arXiv:1711.09123v1 [cs.DC] 24 Nov 2017

A Manifesto for Future Generation Cloud Computing: Research Directions for the Next Decade

Rajkumar Buyya*[1], Satish Narayana Srirama†[2,1], Giuliano Casale[3], Rodrigo Calheiros[4], Yogesh Simmhan[5], Blesson Varghese[6], Erol Gelenbe[3], Bahman Javadi[4], Luis Miguel Vaquero[7], Marco A. S. Netto[8], Adel Nadjaran Toosi[1], Maria Alejandra Rodriguez[1], Ignacio M. Llorente[9], Sabrina De Capitani di Vimercati[10], Pierangela Samarati[10], Dejan Milojicic[11], Carlos Varela[12], Rami Bahsoon[13], Marcos Dias de Assuncao[14], Omer Rana[15], Wanlei Zhou[16], Hai Jin[17], Wolfgang Gentzsch[18], Albert Zomaya[19], and Haiying Shen[20]

[1]University of Melbourne, Australia
[2]University of Tartu, Estonia
[3]Imperial College London, UK
[4]Western Sydney University, Australia
[5]Indian Institute of Science, India
[6]Queen's University Belfast, UK
[7]Dyson, UK
[8]IBM Research, Brazil
[9]Universidad Complutense de Madrid, Spain
[10]Universita degli Studi di Milano, Italy
[11]HP Labs, USA
[12]Rensselaer Polytechnic Institute, USA
[13]University of Birmingham, UK
[14]INRIA, France
[15]Cardiff University, UK
[16]Deakin University, Australia
[17]Huazhong University of Science and Technology, China
[18]UberCloud, USA
[19]University of Sydney, Australia
[20]University of Virginia, USA

November 28, 2017

Abstract

The Cloud computing paradigm has revolutionized the computer science horizon during the past decade and has enabled the emergence of computing as the fifth utility. It has captured significant attention of academia, industries and government bodies. Now, it has emerged as the backbone of modern economy by offering subscription-based services anytime, anywhere following a pay-as-you-go model. This has instigated (1) shorter establishment times for start-ups, (2) creation of scalable global enterprise applications, (3) better cost-to-value associativity for scientific and high performance computing applications, and (4) different invocation/execution models for pervasive and ubiquitous applications. The recent technological developments and paradigms such as serverless computing, software-defined networking, Internet of Things, and processing at network edge are creating new opportunities for Cloud computing. However, they are also posing several new challenges and creating the need for new approaches and research strategies, as well as the re-evaluation of the models that were developed to address issues such as scalability, elasticity, reliability, security, sustainability, and application models.

*Corresponding author; rbuyya@unimelb.edu.au
†Co-led this work with first author; Equal first author; Corresponding author; srirama@ut.ee

The proposed manifesto addresses them by identifying the major open challenges in Cloud computing, emerging trends and impact areas. It then offers research directions for the next decade, thus helping in the realisation of Future Generation Cloud Computing.

Keywords— Cloud computing, scalability, sustainability, InterCloud, data management, cloud economics, application development, Fog computing, serverless computing

1 Introduction

Cloud computing has shaped the way in which software and IT infrastructure are used by consumers and triggered the emergence of computing as the fifth utility [32]. Since its emergence, industry organizations, governmental institutions, and academia have embraced it and its adoption has seen a rapid growth. This paradigm has developed into the backbone of modern economy by providing on-demand access to subscription-based IT resources, resembling not only the way in which basic utility services are accessed but also the reliance of modern society on them. Cloud computing has enabled new businesses to establish in a shorter amount of time, has facilitated the expansion of enterprises across the globe, has accelerated the pace of scientific progress, and has led to the creation of various models of computation for pervasive and ubiquitous applications, among other benefits.

Up to now, there have been three main service models that have fostered the adoption of Clouds, namely Software, Platform, and Infrastructure as a Service (SaaS, PaaS and IaaS). SaaS offers the highest level of abstraction and allows users to access applications hosted in Cloud data centres (CDC), usually over the Internet. This, for instance, has allowed businesses to access software in a flexible manner and has allowed them to avoid incurring in expenses such as license fees and IT infrastructure maintenance. PaaS is tailored for users that require more control over their IT resources and offers a framework for the creation and deployment of Cloud applications that includes features such as programming models and auto-scaling. This has allowed developers to easily create applications that benefit from the elastic Cloud resource model for example. Finally, IaaS offers access to computing resources, usually by leasing Virtual Machines (VMs) and storage space. This layer is not only the foundation for SaaS and PaaS but has also been the pillar of Cloud computing. It has done so by enabling users to access the IT infrastructure they require only when they need it, to adjust the amount of resources used in a flexible way, and to pay only for what has been used, all while having a high degree of control over the resources.

1.1 Motivation and Goals of the Manifesto

Throughout the evolution of Cloud computing and its increasing adoption, not only have the aforementioned models advanced and new ones emerged, but also the technologies in which this paradigm is based (e.g., virtualization) have continued to progress. For instance, the use of novel virtualization techniques such as containers that enable improved utilization of the physical resources and further hide the complexities of hardware is becoming increasingly widespread, even leading to a new service model being offered by providers known as Container as a Service (CaaS). There has also been a rise in the type and number of specialized Cloud services that aid industries in creating value by being easily configured to meet specific business requirements. Examples of these are emerging, easy-to-use, Cloud-based data analytics services and serverless architectures.

Another clear trend is that Clouds are becoming increasingly geographically distributed to support emerging application paradigms. For example, Cloud providers have recently started extending their infrastructure and services to include edge devices for supporting emerging paradigms such as the Internet of Things (IoT) and Fog computing. Fog computing aims to move decision making operations as close to the data sources as possible by leveraging resources on the edge such as mobile base stations, gateways, network switches and routers thus reducing response time and network latencies. Additionally, as a way of fulfilling increasingly complex requirements that demand the composition of multiple services and as a way of achieving reliability and improving sustainability, services spanning across multiple geographically distributed CDCs have also become more widespread.

The adoption of Cloud computing will continue to increase and support for these emerging models and services is of paramount importance. In 2016, the IDG's Cloud adoption report found that 70% of organi-

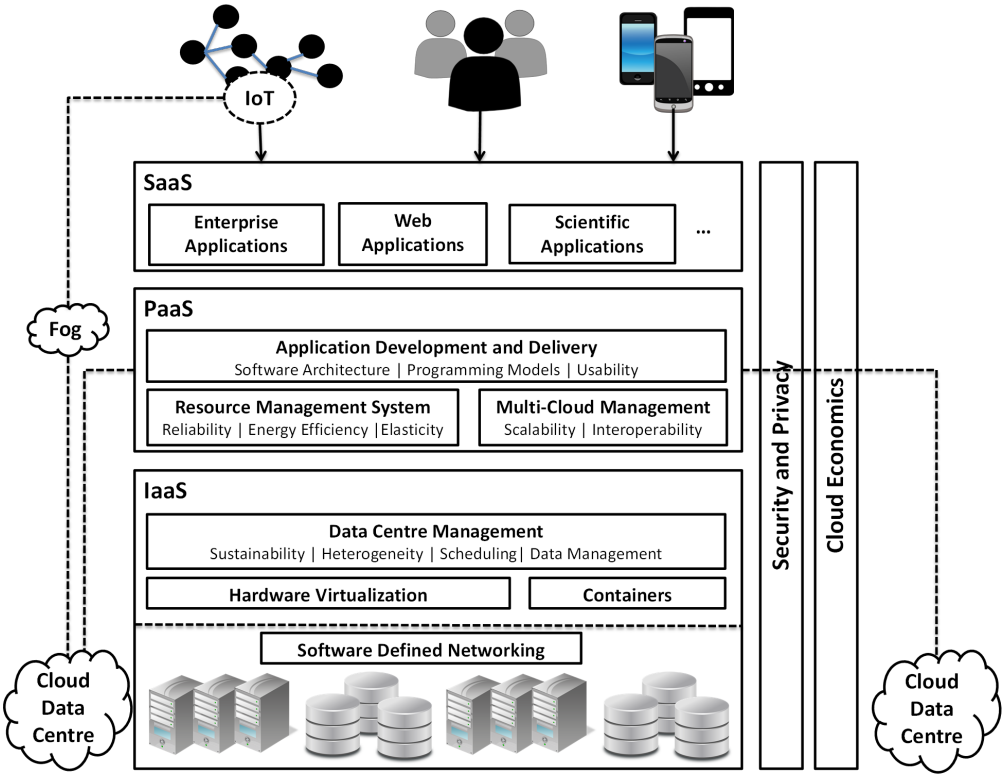

Figure 1: Components of the Cloud computing paradigm

zations have at least one of their applications deployed in the Cloud and that the numbers are growing [90]. In the same year, the IDC's (International Data Corporation) Worldwide Semiannual Public Cloud Services Spending Guide [89] reported that Cloud services were expected to grow from $70 billion in 2015 to more than $203 billion in 2020, an annual growth rate almost seven times the rate of overall IT spending growth. This extensive usage of Cloud computing in various emerging domains is posing several new challenges and is forcing us to rethink the research strategies and re-evaluate the models that were developed to address issues such as scalability, resource management, reliability, and security for the realisation of next-generation cloud computing environments [168].

This comprehensive manifesto brings these advancements together and identifies open challenges that need to be addressed for realising the *Future Generation Cloud Computing*. Given that rapid changes in computing/IT technologies in a span of 4-5 years are common, and the focus of the manifesto is for the next decade, we envision that identified research directions get addressed and will have impact on the next 2-3 generations of utility-oriented Cloud computing technologies, infrastructures, and their applications' services. The manifesto first discusses major challenges in Cloud computing, investigates their state-of-the-art solutions and identifies their limitations. The manifesto then discusses the emerging trends and impact areas, that further drive these Cloud computing challenges. Having identified these open issues, the manifesto then offers comprehensive future research directions in the Cloud computing horizon for the next decade. Figure 1 illustrates the main components of the Cloud computing paradigm and positions the identified trends and challenges, which are discussed further in the next sections.

The rest of the paper is organized as follows: Section 2 discusses the state-of-the-art of the challenges in Cloud computing and identifies open issues. Section 3 discusses the emerging trends and impact areas related to the Cloud computing horizon. Section 4 provides a detailed discussion about the future research directions to address the open challenges of Cloud computing. In the process, the section also mentions how the respective future research directions will be guided and influenced by the emerging trends. Section 5 provides a conclusion for the manifesto.

2 Challenges: State-of-the-Art and Open Issues

As Cloud computing became popular, it has been extensively utilized in hosting a wide variety of applications. It posed several challenges (shown within the inner ring in Figure 2) such as issues with sustainability, scalability, security, and data management. Over the past decade, these challenges were systematically addressed and the state-of-the-art in Cloud computing has advanced significantly. However, there remains several issues open as summarised in the outer ring of Figure 2. The rest of the section identifies and details the challenges in Cloud computing and their state-of-the-art, along with the limitations driving their future research.

2.1 Scalability and Elasticity

Cloud computing differs from earlier models of distributed computing such as grids and clusters, in that it promises virtually unlimited computational resources on demand. At least two clear benefits can be obtained from this promise: first, unexpected peaks in computational demand do not entail breaking service level agreements (SLAs) due to the inability of a fixed computing infrastructure to deliver users' expected quality of service (QoS), and second, Cloud computing users do not need to make significant up-front investments in computing infrastructure but can rather grow organically as their computing needs increase and only pay for resources as needed. The first (QoS) benefit of the Cloud computing paradigm can only be realized if the infrastructure supports *scalable* services, whereby additional computational resources can be allocated, and new resources have a direct, positive impact on the performance and QoS of the hosted applications. The second (economic) benefit can only be realized if the infrastructure supports *elastic* services, whereby allocated computational resources can *follow demand* and by dynamically growing and shrinking prevent over- and under-allocation of resources.

The research challenges associated to *scalable services* can be broken into hardware, middleware, and application levels. Cloud computing providers must embrace parallel computing *hardware* including multi-core, clusters, accelerators such as Graphics Processing Units (GPUs), and non-traditional (e.g., neuromorphic and future quantum) architectures, and they need to present such heterogeneous hardware to IaaS Cloud computing users in abstractions (e.g., VMs, containers) that while providing isolation, also enable performance guarantees. At the *middleware* level, programming models and abstractions are necessary, so that PaaS Cloud computing application developers can focus on functional concerns (e.g., defining *map* and *reduce* functions) while leaving non-functional concerns (e.g., scalability, fault-tolerance) to the middleware layer [94]. At the *application* level, new generic algorithms need to be developed so that inherent scalability limitations of sequential deterministic algorithms can be overcome; these include asynchronous evolutionary algorithms, approximation algorithms, and online/incremental algorithms (see e.g., [53]). These algorithms may trade off precision or consistency for scalability and performance.

Ultimately, the scalability of Cloud computing is limited by the scaling of its components: compute, storage, and interconnects. Computation has been limited by the end of scaling of both Moore's law (doubling the number of transistors every 1.5 year) and Dennard scaling ("the power use stays in proportion with area: both voltage and current scale (downward) with length"). As a consequence the new computational units do not scale anymore, nor does the power use scale. This directly influences the scaling of computation performance and cost of the Cloud. Research in new technologies, beyond CMOS (complementary metal-oxide-semiconductor), is necessary for further scaling. Similar is true for memory. DRAM (Dynamic random-access memory) is limiting the cost and scaling of existing computers and new non-volatile technologies are being explored that will introduce additional scaling of load-store operating memory while reducing the power consumption. Finally, the photonic interconnects are the third pillar that enables so called silicon photonics to propagate photonic connections into the chips improving performance, increasing scale, and reducing power consumption.

On the other hand, the research challenges associated to *elastic services* include the ability to accurately predict computational demand and performance of applications under different resource allocations [93, 155], the use of these workload and performance models in informing resource management decisions in middleware [95], and the ability of applications to scale up and down, including dynamic creation, mobility, and garbage collection of VMs, containers, and other resource abstractions [166]. While virtualization (e.g., VMs) has achieved steady maturity in terms of performance guarantees, ease of use of containers (especially

Figure 2: Cloud computing challenges, state-of-the-art and open issues

quick restarts) has garnered a lot of adoption from the developers community. Programming models that enable dynamic reconfiguration of applications significantly help in elasticity, by allowing middleware to move computations and data across Clouds, between public and private Clouds, and closer to edge resources as needed by future Cloud applications running over sensor networks such as the IoT.

2.2 Resource Management and Scheduling

The scale of modern CDCs has been rapidly growing and current CDCs contain computing and storage devices in the range of tens to hundreds of thousands, hosting complex Cloud applications and relevant data. This makes the adoption of effective resource management and scheduling policies important to achieve high scalability and operational efficiency.

Nowadays, IaaS providers mostly rely on either *static* VM provisioning policies, which allocate a fixed set of physical resources to VMs using bin-packing algorithms, or *dynamic* policies, capable of handling load variations through live VM migrations and other load balancing techniques [119]. These policies can either be reactive or proactive, and typically rely on knowledge of VM resource requirements, either user-supplied or estimated using monitoring data and forecasting.

Resource management methods are also important for platform and service providers to help managing the type and amount of resources allocated to distributed applications, containers, web-services and micro-services. Policies available at this level include for example: 1) auto-scaling techniques, which dynamically scale up and down resources based on current and forecasted workloads; 2) resource throttling methods, to handle workload bursts, trends, smooth auto-scaling transients, or control usage of preemptible VMs (e.g., micro VMs); 3) admission control methods, to handle peak load and prioritize workloads of high-value customers; 4) service orchestration and workflow schedulers, to compose and orchestrate workloads, possibly specialized for the target domain (e.g., scientific data workflows [118]), which make decisions based on their cost-awareness and the constraint requirements of tasks; 5) multi-Cloud load balancers, to spread the load of an application across multiple CDCs.

Even though the area of resource management and scheduling has spawned a large body of research [10, 116], several challenges and limitations remain. For example, existing management policies tend to be intolerant to inaccurate estimates of resource requirements, calling for studying novel trade-offs between policy optimality and its robustness to inaccurate workload information. Further, demand estimation and workload prediction methods can be brittle and it remains an open question whether Machine Learning (ML) and Artificial intelligence (AI) methods can fully address this shortcoming [35]. Another frequent issue is that resource management policies tend to focus on optimizing specific metrics and resources, often lacking a systematic approach to co-existence in the same environment of multiple control loops and cycles, multi-resource fairness, and holistic optimization across layers of the Cloud stack. Risks related to the interplay between security and resource management are also insufficiently addressed in current research work.

2.3 Reliability

Reliability is another critical challenge in Cloud computing environments. Data centres hosting Cloud computing consist of highly interconnected and interdependent systems. Because of their scale, complexity and interdependencies, Cloud computing systems face a variety of reliability related threats such as hardware failures, resource missing failures, overflow failures, network failures, timeout failures and flaws in software being triggered by environmental change. Some of these failures can escalate and devastatingly impact system operation, thus causing critical failures [76]. Moreover, a cascade of failures may be triggered leading to large-scale service disruptions with far-reaching consequences [97]. As organizations are increasingly interested in adapting Cloud computing technology for applications with stringent reliability assurance and resilience requirements [146], there is an urgent demand to develop new ways to provision Cloud services with assured performance and resilience to deal with all types of independent and correlated failures [51]. Moreover, the mutual impact of reliability and energy efficiency of Cloud systems is one of the current research challenges [172].

Although reliability in distributed computing has been studied before [133], standard fault tolerance and reliability approaches cannot be directly applied in Cloud computing systems. The scale and expected reliability of Cloud computing are increasingly important but hard to analyse due to the range

of inter-related characteristics, e.g. their massive-scale, service sharing models, wide-area network, and heterogeneous software/hardware components. Previously, independent failures have mostly been addressed separately, however, the investigation into their interplay has been completely ignored [74]. Furthermore, since Cloud computing is typically more service-oriented rather than resource-oriented, reliability models for traditional distributed systems cannot be directly applied to Cloud computing. So, existing state-of-the-art Cloud environments lack thorough service reliability models, automatic reliability-aware service management mechanisms, and failure-aware provisioning policies.

2.4 Sustainability

Sustainability is the greatest challenge of our century, and it also affects ICT in general [70], which consumes today close to 10% of all electricity consumed world-wide, together with a CO_2 impact that is comparable to that of air-travel. In addition to the energy consumed to operate ICT systems, we know that electronic components are manufactured, and then decommissioned after the end of their useful life-time, with an amount of energy which is 4-5 fold greater than what they consume to operate during their lifetime.

CDC deployments until recently have mainly focused at high performance and have not paid much attention to energy consumption. Thus, today a typical CDC's energy consumption is similar to that of 25,000 households [103]. The total number of operational CDCs worldwide are 8.5 million in 2017 according to IDC. According to Greenpeace, Cloud computing worldwide consumes more energy than most countries and only the four largest economies (USA, China, Russia, and Japan) surpass Clouds in their annual electricity usage. As the energy cost is rapidly increasing, now much research has gone into minimising the amount of energy consumed by Clouds, information systems that exploit Cloud systems, and networks [132, 28].

On the other hand, networks and the Cloud also have a huge potential to save energy in many areas such as smart cities, or to be used to optimise the mix of renewable and non-renewable energy worldwide [148]. However, the energy consumption of Clouds cannot be viewed independently of the QoS that they provide, so that both energy and QoS must be managed jointly. Indeed, for a given computer and network technology, reduced energy consumption is often coupled with a reduction of the QoS that users will experience. In some cases, such as critical or even life-threatening real-time needs, such as Cloud support of search and rescue operations, hospital operations or emergency management, a Cloud cannot choose to save energy in exchange for reduced QoS.

Current Cloud systems and efforts have in the past primarily focused on consolidation of VMs for minimising energy consumption of servers [20]. But other elements of CDC infrastructures, such as cooling systems (close to 35% of energy) and networks, which must be very fast and efficient, also consume significant energy that needs to be optimised by proper scheduling of the traffic flows between servers (and over high-speed networks) inside the data centre [72].

Because of multi-core architectures, novel hardware based sleep-start controls and clock speed management techniques, the power consumption of servers increasingly depends, and in a non-linear manner, on their instantaneous workload. Thus new ML based methods have been developed to dynamically allocate tasks to multiple servers in a CDC or in the Fog [174] so that a combination of violation of SLA, which are costly to the Cloud operator and inconvenient for the end user, and other operating costs including energy consumption, are minimised. Holistic techniques must also address the QoS effect of networks such as packet delays on overall SLA, and the energy effects of networks for remote access to CDC [173]. The purpose of these methods is to provide online automatic, or autonomic and self-aware methods to holistically manage both QoS and energy consumption of Cloud systems.

Recent work [183] has also shown that deep learning with neural networks can be usefully applied in experimental but realistic settings so that tasks are allocated to servers in a manner which optimised a prescribed performance profile that can include execution delays, response times, system throughput, and energy consumption of the CDC. Another approach that maximises the sustainability of Cloud systems and networks involves rationing the energy supply [69] so that the CDC can modulate its own energy consumption and delivered QoS in response, dynamically modifying the processors' variable clock rates as a function of the supply of energy. It has also been suggested that different sources of renewable and non-renewable energy can be mixed [71].

2.5 Heterogeneity

Public Cloud infrastructure has constantly evolved in the last decade. This is because service providers have increased their offerings while continually incorporating state-of-the-art hardware to meet customer demands and maximise performance and efficiency. This has resulted in an inherently heterogeneous Cloud with heterogeneity at three different levels.

The first is at the VM level, which is due to the organisation of homogeneous (or near homogeneous; for example, same processor family) resources in multiple ways and configurations. For example, homogeneous hardware processors with N cores can be organised as VMs with any subset or multiples of N cores. The second is at the vendor level, which is due to employing resources from multiple Cloud providers with different hypervisors or software suites. This is usually seen in multi-Cloud environments [111]. The third is at the hardware architecture level, which is due to employing both CPUs and hardware accelerators, such as GPUs and Field Programmable Gate Arrays (FPGAs) [149].

The key challenges that arise due to heterogeneity are twofold. The first challenge is related to resource and workload management in heterogeneous environments. State-of-the-art in resource management focuses on static and dynamic VM placement and provisioning using global or local scheduling techniques that consider network parameters and energy consumption [44]. Workload management is underpinned by benchmarking techniques that are used for workload placement and scheduling techniques. Current benchmarking practices are reasonably mature for the first level of heterogeneity and are developing for the second level [99, 167]. However, significant research is still required to predict workload performance given the heterogeneity at the hardware architecture level. Despite advances, research in both heterogeneous resource management and workload management on heterogeneous resources remain fragmented since they are specific to their level of heterogeneity and do not work across the VM, vendor and hardware architecture levels. It is still challenging to obtain a general purpose Cloud platform that integrates and manages heterogeneity at all three levels.

The second challenge is related to the development of application software that is compatible with heterogeneous resources. Currently, most accelerators require different (and sometimes vendor specific) programming languages. Software development practices for exploiting accelerators for example additionally require low level programming skills and has a significant learning curve. For example, CUDA or OpenCL are required for programming GPUs. This gap between hardware accelerators and high-level programming makes it difficult to easily adopt accelerators in Cloud software. One open challenge in this area is developing software that is agnostic of the underlying hardware and can adapt based on the available hardware [101].

2.6 Interconnected Clouds

Federated Cloud computing is considered as the next step in the evolution of Cloud computing and an integral part of the new emerging Edge and Fog computing architectures. The federated Cloud model is gaining increasing interest in the IT market, since it can bring important benefits for companies and institutions, such as resource asset optimization, cost savings, agile resource delivery, scalability, high availability and business continuity, and geographic dispersion [31].

"Different Cloud federation types such as Cloud bursting, Cloud brokering, or Cloud peering have been proposed to provide the necessary mechanisms for sharing computing, storage, and networking resources" [123]. Companies are seeking to offload as much of their applications as possible to the public Clouds, driven not only by the economic benefits and shared resources, but also due to the potential freedom to choose among multiple vendors on their terms.

However, although interconnection of Clouds was one of the earliest research problems that was identified in Cloud computing [31, 140, 21], Cloud interoperation continues to be an open issue since the field has rapidly evolved over the last half decade. Cloud providers and platforms still operate in siloes, and their efforts for integration usually target their own portfolio of services. Cloud interoperation should be viewed as the capability of public Clouds, private Clouds, and other diverse systems to understand each other's system interfaces, configurations, forms of authentication and authorization, data formats, and application initialization and customization [152].

Existing public Cloud providers offer proprietary mechanisms for interoperation that exhibit important limitations as they are not based on standards and open-source, and they do not interoperate with other providers. Although there are multiple efforts for standardization, such as Open Grid Forum's (OGF)

Open Cloud Computing Interface (OCCI), Storage Networking Industry Association's (SNIA) Cloud Data Management Interface (CDMI), Distributed Management Task Force's (DMTF) Cloud Infrastructure Management Interface (CIMI), DMTF's Open Virtualization Format (OVF), IEEE's InterCloud and National Institute of Standards and Technology's (NIST) Federated Cloud, the interfaces of existing Cloud services are not standardized and different providers use different APIs, formats and contextualization mechanisms for comparable Cloud services.

State-of-the-art projects such as Aneka [27], on the other hand, have developed middleware and library solutions for integration of different resources (VMs, databases, etc.). However, the problem of such approaches is that they need to operate in the lowest common denominator among the services offered by each provider, and this leads to suboptimal Cloud applications or support at specific service models.

Interoperability and portability have multiple aspects and relate to a number of different components in the architecture of Cloud computing and data centres, each of which needs to be considered in its own right. These include standard interfaces, portable data formats and applications, and internationally recognized standards for service quality and security. The efficient and transparent provision, management and configuration of cross-site virtual networks to interconnect the on-premise Cloud and the external provider resources is still an important challenge that is slowing down the full adoption of this technology [88].

As Cloud adoption grows and more applications are moved to the Cloud, the need for satisfactory InterCloud solutions is likely to grow. Challenges in this area concern how to go beyond the minimum common denominator of services when interoperating across providers (and thus enabling richer Cloud applications); how to coordinate authorization, access, and billing across providers; and how to apply InterCloud solutions in the context of Fog computing and other emerging trends.

2.7 Empowering Resource-Constrained Devices

Cloud services are relevant not only for enterprise applications, but also for the resource constrained devices and their applications. With the recent innovation and development, mobile devices such as smartphones and tablets, have achieved better CPU and memory capabilities. They also have been integrated with a wide range of hardware and sensors such as camera, GPS (Global Positioning System), accelerometer etc. In addition, with the advances in 4G, 5G and ubiquitous WiFi, the devices have achieved significantly higher data transmission rates. This progress has led to the usage of these devices in a variety of applications such as mobile commerce, mobile social networking and location based services. While the advances in the mobiles are significant and they are also being used as service providers, they still have limited battery life and when compared to desktops have limited CPU, memory and storage capacities, for hosting/executing resource-intensive tasks/applications. These limitations can be addressed by harnessing external Cloud resources, which led to the emergence of Mobile Cloud paradigm.

Mobile Cloud has been studied extensively during the past years and the research mainly focused at two of its binding models, the *task delegation* and the *mobile code offloading* [62]. With the task delegation approach, the mobile invokes web services from multiple Cloud providers, and thus faces issues such as Cloud interoperability and requirement of platform specific API. Task delegation is accomplished with the help of middlewares [62]. Mobile code offloading, on the other hand, profiles and partitions the applications, and the resource-intensive methods/operations are identified and offloaded to surrogate Cloud instances (Cloudlets/swarmlets). Typical research challenges here include developing the ideal offloading approach, identifying the resource-intensive methods, and studying ideal decision mechanisms considering both the device context (e.g. battery level and network connectivity) and Cloud context (e.g. current load on the Cloud surrogates) [61, 185]. While applications based on task delegation are common, mobile code offloading is still facing adaptability issues [61].

Correspondingly, IoT has evolved as *"web 4.0 and beyond"* and *"Industry 4.0"*, where physical objects with sensing and actuator capabilities, along with the participating individuals, are connected and communicate over the internet [154]. There are predictions that billions of such devices/*things* will be connected using advances in building innovative physical objects and communication protocols [58]. Cloud primarily helps IoT by providing resources for the storage and distributed processing of the acquired sensor data, in different scenarios. While this *Cloud-centric IoT* model [154, 75] is interesting, it ends up with inherent challenges such as network latencies for scenarios with sub-second response requirements. An additional aspect that arises with IoT devices is their substantial energy consumption, which can be mitigated by the

use of renewable energy [71], but this in turn raises the issue of QoS as the renewable energy sources are generally sporadic. To address these issues and to realize the IoT scenarios, Fog computing is emerging as a new trend to bring computing and system supervisory activities closer to the IoT devices themselves, which is discussed in detail in section 3.2.

2.8 Security and Privacy

Security is a major concern in ICT systems and Cloud computing is no exception. Here, we provide an overview of the existing solutions addressing problems related to the secure and private management of data and computations in the Cloud (confidentiality, integrity, and availability) along with some observations on their limitations and challenges that still need to be addressed.

With respect to the confidentiality, existing solutions typically, encrypt the data before storing them at external Cloud providers (e.g., [79]). Encryption, however, limits the support of query evaluation at the provider side. Solutions addressing this problem include the development of *encrypted database systems* supporting SQL queries over encrypted data (e.g., [9, 136]) and the definition of *indexes* associated with the encrypted data that can be used for partially evaluating queries directly on them (e.g., [4, 46, 79]). However, indexes should be balancing both precision and privacy. Precise indexes offer efficient query execution, but may lead to improper exposure of confidential information. An alternative solution is to use *encryption functions* that support keyword searches at the Cloud provider side over encrypted data, without compromising confidentiality.

Another interesting problem related to the confidentiality and privacy of data arises when considering modern Cloud-based applications (e.g., applications for accurate social services, better healthcare, detecting fraud, and national security) that explore data over multiple data sources with cross-domain knowledge. A major challenge of such applications is to preserve privacy, as data mining tools with cross-domain knowledge can reveal more personal information than anticipated, therefore prohibiting organizations to share their data. A research challenge is the design of theoretical models and practical mechanisms to preserve privacy for cross-domain knowledge [187]. Furthermore, the data collected and stored in the Cloud (e.g., data about the techniques, incentives, internal communication structures, and behaviours of attackers) can be used to verify and evaluate new theory and technical methods (e.g., [82, 160]). A current booming trend is to use ML methods in information security and privacy to analyse Big Data for threat analysis, attack intelligence, virus propagation, and data correlations [81].

Many approaches protecting the confidentiality of data rely on the implicit assumption that any authorized user, who knows the decryption key, can access the whole data content. However, in many situations there is the need of supporting *selective visibility* for *different users*. Works addressing this problem are based on *selective encryption* and on *attribute-based encryption* (ABE) [171]. Policy updates are supported, for example, by *over-encryption*, which however requires the help of the Cloud provider, and by the *Mix&Slice* approach [13], which departs from the support of the Cloud provider and uses different rounds of encryption to provide complete mixing of the resource. The problem of selective sharing has been considered also in scenarios where different parties cooperate for sharing data and to perform distributed computations.

Alternative solutions to encryption have been adopted when associations among the data are more sensitive than the data themselves [41]. Such solutions split data in different fragments stored at different servers or guaranteed to be non linkable. They support only certain types of sensitive constraints and queries and the computational complexity for retrieving data increases.

While all solutions described above successfully provide efficient and selective access to outsourced data, they are exposed to attacks exploiting frequency of accesses to violate data and users privacy. This problem has been addressed by *Private Information Retrieval* (PIR) techniques, which operate on publicly available data, and, more recently by *privacy-preserving indexing techniques* based on, for example, Oblivious RAM, B-tree data structures, and binary search tree [54]. This field is still in its infancy and the development of practical solutions is an open problem.

With respect to the integrity, different techniques such as digital signatures, Provable Data Possession, Proof Of Retrievability, let detecting unauthorized modifications of data stored at an external Cloud provider. Verifying the integrity of stored data by its owner and authorized users is, however, only one of the aspects of integrity. When data can change dynamically, possibly by multiple writers, and queries need to be supported, several additional problems have to be addressed. Researchers have investigated the use of authenticated

data structures (*deterministic* approaches) or insertion of integrity checks (*probabilistic* approaches) [49] to verify the correctness, completeness, and freshness of a computation. Both deterministic and probabilistic approaches can represent promising directions but are limited in their applicability and integrity guarantees provided.

With respect to the availability, some proposals have focused on the problem of how a user can select the services offered by a Cloud provider that match user's security and privacy requirements (e.g., [47]). Typically, the expected behaviours of Cloud providers are defined by SLAs stipulated between a user and the Cloud provider itself. Recent proposals have addressed the problem of exploring possible dependencies among different characteristics of the services offered by Cloud providers [50]. These proposals represent only a first step in the definition of a comprehensive framework that allow users to select the Cloud provider that best fits their needs, and to verify that providers offer services fully compliant with the signed contract.

Advanced *cyberattacks* in the Cloud domain represent a serious threat that may affect the confidentiality, integrity, and availability of data and computations. In particular, the Advanced Persistent Threats (APTs) defers a particular mention. This is an emerging class of cyberattacks that are goal-oriented, highly-targeted, well-organized, well-funded, technically-advanced, stealthy, and persistent. The notorious Stuxnet, Flame, and Red October are some examples of APTs. The APTs poses a severe threat to the Cloud computing domain, as APTs have special characteristics that can disable the existing defence mechanisms of Cloud computing such as antivirus, firewall, intrusion detection, and antivirus [180]. Indeed, APT-based cyber breach instances and cybercrime activities have recently been on the rise, and it has been predicted that a 50% increase in security budgets will be observed to rapidly detect and respond to them [26]. In this context, enhancing the technical levels of cyber defence only is far from being enough [63]. To mitigate the loss caused by APTs, practicable APT-targeting security solutions must be developed.

2.9 Economics of Cloud Computing

Research themes in Cloud economics have centered on a number of key aspects over recent years: (1) pricing of Cloud services – i.e. how a Cloud provider should determine and differentiate between different capabilities they offer, at different price bands and durations (e.g. micro, mini, large VM instances); (2) brokerage mechanisms that enable a user to dynamically search for Cloud services that match a given profile within a predefined budget; (3) monitoring to determine if user requirements are being met, and identifying penalty (often financial) that must be payed by a Cloud provider if values associated with pre-agreed metrics have been violated. The last of these has seen considerable work in the specification and implementation of SLAs, including implementation of specifications such as WS-Agreement.

SLA is traditionally a business concept, as it specifies contractual financial agreements between parties who engage in business activities. [60] observed that up to three SLA parameters (performance, memory, and CPU cycle) are often used. SLA management also relates to the supply and demand of computational resources, instances and services [30, 25]. A related area of *policy-based approaches* is also studied extensively [34]. Policy-based approaches are effective when resource adaptation scenarios are limited in number. As the number of encoded policies grow, these approaches can be difficult to scale. Various optimisation strategies have been used to enable SLA and policy-based resource enforcement.

Another related aspect in Cloud economics has been an understanding of how an organisation migrates current (in-house or externally hosted) infrastructure to Cloud providers, involving the migration of an in-house IT department to a Cloud provider. Migration of existing services needs to take account of both social and economic aspects of how Cloud services are provisioned and subsequently used, and risk associated with uptime and availability of (often) business critical capability. The above context has also been changed with interest in new implementation technologies – such as sub-second billing made possible through container-based deployments (often also referred to as "serverless computing"), such as in Google "functions", AWS Lambda, amongst others. Serverless computing is discussed further in section 3.4.

Licensing is another economics-related issue, which can include annual or perpetual licensing. These can be restrictive for Cloud resources (e.g. not on-demand, limited number of cores, etc.) when dealing with the demands of large business and engineering simulations for physics, manufacturing, etc. ISVs are currently investigating more suitable licensing models for the Cloud (e.g Abaqus; COMSOL; STAR-CCM).

Another challenge in Cloud economics is related to choosing the right Cloud provider. Comparing offerings between different Cloud providers is time consuming and often challenging, as providers do not use the same

terminology when offering computational and storage resources, making a like-for-like comparison difficult. A number of commercial and research grade platforms have been proposed to investigate benefit/limits of Cloud selection, such as PlanForCloud[1], CloudMarketMaker [98], pricing tools from particular providers (e.g. Amazon Cost Calculator[2], and SMI (Service Measurement Index) for ranking Cloud services [68]. Such platforms focus on what the user requires and hide the internal details of the Cloud provider's resource specifications and pricing models. In addition, marketplace models are also studied where users purchase services from SaaS providers that in turn procure computing resources from either PaaS or IaaS providers [8].

2.10 Application Development and Delivery

Cloud computing empowers application developers with the ability to programmatically control infrastructure resources and platform. Several benefits have emerged from this feature, such as the ability to couple the application with auto-scaling controllers and to embed in the code advanced self-* mechanisms for organizing, healing, optimizing, and securing the Cloud application at runtime.

A key benefit of resource programmability is a looser boundary between development and operation, which results in the ability to accelerate the delivery of changes to the production environment. To support this feature, a variety of agile delivery tools and Cloud standards (e.g., TOSCA - Topology and Orchestration Specification for Cloud Applications) are increasingly adopted during Cloud application development [18]. They support release lifecycle automation, including continuous integration, configuration, and testing, among others.

In terms of platform programmability, separation of concerns has helped in tackling the complexity of software development for the Cloud and in accelerating delivery. For example, MapReduce enables application developers to specify functional components of their application, namely *map* and *reduce* functions on their data; while enabling the middleware layers (e.g., Hadoop, Storm) to deal with non-functional concerns, such as parallelization, data locality optimization, and fault-tolerance.

A common issue faced both at resource and platform level is the development of novel methods to cope with the even increasing heterogeneity of the Cloud platforms. For example, in Edge computing, the effort to split applications relies entirely on the developers [39]. Even recent efforts in this area are not fully automated [102]. Even though it is expected that there will be a wide variety and large number of edge devices and applications, there is a shortage of application delivery frameworks and programming models to deliver software spanning both the Edge and the CDC.

Moreover, the accelerated pace of Cloud application delivery allowed by continuous delivery tools and novel programming abstractions increases on the downside the technical debt, i.e., the additional work required at later stages to fix shortcuts and suboptimal decisions taken at Cloud application design. This is because with agile delivery these issues can be fixed in future releases of the Cloud application.

2.11 Data Management

One of the key selling points of Cloud computing is the availability of affordable, reliable and elastic storage, that is collocated with the computational infrastructure. This offers a diverse suite of storage services to meet most common enterprise needs while leaving the management and hardware costs to the IaaS service provider. They also offer reliability and availability through multiple copies that are maintained transparently, along with disaster recovery with storage that can be replicated in different regions. A number of storage abstractions are also offered to suit a particular application need, with the ability to acquire just the necessary quantity and pay for it. File-based storage (Amazon Simple Storage Service (S3), Azure File), block storage services (Azure Blob, Amazon Elastic Block Store (EBS)) of a disk volume, and logical HDD (Hard Disk Drive) and SSD (Solid-state Drive) disks that can be attached to VMs are common ones. Besides these, higher level data platforms such as NoSQL columnar databases, relational SQL databases and publish-subscribe message queues are available as well.

At the same time, there has been a proliferation of Big Data platforms [109] running on distributed VM's collocated with the data storage in the data centre. The initial focus has been on batch processing and NoSQL query platforms that can handle large data volumes from web and enterprise workloads, such as

[1] https://www.planforcloud.com/

[2] https://calculator.s3.amazonaws.com/index.html

Apache Hadoop, Spark and HBase. However, fast data platforms for distributed stream processing such as Apache Storm, Heron, and Apex have grown to support data from sensors and Internet-connected devices. PaaS offerings such as Amazon ElasticMR, Kinesis, Azure HDInsight and Google Dataflow are available as well.

While there has been an explosion in the data availability over the last decade, and along with the ability to store and process them on Clouds, many challenges still remain. Services for data storage have not been adequately supported by services for managing their metadata that allows data to be located and used effectively [124]. Data security and privacy remain a concern (discussed further in section 2.8), with regulatory compliance being increasingly imposed by various governments, as well as leakages due to poor data protection by users. Data is increasingly being sourced from the edge of the network as IoT device deployment grows, and the latency of wide area networks inhibits their low-latency processing. Edge and Fog computing may hold promise in this respect [170].

Even within the data centre, network latencies and bandwidth between VMs, and from VM to storage can be variable, causing bottlenecks for latency-sensitive stream processing and bandwidth-sensitive batch processing platforms. Solutions such as Software Defined Networking (SDN) and Network Functions Virtualization (NFV), which can provide mechanisms required for allocating network capacity for certain data flows both within and across data centres with certain computing operations been performed in-network are needed [113]. Better collocation guarantees of VMs and data storage may be required as well.

There is also increasing realization that a lambda architecture that can process both data at rest and data at motion together is essential [106]. Big Data platforms such as Apache Flink and Spark Streaming are starting to offer early solutions but further investigation is required [186]. Big Data platforms also have limited support for automated scaling out and in on elastic Clouds, and this feature is important for long-running streaming applications with dynamic workloads [108]. While the resource management approaches discussed above can help, these are yet to be actively integrated within Big Data platforms. Fine-grained per-minute billing along with faster VM acquisition time, possibly using containers, can help shape the resource acquisition better. In addition, composing applications using serverless computing can also off-load the resource allocation mechanism to the Cloud platform provider.

2.12 Networking

Cloud data centres are the backbone of Cloud services where application components reside and where service logic takes place for both internal and external users. Successful delivery of Cloud services requires many levels of communication happening within and across data centres. Ensuring that this communication occurs securely, seamlessly, efficiently and in a scalable manner is a vital role of the network that ties all the service components together.

During the last decade, there has been many network based innovations and research that have explicitly explored Clouds networking. However, despite these many advances, there are still many networking challenges that need to be addressed.

One of the main concerns of today's CDCs is their high energy consumption. Nevertheless, the general practice in many data centres is to leave all networking devices always on [84]. In addition, unlike computing servers, the majority of network elements such as switches, hubs, and routers are not designed to be energy proportional and things such as, sleeping during no traffic and adaptation of link rate during low traffic periods, are not a native part of the hardware [117]. Therefore, the design and implementation of methodologies and technologies to reduce network energy consumption and make it proportional to the load remain as open challenges.

Another challenge with CDC networks is related to providing guaranteed QoS. The SLAs of today's Clouds are mostly centered on computation and storage [77]. No abstraction or mechanism enforcing the performance isolation and hence no SLAs beyond best effort is available to capture the network performance requirements such as delay and bandwidth guarantees. Guo et al. [77] propose a network abstraction layer called VDC which works based on a source routing technique within the data centre infrastructure to provide bandwidth guarantees for VMs. Yet, their method does not provide any network delays guarantee.

In addition, Cloud networking is not a trivial task and modern CDCs face similar challenges to building the Internet due to their size [12]. The highly virtualized environment of a CDC is also posing issues that have always existed within network apart from new challenges of these multi-tenant platforms. For example

in terms of scalability, VLANs (Virtual Local Area Network) are a simple example. At present, VLANs are theoretically limited to 4,096 segments. Thus, the scale is limited to approximately 4,000 tenants in a multitenant environment. IPv4 is another example, where some Cloud providers such as Microsoft Azure admitted that they ran out of addresses. This requirement means that the need for network technologies offering high performance, robustness, reliability, flexibility, scalability, and security never ends [12].

2.13 Usability

The Human Computer Interface and Distributed Systems communities are still far from one another. Cloud computing, in particular, would benefit from a closer alignment of these two communities. Although much effort has happened on resource management and back-end related issues, usability is a key aspect to reduce costs of organizations exploring Cloud services and infrastructure. This reduction is possible mainly due to labour related expenses as users can have better quality of service and enhance their productivity. The usability of Cloud [59] has already been identified as a key concern by NIST as described in their Cloud Usability Framework [156], which highlights five aspects: capable, personal, reliable, secure, and valuable. Capable is related to meeting Cloud consumers expectations with regard to Cloud service capabilities. Personal aims at allowing users and organizations to change the look and feel of user interfaces and to customize service functionalities. Reliable, secure, and valuable are aspects related to having a system that performs its functions under state conditions, safely/protected, and that returns value to users respectively. Coupa's white paper [43] on usability of Cloud applications also explores similar aspects, highlighting the importance of usability when offering services in the Internet.

For usability, current efforts in Cloud have mostly focused on encapsulating complex services into APIs to be easily consumed by users. One area where this is clearly visible is High Performance Computing (HPC). Researchers have been creating services to expose HPC applications to simplify their consumptions [87, 40]. These applications are not only encapsulated as services, but also receive Web portals to specify application parameters and manage input and output files.

Another direction related to usability of Cloud that got traction in the last years is DevOps [15, 138]. Its goal is to integrate development (Dev) and operations (Ops) thus aiding faster software delivery (as also discussed in Sections 2.10 and 4.10). DevOps has improved the productivity of developers and operators when creating and deploying solutions in Cloud environments. It is relevant not only to build new solutions in the Cloud but also to simplify the migration of legacy software from on-premise environments to multi-tenancy elastic Cloud services.

3 Emerging Trends and Impact Areas

As Cloud computing and relevant research matured over the years, it lead to several advancements in the underlying technologies such as containers and software defined networks. These developments in turn have led to several emerging trends in Cloud computing such as Fog computing, serverless computing, and software defined computing. In addition to them, other emerging trends in ICT such as Big Data, machine/deep learning and blockchain technology also have started influencing the Cloud computing research and have offered wide opportunities to deal with the open issues in Cloud related challenges. Here we discuss the emerging trends and impact areas relevant in the Cloud horizon.

3.1 Containers

With the birth of Docker [122], the container technology has aroused wide interest both in academia and industry [150]. More and more Internet companies are putting the container technology into their practice, and containers have become the de-facto standard for creating, publishing, and running applications. Containers rely on modern Linux operating systems' kernel facilities such as cgroups, LXC (Linux containers) and libcontainer . Meanwhile, CaaS (*container as a service*) is derived from the traditional Cloud computing [141], which gives birth to a lot of companies supplying the container services. E.g. UberCloud application containers allow a wide variety and selection of containerized applications for several online engineering Marketplaces [2, 73].

As a new type of virtualization technology, container provides two key features. First, containers start up very quickly and their launching time is less than a second. Second, containers have tiny memory footprint and consume a very small amount of resources. Compared with VMs, using containers not only improves the performance of applications, but also allows the host to support more instances simultaneously.

Docker, today's de facto container technology, uses Linux kernel's cgroups and namespaces to run independent "containers" within a physical machine. cgroups provide isolation of resources such as CPU, memory, block I/O and network. On the other hand, namespaces isolate an application's view of the operating environment, that includes process trees, network, user IDs and mounted file systems. Docker contains the libcontainer library as a container reference implementation. Furthermore, with Docker images, the applications can be built once and then deployed anywhere. By packing the application and related dependencies into a Docker image, Docker simplifies the deployment of the application and improves the development efficiency.

Although the container technology provides many benefits, there are still a lot of challenges to be dealt with. First, due to the sharing of kernel, the isolation and security of containers is weaker than VMs [179], which stimulates much interest and enthusiasm of researchers. There are two promising solutions to the problem. One is to leverage the new hardware features, such as the trusted execution support of Intel SGX [11]. The other is to use Unikernel, which is a kind of library operating system [3]. Second, trying to optimize the container performance is an everlasting theme. For example, to accelerate the container startup, Slack is proposed to optimize the storage driver [83]. Last but not least, the management of container clusters based on users QoS requirements is attracting significant attention. The systems for container cluster management such as Kubernetes[3], Mesos [86] and Swarm[4] are emerging as the core software of the Cloud computing platform.

3.2 Fog Computing

The Fog is an extension to the traditional Cloud computing model in that the edge of the network is included in the computing ecosystem to facilitate decision making as close as possible to the data source [24, 165, 67]. The vision of Fog computing is three fold. Firstly, to enable general purpose computing on traffic routing nodes, such as mobile base stations, gateways and routers. Secondly, to add compute capabilities to traffic routing nodes so as to process data as it is transmitted between user devices and a CDC. Thirdly, to use a combination of the former.

There are a number of benefits in using such a compute model. For example, latencies between users and servers can be reduced. Moreover, location awareness can be taken into account for geo-distributed computing on edge nodes. The Fog model inherently lends itself to improving the Quality-of-Service of streaming and real-time applications. Additionally, mobility can be seamlessly supported, wireless access between user devices and compute servers can be enabled and scalable control systems can be orchestrated. These benefits make it an appropriate solution for the upcoming IoT class of applications [175, 170, 48].

Edge and Fog computing are normally used interchangeably, however, they are slightly different, both paradigms rely on local processing power near data sources. In Edge computing, the processing power is given to the IoT device itself, while in the Fog computing, computing nodes (e.g., Dockers and VMs) are placed very close the source of data. The Edge computing paradigm depends on how IoT devices can be programmed to interact with each other and run user defined codes. Unfortunately, standard APIs that provide such functionality are not fully adopted by current IoT sensors/actuators, and thus Fog computing seems to be the only viable/generic solutions to date [127].

The Fog would offer a full-stack of IaaS, PaaS and SaaS resources, albeit not to the full extent as a CDC. Given that a major benefit of the Fog is its closer network proximity to the consumers of the services to reduce latency, it is anticipated that there will be a few Fog data centres per city. But as yet, the business model is evolving and possible locations for Fog resources range from a local coffee shop to mobile cell towers (as in Mobile Edge computing [176]). Additionally, infrastructure provided by traditional private Cloud and independent Fog providers may be employed [36]. Although the concept of Mobile Edge computing is similar to the premise of Fog computing, it is based on the mobile cellular network and does not extend to other traffic routing nodes along the path data travels between the user and the CDC.

[3] https://kubernetes.io/

[4] https://docs.docker.com/swarm/

Advantages of Fog computing include the vertical scaling of applications across different computing tiers. This allows for example, pre-processing the data contained in packets so that value is added to the data and only essential traffic is transmitted to a CDC. Workloads can be (1) decomposed on CDCs and offloaded on to edge nodes, (2) migrated from a collection of user devices on to edge nodes, or (3) aggregated from multiple sensors or devices on an edge node. In the Fog layer, workloads may be deployed via containers in lieu of VMs that require more resources [115, 102].

Cloud vendors have started to use edge locations to deliver security services (AWS Shield, Web Application Firewall Service) closer to users or to modify network traffic (e.g. Lambda@Edge[5]). Cloud providers are also asking customers to deploy on-premise storage and compute capabilities working with the same APIs as the ones they use in their Cloud infrastructure. These have made it possible to deliver the advantages of Fog architectures to the end users. For instance, in Intensive Care Units, in order to guarantee uninterrupted care when faced with a major IT outage, or to bring storage and computing capabilities to poorly connected areas (e.g. AWS Snowball Edge for the US Department of Defense[6]).

Other applications that can benefit from the Fog include smart city and IoT applications that are fast growing. Here, multi-dimensional data, such as text, audio and video are captured from urban and social sensors, and deep-learning models may be trained and perform inferencing to drive real-time decisions such as traffic signalling. Autonomous vehicles such as driverless cars and drones can also benefit from the processing capabilities offered by the Fog, well beyond what is hosted in the vehicle. The Fog can also offer computing and data archival capabilities. Immersive environments such as MMORPG gaming, 3D environment such as HoloLens and Google Glass, and even robotic surgery can benefit from GPGPUs that may be hosted on the Fog.

Many works such as Shi and Dustdar [147], Varghese et al. [169], Chang et al. [36] and Garcia Lopez et al. [67] have highlighted several challenges in Edge/Fog computing. Two prominent challenges that need to be addressed to enhance utility of Fog computing are mentioned here. Firstly, tackling the complex management issues related to multi-party SLAs. To this end, as a first step responsibilities of all parties will need to be articulated. This will be essential for developing a unified and interoperable platform for management since Edge nodes are likely to be owned by different organisations. The EdgeX Foundry[7] project aims to tackle some of these challenges. Secondly, given the possibility of multiple node interactions between a user device and CDC, security will need to be enhanced and privacy issues will need to be debated and addressed [158]. The Open Fog consortium[8] is a first step in this direction.

3.3 Big Data

There is a rapid escalation in the generation of streaming data from physical and crowd-sourced sensors as deployments of IoT, Cyber Physical Systems (CPS) [177], and micro-messaging social networks such as Twitter. This quantity is bound to grow many-fold, and may dwarf the size of data present on the public WWW, enterprises and mobile Clouds. Fast data platforms to deal with data velocity may usurp the current focus on data volume.

This has also seen the rise of in-memory and stream computation platforms such as Spark Streaming, Flink and Kafka that process the data in-memory as events or micro-batches and over the network rather than write to disk like Hadoop [184]. This offers a faster response for continuously arriving data, while also balancing throughput. This may put pressure on memory allocation for VMs, with SSD's playing a greater role in the storage hierarchy.

We are also seeing data acquisition at the edge by IoT and Smart City applications with an inherent feedback loop back to the edge. Video data from millions of cameras from city surveillance, self-driving cars, and drones at the edge is also poised to grow [144]. This makes latency and bandwidth between Edge and Cloud a constraint if purely performing analytics on the Cloud. Edge/Fog computing is starting to complement Cloud computing as a first-class platform, with Cloud providers already offering SDK's to make this easier from user-managed edge devices. While smartphones have already propagated mobile Clouds

[5] http://docs.aws.amazon.com/lambda/latest/dg/lambda-edge.html

[6] https://aws.amazon.com/blogs/publicsector/aws-snowball-edge-for-the-dod-bringing-storage-and-compute-to-tactical-situations/

[7] `https://www.edgexfoundry.org/`

[8] `https://www.openfogconsortium.org/`

where apps cooperatively work with Cloud services, there will be a greater need to combine peer-to-peer computing on the Edge with Cloud services, possibly across data centres. This may also drive the need for more regional data centres to lower the network latency from the edge, and spur the growth of Fog computing.

Unlike structured data warehouses, the growing trend of *"Data Lakes"* encourages enterprises to put all their data into Cloud storage, such as HDFS, to allow intelligence to be mined from it [157]. However, a lack of tracking metadata describing the source and provenance of the data makes it challenging to use them, effectively forming *"data graveyards"*. Many of these datasets are also related to each other through logical relationships or by capturing physical infrastructure, though the linked nature of the datasets may not be explicitly captured in the storage model [23]. There is heightened interest in both deep learning platforms like TensorFlow to mine such large unstructured data lakes, as well as distributed graph databases like Titan and Neo4J to explore such linked data.

3.4 Serverless Computing

Serverless computing is an emerging architectural pattern that changes dramatically the way Cloud applications are designed. Unlike a traditional three-tiered Cloud application in which both the application logic and the database server reside in the Cloud, in a serverless application the business logic is moved to the client; this may be embedded in a mobile app or runs on temporarily provisioned resources during the duration of the request. This translates to the fact that a client does not need to rent resources, for example Cloud VMs for running the server of an application[9]. This computing model implicitly handles the challenges of deploying applications on a VM, such as over/under provisioning cloud VMs for the application, balancing the workload across the resources and ensuring reliability and fault-tolerance. In this case, the actual server is made abstract, such that properties like control, cost and flexibility, which are not conventionally considered are taken into account.

Consequently, serverless computing reduces the amount of backend code, developers need to write, and also reduces administration on Cloud resources. It appears in two forms; Backend as a Service (BaaS) and Functions as a Service (FaaS) [139]. This architecture is currently supported on platforms such as AWS Lambda[10], IBM OpenWhisk[11] and Google Cloud Functions[12].

It is worth noting the term "serverless" may be somehow misleading: it does not mean that the application runs without servers; instead, it means that the resources used by the application are managed by the Cloud provider [16]. In BaaS, the server-side logic is replaced by different Cloud services that carry out the relevant tasks (for example, authentication, database access, messaging, etc.), whereas in FaaS ephemeral computing resources are utilized that are charged per access (rather than on the basis of time, which is typical of IaaS solutions).

FaaS poses new challenges particularly for resource management in Clouds that will need to be addressed. This is because arbitrary code (the function) will need to execute in the Cloud without any explicit specification of resources required for the operation. To make this possible, FaaS providers pose many restrictions about what functions can do and for how long they can operate [16]. For example, they enforce limits on the amount of time a function can execute, how functions can be written, and how the code is deployed [16]. This is restrictive in the types of applications that can make use of current FaaS models. The adoption of the FaaS model will however increase if a wider range of applications can make use of restriction relaxed FaaS models, which will evolve what is now a "niche" service model to an effective method for developing Cloud applications.

A full-fledged general-purpose serverless computing model is still a vision that needs to be achieved. Upcoming research has explored applications that can benefit from serverless computing [181] and platforms that match services offered by providers [85, 153, 120]. As discussed by Hendrickson et al. [85], there are still a number of issues at the middleware layer that need to be addressed that are orthogonal to advances in the area of Cloud computing that are also necessary to better support this model. Despite these challenges, this is a promising area to be explored with significant practical and economic impact. It is predicted by Forbes

[9] `https://d0.awsstatic.com/whitepapers/AWS_Serverless_Multi-Tier_Architectures.pdf`
[10] `https://aws.amazon.com/lambda/`
[11] `https://developer.ibm.com/openwhisk/`
[12] `https://Cloud.google.com/functions/`

that there will be a likely increase of serverless computing since a large number of 'things' will be connected to the edge and data centres[13].

3.5 Software-defined Cloud Computing

Software-defined Cloud Computing is a method for the optimization and automation of configuration process and physical resources abstraction, by extending the concept of virtualization to all resources in a data centre including compute, storage, and network [29]. Virtualization technologies aim to mask, abstract and transparently leverage underlying resources without applications and clients having to understand physical attributes of the resource. Virtualization technologies for computing and storage resources are quite advanced to a large extent. The emerging trends in this space are the virtualization in networking aspects of Cloud, namely Software-defined networking (SDN) and Network functions virtualization (NFV).

The main motivation for SDN, an emerging networking paradigm, is due to the demand/need for agile and cost-efficient computer networks that can also support multi-tenancy [126]. SDN aims at overcoming the limitations of traditional networks, in particular networking challenges of multi-tenant environments such as CDCs where computing, storage, and network resources must be offered in slices that are independent or isolated from one another. Early supporters of SDN were among those believing that networking equipment manufacturers were not meeting their needs particularly in terms of innovation and the development of required features of data centres. There were another group of supporters who aimed at running their network by harnessing the low-cost processing power of commodity hardware.

SDN decouples the data forwarding functions and network control plane, which enables the network to become centrally manageable and programmable [130]. This separation offers the flexibility of running some form of logically centralized network orchestration via the software called SDN controller. The SDN controller provides vendor-neutral open standards which abstract the underlying infrastructure for the applications and network services and facilitates communication between applications wishing to interact with network elements and vice versa [126]. OpenFlow [121] is the de-facto standard of SDN and is used by most of SDN Controllers as southbound APIs for communications with network elements such as switches and routers.

NFV is another trend in networking which is quickly gaining attention with more or less similar goals to SDN. The main aim of NFV is to transfer network functions such as intrusion detection, load balancing, firewalling, network address translation (NAT), domain name service (DNS), to name a few, from proprietary hardware appliances to software-based applications executing on commercial off-the-shelf (COTS) equipment. NFV intends to reduce cost and increase elasticity of network functions by building network function blocks that connect or chain together to build communication services [37].

Apart from networking challenges, SDN and NFV can serve as building blocks of next-generation Clouds by facilitating the way challenges such as sustainability, interconnected Clouds, and security can be addressed. Heller et al. [84] conducted one of the early attempts towards sustainability of Cloud networks using OpenFlow switches and providing network energy proportionality. The main advantage of using NFV is that Cloud service providers can launch new network function services in a more agile and flexible way. In view of that, Eramo et al. [57] proposed a consolidation algorithm based on a migration policy of virtualized network function instances to reduce energy consumption. Google adopted SDN in its B4 network to interconnect its CDC with a globally-deployed software defined WAN [96]. Yan et al. [182] investigate how SDN-enabled Cloud brings us new opportunities for tackling distributed denial-of-service (DDoS) attacks in Cloud computing environments.

3.6 Blockchain

In several industries, blockchain technology [159] is becoming fundamental to accelerate and optimize transactions by increasing their level of traceability and reliability. Blockchain consists of a distributed immutable ledger deployed in a decentralized network that relies on cryptography to meet security constraints [162]. Different parties of a chain have the same copy of the ledger and have to agree on transactions being placed into the blockchain. Cloud computing is essential for blockchain as it can host not only the blockchain nodes, but services created to leverage this infrastructure. Cloud can encapsulate blockchain services in both PaaS and SaaS to facilitate their usage. This will involve also challenges related to scalability as these

[13] http://www.forbes.com/sites/ibm/2016/11/17/three-ways-that-serverless-computing-will-transform-app-development-in-2017/

chains start to grow as technology matures. Cloud plays a key role in the widespread adoption of blockchain with its flexibility for dynamically allocating computing resources and managing storage [42]. An important component of blockchain is to serve as a platform to run analytics on transaction data, which can be mixed with data coming from other sources such as IoT, financial and weather-related services.

Another side of blockchain and Cloud is to consider the direction where the advances in blockchain will assist Cloud computing [14, 64]. It is well known that Cloud is an important platform for collaboration and data exchange. Blockchain can assist Cloud by creating more secure and auditable transaction platform. This is essential for several industries including health, agriculture, manufacturing, and petroleum.

3.7 Machine and Deep Learning

Due to the vast amount of data generated in the last years and the computing power increase, mainly of GPUs, AI has gained a lot of attention lately. Algorithms and models for machine learning and deep learning are relevant for Cloud computing researchers and practitioners. From one side, Cloud can benefit from machine/deep learning in order to have more optimized resource management, and on the other side, Cloud is an essential platform to host machine/deep learning services due to its pay-as-you-go model and easy access to computing resources.

In the early 2000s, autonomic computing was a subject of study to make computing systems more efficient through automation [104]. There, systems would have four major characteristics: self-configuration, self-optimization, self-healing, and self-protection. The vision may become possible with the assistance of breakthroughs in artificial intelligence and data availability. For Cloud, this means efficient ways of managing user workloads, predictions of demands for computing power, estimations of SLA violations, better job placement decisions, among others. Simplifying the selection of Cloud instances [142] or optimising resource selection [17] are well known examples of the use of machine learning for better use of Cloud services and infrastructure. The industry has already started to deliver auto-tuning techniques for many Cloud services so that many aspects of running the application stack are delegated to the Cloud platform. For instance, Azure SQL database has auto-tuning as a built-in feature that adapts the database configuration (e.g. tweaking and cleaning indices [104]).

Several machine learning and deep learning algorithms require large-scale computing power and external data sources, which can be cheaper and easier to acquire via Cloud than using on-premise infrastructure. That is why several companies are providing AI-related services in the Cloud such as IBM Watson, Microsoft Azure Machine Learning, AWS Deep Learning AMIs, Google Cloud Machine Learning Engine, among others. Some of these Cloud services can be enhanced while users consume them. This has already delivered considerable savings for CDCs (e.g. [65]). It can also streamline managed database configuration tuning (Van Aken et al. [164]).

We anticipate a massive adoption of auto-tuners, especially for the SaaS layer of the Cloud. We also foresee the likely advent of new automated tools for Cloud users to benefit from the experience of other users via semi-automated application builders (recommending tools of configurations other similar users have successfully employed), automated database sharders, query optimisers, or smart load balancers and service replicators. As security becomes a key concern for most corporations worldwide, new ML-based security Cloud services will help defend critical Cloud services and rapidly mutate to adapt to new fast-developing threats.

4 Future Research Directions

The Cloud computing paradigm, like the Web, the Internet, and the computer itself, has transformed the information technology landscape in its first decade of existence. However, the next decade will bring about significant new requirements, from large-scale heterogeneous IoT and sensor networks producing very large data streams to store, manage, and analyse, to energy- and cost-aware personalized computing services that must adapt to a plethora of hardware devices while optimizing for multiple criteria including application-level QoS constraints and economic restrictions.

Significant research was already performed to address the Cloud computing technological and adoption challenges, and the state-of-the-art along with their limitations is discussed thoroughly in section 2. The

future research in Cloud computing should focus at addressing these limitations along with the problems hurled and opportunities presented by the latest developments in the Cloud horizon. Thus the future R&D will greatly be influenced/driven by the emerging trends discussed in section 3. Here the manifesto provides the key future directions for the Cloud computing research, for the coming decade.

4.1 Scalability and Elasticity

Scalability and elasticity research challenges for the next decade can be decomposed into hardware, middleware, and application-level.

At the Cloud computing hardware level, an interesting research direction is special-purpose Clouds for specific functions, such as deep learning, e.g. Convolutional Neural Networks (CNNs), Multi-Layer Perceptrons (MLPs), and Long Short-Term Memory (LSTMs) can be deployed for a spectrum of applications, such as image and video recognition. While these applications may appear to be very narrow, their use is increasingly growing. There are numerous examples at control points at airports, for social networking and many other applications. Key Cloud providers are already offering accelerator and special-purpose hardware with increasing usage growth, e.g., Amazon is offering GPUs, Google has been deploying Tensor Processing Units (TPUs) [100] and Microsoft is deploying FPGAs in the Azure Cloud [137]. As new hardware addresses scalability, Clouds need to embrace non-traditional architectures, such as neuromorphic, quantum computing, adiabatic, nanocomputing and many others (see [91]). Research needed includes coming up with appropriate virtualization abstractions and pricing models for special-purpose hardware (e.g., image and video processing as micro-services).

At the Cloud computing middleware level, research is required to further increase reuse of existing infrastructure, to improve speed of deployment and provisioning of hardware and networks for very large scale deployments. This includes algorithms and software stacks for reliable execution of applications with failovers to geographically remote private or hybrid Cloud sites. Research is also needed on InterClouds which will seamlessly enable computations to run on multiple public Cloud providers simultaneously. In order to support HPC applications, it will be critical to guarantee consistent performance across multiple runs even in the presence of additional Cloud users. New deployment and scheduling algorithms need to be developed to carefully match HPC applications with those that would not introduce noise in parallel execution or if not possible use dedicated clusters for HPC [78]. To be able to address large scale communication-intensive applications, further Cloud provider investments are required to support high throughput and low latency networks. Such environment aimed to achieve high levels of performance needs sophisticated management to handle multiple clients and provide sustainable business to Cloud providers. In addition, HPC and engineering applications are converging with Big Data applications. The latter are gaining momentum with the IoT. This moves the narrow area of HPC into much broader use in a spectrum of applications, such as smart cities [143] or industrial IoT [24]. These applications have substantial needs in terms of (near-)real time processing of large scale of data, its intelligent analysis and then closing the loops of control. These applications tie in previous problems of deep learning with large scale (potentially HPC) applications.

At the Cloud computing application level, research is needed in programming models for adaptive elastic mobile decentralized distributed applications as needed by Fog/Edge computing, InterClouds, and the IoT. Separation of concerns will be important to address complexity of software development and delivery models. Functional application aspects should be specified, programmed, tested, and verified modularly. Program specifications may be probabilistic in nature, e.g., when analysing asynchronous data streams. Research is needed in specifying and verifying correctness of non-deterministic programs, which may result, e.g., from online machine learning algorithms. Non-functional aspects, e.g., fault tolerance, should be translucent: they can be completely left to the middleware, or applications should have declarative control over them, e.g., a policy favouring execution away from a mobile device in battery-challenged conditions [19]. *Translucent* programming models, languages, and Application Programming Interfaces (APIs) will be needed to enable tackling the complexity of application development while permitting control of application delivery to future-generation Clouds. Sensor data analytics for mobile IoT devices (e.g., smartphones, vehicles) will require cooperative resource management between centralized CDCs and distributed Edge computing resources for real-time processing. Such a resource management method should be aware of the locations of mobile devices for optimal resource allocation. To deploy applications and services to edge resources, the container technology will be useful due to its small footprint and fast deployment [131]. One research direction

to pursue will be the use of even finer-grained programming abstractions such as the actor model and associated middleware that dynamically reconfigures programs between edge resources and CDCs through transparent migration for users [92]. Declarative policies should dictate application delivery models with differential service, e.g., in emergency scenarios, a high-priority service should be able to pre-empt lower-priority computations by dynamically reallocating resources to guarantee the desired quality of the emergency service.

4.2 Resource Management and Scheduling

The evolution of the Cloud in the upcoming years will lead to a new generation of research solutions for resource management and scheduling. Technology trends such as Fog and serverless computing will increase the level of decentralization of the computation, leading to increased resource heterogeneity and variability in the workloads. Conversely, trends such as software-defined computing and Big Data will come to maturity, leading to novel management paradigms. These technology trends offer many outlets for novel research in resource management and scheduling.

Fog computing poses many manageability questions and will require first a better understanding of the extent by which existing management techniques, such as asymptotic and declarative methods [135, 7], will remain applicable and effective. The knowledge acquired from this exploratory analysis will ultimately lead to Fog-specific management paradigms that can transparently coordinate heterogeneous resources spread across data centres and at the edge, taking into account device mobility, highly dynamic network topology, and privacy and security protection at scale.

In serverless computing, FaaS users expect that their functions will be executed within a specific time, and given that cost is per access, which will require novel methods and metrics for resource management. Given that a single application backed by FaaS can lead to hundreds of hits to the Cloud in a second, an important challenge will be to optimize allocation of resources for each class of service so that revenue is optimized for the provider while all the FaaS QoS expectations are met. This requires to take into consideration soft constraints on execution time of functions and proactive FaaS provisioning to avoid high latency of resource start-up to affect the performance of backed applications.

It is also foreseeable that the emerging SDN paradigm will provide plenty of novel research opportunities in Cloud resource management. By logically centralizing the network control plane, SDNs provide opportunities for more efficient management of resources located in a single administrative domain such as a CDC. SDN also facilitates the joint VM and traffic consolidation, which is a difficult task to do in traditional data centre networks, to optimize energy consumption and SLA satisfaction, thus opening new research outlets [45]. Leveraging SDN together with NFV technologies allows for efficient and on-demand placement and chaining of virtual network functions (VNFs) [38]. The virtualized nature of VNFs also makes orchestration and consolidation of them easier and dynamic deployment of network services possible [110, 134].

Moreover, since Clouds are owned by different organizations, which differ in their usage policies, cost models, and availability patterns for varying loads, resource management and application scheduling in these setups is a complex activity. Cloud providers and consumers also differ in their goals and constraints (e.g., users aim to minimize the cost of computation and providers aim to maximise the profit). To meet these challenges, market-oriented models and methods for resource allocation and regulation of the supply and demand of the available Cloud resources need to be developed.

In addition, it is foreseeable that the ongoing interest for ML, deep learning, and AI applications will help in dealing with the complexity, heterogeneity, and scale, in addition to spawn novel research in established data centre resource management problems such as VM provisioning, consolidation, and load balancing. For example, in scientific workflows the focus so far has been on efficiently managing the execution of platform-agnostic scientific applications. As the amount of data processed increases and extreme-scale workflows begin to emerge, it is important to consider key concerns such as fault tolerance, performance modelling, efficient data management, and efficient resource usage. For this purpose, Big Data analytics will become a crucial tool [52]. For instance, monitoring and analysing resource consumption data may enable workflow management systems to detect performance anomalies and potentially predict failures, leveraging technologies such as serverless computing to manage the execution of complex workflows that are reusable and can be shared across multiple stakeholders.

4.3 Reliability

One of the most challenging areas in Cloud computing systems is reliability as it has a great impact on the QoS as well as on the long term reputation of the service providers. Currently, all the Cloud services are provided based on the cost and performance of the services. The key challenge faced by Cloud service providers is - how to deliver a competitive service that meets end users' expectations for performance, reliability, and QoS in the face of various types of independent as well as temporal and spatial correlated failures. So the future of research in this area will be focused on innovative Cloud services that provide reliability and resilience with assured service performance; which is called Reliability as a Service (RaaS). This requires new modules to be included in the existing Cloud systems such as failure model and workload model and they should be adapted for resource provisioning policies to provide flexible reliability services to the wide range of applications.

One of the future directions in RaaS will be using deep and machine learning for failure prediction. This will be based on failure characterization and developing a model from massive amount of failure datasets. Having a comprehensive failure prediction model will lead to a failure-aware resource provisioning that can guarantee the level of reliability and performance for the user's applications. This concept can be extended as another research direction for the Fog computing where there are several components on the edge. While fault-tolerant techniques such as replication could be a solution in this case, more efficient and intelligent approaches will be required to improve the reliability of new type of applications such as IoT applications.

Another research direction in reliability will be about Cloud storage systems that are now mature enough to handle Big Data applications. However, failures are inevitable in Cloud storage systems as they are composed of large scale hardware components. Improving fault tolerance in Cloud storage systems for Big Data applications is a significant challenge. Replication and Erasure coding are the most important data reliability techniques employed in Cloud storage systems [125]. Both techniques have their own trade-off in various parameters such as durability, availability, storage overhead, network bandwidth and traffic, energy consumption and recovery performance. The future research should include the challenges involved in employing both techniques in Cloud storage systems for Big Data applications with respect to the aforementioned parameters [125]. This hybrid technique applies proactive dynamic data replication of erasure coded data based on node failure prediction, which significantly reduces network traffic and improves the performance of Big Data applications with less storage overhead.

4.4 Sustainability

Sustainability of ICT systems is emerging as a major consideration [70], and increasingly CDCs are being established, with up to 1000 MW of potential power consumption, in or close to areas where there are plentiful sources of renewable energy [22], such as hydro-electricity in northern Norway, and where natural cooling can be available as in areas close to the Arctic Circle. This actually requires new and innovative system architectures that can distribute data centre computing, geographically. To address this, algorithms have been proposed, which rely on geographically distributed data coordination, resource provisioning and energy-aware and carbon footprint-aware provisioning in data centres [80, 55, 105]. In addition, geographical load balancing also can provide an effective approach for demand-response; with careful pricing, electricity providers can motivate Cloud service providers to "follow the renewables" and serve requests through CDCs located in areas where green energy is available [114]. However, placing data centres so far away from the end users places a further burden on the energy consumption and QoS of the networks that connect the end users to the CDCs. Another challenge relates to the very short end-to-end delay that certain operations, such as financial transactions, require.

Unfortunately, high performance and more data processing has always gone hand-in-hand with greater energy consumption. Thus QoS, SLAs and sustainability have to be considered hand-in-hand and managed online. Since all the fast-changing online behaviours cannot be predicted in advance or modelled in a complete manner, adaptive self-aware techniques will be needed to face this challenge. Some progress has been recently made in this direction [173] but further work will be needed. The actual algorithms that may be used will include ML techniques such as those described in [183] which will exploit constant online observation of system parameters that can lead to online decision making that will optimise sustainability while respecting QoS considerations and SLAs.

The Fog can also substantially increase energy consumption because of the greater difficulty of efficient energy management for smaller and highly diverse systems [71]. Thus more attention will need to be directed to such matters in future work including the impact on network QoS and the possibility of adaptive management of the energy supply and demand so as to efficiently share energy among multiple Fog systems. The future work should also focus at maximising the usage of green energy while meeting the QoS expectations of an application, both for the Fog and the Cloud.

CDCs are also drawing increasingly more power with the ever-increasing amount of data that need to be stored and processed. However, reducing energy consumption in networks is a complex problem as saving energy for networking elements often disturbs other aspects such as reliability, scalability, and performance of the network. In addition, it has got considerably lower amount of attention compared to aspects such as computing. The global network awareness and centralized decision-making approach of the SDN provides better opportunity for creating sustainable networks for Clouds. This is perhaps one of the areas that will draw more research efforts and innovations in the next decade.

Furthermore, current sustainability approaches primarily focus on the VM consolidation for minimizing the energy consumption of the servers. Given that other components of CDCs such as storage, networks, and cooling systems together consume around 40% of CDC energy, there is a need for new techniques for energy-efficient management of all resources (including servers, storage, networks, and cooling systems) within CDCs in integrated and holistic manner. Through interplay between IoT-enabled cooling systems and a CDC resource manager, new techniques need to dynamically decide which of the resources to turn-on or turn-off in time and space dimensions based on system load and workload forecasts. This will create Cloud computing environments that are sustainable, save energy, and reduce both cost and carbon footprint.

4.5 Heterogeneity

Heterogeneity on the Cloud was introduced in the last decade, but awaits widespread adoption. As highlighted in Section 2.5, there are currently at least two significant gaps that hinder heterogeneity from being fully exploited on the Cloud. The first gap is between unified management platforms and heterogeneity. Existing research that targets resource and workload management in heterogeneous Cloud environments is fragmented. This translates into the lack of availability of a unified environment for efficiently exploiting VM level, vendor level and hardware architecture level heterogeneity while executing Cloud applications. The manifesto therefore proposes for the next decade an umbrella platform that accounts for heterogeneity at all three levels. This can be achieved by integrating a portfolio of workload and resource management techniques from which optimal strategies are selected based on the requirement of an application.

The second gap is between abstraction and heterogeneity. Current programming models for using hardware accelerators require accelerator specific languages and low level programming efforts. Moreover, these models are conducive for developing scientific applications. This restricts the wider adoption of heterogeneity for service oriented and user-driven applications on the Cloud. One meaningful direction to pursue will be to initiate a community wide effort for developing an open-source high-level programming language that can satisfy core Cloud principles, such as abstraction and elasticity, which are suited for modern and innovative Cloud applications in a heterogeneous environment. This will also be a useful tool as the Fog ecosystem emerges and applications migrate to incorporate both Cloud and Fog resources.

Recently there is also a significant discussion about disaggregated datacentres. Traditionally data centres are built using servers and racks with each server contributing the resources such as CPU, memory and storage, required for the computational tasks. With the disaggregated datacentre each of these resources is built as a standalone resource "*blade*", where these blades are interconnected through a high-speed network fabric. The trend has come into existence as there is significant gap in the pace at which each of these resource technologies individually advanced. Even though most prototypes are proprietary and in their early stages of development, a successful deployment at the data centre level would have significant impact on the way the traditional IaaS are provided. However, this needs significant development in the network fabric as well [66].

4.6 Interconnected Clouds

As the grid computing and web service histories have shown, interoperability and portability across Cloud systems is a highly complicated area and it is clear at this time that pure standardization is not sufficient to address this problem. The use of application containers and configuration management tools for portability, and the use of software adapters and libraries for interoperability are widely used as practical methods for achieving interoperation across Cloud services and products. However, there are a number of challenges [31], and thus potential research directions, that have been around since the early days of Cloud computing and, due to their complexity, have not been satisfactorily addressed so far.

One of such challenges is how to promote Cloud interconnection without forcing the adoption of the minimum common set of functionalities among services: if users want, they should be able to integrate complex functionalities even if they are offered only by one provider. Other research directions include how to enable Cloud interoperation middleware that can mimic complex services offered by one provider by composing simple services offered by one or more providers - so that the choice about the complex service or the composition of simpler services were solely dependent on the user constraints - cost, response time, data sovereignty, etc.

The above raises another important future research direction: how to enable middleware operating at the user-level to identify candidate services for a composition without support from Cloud providers? Given that providers have economic motivation to try to retain all the functionalities offered to their customers (i.e., they do not have motivation to facilitate that only some of the services in a composition are their own), one cannot expect that an approach that requires Cloud providers cooperation might succeed.

However, ubiquitously interconnected Clouds can truly be achieved only when Cloud vendors are convinced that the Cloud interoperability adoption brings them financial and economic benefits. This requires novel approaches for billing and accounting, novel interconnected Cloud suitable pricing methods, along with formation of InterCloud marketplaces [161].

Finally, the emergence of SDNs and the capability to shape and optimize network traffic has the potential to influence research in Cloud interoperation. Google reports that one of the first uses of SDNs in the company was for optimization of wide-area network traffic connecting their data centres [163]. In the same direction, investigation is needed on the feasibility and benefits of SDN and NFV to address some of the challenges above. For example, SDN and NFV can enable better security and QoS for services built as compositions of services from multiple providers (or from geographically distributed services from the same provider) by enforcing prioritization of service traffic across providers/data centres and specific security requirements [88].

4.7 Empowering Resource-Constrained Devices

Regarding future directions for empowering resource-constrained devices, in the mobile Cloud domain, we already have identified that, while task delegation is a reality, code offloading still has adaptability issues. It is also observed that, *"as the device capabilities are increasing, the applications that can benefit from the code offloading are becoming limited"* [154]. This is evident, as the capabilities of smartphones are increasing, to match or benefit from offloading, the applications are to be offloaded to Cloud instances with much higher capacity. This incurs higher cost per offloading. To address this, the future research in this domain should focus at better models for multi-tenancy in Mobile Cloud applications, to share the costs among multiple mobile users. The problem further gets complex due to the heterogeneity of both the mobile devices and Cloud resources.

We also foresee the need for incentive mechanisms for heterogeneous mobile Cloud offloading to encourage mobile users to participate and get appropriate rewards in return. This should encourage in adapting the mobile Cloud pattern to the social networking domain as well, in designing ideal scenarios. In addition, the scope and benefits offered by the emerging technologies such as serverless computing, CaaS and Fog computing, to the mobile Cloud domain, are not yet fully explored.

The incentive mechanisms are also relevant for the IoT and Fog domains. Recently there is significant discussion about the establishment of Fog closer to the *things*, by infrastructure offered by independent Fog providers [36]. These architectures follow the consumer-as-provider (CaP) model. A relevant CaP example in the Cloud computing domain is the MQL5 Cloud Network [1], which utilizes consumer's devices and desktops for performing various distributed computing tasks. Adaptation of such Peer-to-Peer (P2P) and

CaP models would require ideal incentive mechanisms. Further discussion about the economic models for such Micro Data centres is provided in Section 4.9.

The container technology also brings several opportunities to this challenge. With the rise of Fog and Edge computing, it can be predicted that the container technology, as a kind of lightweight running environment and convenient packing tools for applications, will be widely deployed in edge servers. For example, the customized containers, such as Cloud Android Container [178], aimed at Edge computing and offloading features will be more and more popular. They provide efficient server runtime and inspire innovative applications in IoT, AI, and other promising fields.

Edge analytics in domains such as real-time streaming data analytics would be another interesting research direction for the resource constrained devices. The things in IoT primarily deal with sensor data and the Cloud-centric IoT (CIoT) model extracts this data and pushes it to the Cloud for processing. Primarily, Fog/Edge computing came to existence in order to reduce the network latencies in this model. In edge analytics, the sensor data will be processed across the complete hierarchy of Fog topology, i.e. at the edge devices, intermediate Fog nodes and Cloud. The intermediary processing tasks include filtering, consolidation, error detection etc. Frameworks that support edge analytics (e.g. Apache Edgent[14]) should be studied considering both the QoS and QoE (Quality of Experience) aspects.

4.8 Security and Privacy

Security and privacy challenges in Cloud computing offer the following future research directions.

When protecting data with client-side encryption, there is the need for scalable and well-performing techniques that, while not affecting service functionality, can: 1) be easily integrated with current Cloud technology; 2) avoid possible information leakage caused by the solutions (e.g., indexes) adopted for selectively retrieving data or by the encryption supporting queries [128]; 3) support a rich variety of queries. Other challenges are related to the design of solutions completely departing from encryption and based on the splitting of data among multiple providers to guarantee generic confidentiality and access/visibility constraints possibly defined by the users.

Considering the data integrity problem, an interesting research direction consists in designing solutions proving integrity guarantees of data, possibly distributed and stored on multiple Cloud providers.

Moreover, our society is moving towards the Big Data era that introduces many opportunities but also several concerns regarding data protection and privacy. The explosion of data and their variety (i.e., structured, unstructured, and semi-structured formats) make the definition and enforcement of scalable data protection solutions a challenging issue, especially considering the fact that the risk of inferring sensitive information significantly increases in Big Data. Other issues are related to the provenance and quality of Big Data. In fact, tracking Big Data provenance can be useful for: i) verifying whether data came from trusted sources and have been generated and used appropriately; and ii) evaluating the quality of the Big Data, which is particularly important in specific domains (e.g., healthcare). Blockchain technology can be helpful for addressing the data provenance challenge since it ensures that data in a blockchain are immutable, verifiable, and traceable. However, it also introduces novel privacy concerns since data (including personal data) in a blockchain cannot be changed or deleted.

In addition, current solutions for selectively sharing information with different users also present several challenges. Some of these challenges are: 1) the support of write privileges; 2) the support of multiple writers; 3) the efficient enforcement of policies updates in distributed storage systems characterized by multiple and independent Cloud providers; 4) the selective sharing of information among parties involved in distributed computations, thus also taking advantage of the availability of cheaper (but not completely trusted) Cloud providers.

Query privacy is another interesting direction, which deals with the problem of protecting accesses to data. Existing solutions addressing this issue are difficult to apply in real-world scenarios for their computational complexity or for the limited kinds of queries supported. Interesting open issues are the development of scalable and efficient techniques: i) supporting concurrent accesses by different users; and ii) ensuring no improper leakage on user activity and applicability in real database contexts.

The future research should also deal with computation integrity. Since the Cloud providers in charge of computations may not be completely trusted, there is the problem of verifying the integrity of the computa-

[14] http://edgent.apache.org/

tions. Existing solutions are limited in their applicability and integrity guarantees offered. There is then the need to design a generic framework for evaluating the integrity guarantees provided according to the cost that a user is willing to pay to have such guarantees. Furthermore, it is still missing the support of generic queries, possibly involving multiple datasets, with integrity guarantees.

In addition, users often need to select the Cloud providers for the storage and management of their data that better fit their security and privacy requirements. Existing solutions supporting users in this selection process are at their infancy since they consider only limited user-based requirements (e.g., cost and performance requirements only) or pre-defined indicators. An interesting challenge is therefore the definition of a comprehensive framework that allows users both to express different requirements and preferences for the Cloud provider selection, and to verify that Cloud providers offer services fully compliant with the signed contract.

Cloud providers also need to check legitimacy of the requests to tackle issues such as Denial of Service (DoS) or other forms of cyber-attacks. These types of attacks are critical, as a coordinated attack on the Cloud services can be wrongly inferred as legitimate traffic and the resources would be scaled up to handle them. This will result in both the incurred additional costs and waste in energy [151]. Cloud systems should be able to distinguish these attacks and can decide either to drop the additional load or avoid excessive provisioning of resources. This requires extending the existing techniques of DDoS to also include exclusive characteristics of Cloud systems.

Considering the emerging Fog-based scenarios, there are several security and privacy challenges that still need to be investigated. First, lacking (or deferred) central controls is critical for some Fog applications, which may raise privacy and trust issues. Second, Fog computing assumes the presence of trusted nodes together with malicious ones. This requires adapting the earlier research of secure routing, redundant routing and trust topologies performed in the P2P context, to this novel setting [67]. Third, Cloud security research can rely on the idea that all data could be dumped into a data lake and analysed (in near real time) to spot security and privacy problems. This may no longer be possible when devices are not always connected and there are too many of them to make it financially viable to dump all the events into a central locations. This Fog-induced fragmentation of information combined with encryption will foster a new wave of Cloud security research.

4.9 Economics of Cloud Computing

The economics of Cloud computing offers several interesting future research directions. As Cloud computing deployments based on VMs transition to the use of container-based deployments, there is increasing realisation that the lower overheads associated with container deployment can be used to support real-time workloads. Hence, serverless computing capability is now becoming commonplace with Google Cloud Functions, Amazon Lambda, Microsoft Azure Functions and IBM Bluemix OpenWhisk. In these approaches, no computing resources are actually charged for until a function is called. These functions are often simpler in scope and typically aimed at processing data stream-based workloads. The actual benefit of using serverless computing depends on the execution behaviour and types of workloads expected within an application. Eivy [56] outlines the factors that influence the economics of such function deployment, such as: (1) average vs. peak transaction rates; (2) scaling number of concurrent activity on the system, i.e. running multiple concurrent functions with increasing number of users; (3) benchmark execution of serverless functions on different backend hardware platforms, and the overall execution time required for your function.

The combination of stable Cloud resources and volatile user edge resources can reduce the operating costs of Cloud services and infrastructures. However, we expect users to require some incentives to make their devices available at the edge. The availability of Fog and Edge resources, provides the possibility for a number of additional business models and the inclusion of additional category of providers in the Cloud marketplace. We refer to the existence of such systems as Micro Data Centres (MDCs), which are placed between the more traditional data centre and user owned/provisioned resources. Business models include: (1) *Dynamic MDC discovery:* in this model, a user would dynamically be able to choose a MDC provider, according to the MDC availability profile, security credentials, or type. A service-based ecosystem with multiple such MDC providers may be realized, however this will not directly guarantee the fulfilment of the user objectives through integration of externally provisioned services. (2) *Pre-agreed MDC contracts:* in this model, detailed contracts adequately capture the circumstances and criteria that influence the performance

of the MDC provisioned external services. A user's device would have these pre-agreed contracts or SLA with specific MDC operators, and would interact with them preferentially. This also reduces the potential risks incurred by the user. In performance-based contracts, an MDC would need to provide a minimum level of performance (e.g. availability) to the user which is reflected in the associated price. This could be achieved by interaction between MDCs being managed by the same operator, or by MDC outsourcing some of their tasks to a CDC; (3) *MDC federation:* in this model multiple MDC operators can collaborate to share workload within a particular area, and have preferred costs for exchange of such workload. This is equivalent to alliances established between airline operators to serve particular routes. To support such federation, security credentials between MDCs must be pre-agreed. This is equivalent to an extension of the pre-agreed MDC contracts business model, where MDCs across multiple coffee shop chains can be federated, offering greater potential choice for a user; (4) *MDC-Cloud data centre exchange:* in this model a user's device would contact a CDC in the first instance, which could then outsource computation to an MDC if it is unable to meet the required QoS targets (e.g. latency). A CDC could use any of the three approaches outlined above i.e. dynamic MDC discovery, preferred MDCs, or choice of an MDC within a particular group. A CDC operator needs to consider whether outsourcing could still be profitable given the type of workload a user device is generating.

The unpredictable Cloud environment arising due to the use of Fog and Edge resources, and the dynamics of service provisioning in these environments, requires architects to embrace uncertainty. More specifically, architecting for the Cloud needs to strike a reasonable balance between dependable and efficient provision and their economics under uncertainties. In this context, the architecting process needs to incubate architecture design decisions that do not only meet qualities such as performance, availability, reliability, security, compliance, etc. but also seek value through their provision. Research shall look at possible abstractions and formulations of the problem, where competitive and/or cooperative game design strategies can be explored to dynamically manage various players, including Cloud multitenants, service providers, resources etc. Future research should also explore Cloud architectures and market models that embrace uncertainties and provide continuous "win-win" resolutions (for providers, users and intermediaries) for value and dependability.

4.10 Application Development and Delivery

Agile, continuous, delivery paradigms often come at the expense of reduced reasoning at design-time on quality aspects such as SLA compliance, business alignment, and value-driven design, posing for example a risk of adopting the wrong architecture in the early design stages of a new Cloud application. These risks raise a plethora of research challenges on how to continuously monitor and iteratively evolve the design and quality of Cloud applications. The definition of supporting methods, high-level programming abstractions, tools and organizational processes to address these challenges is currently a limiting factor that requires further research. For example, it is important to extend DevOps methods and define novel programming abstractions to include within existing software development and delivery methodologies a support for IoT, edge computing, Big Data, and serverless computing. Early efforts in these directions are already underway [33].

One possibility to extend existing approaches consists in defining novel *architectural styles* and *Cloud-native design patterns* that will replace ad-hoc sub-optimal practices and thereby limit the risks associated with technical debt arising with agile paradigms. The resulting software architectures and patterns may be specific to a given runtime domain, and tolerate changes in contexts, situations, technologies, or service-level agreements.

For example, with serverless computing and FaaS there is the need for developing novel patterns to define services that combine traditional external services along with the serverless computing services. Here the effect and trade-offs of orchestration of such service mixes need to be investigated systematically. The influence of the underpinning choice of Cloud resources (e.g., on-demand, reserved, spot, burstable) need also to be examined.

As another example, bridging in Edge computing the gap between cyber-physical systems (sensors, actuators, control layer) and the Cloud requires patterns to assist developers in building Cloudlets/swarmlets [112]. These are fragments of an application making local decisions and delegating tasks that cannot be solved locally to other Cloudlets/swarmlets in the Cloud [62], which are further discussed in the "Empowering resource-constrained devices" challenge (Section 2.7).

Developing effective Cloud design patterns also requires fundamental research on meta controls for dynamic and seamless switching between these patterns at runtime, based on their value potentials and prospects. Such meta controllers may rely on software models created by the application designers. Proposals in this direction include model-driven engines to facilitate reasoning, what-if analysis, monitoring feedback analysis, and the correct enactment of adaptation decisions [129].

In order to define patterns and architectures that combine multiple paradigms and technologies, it is also important to develop formalisms to describe the workloads and workflows that the application processes, their requirements in terms of performance, reliability, and security, as well as the associated data properties such as velocity, variety, volume, and veracity. Patterns to decide from such requirements include, which technologies to adopt (e.g., blockchain, SDN, Spark, Storm etc.), need to investigated.

4.11 Data Management

The data management challenge in Cloud computing offers the following future research directions.

While Cloud IaaS and PaaS service for storage and data management focus on file, semi-structured and structured data independently, there is not much explicit focus on metadata management for datasets. Unlike structured data warehouses, the concept of "Data Lakes" encourages enterprises to put all their data into Cloud storage, such as HDFS, to allow knowledge to be mined from it. However, a lack of tracking metadata describing the source and provenance of the data makes it challenging to use them. Scientific repositories have over a decade of experience with managing large and diverse datasets along with the metadata that gives a context of use. Provenance that tracks the processing steps that have taken place to derive a data is also essential, for data quality, auditing and corporate governance. S3 offers some basic versioning capability, but metadata and provenance do not yet form a first-class entity in Cloud data platforms.

A key benefit of CDCs is the centralized collocation and management of data and compute at globally distributed data centres, offering economies of scale. The latency to access to data is however a challenge, along with bandwidth limitations across global networks. While Content Distribution Networks (CDN) such as AWS CloudFront cache data at regional level for web and video delivery, these are designed for slow-changing data and there is no such mechanism to write in data closer to the edge. Having Cloud data services at the Fog layer, which is a generalization of CDN is essential. This is particularly a concern as IoT and 5G mobile networks become widespread.

In addition, Cloud storage has adapted to emerging security and privacy needs with support for HIPAA (Health Insurance Portability and Accountability Act of 1996) and other US and EU regulations for data protection. However, enterprises that handle data that is proprietary and have sensitive trade secrets that can be compromised, if it is accessed by the Cloud provider, still remains a concern. While legal protections exist, there are no clear audit mechanisms to show that data has not been accessed by the Cloud provider themselves. Hybrid solutions where private data centres that are located near the public CDCs with dedicated high-bandwidth network allow users to manage sensitive data under their supervision while also leveraging the benefits of public Clouds[15].

Similarly, the interplay between hybrid models and SDN as well as joint optimisation of data flow placement, elasticity of Fog computing and flow routing can be better explored. Moreover, the computing capabilities of network devices can be leveraged to perform in-transit processing. The optimal placement of data processing applications and adaptation of dataflows, however, are hard problems. This problem becomes even more challenging when considering the placement of stream processing tasks along with allocating bandwidth to meet latency requirements.

Furthermore, frameworks that provide high-level programming abstractions, such as Apache Bean, have been introduced in recent past to ease the development and deployment of Big Data applications that use hybrid models. Platform bindings have been provided to deploy applications developed using these abstractions on the infrastructure provided by commercial public Cloud providers such as Google Cloud Engine, Amazon Web Services, and open source solutions. Although such solutions are often restricted to a single cluster or data centre, efforts have been made to leverage resources from the edges of the Internet to perform distributed queries or to push frequently-performed analytics tasks to edge resources. This requires providing means to place data processing tasks in such environments while minimising the network resource usage and latency. In addition, efficient methods are to be investigated which manage resource elasticity

[15] https://cloud.netapp.com/netapp-private-storage

in such scenarios. Moreover, high-level programming abstractions and bindings to platforms capable of deploying and managing resources under such highly distributed scenarios are desirable.

4.12 Networking

Global network view, programmability, and openness features of SDN provide a promising direction for application of SDN-based traffic engineering mechanisms within and across CDC networks. Using SDN within a data centre network, traffic engineering (TE) can be done much more efficiently and intelligently with dynamic flow scheduling and management based on current network utilization and flow sizes [6]. Even though traffic engineering has been widely used in data networks, distinct features of SDN need a novel set of traffic engineering methods to utilize the available global view of the network and flow characteristics or patterns [5]. During the next decade we will also expect to see techniques targeting network performance requirements such as delay and bandwidth or even jitter guarantees to comply with QoS requirements of the Cloud user application and enforce committed SLAs.

SDN may also influence the security and privacy challenges in Cloud. In general, within the networking community, the overall perception is that SDN will help improve security and reliability both within the network-layer and application-layer. As suggested by [107], the capabilities brought by SDN may be used to introduce novel security services and address some of the on-going issues in Clouds. These include but are not limited to areas such as policy enforcement (for example, firewalling, access control, middleboxes), DoS attack detection and mitigation, monitoring infrastructures for fine-grained security examinations, and traffic anomaly detection.

Nevertheless, as a new technology, the paradigm shift brought by SDN brings along new threat vectors that may be used to target the network itself, services deployed on SDNs and the associated users. For instance, attackers may target the SDN controller as the single point of attack or the inter-SDN communications between the control and data plane - threats that did not exist in traditional networks. At the same time, the impact of existing threats may be magnified such as the range of capabilities available to an adversary who has compromised the network forwarding devices [145]. Hence, importing SDN to Clouds may impact the security of Cloud services in ways that have not been experienced or expected, which requires further research in this area.

In addition, recent advances in AI, ML, and Big Data analytics, have great potential to address networking challenges of Cloud computing and automation of the next-generation networks in Clouds. The potential of these approaches along with centralized network visibility and readily accessible network information (e.g., network topology and traffic statistics) that SDN brings into picture, open up new opportunities to use ML and AI in networking. Even though it is still unclear how these can be incorporated into networking projects, we expect to see this as one of the exotic research areas in the following decade.

The emergence of IoT connecting billions of devices all generating data will place major demands on network infrastructure. 5G wireless and its bandwidth increase will also force significant expansion in network capacity with explosion in the number of mobile devices. Even though a key strategy in addressing latency and lower network resource usage is Edge computing, Edge computing itself is not enough to address all the networking demand. To meet the needs of this transition, new products and technologies expanding bandwidth, or the carrying capacity, of networks are required along with advances in faster broadband technologies and optical networking.

4.13 Usability

There are several opportunities to enhance usability in Cloud environments. For instance, it is still hard for users to know how much they will spend renting resources due to workload/resource fluctuations or characteristics. Tools to have better estimations would definitely improve user experience and satisfaction. Due to recent demands from Big Data community, new visualization technologies could be further explored on the different layers of Cloud environment to better understand infrastructure and application behaviour and highlight insights to end users. Easier API management methodologies, tools, and standards are also necessary to handle users with different levels of expertise and interests. User experience when handling data-intensive applications also need further studies considering their expected QoS.

In addition, users are still overloaded with resource and service types available to run their applications. Examples of resources and services are CPUs, GPUs, network, storage, operating system flavour, and all services available in the PaaS. Advisory systems to help these users would greatly enhance their experience consuming Cloud resources and services. Advisory systems to also recommend how users should use Cloud more efficiently would certainly be beneficial. Advices such as whether data should be transferred or visualized remotely, whether resources should be allocated or deleted, whether baremetal machines should replace virtual ones are examples of hints users could receive to make Cloud easier to use and more cost-effective.

4.14 Discussion

As can be observed from the emerging trends and proposed future research directions (summarized in the outer ring of Figure 3), there will be significant developments across all the service models (IaaS, PaaS and IaaS) of Cloud computing.

In the IaaS there is scope for heterogeneous hardware such as CPUs and accelerators (e.g. GPUs and TPUs) and special purpose Clouds for specific applications (e.g. HPC and deep learning). The future generation Clouds should also be ready to embrace the non-traditional architectures, such as neuromorphic, quantum computing, adiabatic, nanocomputing etc. Moreover, emerging trends such as containerization, SDN and Fog/Edge computing are going to expand the research scope of IaaS by leaps and bounds. Solutions for addressing sustainability of CDC through utilization of renewable energy and IoT-enabled cooling systems are also discussed. There is also scope for emerging trends in IaaS, such as disaggregated datacentres where resources required for the computational tasks such as CPU, memory and storage, will be built as standalone resource blades, which will allow faster and ideal resource provisioning to satisfy different QoS requirements of Cloud based applications. The future research directions proposed for addressing the scalability, resource management and scheduling, heterogeneity, interconnected Clouds and networking challenges, should enable realizing such comprehensive IaaS offered by the Clouds.

Similarly, PaaS should see significant advancements through future research directions in resource management and scheduling. The need for programming abstractions, models, languages and systems supporting scalable elastic computing and seamless use of heterogeneous resources are proposed leading to energy-efficiency, minimized application engineering cost, better portability and guaranteed level of reliability and performance. It is also foreseeable that the ongoing interest for ML, deep learning, and AI applications will help in dealing with the complexity, heterogeneity, scale and load balancing applications developed through PaaS. Serverless computing is an emerging trend in PaaS, which is a promising area to be explored with significant practical and economic impact. Interesting future directions are proposed such as function-level QoS management and economics for serverless computing. In addition, future research directions for data management and analytics are also discussed in detail along with security, leading to interesting applications with platform support such as edge analytics for real-time stream data processing, from the IoT and smart cities domains.

SaaS should mainly see advances from the application development and delivery, and usability of Cloud services. Translucent programming models, languages, and APIs will be needed to enable tackling the complexity of application development while permitting control of application delivery to future-generation Clouds. A variety of agile delivery tools and Cloud standards (e.g., TOSCA) are increasingly being adopted during Cloud application development. The future research should focus at how to continuously monitor and iteratively evolve the design and quality of Cloud applications. It is also suggested to extend DevOps methods and define novel programming abstractions to include within existing software development and delivery methodologies, a support for IoT, Edge computing, Big Data, and serverless computing. Focus should also be at developing effective Cloud design patterns and development of formalisms to describe the workloads and workflows that the application processes, and their requirements in terms of performance, reliability, and security are strongly encouraged. It is also interesting to see that even though the technologies have matured, certain domains such as mobile Cloud, still have adaptability issues. Novel incentive mechanisms are required for mobile Cloud adaptability as well as for designing Fog architectures.

Future research should thus explore Cloud architectures and market models that embrace uncertainties and provide continuous "win-win" resolutions, for all the participants including providers, users and intermediaries, both from the Return On Investment (ROI) and satisfying SLA.

Figure 3: Future research directions in the Cloud computing horizon

5 Summary and Conclusions

Cloud computing paradigm has revolutionized the computer science horizon during the past decade and enabled emergence of computing as the fifth utility. It has emerged as the backbone of modern economy by offering subscription-based services anytime, anywhere following a pay-as-you-go model. Thus, Cloud computing has enabled new businesses to establish in a shorter amount of time, has facilitated the expansion of enterprises across the globe, has accelerated the pace of scientific progress, and has led to the creation of various models of computation for pervasive and ubiquitous applications, among other benefits.

However, the next decade will bring about significant new requirements, from large-scale heterogeneous IoT and sensor networks producing very large data streams to store, manage, and analyse, to energy- and cost-aware personalized computing services that must adapt to a plethora of hardware devices while optimizing for multiple criteria including application-level QoS constraints and economic restrictions. These requirements will be posing several new challenges in Cloud computing and will be creating the need for new approaches and research strategies, and force us to re-evaluate the models that were already developed to address the issues such as scalability, resource provisioning, and security.

This comprehensive manifesto brought the advancements together and proposed the challenges still to be addressed in realizing the future generation Cloud computing. In the process, the manifesto identified the current major challenges in Cloud computing domain and summarized the state-of-the-art along with the limitations. The manifesto also discussed the emerging trends and impact areas that further drive these Cloud computing challenges. Having identified these open issues, the manifesto then offered comprehensive future research directions in the Cloud computing horizon for the next decade. The discussed research directions show a promising and exciting future for the Cloud computing field both technically and economically, and the manifesto calls the community for action in addressing them.

Acknowledgement

We thank Adam Wierman (California Institute of Technology), Shigeru Imai (Rensselaer Polytechnic Institute) and Arash Shaghaghi (University of New South Wales, Sydney) for their comments and suggestions for improving the paper.

References

[1] MQL5 Cloud Network. `https://cloud.mql5.com/`, 2017. [Last visited on 23rd October 2017].

[2] Ubercloud application containers. `https://www.TheUberCloud.com/containers/`, 2017. [Last visited on 23rd October 2017].

[3] Unikernels - rethinking cloud infrastructure. `http://unikernel.org/`, 2017. [Last visited on 23rd October 2017].

[4] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu. Order preserving encryption for numeric data. In *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, pages 563–574. ACM, 2004.

[5] I. F. Akyildiz, A. Lee, P. Wang, M. Luo, and W. Chou. A roadmap for traffic engineering in sdn-openflow networks. *Computer Networks*, 71:1–30, 2014.

[6] M. Al-Fares, S. Radhakrishnan, B. Raghavan, N. Huang, and A. Vahdat. Hedera: Dynamic flow scheduling for data center networks. In *NSDI*, volume 10, pages 19–19, 2010.

[7] P. Anderson, G. Beckett, K. Kavoussanakis, G. Mecheneau, and P. Toft. Technologies for large-scale configuration management. *Te chnical re port, The GridWeaver Project*, 2002.

[8] J. Anselmi, D. Ardagna, J. Lui, A. Wierman, Y. Xu, and Z. Yang. The economics of the cloud. *ACM Trans. on Modeling and Performance Evaluation of Computing Systems (TOMPECS)*, 2(4):18, 2017.

[9] A. Arasu, S. Blanas, K. Eguro, R. Kaushik, D. Kossmann, R. Ramamurthy, and R. Venkatesan. Orthogonal security with cipherbase. In *CIDR*, 2013.

[10] D. Ardagna, G. Casale, M. Ciavotta, J. F. Pérez, and W. Wang. Quality-of-service in cloud computing: modeling techniques and their applications. *Jour of Internet Services and Applications*, 5(1):11, 2014.

[11] S. Arnautov, B. Trach, F. Gregor, T. Knauth, A. Martin, C. Priebe, J. Lind, D. Muthukumaran, D. O'Keeffe, M. Stillwell, et al. Scone: Secure linux containers with intel sgx. In *OSDI*, pages 689–703, 2016.

[12] S. Azodolmolky, P. Wieder, and R. Yahyapour. Cloud computing networking: challenges and opportunities for innovations. *IEEE Communications Magazine*, 51(7):54–62, 2013.

[13] E. Bacis, S. De Capitani di Vimercati, S. Foresti, S. Paraboschi, M. Rosa, and P. Samarati. Mix&slice: Efficient access revocation in the cloud. In *ACM SIGSAC Conf. on Computer and Communications Security*, pages 217–228, 2016.

[14] A. Bahga and V. K. Madisetti. Blockchain platform for industrial internet of things. *J. Softw. Eng. Appl*, 9(10):533, 2016.

[15] A. Balalaie, A. Heydarnoori, and P. Jamshidi. Microservices architecture enables devops: migration to a cloud-native architecture. *IEEE Software*, 33(3):42–52, 2016.

[16] I. Baldini, P. Castro, K. Chang, P. Cheng, S. Fink, V. Ishakian, N. Mitchell, V. Muthusamy, R. Rabbah, A. Slominski, et al. Serverless computing: Current trends and open problems. *arXiv preprint arXiv:1706.03178*, 2017.

[17] A. A. Bankole and S. A. Ajila. Predicting cloud resource provisioning using machine learning techniques. In *Electrical and Computer Engineering (CCECE), 2013 26th Annual IEEE Canadian Conference on*, pages 1–4. IEEE, 2013.

[18] L. Bass, I. Weber, and L. Zhu. *DevOps: A Software Architect's Perspective*. Addison-Wesley Professional, 2015.

[19] A. Beloglazov, J. Abawajy, and R. Buyya. Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. *Future generation computer systems*, 28(5):755–768, 2012.

[20] A. Berl, E. Gelenbe, M. Di Girolamo, G. Giuliani, H. De Meer, M. Q. Dang, and K. Pentikousis. Energy-efficient cloud computing. *The computer journal*, 53(7):1045–1051, 2010.

[21] D. Bernstein, E. Ludvigson, K. Sankar, S. Diamond, and M. Morrow. Blueprint for the intercloud-protocols and formats for cloud computing interoperability. In *Int. Conf. on Internet and Web Applications and Services (ICIW'09)*, pages 328–336. IEEE, 2009.

[22] J. L. Berral, Í. Goiri, T. D. Nguyen, R. Gavalda, J. Torres, and R. Bianchini. Building green cloud services at low cost. In *IEEE 34th Int. Conf. on Distributed Computing Systems (ICDCS)*, pages 449–460. IEEE, 2014.

[23] C. Bizer, T. Heath, and T. Berners-Lee. Linked data-the story so far. *Semantic services, interoperability and web applications: emerging concepts*, pages 205–227, 2009.

[24] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli. Fog computing and its role in the internet of things. In *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, pages 13–16. ACM, 2012.

[25] N. Bonvin, T. G. Papaioannou, and K. Aberer. Autonomic sla-driven provisioning for cloud applications. In *IEEE/ACM int. symp. on cluster, cloud and grid computing*, pages 434–443. IEEE Computer Society, 2011.

[26] R. Brewer. Advanced persistent threats: minimising the damage. *Network Security*, 2014(4):5–9, 2014.

[27] R. Buyya and D. Barreto. Multi-cloud resource provisioning with aneka: A unified and integrated utilisation of microsoft azure and amazon ec2 instances. In *Computing and Network Communications (CoCoNet), 2015 International Conference on*, pages 216–229. IEEE, 2015.

[28] R. Buyya, A. Beloglazov, and J. Abawajy. Energy-efficient management of data center resources for cloud computing: a vision, architectural elements, and open challenges. In *Proceedings of the 2010 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 2010)*. CSREA Press, 2010.

[29] R. Buyya, R. N. Calheiros, J. Son, A. V. Dastjerdi, and Y. Yoon. Software-defined cloud computing: Architectural elements and open challenges. In *International Conference on Advances in Computing, Communications and Informatics (ICACCI 2014)*, pages 1–12. IEEE, 2014.

[30] R. Buyya, S. K. Garg, and R. N. Calheiros. Sla-oriented resource provisioning for cloud computing: Challenges, architecture, and solutions. In *Cloud and Service Computing (CSC), 2011 International Conference on*, pages 1–10. IEEE, 2011.

[31] R. Buyya, R. Ranjan, and R. N. Calheiros. Intercloud: Utility-oriented federation of cloud computing environments for scaling of application services. In *Int. Conf. on Algorithms and Architectures for Parallel Processing*, pages 13–31. Springer, 2010.

[32] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic. Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation computer systems*, 25(6):599–616, 2009.

[33] G. Casale, C. Chesta, P. Deussen, E. Di Nitto, P. Gouvas, S. Koussouris, V. Stankovski, A. Symeonidis, V. Vlassiou, A. Zafeiropoulos, et al. Current and future challenges of software engineering for services and applications. *Procedia Computer Science*, 97:34–42, 2016.

[34] E. Casalicchio and L. Silvestri. Mechanisms for sla provisioning in cloud-based service providers. *Computer Networks*, 57(3):795–810, 2013.

[35] I. Casas, J. Taheri, R. Ranjan, and A. Y. Zomaya. Pso-ds: a scheduling engine for scientific workflow managers. *The Journal of Supercomputing*, 73(9):3924–3947, 2017.

[36] C. Chang, S. N. Srirama, and R. Buyya. Indie Fog: An Efficient Fog-Computing Infrastructure for the Internet of Things. *IEEE Computer*, 50(9):92–98, 2017.

[37] M. Chiosi, D. Clarke, P. Willis, A. Reid, J. Feger, M. Bugenhagen, W. Khan, M. Fargano, C. Cui, H. Deng, et al. Network functions virtualisation: An introduction, benefits, enablers, challenges and call for action. In *SDN and OpenFlow World Congress*, pages 22–24, 2012.

[38] D. Cho, J. Taheri, A. Y. Zomaya, and P. Bouvry. Real-time virtual network function (vnf) migration toward low network latency in cloud environments. In *Cloud Computing (CLOUD), 2017 IEEE 10th International Conference on*, pages 798–801. IEEE, 2017.

[39] B.-G. Chun, S. Ihm, P. Maniatis, M. Naik, and A. Patti. Clonecloud: elastic execution between mobile device and cloud. In *Proceedings of the sixth conference on Computer systems*, pages 301–314. ACM, 2011.

[40] P. Church, A. Goscinski, and C. Lefèvre. Exposing hpc and sequential applications as services through the development and deployment of a saas cloud. *Future Generation Computer Systems*, 43:24–37, 2015.

[41] V. Ciriani, S. D. C. D. Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati. Combining fragmentation and encryption to protect privacy in data storage. *ACM Transactions on Information and System Security (TISSEC)*, 13(3):22, 2010.

[42] Cloud Standards Customer Council. Cloud customer architecture for blockchain. Technical report, 2017.

[43] Coupa Software. Usability in enterprise cloud applications. Technical report, Coupa Software, 2012.

[44] S. Crago, K. Dunn, P. Eads, L. Hochstein, D.-I. Kang, M. Kang, D. Modium, K. Singh, J. Suh, and J. P. Walters. Heterogeneous cloud computing. In *Cluster Computing (CLUSTER), 2011 IEEE International Conference on*, pages 378–385. IEEE, 2011.

[45] R. Cziva, S. Jouët, D. Stapleton, F. P. Tso, and D. P. Pezaros. Sdn-based virtual machine management for cloud data centers. *IEEE Transactions on Network and Service Management*, 13(2):212–225, 2016.

[46] E. Damiani, S. Vimercati, S. Jajodia, S. Paraboschi, and P. Samarati. Balancing confidentiality and efficiency in untrusted relational dbmss. In *Proceedings of the 10th ACM conference on Computer and communications security*, pages 93–102. ACM, 2003.

[47] A. V. Dastjerdi and R. Buyya. Compatibility-aware cloud service composition under fuzzy preferences of users. *IEEE Transactions on Cloud Computing*, 2(1):1–13, 2014.

[48] A. V. Dastjerdi and R. Buyya. Fog computing: Helping the internet of things realize its potential. *Computer*, 49(8):112–116, 2016.

[49] S. De Capitani di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati. Efficient integrity checks for join queries in the cloud. *Journal of Computer Security*, 24(3):347–378, 2016.

[50] S. De Capitani di Vimercati, G. Livraga, V. Piuri, P. Samarati, and G. A. Soares. Supporting application requirements in cloud-based iot information processing. In *International Conference on Internet of Things and Big Data (IoTBD 2016)*, pages 65–72. Scitepress, 2016.

[51] J. Dean. Large-scale distributed systems at google: Current systems and future directions. In *The 3rd ACM SIGOPS International Workshop on Large Scale Distributed Systems and Middleware (LADIS 2009) Tutorial*, 2009.

[52] E. Deelman, C. Carothers, A. Mandal, B. Tierney, J. S. Vetter, I. Baldin, C. Castillo, G. Juve, et al. Panorama: an approach to performance modeling and diagnosis of extreme-scale workflows. *The International Journal of High Performance Computing Applications*, 31(1):4–18, 2017.

[53] T. Desell, M. Magdon-Ismail, B. Szymanski, C. Varela, H. Newberg, and N. Cole. Robust asynchronous optimization for volunteer computing grids. In *e-Science, 2009. e-Science'09. Fifth IEEE International Conference on*, pages 263–270. IEEE, 2009.

[54] S. D. C. di Vimercati, S. Foresti, R. Moretti, S. Paraboschi, G. Pelosi, and P. Samarati. A dynamic tree-based data structure for access privacy in the cloud. In *Cloud Computing Technology and Science (CloudCom), 2016 IEEE International Conference on*, pages 391–398. IEEE, 2016.

[55] H. Duan, C. Chen, G. Min, and Y. Wu. Energy-aware Scheduling of Virtual Machines in Heterogeneous Cloud Computing Systems. *Future Generation Computer Systems*, 74:142 – 150, 2017.

[56] A. Eivy. Be wary of the economics of "serverless" cloud computing. *IEEE Cloud Computing*, 4(2):6–12, 2017.

[57] V. Eramo, E. Miucci, M. Ammar, and F. G. Lavacca. An approach for service function chain routing and virtual function network instance migration in network function virtualization architectures. *IEEE/ACM Transactions on Networking*, 2017.

[58] D. Evans. The internet of things: How the next evolution of the internet is changing everything. *CISCO white paper*, 1(2011):1–11, 2011.

[59] C. M. N. Faisal. Issues in cloud computing: Usability evaluation of cloud based application. 2011.

[60] F. Faniyi and R. Bahsoon. A systematic review of service level management in the cloud. *ACM Computing Surveys (CSUR)*, 48(3):43, 2016.

[61] H. Flores, P. Hui, S. Tarkoma, Y. Li, S. Srirama, and R. Buyya. Mobile code offloading: from concept to practice and beyond. *IEEE Communications Magazine*, 53(3):80–88, 2015.

[62] H. Flores and S. N. Srirama. Mobile cloud middleware. *Journal of Systems and Software*, 92:82–94, 2014.

[63] I. Friedberg, F. Skopik, G. Settanni, and R. Fiedler. Combating advanced persistent threats: From network event correlation to incident detection. *Computers & Security*, 48:35–57, 2015.

[64] E. Gaetani, L. Aniello, R. Baldoni, F. Lombardi, A. Margheri, and V. Sassone. Blockchain-based database to ensure data integrity in cloud computing environments. In *ITASEC*, pages 146–155, 2017.

[65] J. Gao and R. Jamidar. Machine learning applications for data center optimization. *Google White Paper*, 2014.

[66] P. X. Gao, A. Narayan, S. Karandikar, J. Carreira, S. Han, R. Agarwal, S. Ratnasamy, and S. Shenker. Network requirements for resource disaggregation. In *OSDI*, pages 249–264, 2016.

[67] P. Garcia Lopez, A. Montresor, D. Epema, A. Datta, T. Higashino, A. Iamnitchi, M. Barcellos, P. Felber, and E. Riviere. Edge-centric computing: Vision and challenges. *ACM SIGCOMM Computer Communication Review*, 45(5):37–42, 2015.

[68] S. K. Garg, S. Versteeg, and R. Buyya. A framework for ranking of cloud computing services. *Future Generation Computer Systems*, 29(4):1012–1023, 2013.

[69] E. Gelenbe. Adaptive management of energy packets. In *Computer Software and Applications Conference Workshops (COMPSACW), 2014 IEEE 38th International*, pages 1–6. IEEE, 2014.

[70] E. Gelenbe and Y. Caseau. The impact of information technology on energy consumption and carbon emissions. *Ubiquity*, 2015(June):1, 2015.

[71] E. Gelenbe and E. T. Ceran. Energy packet networks with energy harvesting. *IEEE Access*, 4:1321–1331, 2016.

[72] E. Gelenbe and C. Morfopoulou. Power savings in packet networks via optimised routing. *Mobile Networks and Applications*, 17(1):152–159, 2012.

[73] W. Gentzsch and B. Yenier. Novel software containers for engineering and scientific simulations in the cloud. *International Journal of Grid and High Performance Computing (IJGHPC)*, 8(1):38–49, 2016.

[74] R. Ghosh, K. S. Trivedi, V. K. Naik, and D. S. Kim. End-to-end performability analysis for infrastructure-as-a-service cloud: An interacting stochastic models approach. In *Dependable Computing (PRDC), 2010 IEEE 16th Pacific Rim International Symposium on*, pages 125–132. IEEE, 2010.

[75] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami. Internet of things (iot): A vision, architectural elements, and future directions. *Future generation computer systems*, 29(7):1645–1660, 2013.

[76] H. S. Gunawi, T. Do, J. M. Hellerstein, I. Stoica, D. Borthakur, and J. Robbins. Failure as a service (faas): A cloud service for large-scale, online failure drills. *University of California, Berkeley*, 3, 2011.

[77] C. Guo, G. Lu, H. J. Wang, S. Yang, C. Kong, P. Sun, W. Wu, and Y. Zhang. Secondnet: a data center network virtualization architecture with bandwidth guarantees. In *Proceedings of the 6th International COnference*, page 15. ACM, 2010.

[78] A. Gupta, P. Faraboschi, F. Gioachin, L. V. Kale, R. Kaufmann, B.-S. Lee, V. March, D. Milojicic, and C. H. Suen. Evaluating and improving the performance and scheduling of hpc applications in cloud. *IEEE Transactions on Cloud Computing*, 4(3):307–321, 2016.

[79] H. Hacigümüş, B. Iyer, C. Li, and S. Mehrotra. Executing sql over encrypted data in the database-service-provider model. In *2002 ACM SIGMOD int. conf. on Management of data*, pages 216–227. ACM, 2002.

[80] A. Hameed, A. Khoshkbarforoushha, R. Ranjan, P. P. Jayaraman, J. Kolodziej, P. Balaji, S. Zeadally, Q. M. Malluhi, N. Tziritas, A. Vishnu, S. U. Khan, and A. Zomaya. A Survey and Taxonomy on Energy Efficient Resource Allocation Techniques for Cloud Computing Systems. *Computing*, 98(7):751–774, July 2016.

[81] Y. Han, T. Alpcan, J. Chan, C. Leckie, and B. I. Rubinstein. A game theoretical approach to defend against co-resident attacks in cloud computing: Preventing co-residence using semi-supervised learning. *IEEE Transactions on Information Forensics and Security*, 11(3):556–570, 2016.

[82] Y. Han, J. Chan, T. Alpcan, and C. Leckie. Using virtual machine allocation policies to defend against co-resident attacks in cloud computing. *IEEE Tran. on Dependable and Secure Computing*, 14(1):95–108, 2017.

[83] T. Harter, B. Salmon, R. Liu, A. C. Arpaci-Dusseau, and R. H. Arpaci-Dusseau. Slacker: Fast distribution with lazy docker containers. In *FAST*, volume 16, pages 181–195, 2016.

[84] B. Heller, S. Seetharaman, P. Mahadevan, Y. Yiakoumis, P. Sharma, S. Banerjee, and N. McKeown. Elastictree: Saving energy in data center networks. In *Nsdi*, volume 10, pages 249–264, 2010.

[85] S. Hendrickson, S. Sturdevant, T. Harter, V. Venkataramani, A. C. Arpaci-Dusseau, and R. H. Arpaci-Dusseau. Serverless computation with openlambda. *Elastic*, 60:80, 2016.

[86] B. Hindman, A. Konwinski, M. Zaharia, A. Ghodsi, A. D. Joseph, R. H. Katz, S. Shenker, and I. Stoica. Mesos: A platform for fine-grained resource sharing in the data center. In *NSDI*, volume 11, pages 22–22, 2011.

[87] Q. Huang. Development of a saas application probe to the physical properties of the earth's interior: An attempt at moving hpc to the cloud. *Computers & Geosciences*, 70:147–153, 2014.

[88] E. Huedo, R. S. Montero, R. Moreno, I. M. Llorente, A. Levin, and P. Massonet. Interoperable federated cloud networking. *IEEE Internet Computing*, 21(5):54–59, 2017.

[89] IDC. Worldwide semiannual big data and analytics spending guide. `http://www.idc.com/getdoc.jsp?containerId=prUS42321417`, Feb 2017.

[90] IDG Enterprise. 2016 idg enterprise cloud computing survey. `https://www.idgenterprise.com/resource/research/2016-idg-enterprise-cloud-computing-survey/`, 2016.

[91] IEEE. IEEE Rebooting Computing. `https://rebootingcomputing.ieee.org/`, 2017.

[92] S. Imai, T. Chestna, and C. A. Varela. Elastic scalable cloud computing using application-level migration. In *Utility and Cloud Computing (UCC), 2012 IEEE Fifth International Conference on*, pages 91–98. IEEE, 2012.

[93] S. Imai, T. Chestna, and C. A. Varela. Accurate resource prediction for hybrid iaas clouds using workload-tailored elastic compute units. In *Utility and Cloud Computing (UCC), 2013 IEEE/ACM 6th International Conference on*, pages 171–178. IEEE, 2013.

[94] S. Imai, P. Patel, and C. A. Varela. Developing elastic software for the cloud. *Encyclopedia on Cloud Computing*, 2016.

[95] S. Imai, S. Patterson, and C. A. Varela. Maximum sustainable throughput prediction for data stream processing over public clouds. In *Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, pages 504–513. IEEE Press, 2017.

[96] S. Jain, A. Kumar, S. Mandal, J. Ong, L. Poutievski, A. Singh, S. Venkata, J. Wanderer, J. Zhou, M. Zhu, et al. B4: Experience with a globally-deployed software defined wan. *ACM SIGCOMM Computer Communication Review*, 43(4):3–14, 2013.

[97] B. Javadi, J. Abawajy, and R. Buyya. Failure-aware resource provisioning for hybrid cloud infrastructure. *Journal of parallel and distributed computing*, 72(10):1318–1331, 2012.

[98] B. Javed, P. Bloodsworth, R. U. Rasool, K. Munir, and O. Rana. Cloud market maker: An automated dynamic pricing marketplace for cloud users. *Future Generation Computer Systems*, 54:52–67, 2016.

[99] B. Jennings and R. Stadler. Resource management in clouds: Survey and research challenges. *Journal of Network and Systems Management*, 23(3):567–619, 2015.

[100] N. P. Jouppi, C. Young, N. Patil, D. Patterson, G. Agrawal, R. Bajwa, S. Bates, S. Bhatia, N. Boden, A. Borchers, et al. In-datacenter performance analysis of a tensor processing unit. *arXiv preprint arXiv:1704.04760*, 2017.

[101] C. Kachris, D. Soudris, G. Gaydadjiev, H.-N. Nguyen, D. S. Nikolopoulos, A. Bilas, N. Morgan, C. Strydis, et al. The vineyard approach: Versatile, integrated, accelerator-based, heterogeneous data centres. In *International Symposium on Applied Reconfigurable Computing*, pages 3–13. Springer, 2016.

[102] Y. Kang, J. Hauswald, C. Gao, A. Rovinski, T. Mudge, J. Mars, and L. Tang. Neurosurgeon: Collaborative intelligence between the cloud and mobile edge. In *Proceedings of the Twenty-Second International Conference on Architectural Support for Programming Languages and Operating Systems*, pages 615–629. ACM, 2017.

[103] J. M. Kaplan, W. Forrest, and N. Kindler. Revolutionizing data center energy efficiency. Technical report, Technical report, McKinsey & Company, 2008.

[104] J. O. Kephart and D. M. Chess. The vision of autonomic computing. *Computer*, 36(1):41–50, 2003.

[105] A. Khosravi and R. Buyya. Energy and carbon footprint-aware management of geo-distributed cloud data centers: A taxonomy, state of the art. *Advancing Cloud Database Systems and Capacity Planning With Dynamic Applications*, page 27, 2017.

[106] M. Kiran, P. Murphy, I. Monga, J. Dugan, and S. S. Baveja. Lambda architecture for cost-effective batch and speed big data processing. In *IEEE Intl Conf. on Big Data*, pages 2785–2792. IEEE, 2015.

[107] D. Kreutz, F. M. Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig. Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, 103(1):14–76, 2015.

[108] A. G. Kumbhare, Y. Simmhan, M. Frincu, and V. K. Prasanna. Reactive resource provisioning heuristics for dynamic dataflows on cloud infrastructure. *IEEE Transactions on Cloud Computing*, 3(2):105–118, 2015.

[109] R. Kune, P. K. Konugurthi, A. Agarwal, R. R. Chillarige, and R. Buyya. The anatomy of big data computing. *Software: Practice and Experience*, 46(1):79–105, 2016.

[110] T.-W. Kuo, B.-H. Liou, K. C.-J. Lin, and M.-J. Tsai. Deploying chains of virtual network functions: On the relation between link and server usage. In *Computer Communications, IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on*, pages 1–9. IEEE, 2016.

[111] H. A. Lagar-Cavilla, J. A. Whitney, A. M. Scannell, P. Patchin, S. M. Rumble, E. De Lara, M. Brudno, and M. Satyanarayanan. Snowflock: rapid virtual machine cloning for cloud computing. In *Proceedings of the 4th ACM European conference on Computer systems*, pages 1–12. ACM, 2009.

[112] E. A. Lee, B. Hartmann, J. Kubiatowicz, T. S. Rosing, J. Wawrzynek, D. Wessel, J. Rabaey, K. Pister, A. Sangiovanni-Vincentelli, S. A. Seshia, et al. The swarm at the edge of the cloud. *IEEE Design & Test*, 31(3):8–20, 2014.

[113] G. Liu and T. Wood. Cloud-scale application performance monitoring with sdn and nfv. In *Cloud Engineering (IC2E), 2015 IEEE International Conference on*, pages 440–445. IEEE, 2015.

[114] Z. Liu, M. Lin, A. Wierman, S. Low, and L. L. Andrew. Greening geographical load balancing. *IEEE/ACM Transactions on Networking (TON)*, 23(2):657–671, 2015.

[115] M. Liyanage, C. Chang, and S. N. Srirama. mePaaS: mobile-embedded platform as a service for distributing fog computing to edge nodes. In *Parallel and Distributed Computing, Applications and Technologies (PDCAT), 2016 17th International Conference on*, pages 73–80. IEEE, 2016.

[116] R. V. Lopes and D. Menascé. A taxonomy of job scheduling on distributed computing systems. *IEEE Transactions on Parallel and Distributed Systems*, 27(12):3412–3428, 2016.

[117] P. Mahadevan, P. Sharma, S. Banerjee, and P. Ranganathan. A power benchmarking framework for network devices. *NETWORKING 2009*, pages 795–808, 2009.

[118] M. Malawski, G. Juve, E. Deelman, and J. Nabrzyski. Algorithms for cost-and deadline-constrained provisioning for scientific workflow ensembles in iaas clouds. *Future Generation Computer Systems*, 48:1–18, 2015.

[119] Z. Á. Mann. Allocation of virtual machines in cloud data centers-a survey of problem models and optimization algorithms. *ACM Computing Surveys (CSUR)*, 48(1):11, 2015.

[120] G. McGrath and P. R. Brenner. Serverless computing: Design, implementation, and performance. In *Distributed Computing Systems Workshops (ICDCSW), 2017 IEEE 37th International Conference on*, pages 405–410. IEEE, 2017.

[121] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner. Openflow: enabling innovation in campus networks. *ACM SIGCOMM Computer Communication Review*, 38(2):69–74, 2008.

[122] D. Merkel. Docker: lightweight linux containers for consistent development and deployment. *Linux Journal*, 2014(239):2, 2014.

[123] R. Moreno-Vozmediano, R. S. Montero, and I. M. Llorente. Iaas cloud architecture: From virtualized datacenters to federated cloud infrastructures. *Computer*, 45(12):65–72, 2012.

[124] K.-K. Muniswamy-Reddy and M. Seltzer. Provenance as first class cloud data. *ACM SIGOPS Operating Systems Review*, 43(4):11–16, 2010.

[125] R. Nachiappan, B. Javadi, R. Calherios, and K. Matawie. Cloud storage reliability for big data applications: A state of the art survey. *Journal of Network and Computer Applications*, 2017.

[126] T. D. Nadeau and K. Gray. *SDN: Software Defined Networks: An Authoritative Review of Network Programmability Technologies*. " O'Reilly Media, Inc.", 2013.

[127] Y. Nan, W. Li, W. Bao, F. C. Delicato, P. F. Pires, and A. Y. Zomaya. Cost-effective processing for delay-sensitive applications in cloud of things systems. In *Network Computing and Applications (NCA), 2016 IEEE 15th International Symposium on*, pages 162–169. IEEE, 2016.

[128] M. Naveed, S. Kamara, and C. V. Wright. Inference attacks on property-preserving encrypted databases. In *22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 644–655. ACM, 2015.

[129] E. D. Nitto, P. Matthews, D. Petcu, and A. Solberg. Model-driven development and operation of multi-cloud applications: The modaclouds approach. 2017.

[130] Open Networking Foundation. Software-defined networking (sdn) definition. `https://www.opennetworking.org/sdn-resources/sdn-definition`, 2017. [Last visited on 23rd October 2017].

[131] C. Pahl and B. Lee. Containers and clusters for edge cloud architectures–a technology review. In *Future Internet of Things and Cloud (FiCloud), 2015 3rd International Conference on*, pages 379–386. IEEE, 2015.

[132] B. Pernici, M. Aiello, J. vom Brocke, B. Donnellan, E. Gelenbe, and M. Kretsis. What is can do for environmental sustainability: A report from caise'11 panel on green and sustainable is. *CAIS*, 30:18, 2012.

[133] J. E. Pezoa and M. M. Hayat. Performance and reliability of non-markovian heterogeneous distributed computing systems. *IEEE Transactions on Parallel and Distributed Systems*, 23(7):1288–1301, 2012.

[134] C. Pham, N. H. Tran, S. Ren, W. Saad, and C. S. Hong. Traffic-aware and energy-efficient vnf placement for service chaining: Joint sampling and matching approach. *IEEE Transactions on Services Computing*, 2017.

[135] G. Pollock, D. Thompson, J. Sventek, and P. Goldsack. The asymptotic configuration of application components in a distributed system. 1998.

[136] R. A. Popa, C. Redfield, N. Zeldovich, and H. Balakrishnan. Cryptdb: protecting confidentiality with encrypted query processing. In *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles*, pages 85–100. ACM, 2011.

[137] A. Putnam, A. M. Caulfield, E. S. Chung, D. Chiou, K. Constantinides, J. Demme, H. Esmaeilzadeh, J. Fowers, G. P. Gopal, J. Gray, et al. A reconfigurable fabric for accelerating large-scale datacenter services. In *2014 ACM/IEEE 41st Int. Symp. onComputer Architecture (ISCA)*, pages 13–24. IEEE, 2014.

[138] M. Rajkumar, A. K. Pole, V. S. Adige, and P. Mahanta. Devops culture and its impact on cloud delivery and software development. In *Int. Conf. on Advances in Computing, Communication, & Automation (ICACCA)*. IEEE, 2016.

[139] M. Roberts. Serverless architectures. `https://martinfowler.com/articles/serverless.html`, 2016.

[140] B. Rochwerger, D. Breitgand, E. Levy, A. Galis, K. Nagin, I. M. Llorente, R. Montero, Y. Wolfsthal, E. Elmroth, J. Caceres, et al. The reservoir model and architecture for open federated cloud computing. *IBM Journal of Research and Development*, 53(4):4–1, 2009.

[141] B. Ruan, H. Huang, S. Wu, and H. Jin. A performance study of containers in cloud environment. In *Advances in Services Computing: 10th Asia-Pacific Services Computing Conference, APSCC 2016, Zhangjiajie, China, November 16-18, 2016, Proceedings 10*, pages 343–356. Springer, 2016.

[142] F. Samreen, Y. Elkhatib, M. Rowe, and G. S. Blair. Daleel: Simplifying cloud instance selection using machine learning. In *Network Operations and Management Symposium (NOMS), 2016 IEEE/IFIP*, pages 557–563. IEEE, 2016.

[143] E. F. Z. Santana, A. P. Chaves, M. A. Gerosa, F. Kon, and D. Milojicic. Software platforms for smart cities: Concepts, requirements, challenges, and a unified reference architecture. *arXiv preprint arXiv:1609.08089*, 2016.

[144] M. Satyanarayanan, P. Simoens, Y. Xiao, P. Pillai, Z. Chen, K. Ha, W. Hu, and B. Amos. Edge analytics in the internet of things. *IEEE Pervasive Computing*, 14(2):24–31, 2015.

[145] A. Shaghaghi, M. A. Kaafar, and S. Jha. Wedgetail: An intrusion prevention system for the data plane of software defined networks. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, pages 849–861. ACM, 2017.

[146] Y. Sharma, B. Javadi, W. Si, and D. Sun. Reliability and energy efficiency in cloud computing systems: Survey and taxonomy. *Journal of Network and Computer Applications*, 74:66–85, 2016.

[147] W. Shi and S. Dustdar. The promise of edge computing. *Computer*, 49(5):78–81, 2016.

[148] J. Shuja, R. W. Ahmad, A. Gani, A. I. A. Ahmed, A. Siddiqa, K. Nisar, S. U. Khan, and A. Y. Zomaya. Greening emerging it technologies: techniques and practices. *Journal of Internet Services and Applications*, 8(1):9, 2017.

[149] M. Singhal, S. Chandrasekhar, T. Ge, R. Sandhu, R. Krishnan, G.-J. Ahn, and E. Bertino. Collaboration in multicloud computing environments: Framework and security issues. *Computer*, 46(2):76–84, 2013.

[150] S. Soltesz, H. Pötzl, M. E. Fiuczynski, A. Bavier, and L. Peterson. Container-based operating system virtualization: a scalable, high-performance alternative to hypervisors. In *ACM SIGOPS Operating Systems Review*, volume 41, pages 275–287. ACM, 2007.

[151] G. Somani, M. S. Gaur, D. Sanghi, M. Conti, and R. Buyya. Ddos attacks in cloud computing: issues, taxonomy, and future directions. *Computer Communications*, 107:30–48, 2017.

[152] B. Sotomayor, R. S. Montero, I. M. Llorente, and I. Foster. Virtual infrastructure management in private and hybrid clouds. *IEEE Internet computing*, 13(5), 2009.

[153] J. Spillner. Snafu: Function-as-a-service (faas) runtime design and implementation. *arXiv preprint arXiv:1703.07562*, 2017.

[154] S. N. Srirama. Mobile web and cloud services enabling Internet of Things. *CSI transactions on ICT*, 5(1):109–117, 2017.

[155] S. N. Srirama and A. Ostovar. Optimal resource provisioning for scaling enterprise applications on the cloud. In *6th International Conference on Cloud Computing Technology and Science (CloudCom)*, pages 262–271. IEEE, 2014.

[156] B. Stanton, M. Theofanos, and K. P. Joshi. Framework for cloud usability. In *International Conference on Human Aspects of Information Security, Privacy, and Trust*, pages 664–671. Springer, 2015.

[157] B. Stein and A. Morrison. The enterprise data lake: Better integration and deeper analytics. *PwC Technology Forecast: Rethinking integration*, 1:1–9, 2014.

[158] I. Stojmenovic, S. Wen, X. Huang, and H. Luan. An overview of fog computing and its security issues. *Concurrency and Computation: Practice and Experience*, 28(10):2991–3005, 2016.

[159] M. Swan. *Blockchain: Blueprint for a new economy.* " O'Reilly Media, Inc.", 2015.

[160] Z. Tari, X. Yi, U. S. Premarathne, P. Bertok, and I. Khalil. Security and privacy in cloud computing: vision, trends, and challenges. *IEEE Cloud Computing*, 2(2):30–38, 2015.

[161] A. N. Toosi, R. N. Calheiros, and R. Buyya. Interconnected cloud computing environments: Challenges, taxonomy, and survey. *ACM Computing Surveys (CSUR)*, 47(1):7, 2014.

[162] D. K. Tosh, S. Shetty, X. Liang, C. A. Kamhoua, K. A. Kwiat, and L. Njilla. Security implications of blockchain cloud with analysis of block withholding attack. In *Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, pages 458–467. IEEE Press, 2017.

[163] A. Vahdat, D. Clark, and J. Rexford. A purpose-built global network: Google's move to sdn. *Queue*, 13(8):100, 2015.

[164] D. Van Aken, A. Pavlo, G. J. Gordon, and B. Zhang. Automatic database management system tuning through large-scale machine learning. In *Proceedings of the 2017 ACM International Conference on Management of Data*, pages 1009–1024. ACM, 2017.

[165] L. M. Vaquero and L. Rodero-Merino. Finding your way in the fog: Towards a comprehensive definition of fog computing. *ACM SIGCOMM Computer Communication Review*, 44(5):27–32, 2014.

[166] C. A. Varela and G. Agha. *Programming Distributed Computing Systems: A Foundational Approach.* MIT Press, 2013.

[167] B. Varghese, O. Akgun, I. Miguel, L. Thai, and A. Barker. Cloud benchmarking for maximising performance of scientific applications. *IEEE Transactions on Cloud Computing*, 2016.

[168] B. Varghese and R. Buyya. Next generation cloud computing: New trends and research directions. *Future Generation Computer Systems*, 2017.

[169] B. Varghese, N. Wang, S. Barbhuiya, P. Kilpatrick, and D. S. Nikolopoulos. Challenges and opportunities in edge computing. In *Smart Cloud (SmartCloud), IEEE International Conference on*, pages 20–26. IEEE, 2016.

[170] P. Varshney and Y. Simmhan. Demystifying fog computing: Characterizing architectures, applications and abstractions. In *International Conference on Fog and Edge Computing (ICFEC)*, 2017.

[171] S. D. C. D. Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati. Encryption policies for regulating access to outsourced data. *ACM Transactions on Database Systems (TODS)*, 35(2):12, 2010.

[172] K. V. Vishwanath and N. Nagappan. Characterizing cloud computing hardware reliability. In *Proceedings of the 1st ACM symposium on Cloud computing*, pages 193–204. ACM, 2010.

[173] L. Wang, O. Brun, and E. Gelenbe. Adaptive workload distribution for local and remote clouds. In *Systems, Man, and Cybernetics (SMC), 2016 IEEE International Conference on*, pages 003984–003988. IEEE, 2016.

[174] L. Wang and E. Gelenbe. Adaptive dispatching of tasks in the cloud. *IEEE Transactions on Cloud Computing*, 2015.

[175] N. Wang, B. Varghese, M. Matthaiou, and D. S. Nikolopoulos. Enorm: A framework for edge node resource management. *IEEE Transactions on Services Computing*, 2017.

[176] S. Wang, R. Urgaonkar, M. Zafer, T. He, K. Chan, and K. K. Leung. Dynamic service migration in mobile edge-clouds. In *IFIP Networking Conference (IFIP Networking), 2015*, pages 1–9. IEEE, 2015.

[177] W. Wolf. Cyber-physical systems. *Computer*, 42(3):88–89, 2009.

[178] S. Wu, C. Niu, J. Rao, H. Jin, and X. Dai. Container-based cloud platform for mobile computation offloading. In *Parallel and Distributed Processing Symposium (IPDPS), 2017 IEEE International*, pages 123–132. IEEE, 2017.

[179] M. G. Xavier, M. V. Neves, F. D. Rossi, T. C. Ferreto, T. Lange, and C. A. De Rose. Performance evaluation of container-based virtualization for high performance computing environments. In *Parallel, Distributed and Network-Based Processing (PDP), 2013 21st Euromicro International Conference on*, pages 233–240. IEEE, 2013.

[180] L. Xiao, D. Xu, C. Xie, N. B. Mandayam, and H. V. Poor. Cloud storage defense against advanced persistent threats: A prospect theoretic study. *IEEE Journal on Selected Areas in Communications*, 35(3):534–544, 2017.

[181] M. Yan, P. Castro, P. Cheng, and V. Ishakian. Building a chatbot with serverless computing. In *Proceedings of the 1st International Workshop on Mashups of Things and APIs*, page 5. ACM, 2016.

[182] Q. Yan, F. R. Yu, Q. Gong, and J. Li. Software-defined networking (sdn) and distributed denial of service (ddos) attacks in cloud computing environments: A survey, some research issues, and challenges. *IEEE Communications Surveys & Tutorials*, 18(1):602–622, 2016.

[183] Y. Yin, L. Wang, and E. Gelenbe. Multi-layer neural networks for quality of service oriented server-state classification in cloud servers. In *2017 Int. Joint Conf. on Neural Networks (IJCNN),*, pages 1623–1627. IEEE, 2017.

[184] M. Zaharia, M. Chowdhury, T. Das, A. Dave, J. Ma, M. McCauley, M. J. Franklin, S. Shenker, and I. Stoica. Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing. In *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*, pages 2–2. USENIX Association, 2012.

[185] B. Zhou, A. V. Dastjerdi, R. Calheiros, S. Srirama, and R. Buyya. mCloud: A Context-aware offloading framework for heterogeneous mobile cloud. *IEEE Transactions on Services Computing*, 10(5):797–810, 2017.

[186] Q. Zhou, Y. Simmhan, and V. Prasanna. Knowledge-infused and consistent complex event processing over real-time and persistent streams. *Future Generation Computer Systems*, 76:391–406, 2017.

[187] T. Zhu, G. Li, W. Zhou, and S. Y. Philip. Differentially private data publishing and analysis: a survey. *IEEE Transactions on Knowledge and Data Engineering*, 2017.