internet, Interactive Gaming, Virtual Reality, and Industrial Internet. All these require extremely low latency for some application components. As a consequence, physical limitations (i.e., speed of light) prohibit the execution of these components in the traditional "deep" cloud. Another set of use cases that is likely to heavily rely on edge computing is "massive" Internet of Things (IoT) – that is, IoT where a large number of devices such as sensors send a large amount of data upstream.

Market consensus is that pre-filtering of this data at the edge is necessary in order to make the overall system scalable without overloading the network and deep cloud compute resources. This makes edge presence generically critical for the success of 5G, as well as the related MEC standard. Moreover, as a recent white paper from ETSI demonstrates,[1] MEC can play an even more critical role in the industry's move toward 5G. It enables deployment of 5G applications over the existing 4G infrastructure, thus smoothing the investment curve required to get to 5G and allowing Telco operators to better align expenditures associated with the deployment of 5G with actual 5G-related revenue streams.

Making 5G a reality involves a highly diverse stakeholder ecosystem that not only includes Telco operators, vendors, application, and content providers, but also start-ups and developer communities, as well as local government and other public entities. To many of these, enabling MEC implies a significant investment in developing and deploying new infrastructure, re-architecting existing applications and integrating these with MEC services, understanding the positive impacts that a large number of micro data centers can have on local economies as well as the planning needed to allow these to spread. All these diverse entities are looking for a comprehensive understanding of the benefits of MEC from their perspective and guidance on how to solve the challenges related to 5G.

This book provides a complete and strategic overview of MEC. It covers network and technology aspects, describes the market scenarios from the point of view of different stakeholders, and analyzes deployment aspects and actions to engage the ecosystem.

As the earlier discussion makes clear, MEC exists in and supports a highly complex "5G world" in which technologists and non-technology decision makers must act in concert and do so within a large interconnected ecosystem of which MEC is just one, albeit important, part.

The book is divided into three sections, with several chapters in each, to address these three key aspects – a technology-focused section, a market-focused section, and an ecosystem-focused section.

Note

1 www.etsi.org/images/files/ETSIWhitePapers/etsi_wp24_MEC_deployment_in_4G_5G_FINAL.pdf

PART 1

MEC and the Network

1

From Cloud Computing to Multi-Access Edge Computing

This chapter introduces multi-access edge computing (MEC) from a network perspective, starting from the historical background of cloud computing, and then considering new trends (including open innovation, network softwarization, and convergence between Telco and information technology [IT]) that drove the evolution toward the edge.

Let us start, as the saying goes, in the beginning. What is MEC? To understand this properly, we actually need to start from the end, with the letter "C" denoting "Computing." This book is primarily about computing, which currently means it is about "the Cloud" (whatever that means – we'll get to that a bit later). Then, there is the "E" denoting "Edge." Chances are you think you know what "edge computing" is, but we are willing to bet that your definition is too narrow. If that's true, we hope that at the very least, this book helps broaden your "edge horizons." Then, there is the "M" which stands for "Multi-access." Here, the "access" is important – it is "edge computing" (whatever that means) which is somehow "connected" to an "access" – that is, a network that users and other client devices (e.g., those billions of things that the Internet of Things [IoT] is being built from) use to "access" the Internet. The "multi" in retrospect is the least important part of the acronym, a designation indicating that MEC technologies can be used in all kinds of networks (mobile, Wi-Fi, fixed access) to enable all kinds of applications.

And so, it appears, MEC is a kind of a chimera – a cloud technology that's located away from the cloud and that has something important to do with networking. It is also a chimera because – as we shall see – it comes about from the convergence of several disparate trends that have been around for some time. But this does not mean that MEC as a field is uncoordinated, disjointed, and nonfunctional – no more so than the mythical Chimera was.

As a brief digression for those readers with a theoretical bent, MEC represents a practical convergence of computing, communication, and – through its critical importance in enabling the industrial IoT – control: the holy grail of modern information sciences. And for those of you with a more philosophical bent, it is also a vibrant illustration of the efficiency and robustness of decentralized decision-making as compared to centralized "optimized" approaches.

1.1 To Edge or Not to Edge

A proper place to start seems to be the question of why one even needs edge computing in general and MEC in particular. Much has been written on the subject, but it can be summarized as follows: there are applications for which the traditional cloud-based application hosting environment simply does not work. This can happen for a number of reasons, and some of the more common of these are:

- The application is latency sensitive (or has latency-sensitive components) and therefore cannot sustain the latency associated with hosting in the traditional cloud.
- Application clients generate significant data which requires processing, and it is not economical, or, perhaps even not feasible to push all this data into the cloud.
- There are requirements to retain data locally, for example, within the enterprise network.

A big driver for edge computing is the IoT, where edge computing is commonly referred to as *fog computing*. NIST (National Institute of Standards and Technology), in its "Fog Computing: Conceptual Model" report [8], makes the following statement:

> Managing the data generated by Internet of Things (IoT) sensors and actuators is one of the biggest challenges faced when deploying an IoT system. Traditional cloud-based IoT systems

are challenged by the large scale, heterogeneity, and high latency witnessed in some cloud ecosystems. One solution is to decentralize applications, management, and data analytics into the network itself using a distributed and federated compute model.

Moreover, IoT is rapidly developing into a significant driver of edge computing revenue – as evidenced by Microsoft's edge cloud solution called "Azure IoT Edge."

However, IoT is just one of the several types of applications that require edge presence. In a white paper that has been widely influential in defining what "5G" is, the Next Generation Mobile Networks (NGMN) alliance lists eight classes of 5G applications that define 5G user experience and drive requirements on 5G mobile networks [9]. These include:

- Pervasive Video
- 50+ Mbps Everywhere
- High-Speed Train
- Sensor Networks
- Tactile Internet
- Natural Disaster
- E-Health Services
- Broadcast Services

A rough top-level analysis of these categories leads to a conclusion that most of them either require edge computing or significantly benefit from it. Indeed, we can make the following statements:

- Pervasive Video: edge computing can be used to significantly reduce backhaul/core network loading by edge caching and video processing and transcoding at the edge.
- High-Speed Train: such "high-speed" environments will almost certainly require application presence "on the train" to avoid dealing with network limitations associated with connectivity from a high-speed platform to a stationary network.
- Sensor Networks: The massive IoT problem of collecting and processing massive amounts of data, which is a primary focus of fog computing, lies in this category.

- Tactile Internet: use cases and applications in this category are known to require end-to-end latencies as low as 1 msec. In most networks, the physical limitations imposed by the speed of light make it impossible to achieve such latencies without edge computing.
- Natural Disaster: supporting these use cases requires deploying networks on a "connectivity island" (i.e., with limited/intermittent or even absent connectivity to the Internet). Thus, any applications have to run at the edge.
- Broadcast Services: these can benefit significantly when content can be present at the edge, as that would save significant network traffic. Moreover, edge-based contextualization of broadcast can improve what is made available in each particular area.

Clearly, edge computing is a key enabling technology for 5G, something that was recognized as early as the NGMN paper, which lists "Smart Edge Node" as a "Technology Building Block" and lists its use to run core network services close to the user as well as its potential use for application services such as edge caching.

However, focused as it was on mobile *networks*, what the NGMN paper missed is that because its "Smart Edge Node" is a landing zone for applications, it really needs to become a kind of "cloud node." This theme was picked up by ETSI (European Telecommunications Standards Institute) in the white paper "Mobile Edge Computing: A Key Technology Towards 5G" and in the creation of an Industry Specification Group (ISG) focused on what was called mobile edge computing (MEC) [10]. Within a few years, the group was renamed to multi-access edge computing (keeping the MEC abbreviation) to recognize the fact that its work was applicable across all types of access

> MEC thus represents a key technology and architectural concept to enable the evolution to 5G, since it helps advance the transformation of the mobile broadband network into a programmable world and contributes to satisfying the demanding requirements of 5G in terms of expected throughput, latency, scalability and automation.

networks: mobile (3GPP defined) as well as Wi-Fi, fixed access, etc. Again, why paraphrase when we can just quote:

One thing that was missed, or rather not highlighted by all this work, is that edge computing – specifically MEC – is not just a "5G technology." In fact, MEC is a critical tool in enabling operators to launch 5G applications on their existing 4G networks. This can have a significant impact on the business side of MEC – something discussed in detail in Ref. [11] and also in our discussion of the economic and business aspect of MEC in Chapter 3.

To conclude this brief introductory discussion, let us summarize the themes: edge presence is needed to make the full world of 5G worknetwork. This includes IoT, which is the focus of many initial deployments, but encompasses a much broader set of applications, use cases, and markets. MEC enables such edge presence by creating a cloud-like application landing zone within the access network – that is, as close to the client devices as possible. It is therefore a key enabler of the emerging world of computing – 5G, IoT, AR/VR, etc. This book expands on these themes and examines in some detail what they mean, the various ecosystem players, challenges and opportunities, as well as provides an overview of the key technologies involved. However, we must start by actually explaining what MEC is – or is not – and this is what we turn to next.

1.2 The Cloud Part of MEC

Recall, a page or two back, we noted that the primary letter in the "MEC" abbreviation is the last one – "C" denoting "computing", but really denoting "cloud." And so, we begin by looking at the cloud computing aspects of MEC. The Wikipedia page on "Cloud Computing"[1] defines it as follows.

> **Cloud computing** is an IT paradigm that enables ubiquitous access to shared pools of configurable system resources and higher level services that can be rapidly provisioned with minimal management effort, often over the Internet. Cloud computing relies on sharing of resources to achieve coherence and economies of scale, similar to a public utility.

As the same page notes, the term was popularized by Amazon Web Services (AWS) in the mid-2000s but can be dated to at least another decade prior. So, the cloud computing aspect of edge computing seems to be a well-known thing. Indeed, one of the main goals of edge computing is to "enable ubiquitous access to shared pool of configurable system resources and higher-level services that can be rapidly provisioned." A detail-oriented reader may wonder why the quote stops where it does, and indeed this is not accidental.

So, let us consider what is *not* requoted, notably "sharing of resources to achieve coherence and economies of scale." In fact, achieving what's behind these two short terms required significant advances which took several decades to be realized to a point where cloud computing became an economically viable business:

- Separation of physical hardware and applications through virtualization. This made it possible to migrate application workloads between different hardware platforms without requiring existence of different SW builds for each particular type of HW.
- Convergence to a few, industry-standardized "compute architectures" – primarily the Intel x86 architecture, so that the vast majority of applications that are virtualized are built with the assumption of an Intel-architecture–based processing underlying it.
- Development of high-speed Internet, which made possible transfer of large amounts of data and computation outside private enterprise networks.
- Development of the World Wide Web, which enabled name-based resource access paradigms. (It is unlikely that cloud computing would work well if our applications had to rely on IP for resource addressing, since IP addresses are naturally tied – that is, "pinned" – to particular HW.)
- Introduction, notably by AWS, of REST-API–based service management framework – using the World Wide Web transport mechanism (HTTP).
- An economic environment that made possible the deployment of massively sized data centers that could be turned into shared public clouds (again, led by AWS).

The above developments and resulting technologies made it possible for enterprises and cloud providers to put together systems which presented compute resources to applications as a homogeneous pool of uniform abstract resources (vCPUs, virtual RAM, virtual disk storage) that can be consumed as without regard as to their physical origin. To highlight how difficult this task really is, we note that even today the x86 CPU architecture remains the single most prevalent virtualized architecture. While ARM-based processors are ubiquitous across many industries, including, notably the telecom industry, the same is not true for ARM virtualization, which is significantly less used than x86-based virtualization. Furthermore, if you have a high-performance computing application that requires access to GPUs, then you are out of luck. Notwithstanding the extensive adoption of GPUs for a wide range of applications, GPU virtualization support is only now being developed by, for example, OpenStack. This lack of adoption is not due to any deep technical challenges. In the case of ARM, for example, the virtualization technology has been around for a while. This is because economies of scale are required for a cloud system to make sense, and enabling such a scale requires a timely convergence of a lot of factors, including existence of extensive ecosystems of applications and tools that can come together "at the right time" to make the cloud work.

So, is MEC a cloud technology? Absolutely. It is all about abstraction of compute (and storage) resources in exactly the same way as traditional cloud technology. Like traditional cloud technology, MEC leverages modern resource access paradigms (specifically Web-based resource access) and management framework; for example, all ETSI MEC APIs are defined to be RESTful and use HTTP as the default transport. However, it differs from a traditional cloud in a key aspect: scale. We are not implying that MEC (and more broadly, edge computing) lacks the same scale as traditional cloud computing. Rather, the nature of scale, and therefore the challenges that scaling presents, are of a different nature. To properly understand this, we need to examine the second component of MEC – the letter "E" denoting "Edge."

1.3 The Edge Part of MEC

There are important reasons why edge computing is now moving into the forefront of the conversation both in cloud computing and in Telco – and it is not because it is a hot new technological development. The origins of edge computing can be traced at least to Edge content distribution networks (CDNs), which were developed in the early to mid-2000s. A good brief summary can be found on the CloudFlare site (www.cloudflare.com/learning/cdn/glossary/edge-server/), which states the following:

> An edge server is a type of edge device that provides an entry point into a network. Other edges devices include routers and routing switches. Edge devices are often placed inside Internet exchange points (IxPs) to allow different networks to connect and share transit.

The CloudFlare site also provides a nice illustration, which is reproduced in Figure 1.1.

Here we make a key observation. The "Edge CDN" server as defined by CloudFlare – and indeed as is commonly used by others – is actually located at the farthermost point of the communication services provider (CSP) network (the Internet Service Provider/ISP being a special case of CSP) from the user. In part, this was dictated by a simple necessity – Edge CDN providers such as CloudFlare and Akamai simply could not get their devices any closer, since that would have meant leaving the Internet and placing their devices inside the CSP's proprietary networks.

This led to a natural next step, briefly explored, of developing a CSP-owned "Edge CDN" located as close to the user as possible.

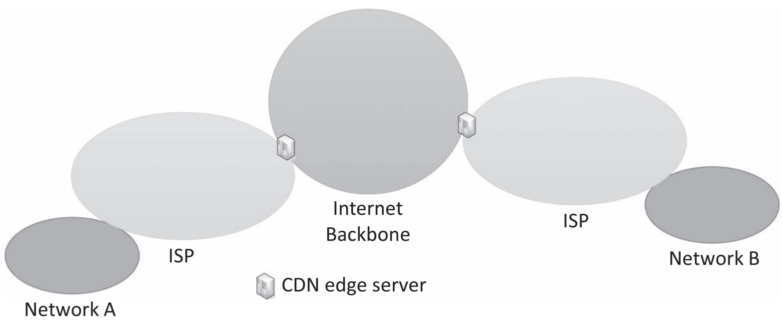

Figure 1.1 Illustration of Edge CDN. (Elaboration from CloudFlare www.cloudflare.com/learning/cdn/glossary/edge-server.)

In the case of mobile networks, this would mean locating it at the radio access network (RAN), or rather on the "S1" interface connecting the RAN to the core network. Unfortunately, doing so would require intercepting the normal flow of traffic which, in a pre-5G mobile core network, is tunneled between the RAN (e.g., the base station) and the Packet Gateway (PGW). The PGW is located at the farthest edge of the mobile network, that is, precisely where a "normal" Edge CDN is located.[2] The situation is illustrated in Figure 1.2, which shows a highly simplified diagram of the 4G core network (called the "evolved packet core" or "ePC"). The ePC interfaces are labelled, but of these, the S1 interfaces (note that there are two!) and the SGi interface are going to be really important, as they will reoccur at various points of discussion in this book as we encounter "MEC on the S1" and "MEC on the SGi" implementation options. The S1-U interface – for S1 User Plane – connects the RAN to the Serving Gateway (SGW) and carries user traffic, while the S1-MME interface connects the RAN to the mobility management entity (MME) and carries traffic which controls the various aspects by which end user devices access the RAN. It should properly be considered the "control plane." The SGi interface is the "architectural reference" the 3GPP gives to the "plain vanilla IP traffic" interface in and out of the ePC. While Edge CDNs could be placed in other logical locations in the ePC (e.g., on the S5, within the SGW), generally speaking, the assumption is that if you are going to move into the mobile network, you want to be at the RAN, that is, on the S1.

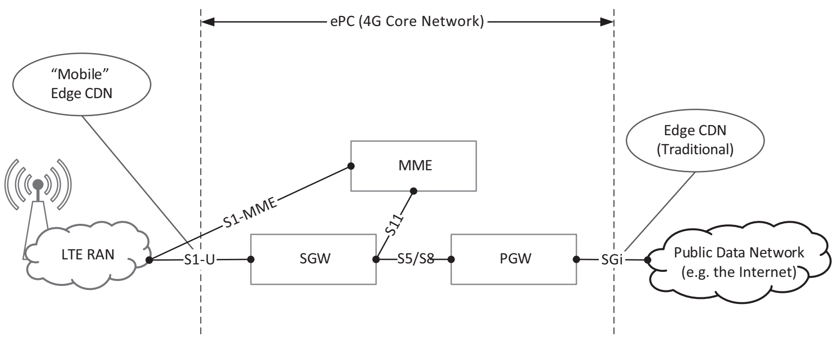

Figure 1.2 Highly simplified 4G core network showing Edge CDN locations.

The need to have an "Edge CDN" node "on the S1," as well as the potential to offer other services there, led to the development of both a standards-defined architecture (LIPA) and a "transparent" local breakout approach, in which the tunnels on the S1 are broken. Both approaches were the subject of active development and productization in the mid-2000s and, anticipating MEC, were used for applications much broader than just "edge data caching." See Ref. [3] for an example that is completely different from edge caching as well as for a good discussion of how locating processing "on the S1" works. It should be noted here that in the 5G architecture, as currently being defined by 3GPP, placement of an application function next to the RAN will be supported natively via a properly located user-plane function (UPF). We shall discuss this in more detail further ahead.

Returning, however, to the traditional Edge CDN, we note that its location at the far edge of the CSP network was dictated by an additional factor – how these entities work. At least in part they use sophisticated algorithms to analyze user population statistics and content request statistics, and attempt to predict which content is likely to be requested by which user populations. This only works if the user and content population statistics are sufficiently large and if the storage at the "edge" sites is also sufficiently large to hold a significant amount of content. Moving the Edge CDN closer to the user also reduces the size (and thus, statistical sample) of the user population served at any given time as well as the amount of content (again statistical sample) that passes the caching point. This could be counteracted by increasing the size of the cache; unfortunately, the economics of moving to the edge dictate the exact opposite – the amount of storage has to be reduced. Thus, a traditional approach to edge caching would not work much closer than the edge of the CSP network.

The way out of this is to take advantage of the rich context which being at the very far edge offers. For example, a very small cell serving a coffee shop is highly contextualized – its user population is highly likely to be into coffee and have a particular profile associated with that coffee shop. If that context can be exposed to the application, the application might just know what to do about it. If it is then provided a landing zone for its content at the small cell, it might well know what to do with this storage. This idea was recognized by the Small Cells community; the work of the Small Cells Forum in this area is documented in Refs. [4–7]. However, its realization would have to wait for edge computing to come into its own – after all, once a landing zone for content was provided for application, a natural next step is to also have a landing zone for the application's compute – at least for some components that could run at the edge.

Clearly, then, the idea of doing something at the edge is neither new nor are some of the technologies enabling it, but that still leaves open the question "what is the edge?" or, maybe, "where is the edge?" or, perhaps, "where is the boundary between the edge and the traditional cloud?" Let us now make the following bold statement. One of the more difficult questions you – the reader – may ever be asked is: "what is the 'edge' in the context of 'edge computing?'" It's easy to give some good examples. Moreover, if you are deeply immersed in your own field, you may think that *your* example is actually *the* edge. But then, talking to an acquaintance working in a related field, you find, to your surprise, that their "edge" is different from yours and that in fact defining where that "deep" cloud ends and the "edge" starts is difficult. Nevertheless, this is something we need to do – otherwise, the rest of this book is moot. We cannot very well write a book (or an essay for that matter) useful to a broad audience unless we have some common understanding with that audience of what it is that the book is about.

1.4 The Access Part of MEC

As much as we want to realize, we need to start with good examples. Amazon seems to have a clear definition of what edge is – just look at Greengrass. Microsoft more or less agrees with them, ergo Azure IoT Edge or AzureStack. This leads us to the following definition: edge is the extension of a public cloud to on-premise deployments. The idea is to provide the enterprise customer a unified management experience across both "deep" public and on-premise edge cloud and seamless automated workload migration (subject, of course, to size and other constraints).[3] In fact, in the world of enterprise computing as it stands today, this is a pretty good definition – and thus a good starting point for us.

What's missing? One instance of edge computing which is present in most enterprises and which doesn't quite fit this definition is customer premises equipment (CPE). A CPE is a component of wide-area network (WAN) infrastructure, provided by the CSP to its (usually) enterprise customers. It resides at a customer's site and terminates the WAN links into that site. Its purpose is to provide secure and reliable access between the WAN communication capabilities and the local area network (LAN) at that particular site. As such, a CPE typically comprises the switching and smart routing functionality for moving traffic between the WAN and LAN and also for load balancing across WAN links when multiple such links are present, as is often the case. Additionally, a firewall is present, providing industry standard security services to traffic passing to/from the enterprise. Both the switching/routing and firewall functionalities are typically policy-configurable and support QoS differentiation, integration with enterprise policy systems, etc. In some cases, additional functionality such as CSP-required support for charging, regulatory compliance, or Wi-Fi AP (Access Point) and LTE eNB may also be part of the CPE.

The traditional CPE consisted of discrete vertical (HW+SW) implementation of all these components. Often, but not always, these were packaged into a single "box" so that the CPE appeared as a single device to the user. Nevertheless, the internal of the CPE remained HW-based, making it inflexible and costly to upgrade, maintain, and fix.

Recently, the industry has been moving toward a flexible, configurable, and often virtualized WAN approach, usually referred to as software-defined WAN (SD-WAN). As part of this trend, there is an increasing move to replace the traditional CPE with a flexible "universal" CPE (uCPE) where all the CPE applications are virtualized and run on a generic compute platform. Figure 1.3 shows an example of a typical uCPE with a WLAN controller application in addition to the standard uCPE applications.

As is the case with other applications, virtualization brings about significant advantages: the ability to remotely monitor, maintain, and upgrade the various uCPE applications and to do so frequently and inexpensively (what used to be an HW change now becomes an SW upgrade); the ability to deliver different CPE appliances (i.e., different max throughput, max connections, users, WAN

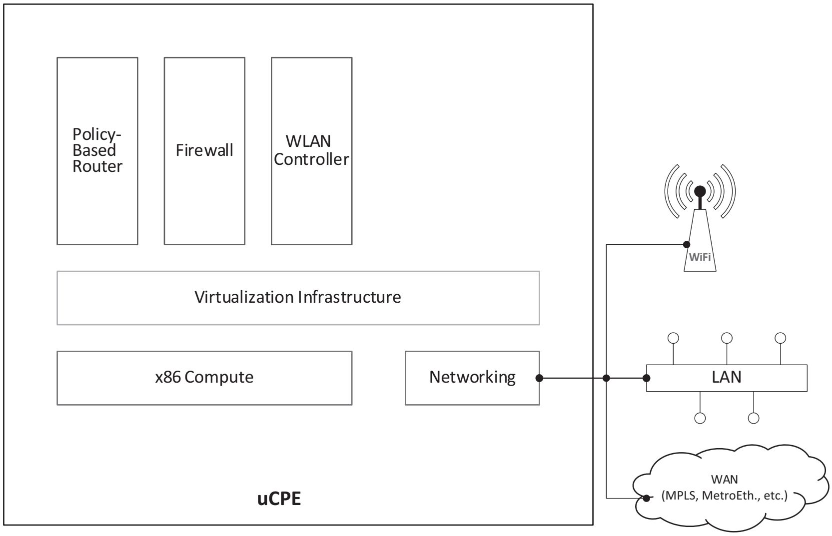

Figure 1.3 An example of a uCPE.

links, etc.) on a common platform – in fact, the ability to morph one type of CPE into another (hence, the "universal" moniker). For example, the uCPE device in Figure 1.3 may have been shipped to a customer site without the WLAN controller application. At some point, while the device is at the site, the customer requests the addition of the WLAN controller functionality. The appropriate application is pushed to the device remotely and activated, with the customer needing only to turn on and connect the WLAN AP. Provided the compute platform in the uCPE is sufficiently capable to run this additional application, all of this is done remotely, without the downtime of the WAN functions (since the other applications do not need to be taken down) and with minimal customer interaction.

At this point, it appears that uCPE is essentially yet another virtualized application (or rather a set of such applications) running on generic compute – and it should therefore be possible for uCPE to run on the same on-premise edge cloud that the enterprise runs its other applications, for example, on AWS's or Azure's edge solution. However, the applications running on uCPE differ from standard cloud applications in several critical ways.

1.4.1 Real-Time Data Processing

Operations on the real-time data stream is a key aspect of these applications – as such, they interact heavily with the “networking” component of the physical infrastructure. One common way that this is done is via data plane virtualization, that is, the Ethernet switch is virtualized on the x86 compute (Open vSwitch is an example) and the “networking” is just the physical layer function. The higher layer applications (PBR, Router, etc.) then interact with this virtualized switch. A second common approach is using SDN, in which case the “networking” component is a programmable switch (e.g., an OpenFlow switch), controlled by an SDN within the virtualization infrastructure with the higher layer applications interacting with the SDN controller.

Whatever the approach to networking is taken, the upshot of the need to operate on real-time data is that the uCPE applications are subject to very different operational requirements than traditional “IT” applications that are operated – and virtualized – by an enterprise. Application downtime – even very short downtime – can result in significant service disruption and data loss. Load balancing and resiliency through redundancy of application instances is difficult – sure, you can run multiple PBR instances on multiple compute nodes, but you cannot duplicate the data being processed by one instance at any given time.

1.4.2 SLAs and Regulatory Requirements and Critical Infrastructure

A related aspect of the uCPE applications has to do with the requirement that these are tasked to satisfy for the CSP. These include contractual service-level agreements (SLAs) – essentially performance guarantees that are associated with the connectivity services provided by the CSP to the Enterprise. Even more significantly, these communication links can often be part of critical infrastructure, that is, infrastructure where downtime may have significant and even catastrophic impact on business performance, or, in the case of public infrastructure, peoples’ lives. Finally, in the case of public infrastructure, the uCPE performance is subject to various regulations –from its potential role as critical infrastructure to lawful intercept, etc. All of these cannot withstand the kind of failures that are easily addressed for “IT” applications through the now standard approaches of redundancy and load balancing.

1.4.3 Network Function Virtualization

The result of these vastly different requirements is the development of an understanding that these represent a different type of virtual applications and that the infrastructure for enabling such functions must be different as well. This different approach to virtualization is now called network function virtualization (NFV), and it is well accepted that the virtualization and management infrastructure it requires is different from the traditional virtualization world of "IT" and "Web" applications.

One key example of a management framework for such an infrastructure has been developed by ETSI's NFV ISG, and it is a framework that will be examined deeper in this book. However, as we do with the difference between "edge computing," "MEC," and a bit later "ETSI MEC," we caution the reader not to confuse the general concept of NFV with the particular management framework of ETSI NFV – the latter is both a subset of the former (with a focus on management aspects) and a specific example of a generic concept.

1.4.4 Not Your Brother's IT Cloud

So now, we can return to the question we asked some time back – why shouldn't it be possible to run uCPE applications on the same on-premise edge cloud that the enterprise runs its other applications on? The answer is quite simply because the enterprise edge clouds are just that and are not NFV clouds. They usually cannot run NFV infrastructure – and this is indeed true of, for example, the current implementation of Amazon and Microsoft Edge Environments. As a result, most NFV deployments utilize OpenStack. Moreover, a typical enterprise application does not expose the networking aspects that are required by NFV. The compute clusters (and the underlying networking) are architected to different requirements for enterprise and NFV, and thus, one should expect a different architecture to result. The NFV applications are "packaged" as virtual network functions (VNFs) which generally require significant management over and above what a standard virtualization stack (e.g., OpenStack or VMware) offers – thus the need for SW such as VNF Managers and NFV Orchestrators. Service orchestration is done using tools far more sophisticated than what Ansible or Puppet provides – not because these are bad tools – quite the contrary, their success in the enterprise world speaks wonders of how good they are. They simply do not work in the NFV space.

Clearly then, the uCPE is a different kind of beast than a typical enterprise application, but is it just an exception, a special case of something that does not have to be generalized? The answer is no, although the uCPE is probably the only example of "multi-access" edge computing that runs within an enterprise. Here are just some *locations* within a CSP network where edge clouds may be deployed – customer premises (just discussed for the enterprise, but not for residential), antenna towers, fiber aggregation points, Central Offices (COs), mobile telephone switching stations (MTSOs). The *applications* that may run in these locations include the distributed units (DUs) and centralized units (CUs) of a cloud radio access network (CRAN), packet care (ePC), border network gateways, mobile network video optimization, deep packet inspection, etc. All of these are VNFs and thus behave and have requirements akin to those of the uCPE, and differ from traditional "IT" applications in the same way that uCPE differs from traditional uCPE applications.

This does beg the question, why are there so many *locations* running so many "weird applications"? After all, it does seem like Enterprises and Web Services, as enabled by, for example, Amazon and Microsoft, are making do with just two *locations* – the cloud and the premise. Why is Telco so different? A big part of the answer is that in the traditional IT/Web services world, there are only two entities: *the cloud* and *the enterprise's premises*. The rest is just a pipe. Since the emergence of the Internet, the players in this "over-the-top" (OTT) space have never had to worry about how the pipe worked. It just did. This was – and remains – the magic of the Internet.

However, that "pipe" is actually a highly sophisticated global engineered system (perhaps the most sophisticated such system created by mankind) in which multiple highly sophisticated components are used to ensure that both the critical communication and a request for a YouTube video receive the expected QoS. And once we start talking about virtualization of the components that make up this infrastructure, we can no longer ignore its complexity – it is exposed to us, and we have to deal with it.

Moreover, this globally engineered system relies on a massively distributed infrastructure – perhaps the only distributed infrastructure of that scale in the world. Hence, it is "edge-native" – perhaps as much as 90% of Telco systems is edge. As these components are virtualized and migrated to a generic compute platform, each becomes a cloud point-of-presence. And so, voila, we have so many options.

This raises another question: does a CSP really need all of these options? For any single CSP, chances are the answer is no. Each provider is likely to pick a few edge cloud "locales" – perhaps as few as one. The choice is driven by the specifics of each CSPs: the architecture of their network – RAN, access, etc.; the kind of population demographics they are serving; and the use cases needed to enable and the business cases associated with these. All of these can vary tremendously from one operator to another and so will their definition of "edge." Moreover, they also change over time – hence the need for architectural malleability in each operator's edge architecture. Given all this variability across the CSPs in our industry, we as "the Telco industry" do need to enable all of these various options. This means that we need to develop infrastructure, standards, management frameworks, etc. that can address all of these options in a simple, unified, and *highly scalable* way.

1.5 Who Needs Standards Anyway?

Here it is worthwhile to make a brief digression and discuss the role of standardization. If you are a "Telco person," your reaction is probably "why?", "of course we need standards!" However, if you are a "cloud person," your reaction is "why?", "I haven't had the need to bother with standards so far." And so, when it comes to MEC, we once again have a set of divergent opinions on a topic of potentially key importance, which is why we need this digression.

To understand the underlying cause of this difference, we need to, once again, consider how traditional "IT" and "Web" applications are developed. The development team makes a number of key decisions: architectural approach (e.g., microservices), development and operations philosophy (e.g., DevOps), compute platform (most likely x86 if you plan to virtualize), and so on. Among these is the cloud provider/stack. Some common choices may be AWS, Azure, Kubernetes, Mesosphere, OpenStack, etc.[4] Each one of these comes with its own approach to management services, which means, its own APIs, which the configuration, management and assurance scripts and services will have to utilize. However, this is not a problem; after all, the development team is going to pick just one, maybe two. Chances are the team is already familiar with the environment, but even if it's completely new, after some learning curve, you are up and running. Moreover, if you go with a widely used platform, there are plenty of good tools out there to help, both in the Open Source and in commercial SW space.

Let's translate this experience to MEC, keeping in mind that MEC is about the CSP provider's edge cloud – that is, the CSP becomes a cloud provider. Just for simplicity, let's restrict our attention to the United States. At the time of writing of this book, the United States has four major mobile carriers: Verizon, AT&T, Sprint, and T-Mobile. It also has several major independent broadband/cable providers: Comcast, Spectrum, Time Warner, and CenturyLink. With MEC, each one of these becomes an edge cloud provider, which appears to be similar to the AWS, Azure, Kubernetes, OpenStack, etc. ecosystem listed earlier, except for one critical point: *you can't just select one or two. As an application developer, you have to be able to work with all of these.*

The reasons become apparent if you think about your users: they have to be able to reach your cloud instances. However, while it is reasonable to expect that most of your users are able to reach AWS most of the time and also are able to reach your private cloud running OpenStack all the time, it is not reasonable to expect that customers of Operator 1 (Op1) are able to reach Operator 2's (Op2) edge cloud. And even if they are, Op2's edge cloud is NOT an edge cloud for Op1 customers. To reach it, their communication has to "leave" Op1's network, traverse the Internet, and then enter Op2's cloud. By this point, any edge benefits (proximity, low latency, minimization of network BW, etc.) are lost – in fact, the Op1 customers would be better off accessing a public cloud–hosted instance. We have illustrated this in Figure 1.4.

This creates a need for the application developers to be able to work with most of the CSPs in any geography where the application needs to be able to be present at the edge. However, for all but the largest application developers, such scaling is simply not feasible – and even

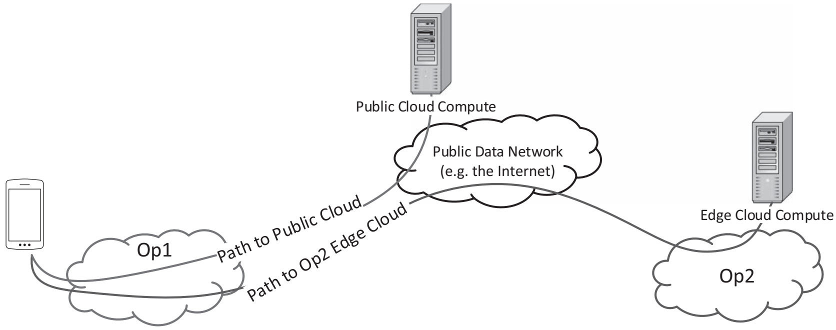

Figure 1.4 Illustrating paths to public and another operator's edge clouds.

where it is feasible, the economics involved are unfavorable. This is a problem that we will return to several times in this book, and so it will be useful to give it a short name – let's call it the Application Developers' Scaling Problem (ADSP). The ADSP has many aspects that need attention – how to set up the appropriate business relationships, specifying where to deploy applications, management of application instances, etc. One of the particular interests in this discussion is the technical problem – how can an application developer write its SW *once* and be assured that it will properly work *at every edge cloud*.

Fortunately, this problem is but a special case of a well-known problem in the communication industry – that of multi-vendor interoperability. Think, for example, of the various types of Wi-Fi devices made by various manufacturers and coming in various shapes and sizes working with Wi-Fi Access Points, also made by different manufacturers and coming in completely different shapes and sizes (from a home Wi-Fi router to Enterprise WLAN to Soft APs increasingly present in printers, etc.). And standards are the means used by the communication industry to address such a problem. When successful (and many standards, like many technologies are not), standards can enable tremendous growth of a new ecosystem, driving completely new applications and businesses. Witness the success of the IEEE 802.11 standard, which underlies Wi-Fi, or the 3GPP set of standards that have been the foundation of the global mobile communication industry since the days of GSM.

What is needed, therefore, to address the technical aspect of ADSP is a standard – more precisely, a standardized interface between the application and the MEC cloud. As we shall see further on ETSI's

MEC, standards define just such an interface. Moreover, ADSP is not the only aspect of MEC where multi-vendor interoperability is an issue, as we shall see in some of the subsequent chapters.

1.6 All We Need Is Open Source?

There is a sense in our community that while industry standard interface definitions are important, these do not have to come from traditional "standards." Instead, they can come from a development community, for example, from an Open Source project. Moreover, having these come from an Open Source project is better because the result is not just a document, but running code.

Let us agree on one point here: open source has had and is having a tremendous impact. Open source dramatically lowers barriers to entry into a market and in doing so enables small, nimble, and highly innovative companies to play on more equal footing with large established players. It does this by providing a foundation on which to build on, allowing a small team (or a large one) to focus only on those areas where they can provide value add. A dramatic example: TensorFlow allows developers to deploy a machine learning algorithm with somewhere around 20 lines of fairly straightforward Python code. This means that data companies no longer need to spend time and resources on developing the mechanics of neural network processing; rather, they can focus on where their value add is – data science.

Still, none of this means that open source serves the role that standards serve. In fact, *it does not!* To illustrate this point, let us pick on OpenStack. Not because it's bad – quite the opposite, because it is so good and so successful of an open source project, it is a good place to make a point. As anyone who has worked with OpenStack knows, No it does not – we develop to the API, but not the API itself.

- Which version of OpenStack you are developing and which APIs you need.
- Which OpenStack you are developing (the true open source one, RedHat, Mirantis, etc.)

Yes, they are *almost* the same – but *almost* is not quite the same as *the same*. And when you are a small company, having 100 edge clouds that present *almost* the same interfaces still leaves you with the challenge of scaling to integrate with 100 different – slightly, but still different – implementations. In other words, you are still missing *a standard*.

The simple truth here is that standards cannot replace open source – they don't tell you how to build anything, but open source cannot replace standards. In an ideal world, open source projects would utilize standardized interfaces in those areas where these are needed, that is, where large-scale inter-vendor interoperability needs are expected. This would, then, deliver the benefits of both worlds to the industry.

1.7 Looking Ahead … in More Ways than One

In this introductory chapter, we've tried to do a number of things: give you, the reader, a sense of why edge is important; define what edge is; and talk about the role of the various ecosystem players, including standards and open source. This was a high-level overview – the rest of the book delves much deeper into these topics and more – and it would have been both impossible and silly to try and write the whole book in the Introduction.

We also hope to have provided enough of a historical perspective to give you a sense that edge computing, in general, and MEC, specifically, is not a radical new idea out of the blue. Rather, it is a synthesis of several existing and long developing strands which came together to address an emerging need. This coming together in a timely manner is not fortuitous. It is driven by that very same timeliness, that is, by the maturity of technology to make edge computing economically feasible combined with an emergence of potential market needs.

Having said that, it would be wrong to allow you to walk away from the Introduction thinking that the major issues around edge computing have already been solved – that is far from the truth. As we shall see in some of the forthcoming chapters, what has been largely solved is the technical problem of how to stand up a single MEC site – that is, a single edge cloud at an access network, with some form of brake-out of access network traffic and some simple applications running on it. In other words, we – the broad Telco and cloud computing community – know how to build a proof-of-concept and a field trial.

Alas, once we go to production, we will have to manage thousands of small clouds. No one really knows how to do that. The Googles of the world manage tens of large clouds, not thousands of small ones. In many cases, these clouds will run workloads from entities that do not trust each other – in the formal definition of "trust," with some of these entities running SW applications that constitute critical, real-time regulated infrastructure – and others running stuff like games. How does that work? There are some potential answers, but no "best practices" – since there have been no "practices" at scale to select some "best" ones.

When these clouds do form components of critical infrastructure – or maybe just "important" infrastructure – how do they fail? What do we need to do to localize failures? Recall that IoT is a major use case for edge computing, which means edge computing will be pervasive in our lives. As many recent examples show, complex systems fail in complex ways that we do not understand and do so catastrophically more often than we think (see, e.g., Nicholas Taleb's discussion in books such as *Antifragile* [13]). Massively distributed computing makes these systems much more complex. If any of you, our readers, are looking for a challenging and impactful Ph.D. topic, this must be it.

Beyond failures, we also need to look at security – but more than just traditional cloud security. While we worry about access to our data in the "cloud" in the cyber-sense, that is, someone gaining access through user credentials theft, obtaining admin access into the management system, etc., with edge computing, we also need to worry about the physical access – quite literally, an unauthorized access to one of the many cloud sites. Prevention – while critical – can only go so far. With such a scale and most sites located in intrinsically less secure locations than a cloud provider's data center, this will happen. How can this be detected, how is sensitive data protected, and what are the recovery mechanisms?

At the same time, edge computing can be a tremendous asset in a security system. For example, Anycast IP routing is well known to be an effective mitigation strategy against Denial-of-Service (DoS) attacks – particularly, distributed DoS attacks. However, the effectiveness of Anycast-based DDoS mitigation depends on there being a highly distributed infrastructure – not just at the end points (presumably, a cloud provider can provide many end points on many compute nodes in a data center), but along the path to the end points as well. This is hard for physical reasons – eventually, the traffic must converge on one or a few physical sites where the cloud is. Not so with edge computing – the inherent scale and physical distribution of a large MEC network is a natural fit to an Anycast-based DDoS defense.

Then, there is the challenge of writing applications that take advantage of the edge. We know they should probably rely on RESTful microservices, but really not much more. From very practical approaches (see, e.g., Ref. [12]) to fundamental questions of the underlying power of distributed computation [14] (i.e., whether and how it can fully realize everything that a Turing machine can), the challenge of distributed cloud computing has not yet received extensive study – although the extensive existing work on distributed computing is bound to be relevant and perhaps the edge is where networking/computing paradigms such as ICN will find their true application. It is not an accident that Amazon named its serverless compute "Lambda functions" and that it is precisely "Lambda functions" that AWS Greengrass enables at the edge today.

The societal impact of the edge is not understood at all. From such mundane topics as business cases and strategies – which we shall address in some more details – to deeper issues of how pervasive edge cloud could impact society, very little has been studied and understood. However, it is clear that by putting flexible general-purpose computing at the edge and enabling the appropriate communication and management with it, MEC can have significant impact. From bringing cloud to the underserved areas of the world, to initiatives such as Sigfox's "Seconds to Save Lives" (https://sigfoxfoundation.org/seismic-alert/), MEC – often combined with IoT – can reshape our lives in ways that we cannot imagine today.

This brings us to the last point of this discussion – while some uses of MEC are going to happen because the society needs them, most need to be driven by solid business consideration. In short, all parties need to understand how they are going to make money. This is especially true because MEC needs to be huge so as to deliver on its promise. A dozen MEC sites in a downtown of a city is a pilot deployment, not a commercially viable enterprise. Such scale demands a huge amount of investment – and huge investment is unlikely to happen without an understanding of how a return on such investment is made. And even these business aspects of MEC are now well understood today, something which we will also explore in more depth further in this book.

Notes

1 https://en.wikipedia.org/wiki/Cloud_computing
2 Refer to, e.g., Refs. [1,2] for an excellent background on 4G networks.
3 This is the clear and commonly understood goal, even if the solutions presently offered are somewhat short of it.
4 Our mixing of VM (Virtual Machine) and container-based environments is intentional – for this discussion, it doesn't matter.
1 Note: the concept of network slice is not deeply discussed in this book, but the interested reader can find relevant resources in 3GPP specifications for 5G system architecture [70].
2 In principle, the MEC architecture [59] also permits stand-alone implementations, of course based on virtualized infrastructure, but not necessarily in NVF environments [60]. The second possibility is better clarified and defined in the second phase of MEC standardization [61].
3 See Vodafone's blog post: www.vodafone.com/content/index/what/technology-blog/video-streaming-experience.html#
4 See also Saguna website: www.saguna.net/blog/improving-video-streaming-customer-experience-with-multi-access-edge-computing-research-results-from-vodafone-and-saguna/
5 Interested readers can refer to the following 3GPP specifications, relevant to network slicing: TS 23.501 (on 5G System Architecture, ref. [70]), TS 22.261 (on 5G Requirements, Ref. [71]), and TS 28.531/28.532 (on 5G Slice Management, Ref. [72,73]).
6 In fact, according to 3GPP specifications, the packet delay budget (PDB) defines an upper bound for the time that a packet may be delayed between the UE and the UPF. This means that the N6 reference point toward the DN (thus until the MEC App) is not included, by definition, is this delay budget. On the contrary, E2E performances requirements should be determined by considering the overall path of user traffic packets, thus including N6 and the MEC app.
7 In TS 28.531, an NST is defined by 3GPP as a subset of attributes' values used for creation of instances of network slice information object class (IOC). The content of NST is not planned to be standardized by 3GPP, that is, it is defined by MNO and vendor.
8 www.gsma.com/futurenetworks/wp-content/uploads/2018/07/1_2_GSMA-Progress-of-5G-Network-Slicing_GSMA-NEST_vice-chair.pdf
1 www.zdnet.com/article/iot-devices-will-outnumber-the-worlds-population-this-year-for-the-first-time/
2 www.mckinsey.com/industries/telecommunications/our-insights/a-future-for-mobile-operators-the-keys-to-successful-reinvention

1 www.grandviewresearch.com/press-release/global-edge-computing-market
2 www.lfedge.org/projects/akraino/
3 https://telecominfraproject.com/edge-computing/
4 www.lanner-america.com/blog/multi-access-edge-computing-part-2-security-challenges-protecting-securing-mec/
5 www.qualcomm.com/media/documents/files/private-lte-networks.pdf
1 When talking about Internet of Things, some people also refer to Fog.
2 A careful reader may notice that sometimes, a company can be, in principle, categorized in more than one vertical. In addition to that, it may happen that companies expand their business and start focusing on areas where initially they were not present. For these reasons, the above categorization should not be considered as a limitation, but just as a tool to identify the main impacts on MEC coming from very heterogeneous requirements.
3 On the other hand, data transport saving is also one important advantage of edge computing. Nevertheless, this aspect should be considered more as a benefit from the point of view of the operator, which is in fact operating the communicating network, instead of the industry vertical.
1 For further deepening, Ref. [90] provides a detailed analysis of the improvements and cost savings enabled by this relatively recent data center strategy over the years.
2 NOTE: this movement is relatively recent; however, the reader should keep in mind that hyperscale computing is not just an American phenomenon; it is global, with leading players around the world (e.g., including companies well-known under the acronym of BAT, i.e., Baidu, Alibaba, and Tencent, driving significant innovation across Asia).
3 www.intel.com/content/www/us/en/architecture-and-technology/rack-scale-design-overview.html
4 More info for Cloud IoT Edge can be found here: https://cloud.google.com/iot-edge/
5 https://root-nation.com/gadgets-en/smartphones-en/en-ai-in-smartphones/
6 www.cloudcooler.co.uk/edge-computing
7 https://en.wikipedia.org/wiki/Business_Model_Canvas
1 https://en.wikipedia.org/wiki/Open_innovation
2 https://tmt.knect365.com/edge-computing-congress/etsi-mec-hackathon
3 https://edgegravity.ericsson.com/application-providers/
4 www.your-now.com/our-solutions/share-now
5 www.continental-corporation.com/en/press/press-releases/2019-03-21-car2mec-168160
6 https://aws.amazon.com/lambda/edge/
7 These should be intended as “smart” edge devices, of course. In fact, Greengrass requires at least 1 GHz of compute (either Arm or x86), 128 MB of RAM, plus additional resources for OS, message throughput, and AWS Lambda execution. According to Amazon, “Greengrass Core can run on devices that range from a Raspberry Pi to a server-level appliance”.

8 https://mecwiki.etsi.org/index.php?title=PoC_Framework
9 https://mecwiki.etsi.org/index.php?title=MEC_Hackathon_Framework
10 https://mecwiki.etsi.org/index.php?title=MEC_Deployment_Framework
11 www.etsi.org/newsroom/press-releases/1548-2019-02-etsi-multi-access-edge-computing-opens-new-working-group-for-mec-deployment
12 https://wiki.akraino.org/display/AK/MEC+API++Framework
13 https://wiki.openstack.org/wiki/Edge_Computing_Group

References

1. E. Dalhman, et al., *4G, LTE-Advanced-Pro and the Road to 5G*, 3rd Ed., Academic Press, Cambridge, MA, 2016.
2. M. Olsson, et al., *EPC and 4G Evolved Packet Networks*, 2nd Ed., Academic Press, Cambridge, MA, 2013.
3. J. Cartmell, et al., Local Selected IP Traffic Offload Reducing Traffic Congestion within the Mobile Core Network, in *Proceedings IEEE CCNC 2013*, Las Vegas, Nevada, USA, 2013.
4. SCF046, "Small Cell Services." Available at: http://scf.io/en/documents/046_Small_cell_services.php.
5. SCF084, "Small Cell Zone Services: RESTfule Bindings." Available at: http://scf.io/en/documents/084_-_Small_Cell_Zone_services_RESTful_Bindings.php.
6. SCF091, "Small Cell Application Programmer's Guide." Available at: http://scf.io/en/documents/091_-_Small_cell_application_programmers_guide.php.
7. SCF014, "Edge Computing Made Simple." Available at: http://scf.io/en/documents/014_-_Edge_Computing_made_simple.php.
8. NIST Special Publication 500-325, "Fog Computing Conceptual Model: Recommendations of the National Institute of Standards and Technology," 03/2018. Available at: https://doi.org/10.6028/NIST.SP.500–325.
9. NGMN, "5G White Paper," 2015. Available at: www.ngmn.org/fileadmin/ngmn/content/downloads/Technical/2015/NGMN_5G_White_Paper_V1_0.pdf.
10. ETSI, "Mobile Edge Computing: A Key Technology Towards 5G," 2015. Available at: www.etsi.org/images/files/ETSIWhitePapers/etsi_wp11_mec_a_key_technology_towards_5g.pdf.

11. ETSI, "MEC Deployments in 4G and Evolution Towards 5G," 2018. Available at: www.etsi.org/images/files/ETSIWhitePapers/etsi_wp24_MEC_deployment_in_4G_5G_FINAL.pdf.
12. ETSI, "Developing Software for Multi-Access Edge Computing," 2017. Available at: www.etsi.org/images/files/ETSIWhitePapers/etsi_wp20_MEC_SoftwareDevelopment_FINAL.pdf.
13. Nassim Nicholas Taleb, *Antifragile*, Random House, New Yok, 2014.
14. M. Sifalakis, et al., An Information Centric Network for Computing the Distribution of Computations, in *Proceedings ACM-ICN 2014*, Paris, France, 2014.
15. ETSI GS MEC 003, "Mobile Edge Computing (MEC); Framework and Reference Architecture," v. 1.1.1, 03/2016. Available at: www.etsi.org/deliver/etsi_gs/MEC/001_099/003/01.01.01_60/gs_MEC003v010101p.pdf.
16. ETSI GS MEC 011, "Mobile Edge Computing (MEC); Mobile Edge Platform Application Enablement," v. 1.1.1, 07/2017. Available at: www.etsi.org/deliver/etsi_gs/MEC/001_099/011/01.01.01_60/gs_MEC011v010101p.pdf.
17. ETSI GS MEC 009, "Mobile Edge Computing (MEC); General principles for Mobile Edge Service APIs," v. 1.1.1, 07/2017, Available at: www.etsi.org/deliver/etsi_gs/MEC/001_099/009/01.01.01_60/gs_MEC009v010101p.pdf.
18. ETSI GS MEC 012, "Mobile Edge Computing (MEC); Radio Network Information API," v. 1.1.1, 07/2017, Available at: www.etsi.org/deliver/etsi_gs/MEC/001_099/012/01.01.01_60/gs_MEC012v010101p.pdf.
19. ETSI GS MEC 013, "Mobile Edge Computing (MEC); Location API," v. 1.1.1, 07/2017, Available at: www.etsi.org/deliver/etsi_gs/MEC/001_099/013/01.01.01_60/gs_MEC013v010101p.pdf.
20. OMA-TS-REST-NetAPI-ZonalPresence-V1-0-20160308-C, "RESTful Network API for Zonal Presence."
21. OMA-TS-REST-NetAPI-ACR-V1-0-20151201-C, "RESTful Network API for Anonymous Customer Reference Management."
22. SCF084, "Small Cell Zone Services: RESTfule Bindings." Available at: http://scf.io/en/documents/084_-_Small_Cell_Zone_services_RESTful_Bindings.php.
23. SCF 152, "Small Cell Services API." Available at: http://scf.io/en/documents/152_-_Small_cell_services_API.php.
24. ETSI GS MEC 014, "Mobile Edge Computing (MEC); UE Identity API," v. 1.1.1, 02/2018. Available at: www.etsi.org/deliver/etsi_gs/MEC/001_099/014/01.01.01_60/gs_MEC014v010101p.pdf.
25. ETSI, "MEC in an Enterprise Setting: A Solution Outline," 2018. Available at: www.etsi.org/images/files/ETSIWhitePapers/etsi_wp30_MEC_Enterprise_FINAL.pdf.
26. ETSI GS MEC 015, "Mobile Edge Computing (MEC); Bandwidth Management API," v. 1.1.1, 07/2017. Available at: www.etsi.org/deliver/etsi_gs/MEC/001_099/015/01.01.01_60/gs_MEC015v010101p.pdf.

27. OpenStack, "Cloud Edge Computing: Beyond the Data Center." Available at: www.openstack.org/assets/edge/OpenStack-EdgeWhitepaper-v3-online.pdf (accessed Jan. 2019).
28. VmWare Project Dimension, www.vmware.com/products/project-dimension.html (accessed Jan. 2019).
29. ETSI GS MEC 010-1, "Mobile Edge Computing (MEC); Mobile Edge Management; Part 1: System, host and platform management," v. 1.1.1, 10/2017. Available at: www.etsi.org/deliver/etsi_gs/MEC/001_099/01001/01.01.01_60/gs_MEC01001v010101p.pdf.
30. ETSI GS MEC 010-2, "Mobile Edge Computing (MEC); Mobile Edge Management; Part 2: Application Lifecycle, Rules and Requirements Management," 07/2017. Available at: www.etsi.org/deliver/etsi_gs/MEC/001_099/01002/01.01.01_60/gs_MEC01002v010101p.pdf.
31. ETSI GS MEC 016, "Mobile Edge Computing (MEC); UE Application Interface," v. 1.1.1, 09/2017. Available at: www.etsi.org/deliver/etsi_gs/MEC/001_099/016/01.01.01_60/gs_MEC016v010101p.pdf.
32. ETSI GR MEC 017, "Mobile Edge Computing (MEC); Deployment of Mobile Edge Computing in an NFV Environment," v. 1.1.1, 02/2018. Available at: www.etsi.org/deliver/etsi_gr/MEC/001_099/017/01.01.01_60/gr_MEC017v010101p.pdf.
33. ETSI GS MEC-IEG 004, "Mobile-Edge Computing (MEC); Service Scenarios," v. 1.1.1, 11/2015. Available at: www.etsi.org/deliver/etsi_gs/MEC-IEG/001_099/004/01.01.01_60/gs_MEC-IEG004v010101p.pdf.
34. ETSI GS MEC 002, "Multi-access Edge Computing (MEC); Phase 2: Use Cases and Requirements," v. 2.1.1, 10/2018. Available at: www.etsi.org/deliver/etsi_gs/MEC/001_099/002/02.01.01_60/gs_MEC002v020101p.pdf.
35. Hewlett Packard Enterprise, "Edge Video Analytics. HPE Edgeline IoT Systems with IDOL Media Server Enable Exceptional Scene Analysis & Object Recognition Performance." Available at: https://support.hpe.com/hpsc/doc/public/display?docId=emr_na-c05336736&docLocale=en_US.
36. HPE, AWS, Saguna, "A Platform for Mobile Edge Computing." Available at: www.saguna.net/blog/aws-hpe-saguna-white-paper-platform-for-mobile-edge-computing/ (accessed Jan. 2019).
37. 3GPP TR 22.866, "Study on Enhancement of 3GPP Support for 5G V2X Services (Release 16)," v. 16.2.0, 12/2018.
38. ETSI GS MEC 022, "Multi-access Edge Computing (MEC); Study on MEC Support for V2X Use Cases," v. 2.1.1, 09/2018.
39. ETSI, "Cloud RAN and MEC: A Perfect Pairing," 2018. Available at: www.etsi.org/images/files/ETSIWhitePapers/etsi_wp23_MEC_and_CRAN_ed1_FINAL.pdf.
40. ETSI, "MEC Deployments in 4G and Evolution Towards 5G," 2018. Available at: www.etsi.org/images/files/ETSIWhitePapers/etsi_wp24_MEC_deployment_in_4G_5G_FINAL.pdf.
41. ETSI, "MEC in 5G Networks," 2018. Available at: www.etsi.org/images/files/ETSIWhitePapers/etsi_wp28_mec_in_5G_FINAL.pdf.

42. 3GPP TS 23.501 V15.3.0, “3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; System Architecture for the 5G System; Stage 2 (Release 15)”.
43. 3GPP TS 29.500 V15.1.0, “3rd Generation Partnership Project; Technical Specification Group Core Networks and Terminals; 5G System; Technical Realization of Service Based Architecture; Stage 3 (Release 15)”.
44. STL Partners, HPE, Intel, “Edge Computing, 5 Viable Telco Business Models,” 11/2017. Available at: https://h20195.www2.hpe.com/v2/getpdf.aspx/a00029956enw.pdf (accessed Jan. 2019).
45. “Report ITU-R M.2370-0, IMT Traffic Estimates for the Years 2020 to 2030,” 07/2015. Available at: www.itu.int/dms_pub/itu-r/opb/rep/r-rep-m.2370-2015-pdf-e.pdf.
46. “Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016–2021 White Paper,” 02/2017. Available at: www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html.
47. “Cisco Global Cloud Index: Forecast and Methodology, 2016–2021 White Paper,” 11/2018. Available at: www.cisco.com/c/en/us/solutions/collateral/service-provider/global-cloud-index-gci/white-paper-c11-738085.html.
48. “Ericsson Mobility Report,” 11/2018. Available at: www.ericsson.com/assets/local/mobility-report/documents/2018/ericsson-mobility-report-november-2018.pdf.
49. “Brian Krzanich, Talk at InterDrone,” Las Vegas, NV, September 6–8 2017. Available at: https://uavcoach.com/intel-ceo-interdrone/.
50. Recommendation ITU-R M.2083, “IMT Vision – Framework and Overall Objectives of the Future Development of IMT for 2020 and Beyond,” 09/2015.
51. ITU WP 5D, Draft new Report ITU-R M. [IMT-2020.TECH PERF REQ] – “Minimum Requirements Related to Technical Performance for IMT-2020 Radio Interface(s),” 02/2017.
52. B. Barani, “Network Technologies,” European Commission, Panel 5G: From Research to Standardisation, Austin, 8 December 2014.Available at: www.irisa.fr/dionysos/pages_perso/ksentini/R2S/pres/Bernard-EC-Panel-R2S-2014.pdf.
53. GSMA, “5G Spectrum – GSMA Public Policy Position,” 11/2018. Available at: www.gsma.com/spectrum/wp-content/uploads/2018/11/5G-Spectrum-Positions.pdf.
54. A. Manzalini, “Multi-access Edge Computing: Decoupling IaaS-PaaS for Enabling New Global Ecosystems,” Berlin, 19–20 September 2018. Edge Computing Congress, 2018.
55. ETSI ISG MEC, Available at: www.etsi.org/technologies/multi-access-edge-computing.
56. ETSI White Paper, “Mobile Edge Computing: A Key Technology Towards 5G,” 09/2015. Available at: www.etsi.org/images/files/ETSIWhitePapers/etsi_wp11_mec_a_key_technology_towards_5g.pdf.
57. ETSI ISG NFV, Available at: www.etsi.org/technologies/nfv

58. ETSI GS MEC 002 V1.1.1 (2016-03), "Mobile Edge Computing (MEC); Technical Requirements." Available at: www.etsi.org/deliver/etsi_gs/MEC/001_099/002/01.01.01_60/gs_MEC002v010101p.pdf.
59. ETSI GS MEC 003 V1.1.1 (2016-03), "Mobile Edge Computing (MEC); Framework and Reference Architecture." Available at: www.etsi.org/de-liver/etsi_gs/MEC/001_099/003/01.01.01_60/gs_MEC003v010101p.pdf.
60. ETSI MEC 017, "MEC in NFV Deployment." Available at: www.etsi.org/de-liver/etsi_gr/MEC/001_099/017/01.01.01_60/gr_mec017v010101p.pdf.
61. ETSI GS MEC 003 V2.1.1 (2019-01), "Multi-access Edge Computing (MEC); Framework and Reference Architecture." Available at: www.etsi.org/deliver/etsi_gs/MEC/001_099/003/02.01.01_60/gs_mec003v020101p.pdf.
62. 3GPP SA2, "Study on Enhancement of Support for Edge Computing in 5GC," S2–1902838- Study Item Description, 03/2019. Available at: www.3gpp.org.
63. 3GPP SA5, "Study on Management Aspects of Edge Computing," xx. Available at: www.3gpp.org/ftp/SPecs/archive/28_series/28.803/.
64. 3GPP SA6. "Study on Application Architecture for Enabling Edge Applications," S6–190238- Study Item Description, 01/2019. Available at: www.3gpp.org.
65. 5GAA White Paper, "Toward Fully Connected Vehicles: Edge Computing for Advanced Automotive Communications," 12/2017. Available at: http://5gaa.org/wp-content/uploads/2017/12/5GAA_T-170219-whitepaper-EdgeComputing_5GAA.pdf.
66. ETSI White Paper, "MEC Deployments in 4G and Evolution Towards 5G," 02/2018. Available at: www.etsi.org/images/files/ETSIWhitePapers/etsi_wp24_MEC_deployment_in_4G_5G_FINAL.pdf.
67. ETSI White Paper, "MEC in 5G Networks," 06/2018. Available at: www.etsi.org/images/files/ETSIWhitePapers/etsi_wp28_mec_in_5G_FINAL.pdf.
68. ETSI ISG MEC, DGR/MEC-0031 Work Item, "MEC 5G Integration." Available at: https://portal.etsi.org/webapp/workProgram/Report_Schedule.asp?WKI_ID=56729.
69. NGMN Alliance, "Description of Network Slicing Concept," 01/2016.
70. 3GPP TS 23.501 V15.4.0 (2018-12), "System Architecture for the 5G System." Available at: www.3gpp.org/ftp/specs/archive/23_series/23.501/.
71. 3GPP TS 22.261 V16.6.0 (2018-12), "Service Requirements for the 5G System." Available at: www.3gpp.org/ftp/specs/archive/22_series/22.261/.
72. 3GPP TS 28.531 V16.0.0 (2018-12), "Management and Orchestration; Provisioning." Available at: www.3gpp.org/ftp/Specs/Archive/28_series/28.531/.
73. 3GPP TS 28.532 V15.1.0 (2018-12), "Management and Orchestration; Generic Management Services." Available at: www.3gpp.org/FTP/Specs/archive/28_series/28.532/.

74. DGR/MEC-0024 Work Item, "MEC Support for Network Slicing." Available at: https://portal.etsi.org/webapp/WorkProgram/Report_WorkItem.asp?wki_id=53580.
75. GSMA Network Slicing Task Force, Available at: www.gsma.com/futurenetworks/technology/understanding-5g/network-slicing/.
76. STL Partners, HPE, Intel, "Edge Computing, 5 Viable Telco Business Models," 11/2017. Available at: https://h20195.www2.hpe.com/v2/getpdf.aspx/a00029956enw.pdf (accessed Jan. 2019).
77. "Chetan Sharma consulting, Operator's Dilemma (And Opportunity): The 4th Wave; White Paper," 2012. Available at: www.chetansharma.com/publications/operators-dilemma-and-opportunity-the-4th-wave/.
78. "Digital TV Research, Global OTT Revenue Forecasts," 11/2018. Available at: www.digitaltvresearch.com/ugc/press/245.pdf.
79. GSMA, Network 2020 Programme, "Unlocking Commercial Opportunities – From 4G Evolution to 5G," 02/2016. Available at: www.gsma.com/futurenetworks/wp-content/uploads/2016/02/704_GSMA_unlocking_comm_opp_report_v5.pdf.
80. 5G-ACIA, "5G Non-Public Networks for Industrial Scenarios," White Paper, 03/2019. Available at: www.5g-acia.org/index.php?id=6958.
81. Riot Research, "Connected Car Revenue Forecast 2017–2023," 02/2018. Available at: https://rethinkresearch.biz/wp-content/uploads/2018/02/Connected-Car-Forecast.pdf.
82. Cisco, "Cisco Visual Networking Index: Forecast and Trends, 2017–2022 White Paper." Available at: www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-741490.html (accessed Feb. 2018).
83. Allied Market Research, "Global Factory Automation Market Overview," 06/2018. Available at: www.alliedmarketresearch.com/factory-automation-market.
84. Zion Market Research, "Global Smart Manufacturing," 05/2018. Available at: www.zionmarketresearch.com/report/smart-manufacturing-market.
85. Markets and Markets, "eHealth Market," 03/2018. Available at: www.marketsandmarkets.com/Market-Reports/ehealth-market-11513143.html?gclid=EAIaIQobChMIsO3k5eG-4QIVj-F3Ch0v5Qf2EAAYASAAEgIDIfD_BwE.
86. IoT Analytics, "State of the IoT 2018: Number of IoT Devices Now at 7B – Market Accelerating," 08/2018. Available at: https://iot-analytics.com/state-of-the-iot-update-q1-q2-2018-number-of-iot-devices-now–7b/.
87. iGR White Paper, "The Business Case for MEC in Retail: A TCO Analysis and Its Implications in the 5G Era," 07/2017. Available at: https://networkbuilders.intel.com/blog/opportunities-for-multi-access-edge-computing-and-5g-in-retail-total-cost-of-ownership-analysis.
88. Ericsson Technology Review, "Distributed Cloud – A Key Enabler of Automotive and Industry 4.0 Use Cases," 11/2018. Available at: www.ericsson.com/en/ericsson-technology-review/archive/2018/distributed-cloud.

89. "Cisco Global Cloud Index: Forecast and Methodology, 2016–2021 White Paper," 11/2018. Available at: www.cisco.com/c/en/us/solutions/collateral/service-provider/global-cloud-index-gci/white-paper-c11-738085.html.
90. Intel White Paper, "Data Center Strategy Leading Intel's Business Transformation," 11/2017. Available at: www.intel.com/content/dam/www/public/us/en/documents/white-papers/data-center-strategy-paper.pdf.
91. D. Sabella, "5G Experimental Activities in Flex5Gware Project – Focus on RAN Virtualization," IEEE 5G Berlin Summit, Fraunhofer-Forum Berlin, 2 November 2016. Available at: www.5gsummit.org/berlin/docs/slides/Dario-Sabella.pdf.
92. D. Thanh, I. Jørstad, "The Mobile Phone: Its Evolution from a Communication Device to a Universal Companion," 2005. Available at: www.researchgate.net/publication/239608908_The_mobile_phone_Its_evolution_from_a_communication_device_to_a_universal_companion.
93. N. Islam, R. Want, "Smartphones: Past, Present, and Future," *IEEE Pervasive Computing*, vol. 13, issue 4, October–December 2014. Available at: https://ieeexplore.ieee.org/document/6926722.
94. P. Rubin, "Wired, Eye Tracking Is Coming to VR Sooner than You Think. What Now?" 23 March 2018. Available at: www.wired.com/story/eye-tracking-vr/.
95. R. Berger, "Digital Factories: The Renaissance of the U.S. Automotive Industry." Available at: www.rolandberger.com/en/Publications/pub_digital_factories.html.
96. Ericsson, Technology Trends 2018, "Five Technology Trends Augmenting the Connected Society," 2018. Available at: www.ericsson.com/en/ericsson-technology-review/archive/2018/technologytrends-2018.
97. ETSI GS MEC 002 V2.1.1 (2018-10), "Multi-access Edge Computing (MEC); Phase 2: Use Cases and Requirements." Available at: www.etsi.org/deliver/etsi_gs/MEC/001_099/002/02.01.01_60/gs_MEC002v020101p.pdf.
98. Intel White Paper, "Intel IT's Data Center Strategy for Business," 2014. Available at: www.insight.com/content/dam/insight-web/en_US/article-images/whitepapers/partner-whitepapers/intel-it-s-data-center-strategy-for-business-transformation.pdf.
99. EARTH Project Deliverable D2.1, "Economic and Ecological Impact of ICT." Available at: https://bscw.ict-earth.eu/pub/bscw.cgi/d38532/EARTH_WP2_D2.1_v2.pdf
100. G. Auer et al., "Cellular Energy Efficiency Evaluation Framework," Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd; 15–18 May 2011; Budapest, Hungary, 2011. ISSN: 1550–2252. Available at: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5956750.

101. D. Sabella et al., "Energy Efficiency Benefits of RAN-as-a-Service Concept for a Cloud-Based 5G Mobile Network Infrastructure," *IEEE Access*, vol. 2, December 2014; ISSN: 2169–3536, doi:10.1109/ACCESS.2014.2381215.
102. China Mobile Research Institute, "C-RAN The Road Towards Green RAN," White Paper, Version 2.5, 10/2011. Available at: https://pdfs.semanticscholar.org/eaa3/ca62c9d5653e4f2318aed9ddb8992a505d3c.pdf.
103. D. Rapone, D. Sabella, M. Fodrini, "Energy Efficiency Solutions for the Mobile Network Evolution Toward 5G: An Operator Perspective," The Fourth IFIP Conference on Sustainable Internet and ICT for Sustainability, SustainIT2015, Madrid, 15 April 2015. Available at: www.networks.imdea.org/sustainit2015/t-program.html.
104. D. Sabella et al., "Energy Management in Mobile Networks Towards 5G," In: M. Z. Shakir, M. A. Imran, K. A. Qaraqe, M.-S. Alouini, A. V. Vasilakos, (eds.), *Energy Management in Wireless Cellular and Ad-hoc Networks*, Volume 50 of the series Studies in Systems, Decision and Control, Springer, Switzerland, 2015, pp. 397–427.
105. ETSI GS MEC-IEG 005 V1.1.1 (2015-08), "Mobile-Edge Computing (MEC); Proof of Concept Framework." Available at: www.etsi.org/deliver/etsi_gs/MEC-IEG/001_099/005/01.01.01_60/gs_MEC-IEG005v010101p.pdf.
106. Akraino Edge Stack website, www.lfedge.org/projects/akraino/.
107. OSF Edge Computing Group website, www.openstack.org/edge-computing/.
108. 5G Infrastructure Public Private Partnership (5G PPP) website, https://5g-ppp.eu/.
109. 5G Automotive Association (5GAA) website, https://5gaa.org/.
110. ETSI ISG MEC, DGS/MEC-0030V2XAPI' Work Item "Mobile-Edge Computing (MEC); MEC V2X API" Available at: https://portal.etsi.org/webapp/WorkProgram/Report_WorkItem.asp?wki_id=54416.
111. L. Baltar, M. Mueck, D. Sabella, "Heterogeneous Vehicular Communications – Multi-Standard Solutions to Enable Interoperability", 2018 IEEE Conference on Standards for Communications and Networking (CSCN), Paris (France), 29-31 Oct. 2018. Available at: https://ieeexplore.ieee.org/document/8581726.